Examine^{3D}

A 3D computer-aided engineering
analysis package for underground
excavations in rock

USERS' MANUAL
Version 4.0

Acknowledgments

EXAMINE^{3D} is a computer-aided engineering analysis package for underground structures, developed by the Rock Engineering Group, Department of Civil Engineering, University of Toronto. It currently includes modules for geometric modeling, surface meshing, elastic stress analysis based on the boundary element method, and data visualization/interpretation. The initial funding for the development of EXAMINE^{3D} was generously provided by:

NORANDA MINERALS INC.
ONTARIO GOVERNMENT (URIF)

Additional funding (1995-97) was provided by CANMET through the Canada/Ontario Mineral Development Agreement.

The principal contributors to the EXAMINE^{3D} project are:

Overall Conceptual Design:	John H. Curran and Brent T. Corkum
Programming of EXAMINE^{3D} modules:	
Boundary Element Stress Analysis	Vijayakumar Sinnathurai & Sanjiv Shah
Geometric Modeling/Surface Meshing	Brent T. Corkum
Data Visualization/Interpretation	Brent T. Corkum

The contributions of the following people are also gratefully acknowledged:

Will Bawden	Paul Young	Goodluck OfoegbuIgor Pashutinski	
Yves Potvin	Shawn Maxwell	James McLeod	Thamer Yacoub
Don Grant	Mike Neumann	Kathir Mailvaganam	
Evert Hoek	Edward Dzik	Murray Grabinsky	
Derek Martin	Mark Diederichs	Andrew Wyllie	

Contents

1. Getting Started

EXAMINE^{3D} is a general purpose program for the modeling and visualization of three dimensional geometry and data. Most of its functionalities are geared towards the generation of input data for, and the visualization of analysis results from, a companion program COMPUTE$^{3D\text{-}BEM}$ (a three dimensional boundary element stress analysis program), which is supplied with EXAMINE^{3D}. On the other hand, its use extends far beyond the boundary element program. It has been used successfully to visualize three dimensional finite element data and microseismic monitoring data for mines. The program is generally applicable to any three dimensional data, as long as the data is written in the appropriate format. The format of EXAMINE^{3D} files is described later in this manual.

New users of EXAMINE^{3D} are encouraged to complete all the tutorial exercises in Chapters 6 thru 15 (after going through Chapter 1); those already familiar with EXAMINE^{3D} should still read Chapter 1 and any other required section of the Manual.

1.1 System Requirements

- Win32 windows version for Pentium PC's running Microsoft Windows 95/NT. Minimum 16MB of memory (32MB recommended) is required.
- XDOS version for Pentium PC's running DOS/Win 3.1. Minimum 16MB (32MB recommended) of memory, a mouse and color VGA display are required. VESA Super VGA and 8514A compatible graphics adapters are also supported.
- Silicon Graphics version for machines running IRIX (UNIX) 5.3 or later.

1.2 Installation

To install EXAMINE^{3D}, insert the CD/diskette into the appropriate drive and run

```
setup.exe
```

The installation program is friendly and easy to follow. When it is completed, the following files and subdirectories will be contained in your installation directory:

C3.EXE	COMPUTE$^{3D\text{-}BEM}$, a linear elastic isotropic homogeneous boundary element program for calculating stresses around underground excavations.

E3.EXE	EXAMINE^{3D}, a data visualization and modeling program for underground excavations in rock.
E3.CFG	The EXAMINE^{3D} Configuration File. Settings in this file may be modified using an ASCII (DOS text) editor.
EXAMPLES	A directory containing a variety of example and tutorial files.
UTILITIES	A directory containing the source code, executables and documentation for a variety of utilities. Of primary interest is the dxf to geo converter "DXFGEO" which will convert Autocad™ 3D .DXF files to a form that EXAMINE^{3D} can read.
TEXTURES	A directory containing background textures for rendered images. The user may place their own GIF file in this directory for use as a background image.

Also contained within this directory are various support files and directories. The user should not remove these files or directories as they are important for the proper execution of EXAMINE^{3D}. The user should not try to execute any of these files, as the result might be quite unpredictable.

1.3 Starting EXAMINE^{3D}

Select the EXAMINE^{3D} icon in the Start→Rocscience→Examine3D menu.

1.4 EXAMINE^{3D} Screens

The first screen presented by EXAMINE^{3D} is the *welcome screen*, shown in Figure 1.1. Three menu items, **Modeler, Interpret** and **Exit** are available from this screen. The **Modeler** prepares or reviews a model; **Interpret** leads to the group of functions for the examination of analysis results; and **Exit** activates the *closing down* dialogue for exiting from EXAMINE^{3D}.

1.4.1 The Modeler Screen

The screen shown in Figure 1.2 is presented when **Modeler** is selected from the welcome screen. The modeler screen consists of

1. four view windows, each giving a different view of the model;

2. a menu bar at the base of the two bottom windows; and

3. an information bar at the bottom of the screen.

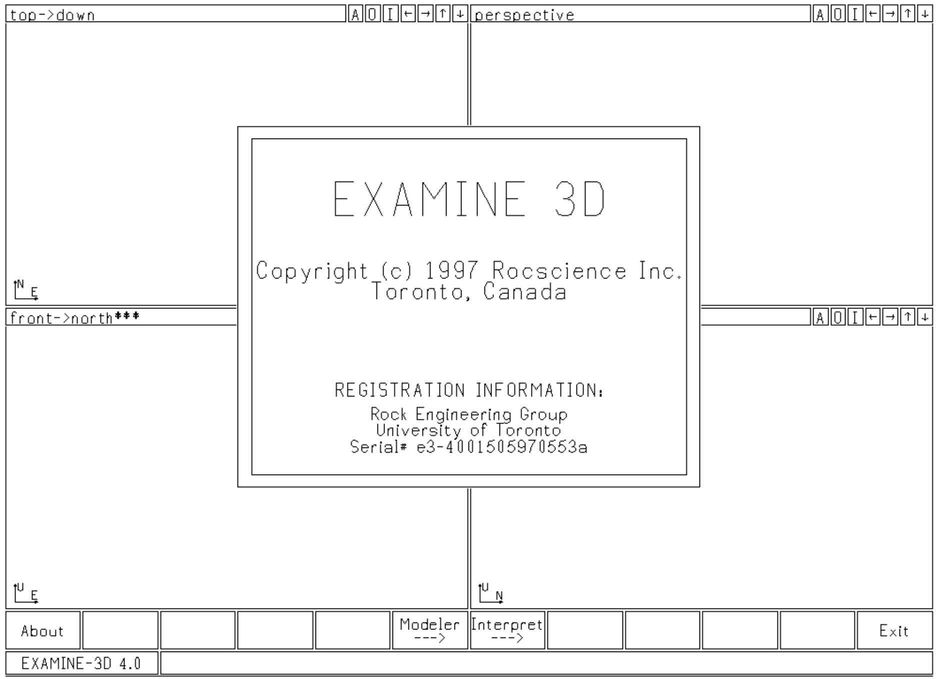


Figure 1.1: EXAMINE^{3D} WELCOME **screen**

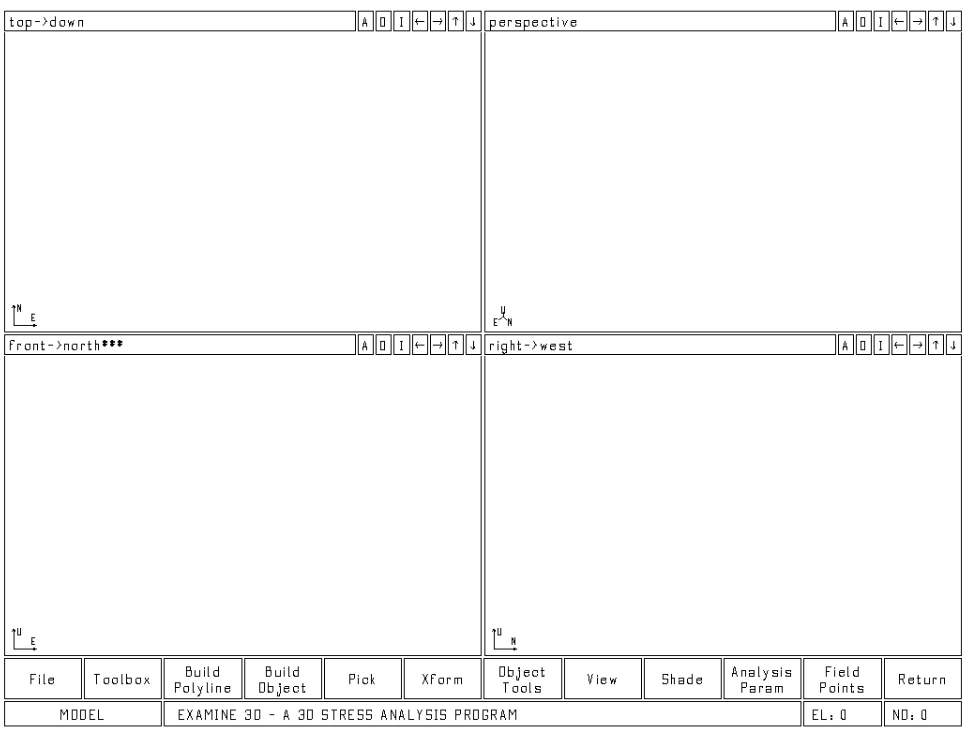


Figure 1.2: EXAMINE^{3D} MODELER **screen**

1.4.1.1 The Information Bar

The information bar is divided into four boxes:

1. A short box at the extreme left, containing the word MODEL, which identifies the current screen as the modeler screen.

2. Two short boxes at the extreme right containing EL: 0 and ND: 0. The element counter EL counts the number of boundary elements in the model; the node counter ND counts the number of boundary element nodes in the model. Both counters are updated as elements and/or nodes are created. They are both set to zero in Fig. 1.2 because no elements or nodes had been created when the screen was printed.

3. The long rectangular box in the middle of the information bar is the *dialogue box*. Messages directed at the user are printed in the dialogue box; user's responses entered from the keyboard are echoed in the box.

1.4.1.2 The Menu Bar

EXAMINE^{3D} is a menu-driven program; in general, all the functions required to execute a given task are available through the menu bar. The modeler screen menu bar presents the *Main menu* items through which any of the **Modeler** group of functions can be activated. A *Pop-up menu* is obtained by clicking on a main menu item. To select a pop-up menu item, first click and hold the left mouse button with the mouse pointer on the main menu item; drag the mouse pointer to the pop-up menu item; then release the mouse button. For example, to select the **autoscale** function from the **View** menu, click and hold the left mouse button, with the mouse pointer on **View**; drag the mouse pointer to **autoscale** on the pop-up menu; then release the button. In this manual, the process just described is indicated by the directive "Select **View + autoscale**".

Some pop-up menu items open up *sub-menus* when they are selected. For example, the SELECT OPTIONS sub-menu is opened by selecting **Toolbox + setup options**. All the main menu, pop-up menu, and sub-menu items are described in detail in this manual.

1.4.1.3 View Windows

Four views of a rectangular box are shown in Fig 1.3, to help describe the four view windows. The upper left window gives a *top* (looking down) view of the model. Only a projection of the model onto an E-N plane (E for East, N for North)[1] is visible through this window. The upper right window is the *perspective* view window. It gives a three-dimensional view of the model. The lower left window is the front (looking north) view window, which shows a projection of the model on an E-U (U for Up) plane. The lower right window is the *right* (looking west) view window. It shows a projection of the model on an N-U plane.

The coordinate axes directions are identified with arrows at the lower left corner of each window. Also, the view offered through each window is identified at its upper left corner. The seven small boxes at the upper right corner of each window enable the window to be resized or moved, relative to the model. Clicking on $\boxed{\text{A}}$ causes the window to be resized and relocated to just fit and center the model. The $\boxed{\text{A}}$ button is similar to the **autoscale** function, except that the former acts on one window, whereas the latter acts on the entire display. The zoom out button $\boxed{\text{O}}$ enlarges the window relative to the model, thereby bringing more of the model into view. The zoom in button $\boxed{\text{I}}$ focuses the window on decreasing portions of the model, thereby showing more details. Each of the arrow buttons moves the window, relative to the model, in a direction opposite to the arrow (which has the appearance of moving the model in the direction of the arrow).

[1] Coordinate System is discussed in a subsequent section.

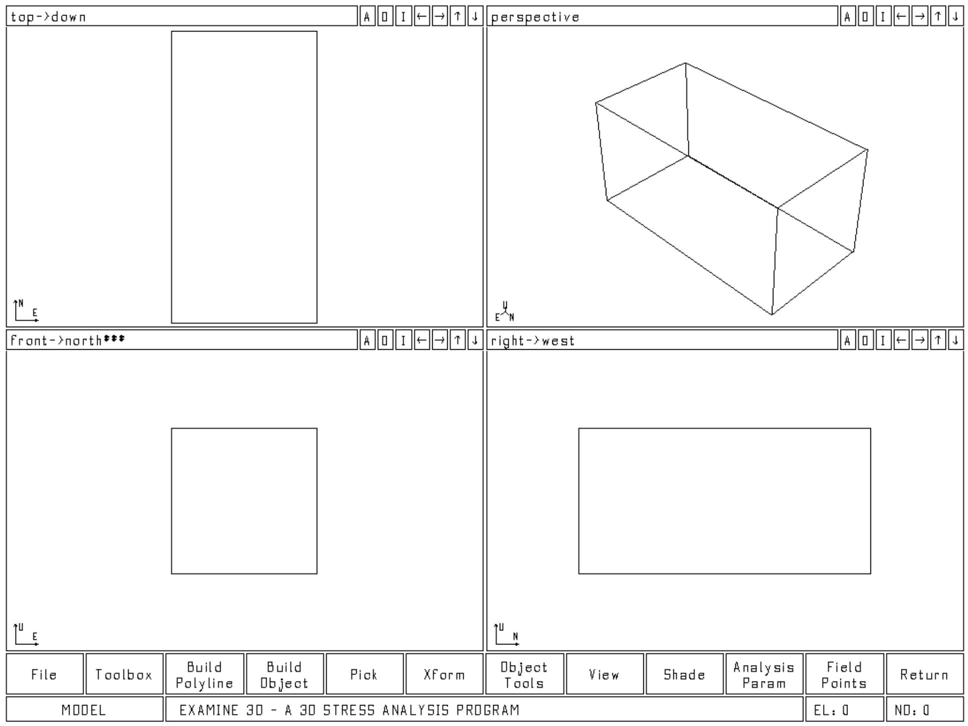


Figure 1.3: EXAMINE^{3D} MODELER **screen, showing different views of a rectangular box**

The greenish bar at the top of each window is used to toggle between the *multi-view* display (in which all four windows are visible) and the *maximized single window* display (in which only one window is visible, and enlarged to fill the screen).

Each of the three orthogonal views (right, front and top views) represents a plane; the location of a point relative to the coordinate axis normal to a plane cannot be changed by moving on the plane. For example, the U-coordinate of a point cannot be changed through the top view (E-N coordinate) window; similarly, only the E-U coordinates can be changed through the front view window; and only the N-U coordinates can be changed through the right view window.

1.4.2 The Stress Interpreter Screen

The **Stress Interpreter** DATA SELECTION screen is shown in Figure 1.4. This screen is obtained by selecting **Interpret** from the WELCOME screen (Figure 1.1), and then **Stress** (from the DATASETS screen) and loading the EXAMINE^{3D} file containing the data to be interpreted. The screen is the same as the MODELER screen (Figure 1.3), except for the following differences:

1. The information bar (at the bottom of the screen) does not show the EL and ND counters.

2. The menu bar shows the available results variables which can be viewed, and five other menu items.

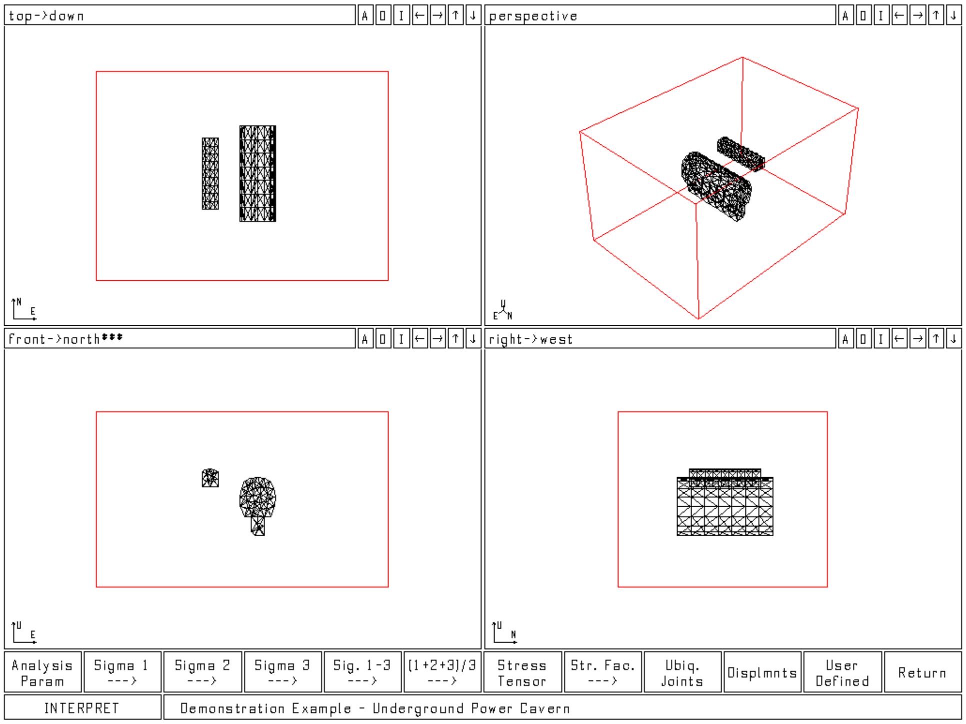


Figure 1.4: EXAMINE^{3D} DATA SELECTION **screen**

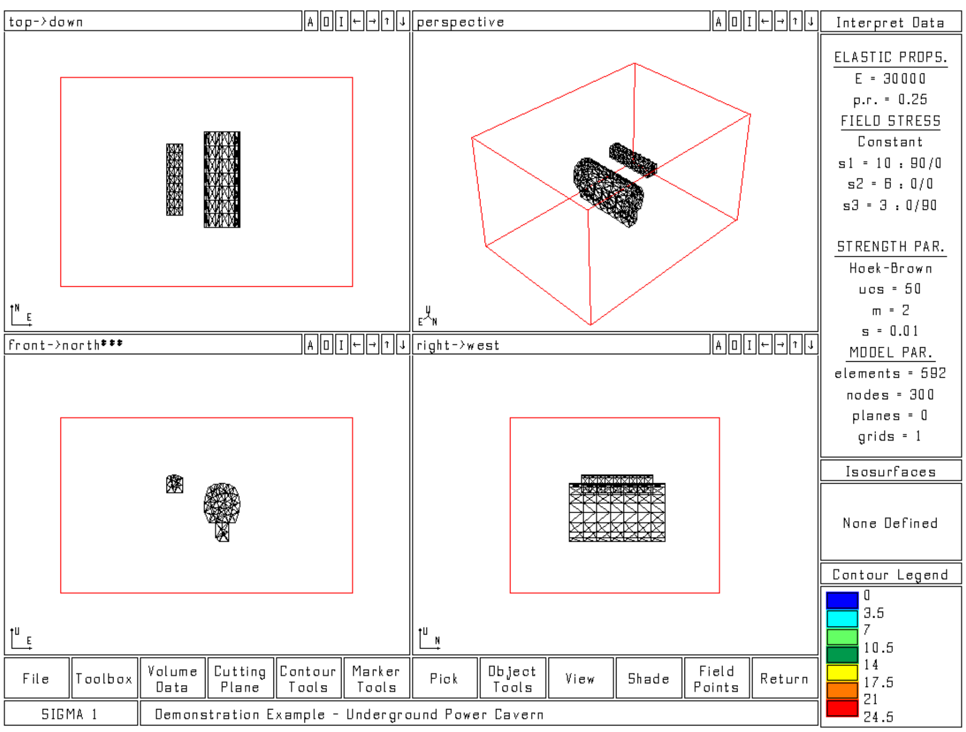


Figure 1.5: EXAMINE^{3D} DATA INTERPRETATION **screen**

The next screen (the **Interpret** DATA INTERPRETATION screen) is obtained by clicking on one of the results variables (in the menu bar). The screen obtained by clicking on **Sigma 1** is shown in Figure 1.5. Notice that the left end of the information box now shows the name of the *current results variable* (in this case **Sigma 1**). The name displayed in this box may be modified by the user through the Interpret function **Toolbox + setup options**, described in Section 3.7. An information column is now also given, at the right end of the screen. The top part of the column summarizes the elastic properties, field stress, strength parameters, and model parameters. This is followed by the Isosurface Legend and then by the Contour Legend.

All other properties of the screen are the same as was described for the modeler screen.

1.5 Coordinate System

EXAMINE^{3D} can use either the Cartesian or Compass coordinate systems. The Compass system is the default. It defines a point in space with respect to its distance in the North (N), Up (U), and East (E) directions, relative to an origin. Hence, based on this system, the location of a point is defined in terms of its N-U-E coordinates. The Cartesian system is equivalent to the N-U-E system, except that the names X, Y, and Z replace N, U, and E, respectively. The N-U-E system is used in this manual; the names X, Y, and Z could be substituted for N, U, and E, respectively, without requiring any other change.

Data files written in orthogonal coordinate systems different from the EXAMINE^{3D} system (N-U-E or X-Y-Z) must be translated as they are read into EXAMINE^{3D}. The program provides a painless procedure for doing this, using the **coord transform** function, available through the **File** menu.

1.5.1 Entering Coordinates

The coordinates of a point, when required, can be given either by (1) typing in the N-U-E values, or (2) by clicking the mouse button when the mouse pointer is at the required point. When entering coordinates with the mouse, recall that only two coordinates can be modified through any one of the orthogonal view windows; all three coordinates can be modified through any two of the three windows. It is likely that most users will prefer entering coordinates with the mouse. Therefore, EXAMINE^{3D} provides three functions for constraining the category of points selectable with the mouse. The three functions are *grid snap*, *vertex snap* and *orthogonal snap*. They are available only when they are needed, during which time the current status of each (*on* or *off*) is shown in the dialogue box.

1.5.1.1 Grid Snap

The grid snap function, when it is available, can be toggled on or off by pressing `s`. The current status of the function is shown in the dialogue box as `s=on` or `s=off`. When it is on, grid snap returns (as its value) the coordinates of the grid intersection closest to the mouse pointer when the mouse is clicked. Therefore, only grid intersections can be picked with the mouse when grid snap is on. The grid density (defined using the **Grid Spacing** function) can be used to change the number of points that are selectable with the mouse under grid snap.

1.5.1.2 Vertex Snap

When the vertex snap function is available, its status is shown as `vn=on` or `vn=off` in the dialogue box. It can then be toggled on or off by pressing `v` . Vertex snap, when it is on, returns (as its value) the coordinates of the *vertex* or *node* closest to the mouse pointer when the mouse is clicked.

1.5.1.3 Orthogonal Snap

The status of the orthogonal snap function is shown as `o=on` or `o=off` in the dialogue box, when the function is available. It can be toggled on or off by pressing `o` . When it is on, lines drawn on the screen are restricted to horizontal and vertical: clicking the *left mouse button* causes a *horizontal line* to be drawn from the most recently marked point to the current *vertical* coordinate of the mouse pointer; clicking *the right mouse button* causes a *vertical line* to be drawn from the most recently marked point to the current *horizontal* coordinate of the mouse pointer. Recall that the view window in which the mouse pointer is located determines which coordinate axes are horizontal and vertical.

1.6 Hot Keys

There are six 'hot' keys in EXAMINE^{3D}:

1. the Ctrl-X key causes immediate exit from EXAMINE^{3D}; nothing is saved and no warning or query is issued;
2. the Ctrl-I key invokes the image capture option, which enables the user to store the contents of the current screen as a GIF, PCX, BMP, PPM, or TGA image file;
3. the Ctrl-P key invokes the print screen function, which enables the user to print the contents of the current screen;
4. the ESC key terminates the current option, returning control to the next higher option from which the current one was selected; it may be necessary to press ESC a few times in order to exit from a menu item, depending on the number of levels of options and sub-options from which exit is being requested;
5. the F3 keyboard function key enables the user to view the amount of available memory left.
6. the F6 keyboard function key enables the user to toggle cursor tracking (see 2.2.1.3) on or off;
7. the F7 keyboard function key toggles grid lines on or off.

1.7 Capturing Image Files From EXAMINE^{3D}

The user may capture the screen to an image file for import into word processor documents, paint programs, or image editing tools. The formats supported are:

Paintbrush PCX
Compuserve GIF
Targa TGA
Windows BMP
Pbmplus PPM

Once the image you want to capture is on the screen, press Ctrl-I (the *control* and *i* keys simultaneously) or select **File + export image**. Choose the image file type and name in the popup menu. A dialog will then appear that will request certain information about the format of the image you wish to capture. After you have selected the format the program will capture the screen to the image file. The file can then be imported into your favorite word processor or image editor. This method for creating output is the preferred method as it will generate the best quality images.

1.8 Typographical Convention

The following convention is used in this manual:

The <enter> button is represented by the symbol ↵.

<u>Explanations and Instructions:</u>

Explanations and/or instructions are given in 10 point times roman font, as in this sentence.

<u>Dialogue Boxes</u>:

The program's requests are framed and written in 8 point courier font, as in:

```
Enter number of elements for highlighted segments [1]:
```

Names of variables, for which the program substitutes actual values at run time, are given in 8 point italic courier font, enclosed in square brackets, as in:

```
Enter length of extrusion [default]:
```

The user's responses are framed and written in courier font (uppercase or lowercase), and often preceded by the word "Press" or "Enter", as in

Press $\boxed{\texttt{S}}$ Uppercase letters are used here for the purpose of emphasis only. EXAMINE^{3D} does not distinguish between lowercase and uppercase letters when accepting responses from the keyboard. For example, pressing either s or S will produce the same effect.

Sometimes, for 'yes' or 'no' responses, the frame is omitted, ie.,

Enter Y↵ is equivalent to Enter $\boxed{\texttt{Y}}$ ↵

Note that the enter symbol ↵ is explicitly included wherever the <enter> (or <return>) button is required, as shown above.

The names of variables, for which actual values have to be substituted by the user, are given in italic enclosed in square brackets, as in:

Enter [*coordinates*]

or sometimes (in the tutorial chapters) the values are given explicitly, as in

`0 0 -15`

Menu Items

Menu items are written in bold times roman font, often preceded by the word “Select” in normal type, as in:

Select **Modeler**

The word “Select”, in this use, means *Move the mouse pointer to the menu item and click the left mouse button*. When two menu items are joined with the addition symbol ‘**+**’, then the second menu item should be selected from the *Pop-up* menu obtained by selecting the *main* (first) menu item. The left mouse button should be held down when the mouse pointer is being moved from the main menu item to the pop-up menu item. For example,

Select **View + autoscale**

means *Move the mouse pointer to* **View**, *click and hold the left mouse button, then drag the mouse pointer to* **autoscale** *and release the button*.

2. The Modeler

The EXAMINE^{3D} modeler is activated by selecting **Modeler** from the welcome screen (Figure 1.1). It opens access to a group of functions for generating three dimensional geometry and boundary element discretization for underground structures. The model generated can be visualized from all conceivable angles, either in *wire frame* or *shaded* mode. Furthermore, the boundary element discretization data can be written directly to a file for the boundary element stress analysis program COMPUTE$^{3D\text{-}BEM}$.

All the functions required for any of these tasks are accessed though the **Modeler** menu bar (Fig 1.2). Each of the menu items will now be described.

2.1 Modeler Menu Item: File

print
export image
coord transform
append to model
save file
open file

The **File** menu item opens access to a group of six file management functions; the pop-up menu is illustrated at left.

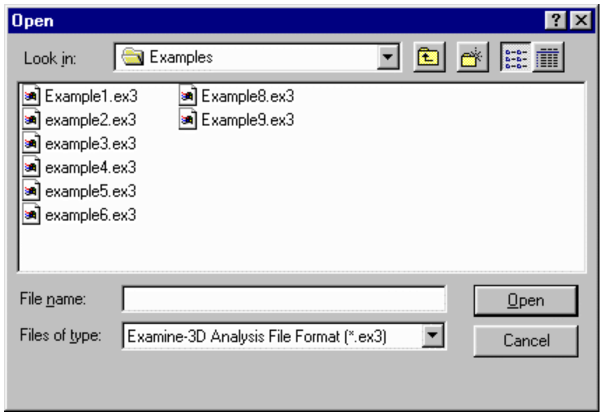
File	Toolbox	Build Polyline	Build Object	Pick	Xform	Object Tools	View	Shade	Analysis Param	Field Points	Return

2.1.1 File + open file

This function reads EXAMINE^{3D} files, stored under the name *filename*.EX3, where *filename* is user-assigned and the extension .EX3 is automatically appended by the program.

Such files contain the input data for COMPUTE$^{3D\text{-}BEM}$, which consists of the boundary element data, material parameters, and the field points at which analysis results are required.

The following dialog is displayed when **File + open file** is selected:



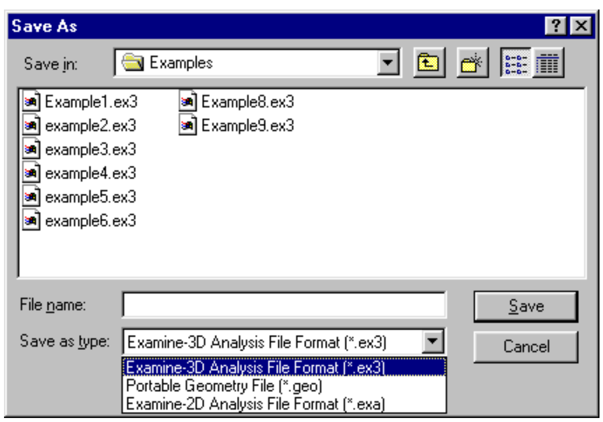
Use the dialog to navigate to the folder containing the EXAMINE^{3D} data file. Then select it or type in the name. The following message is displayed thereafter:

```
Reading the [filename] data file, Please Wait . . .
```

When the data is fully read in, the dialogue box clears to indicate that EXAMINE^{3D} is ready for further instructions.

2.1.2 File + save file

The following dialog is displayed when **File + save file** is selected:



As shown in the figure you may store a file in one of three file formats. If you select the EXAMINE^{3D} file format, a file capable of being analyzed by the boundary element method is written. If you select a portable geometry file, an ascii file containing just the geometry is written. The purpose of this file is to allow the transfer of geometry from/to other programs. The format of the file is described in Section 5.6. You may also write an EXAMINE^{2D} file. EXAMINE^{2D} is a two-dimensional boundary element package.

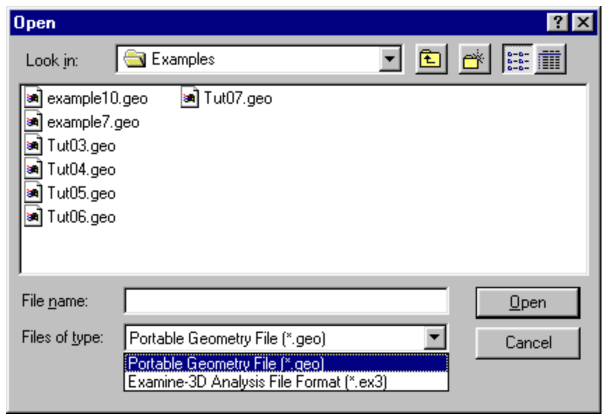
Use the dialog to navigate to the folder to store the data file. Then select it or type in the name.

2.1.3 File + append to model

The **append to model** function reads data from a .EX3 or .GEO file and appends it to the current data base.

Complex structures are often built up in parts, each of which is stored in a separate .EX3 or .GEO file. The **append to model** function is used to put such parts together.

The following dialog is displayed when **File + append to model** is selected:



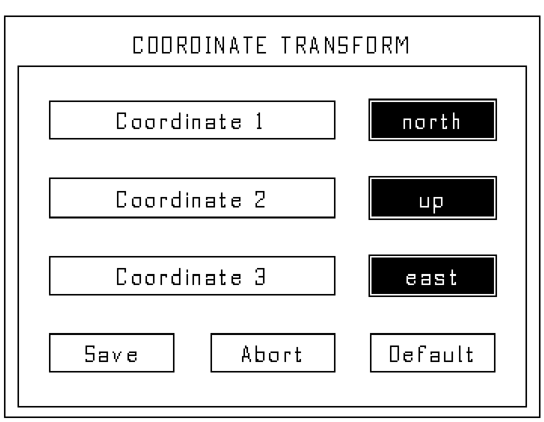
When the data is fully read, it is properly merged with the existing data, and the result is displayed. Material parameter assignments in the current database supersede those in the appended file; secondly, any common nodes and elements are filtered out.

2.1.4 File + coord transform

The function **coord transform** is used to specify the orthogonal coordinate system for an external file.

EXAMINE^{3D} sets up a coordinate transformation for the specified system, so that data written to, or read from, the external file will be properly interpreted. The **coord transform** function must be executed before reading from, or writing to, the external file.

To execute this function, select **File + coord transform**:



The COORDINATE TRANSFORM sub-menu, shown in the sketch on the left, is displayed. It is assumed that the coordinates of a point are given in the external file in terms of three values: Coordinate 1, Coordinate 2, and Coordinate 3. Each coordinate is associated with one of three orthogonal directions. The coordinate system is specified by identifying the appropriate directions in the sub-menu.

Click repeatedly on **Coordinate 1**, until the name displayed in the rectangular box on its right correctly describes the direction of Coordinate 1 in the external file.

Do the same for **Coordinate 2**, to set its direction.

Set the direction of **Coordinate 3** the same way.

Then click on **Save** to effect the settings and exit from the **coord transform** function.

While in the sub-menu, clicking on **Abort** aborts the function; and clicking on **Default** resets the coordinates to North-Up-East.

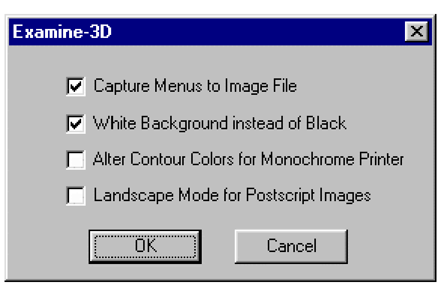
2.1.5 File + export image

Use this function to capture the current EXAMINE^{3D} window to an image file. The following dialog is displayed when **File + export image** is selected:

Use the dialog to navigate to the folder to store the image file. Then select it or type in the name. EXAMINE^{3D} supports the following image file formats:

Paintbrush PCX
Compuserve GIF
Targa TGA
Windows BMP
Pbmplus PPM

Select Save and the following dialog is displayed:



If you choose to capture the menus to the image file, then all frames, menus and buttons are captured to the image file. Otherwise, just the contents of the view windows are placed in the image file. Reversing black and white ensures that the background is white instead of black. For eventual hardcopy output on paper this is most likely what you want. Capturing the image for a black and white printer causes the program to alter some colors so that the image when printed on a black and white printer looks the best.

Make your selections and press OK. The program then proceeds in creating the image file.

2.1.6 File + print

This function is used to send the complete contents of the EXAMINE^{3D} window to a printer. The printer must be first defined as a system resource by the operating system. In Windows 95/NT this is done through the Printers option in the Control Panel. When you select this option the standard Windows dialog appears asking you to select a printer. After you select the printer and press OK, the program will proceed in making a hardcopy of the current EXAMINE^{3D} window.

2.2 Modeler Menu Item: Toolbox

volume & area
geometry slicer
setup options
store status
retrieve status
object check

The **Toolbox** menu item opens access to a group of six miscellaneous functions, as shown in the pop-up menu at left.

File	Toolbox	Build Polyline	Build Object	Pick	Xform	Object Tools	View	Shade	Analysis Param	Field Points	Return

2.2.1 Toolbox + volume & area

This function is used to calculate the volume and surface area of the current three-dimensional model. For models which are not closed, the volume calculated is invalid but the surface area is still correct.

Select **Toolbox + volume&area**, and the following message will be displayed:

```
Volume = [num] m3, Surface Area = [num] m2
```

2.2.2 Toolbox + geometry slicer

This option is used to generate the intersection polyline between the current three-dimensional model and a user supplied cutting plane. The polyline may then be exported through the **store geo** option. Users may then use the DXFGEO utility to convert the GEO file to a DXF file for use in Autocad™ (see Chapter 5).

EXAMINE^{3D} provides six methods for defining cutting planes; when **Toolbox + geometry slicer** is selected, the user is prompted to select one of the methods, as follows:

```
Enter Input Method (1=plane,2=3pt plane,3=North,4=Up,5=East,6=FP plane) [default]:
```

2.2.2.1 Input Option 1: Plane

This option allows the user to define the cutting plane by giving the coordinates of two diagonally opposite corners. To select this option, enter 1 ↵ in response to the above message. The following is displayed, along with the Go buttons:

```
Input plane point #1 [N,U,E: snaps]:
```

Either: type the coordinates of the first corner, followed by ⏎;

Or: click on any point to place the first corner there.

The selected point is marked with a red star; its location can be modified, either by entering alternative coordinates through the keyboard, or by clicking with the mouse. If using the mouse, notice that the coordinate snap functions are available; as was explained in Section 1.5.1, they constrain the category of points selectable with the mouse. Furthermore, recall that only two coordinates can be modified through any one (and all three coordinates through any two) of the orthogonal view windows.

When the selected point is satisfactory, click on [Go] to accept it. The dialogue box updates to request the second corner, as follows:

```
Input plane point #2 [N,U,E: snaps]:
```

Enter the coordinates of the second corner, either from the keyboard, or with the mouse. The rectangle formed by the two points (corners) is displayed; it may be adjusted by modifying the coordinates of the second point, following the procedure described for the first point.

When the rectangle is satisfactory, click on [Go] to accept it. The program will then create the polyline defining the intersection of the plane with the three-dimensional geometry.

2.2.2.2 Input Option 2: Plane (3pt)

This option allows the user to define the plane by giving the coordinates of any three different points. Enter [2] ⏎, and the following message will appear, along with the [Go] buttons:

```
Input plane point #1 [N,U,E: vn=off, s=off, o=off]:
```

Either: type the coordinates of the first corner, followed by ⏎;

Or: click on any point to place the first corner there.

The location of the point can be adjusted, either by entering alternative coordinates from the keyboard, or by clicking with the mouse (taking advantage of the snap functions and multi-view windows, as was explained above). When the selected point is satisfactory, click on [Go] to accept it. The dialogue box updates to request the second point, as follows:

```
Input plane point #2 [N,U,E: vn=off, s=off, o=off]:
```

Enter the coordinates of the second point, either from the keyboard, or with the mouse. A yellow line is displayed, joining the second point to the first one. The location of the second point can be adjusted, using the same procedure described above for the first one. When the point is satisfactory, click on [Go] to accept it. The dialogue box updates to request the third point, as follows:

```
Input plane point #3 [N,U,E: vn=off, s=off, o=off]:
```

Enter the coordinates of the third point. The rectangle defined by the three points is displayed. Its size and orientation can be adjusted by relocating the third point. When the rectangle is satisfactory, click on [Go] to accept it. The program will then create the polyline defining the intersection of the plane with the three-dimensional geometry.

2.2.2.3 Input Option 3: North

This option provides for the plane to be entered as a vertical plane oriented perpendicular to the North axis. Enter 3 ↵; the following message appears:

```
Enter Northing:
```

Enter the location of the plane in the northing direction. The program will then create the polyline defining the intersection of the plane with the three-dimensional geometry.

2.2.2.4 Input Option 4: Up

This option provides for the plane to be entered as a horizontal plane oriented perpendicular to the Up axis. Enter 4 ↵; the following message appears:

```
Enter Elevation:
```

Enter the elevation of the plane in the up direction. The program will then create the polyline defining the intersection of the plane with the three-dimensional geometry.

2.2.2.5 Input Option 5: East

This option provides for the plane to be entered as a vertical plane oriented perpendicular to the East axis. Enter 5 ↵; the following message appears:

```
Enter Easting:
```

Enter the location of the plane in the east direction. The program will then create the polyline defining the intersection of the plane with the three-dimensional geometry.

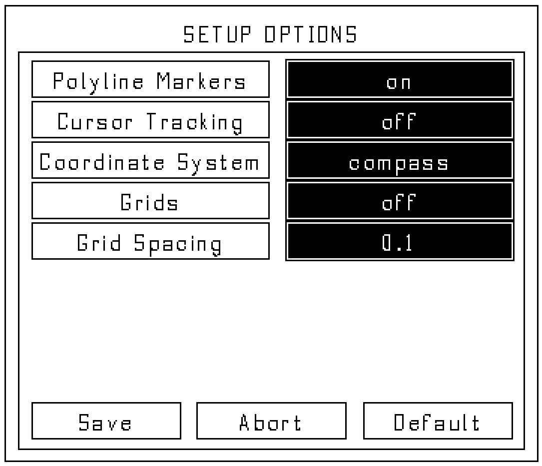
2.2.2.6 Input Option 6: Currently defined Field Point Plane

This option provides for the plane to be entered using an already existing field point plane (Section 2.11.1). Enter 6 ↵; the following message will appear if more than one field point plane exists, otherwise the one and only field point plane is automatically selected:

```
Select cutting plane [Esc=abort]:
```

If prompted, select a field point plane. The program will then create the polyline defining the intersection of the plane with the three-dimensional geometry.

2.2.3 Toolbox + setup options



The **setup options** function opens the SETUP OPTIONS sub-menu, shown in the sketch on the left. The default setting for each of the functions is shown. Any changes made in the sub-menu are effected by selecting **Save**, which also causes an exit. Selecting **Abort** causes exit without effecting any changes. The default setting is restored by clicking on **Default**.

2.2.3.1 Polyline Markers

Polyline markers are the blue stars, and greenish yellow disc and circle, which mark the polyline vertices. Click on **Polyline Markers** to toggle them on or off.

2.2.3.2 Cursor Tracking

This function continuously displays two coordinates of the mouse pointer location, as the pointer moves in any of the orthogonal view windows. The coordinates are displayed in the greenish bar at the top of the particular window. Click on **Cursor Tracking** to toggle the function on or off.

The F6 keyboard function key can also be used at any time to toggle cursor tracking on or off.

2.2.3.3 Coordinate System

Toggle the coordinate system between *compass* and *Cartesian*, by clicking on **Coordinate System**.

2.2.3.4 Grids

Grid lines are toggled on or off by clicking on **Grids**.

The F7 keyboard function key can also be used at any time to toggle grid lines on or off.

2.2.3.5 Grid Spacing

To set the grid spacing, click on **Grid Spacing**, and type the required value, followed by ↵.

2.2.4 Toolbox + store status

This function is used to store the eye position relative to a model, so that the model can be examined from the same viewpoint when so-desired. Along with the current view, **setup options** (see above) are also stored.

Start by selecting **Toolbox + store status**, which causes the following message to be displayed:

```
Storing STATUS file, enter filename:
```

Enter *filename* ↵, to cause the viewpoint status to be stored in a file named *filename*.STA.

2.2.5 Toolbox + retrieve status

This function is used to read data from a .STA file. Such a file stores the **eye+target** information required to examine a model from a specific viewpoint. See also **View + eye+target**

Start by selecting **Toolbox + retrieve status**, which gives the following message:

```
Retrieving STATUS file, enter filename:
```

Either: enter *filename* ↵, to specify *filename*.STA as the input file, where *filename* includes full DOS path

Or: click the left mouse button to obtain a list of the .STA files in the working directory. Then click on the required *filename* to select it.

After the file is read, the existing model (or any one read in subsequently) is rotated to present the viewpoint specified in the .STA file. The viewpoint remains the same until it is changed using any of the **View** options (Section 2.8) or **Shade + animate**, or until the current EXAMINE^{3D} session is terminated.

2.2.6 Toolbox + object check

The **object check** function is used to check boundary element discretizations, to ensure that basic rules have not been violated. It is advisable to run **object check** on a mesh before submitting it to COMPUTE$^{3D\text{-}BEM}$ for stress analysis.

To start, select **Toolbox + object check**. The function presents a series of options, prompting the user each time for permission to proceed:

```
Check object, element, and node numbering (y):
```

This option checks the numbering of nodes, elements and objects. It should be done, unless it has in a previous run (and **object check** is being run again to check something else).

Press ↵ to accept it, or N↵ to reject.

```
Check for zero area elements (y):
```

Zero area elements may occur in the unusual case of two nodes being assigned the same (or very nearly the same) coordinates. The option should be run, unless it has been run previously.

Press ↵ to accept, or N↵ to reject.

```
Check triangular element base/height ratios (y):
```

This option checks the aspect ratio of elements. It should be run, unless it has been run previously, or the mesh is so simple that the shape of all elements can be examined visually. Values of base/height ratio between 0.1 and 10 are considered acceptable. Values outside this range may give bad analysis results.

Press ↵ to accept, or N↵ to reject. If it is run, the result is displayed as follows:

```
Maximum ratio = [rmax]; element [Nel] - Press Enter to Continue
```

and the element number [*Nel*], at which the maximum ratio [*rmax*] is obtained, is highlighted red.

Press ↵ to continue,

```
Check for invalid overlapping elements (y):
```

If two objects (see Section 2.5.4) generated separately are subsequently joined (face-to-face), it is possible that elements on the adjoining faces overlap with the nodes of some elements lying on the face of others. Such element overlap would cause numerical difficulties. This option of **object check** looks for such situations.

Press ↵ to accept it. If any such overlap is detected, the following message is displayed:

```
[Num] Invalid Overlapping Elements Found/Picked -- Press Enter
```

The offending elements are highlighted and the view windows are **autoscaled** to the elements. To correct the problem, delete all elements on the adjoining faces; the two objects may then be joined using the **open edge polyline** (Section 2.3.6) and **face** (Section 2.4.3) functions.

To proceed with **object check**, press ↵; the function then proceeds with the intersection check as follows (it also proceeds with the intersection check if no invalid overlapping elements are detected):

```
Check for invalid intersecting elements (y):
```

If two objects (see Section 2.5.4) generated separately intersect, such element intersection would cause numerical difficulties. This option of **object check** looks for such situations.

Press ↵ to accept it. If any such intersection is detected, the following message is displayed:

```
Invalid Intersecting Elements - Press Enter to Continue…
```

The offending elements are automatically picked. Consider aborting the **object check** and using the **View** (Section 2.8) functions to identify the problem. To correct the problem you must redefine your geometry in such a way as to remove the intersecting elements.

If you wish to proceed with **object check**, press ↵; the function proceeds with the leaky check as follows (it also proceeds with the leaky check if no invalid intersecting elements are detected):

```
Check for leaky objects (y):
```

In a properly constituted boundary element mesh, each of the structures modeled is entirely “wrapped” with boundary elements (except for free surfaces). A structure is considered NOT LEAKY if it is so wrapped; otherwise, it is LEAKY, in which case all the holes must be “sealed” with boundary elements before the mesh is submitted to COMPUTE$^{\text{3D-BEM}}$.

This option should be run, unless it has been done previously.

Press ↵ to accept or N↵ to reject. If it is run, the following message will be displayed at the end:

```
Geometry is NOT LEAKY
```

if that is the finding; otherwise, the display would be:

```
Geometry is LEAKY and invalid for analysis - Press Enter
```

The first message indicates an acceptable mesh. For the second case, the boundaries of the holes detected are highlighted red; if the highlighted boundaries surround a free surface, then the LEAKY message should be ignored; otherwise, the mesh should be modified.

It is not usually necessary to run **object check** on a free surface mesh. If it is done, the normals on the free surface should be re-checked visually (see **Shade + shade options**) to ensure that they point into the rock.

The second message `Geometry is LEAKY and invalid for analysis - Press Enter` is also given if superimposed faces are detected, superimposed Mesh generation for a face, using the face function, may inadvertently be performed more than once, in which case the face elements would be superimposed on each other. When this is detected by the leaky check, the above message is displayed, and the boundaries of the face are highlighted red.

Respond with ↵; then the following is displayed:

```
Check for and delete overlying faces near red line -- Press Enter
```

Press ↵; then use **Pick + component** and **Object Tools + delete picked** to select and delete all but one of the superimposed faces. Thereafter, repeat the leaky check.

If the geometry is NOT LEAKY, the **object check** function terminates; otherwise, it proceeds with the clockwise elements check, as follows:

```
Check all objects for clockwise elements (y):
```

Boundary element nodes should be numbered counter-clockwise, for their normals to point away from the opening (for example) being modeled. This option conducts that check; it should be run, unless it has been done previously.

Press ↵ to accept, or N↵ to reject. Its progress is reported in the dialogue box (when it is run).

2.3 Modeler Menu Item: Build Polyline

open edge polyline
nodeline-> polyline
polyline-> nodeline
continue polyline
edit polyline
new polyline

The **Build Polyline** menu item opens access to a group of six polyline construction and editing functions.
A polyline consists of a group of one or more progressively connected (i.e., end-to-end) line segments arbitrarily oriented in space. Polylines are the basic units used to define three dimensional shapes in EXAMINE^{3D}. For example, the cross-section of an underground opening would initially be represented as a polyline.

File	Toolbox	Build Polyline	Build Object	Pick	Xform	Object Tools	View	Shade	Analysis Param	Field Points	Return

Each polyline consists of line *segments* connected at *vertices*. The number of segments depends on the amount of detail admitted in the shape being represented. For example, three or more segments can be used to represent a circle; the larger the number of segments, the closer the shape of the polyline approaches that of a circle, and the larger the amount of information required to define the polyline.

2.3.1 Build Polyline + new polyline

The **new polyline** function is used to construct polylines, starting from scratch. The following message is displayed when **Build Polyline + new polyline** is selected:

```
Select Point [N,U,E; vn=off, s=off, o=off, c,u,i,e, [NP] ]:
```

At the same time, a button labeled [Go] appears at the lower right corner of each window. The user is required to define the polyline vertices (points in space) by supplying their N-U-E coordinates.

To specify the coordinates of a point:

Either: type the N-U-E values, separated with comma (,) or space, and press ↵;

Or: click on a desired point through any of the windows, to extract its coordinates.

When a point is entered, the counter [*NP*] (last parameter in the dialogue box message) updates to indicate the total number of points entered so far. It is also updated each time a vertex is erased.

The vertex snap, grid snap and orthogonal snap functions (see Section 1.5.1) are available; hence their status is shown in the dialogue box. Recall that pressing [V], [S] or [O] toggles vertex snap, grid snap or orthogonal snap, respectively, on or off.

The other parameters (`c,u,i,e`), activate different options of the function, as described below:

2.3.1.1 Exit Options

Three exit options are provided for terminating the **new polyline** function:

Either: press [c] to close the polyline (i.e., connect the most recent vertex to the first vertex) and exit; minimum of three vertices are required to form a closed polyline;

Or: click on [Go] to accept the polyline as is (closed or not) and exit;

Or: press ESC to abort the function; the polyline is discarded.

2.3.1.2 Deleting Vertices

Pressing [u] , the *undo* option, causes the most recent vertex to be erased. It can be pressed repeatedly, causing the most recent among the existing vertices to be erased each time.

2.3.1.3 Editing Vertices

The location of the *current vertex* can be modified by invoking the *edit* mode of the **new polyline** function. Only the current vertex can be edited using this approach; the **edit polyline** function is required to edit previous vertices. To relocate the current vertex, press [e] ; the following message is displayed:

```
Edit Current Point [N,U,E: vn=off, s=off, o=off]:
```

The location of the vertex can then be adjusted by clicking on the required new location in any of the windows. Recall that only two coordinates can be modified through any one (and all three coordinates through any two) of the orthogonal view windows: The E and U coordinates can be adjusted through the front view window, without changing the N coordinate; the N and U coordinates can be adjusted through the right view window, without changing the E coordinate; and the E and N coordinates can be adjusted through the top view window, without changing the U coordinate.

When the location of the point is satisfactory, click on [Go] in any window, to accept the point and exit from the edit mode. The dialogue box reverts to

```
Select Point [N,U,E; vn=off, s=off, o=off, c,u,i,e, [NP] ]:
```

2.3.1.4 Making a Circular Polyline

Circular polylines are constructed using the [i] option of the **new polyline** function. The user is prompted to give the radius and number of segments. To start, select **Build Polyline + new polyline**. The following is displayed:

```
Select Point [N,U,E; vn=off, s=off, o=off, c,u,i,e, [NP] ]:
```

Press [I]

```
Enter circle radius [default]:
```

Enter [*r*]⏎, or press ⏎ to accept the [*default*], where [*r*] is the desired radius.

```
Enter number of line segments/circle [default]:
```

Enter [*n*] ↵, or press ↵ to accept the [*default*], where [*n*] is the desired number of segments. The acceptable minimum is 3 (which would give a triangular polyline). Usually, a minimum of 8 would be required to obtain a circular polyline.

The resulting circle is centered at (0,0,0) and oriented normal to the E-coordinate axis. It can be rotated to the desired orientation, and moved to the desired location, using the **rotate** and **move** functions, respectively, available through the **Xform** menu.

2.3.1.5 Appearance of a Polyline

The default color of polylines is blue. The vertices are marked with blue stars; the first vertex is marked with a solid green circle, and the second with a hollow green circle, thus establishing the order of vertices. These markers can be toggled off or on using **Toolbox + setup options**. When polylines appear crowded, turning the markers off gives a cleaner picture.

2.3.2 Build Polyline + edit polyline

This function is used to relocate any or all of the vertices of a polyline, add new vertices, or delete existing ones. It is invoked by selecting **Build Polyline + edit polyline**, which causes the following message to be displayed:

```
pick a polyline vertex to move [d=delete, a=add, f=1st pt, Enter=done]:
```

2.3.2.1 Relocating a Vertex

To relocate a vertex, click on it in response to the above message. The selected vertex is highlighted with a yellow circle, and the following message is displayed:

```
New N,U,E Location (Enter=done):[N,U,E: vn=off, s=off, o=off]:
```

<u>Either</u>: type the coordinates of the new location, followed by ↵;

<u>Or</u>: click on a point to select it as the new location.

The highlighted vertex is relocated as soon as a new point is entered, and the above message is redisplayed. Modify the point as desired, using either of the above methods; the vertex moves to the new location each time.

The vertex, grid and orthogonal snap functions (Section 1.5.1) are available, and can be toggled on and off by pressing $\boxed{v}$, $\boxed{s}$, and $\boxed{o}$, respectively. If using the mouse, recall that the snap functions constrain the category of selectable points; also recall that only two coordinates can be modified through any one (and all three coordinates through any two) of the orthogonal view windows.

When the location of the highlighted vertex is satisfactory, press ↵ to accept it. Its marker reverts to a blue star, and the dialogue box message reverts to

```
pick a polyline vertex to move [d=delete, a=add, f=1st pt, Enter=done]:
```

2.3.2.2 Deleting a Vertex

In order to delete a vertex, press `d` in response to the above message. The following message will be displayed:

```
pick a polyline vertex to delete [d=delete, a=add, f=1st pt, Enter=done]:
```

Delete any vertex by clicking on it. It is deleted immediately. Therefore, before clicking on a vertex be sure that you do want to delete it; the delete process is irreversible by itself. On the other hand, new vertices can be inserted (on existing segments) using the *add* option; furthermore, polylines can be extended (i.e., adding both vertices and segments) using the **continue polyline** function.

After a vertex is deleted, the dialogue box message reverts to

```
pick a polyline vertex to move [d=delete, a=add, f=1st pt, Enter=done]:
```

thereby exiting from the *delete* option. Exit from the delete option without deleting a vertex, can be effected by pressing ⏎.

2.3.2.3 Adding a Vertex

To add a vertex (to an existing segment), press `a` in response to the above message. The following message is displayed:

```
pick a polyline edge to insert a vertex on [d=delete, a=add, f=1st pt, Enter=done]:
```

A polyline edge is the line segment joining any two vertices. Click on the edge at which a new vertex is desired. A vertex is inserted at its midpoint, and the dialogue box message reverts to:

```
pick a polyline vertex to move [d=delete, a=add, f=1st pt, Enter=done]:
```

The new vertex can be relocated as was described earlier. Exit from the add option without adding a vertex, can be effected by pressing ⏎.

2.3.2.4 Re-Assigning the First Vertex

The first vertex is the one marked with a solid green circle. It can be relocated, like any other vertex, using the procedure described in Section 2.3.2.1. The status of "first point" can also be re-assigned to another vertex. To do this, press `f` in response to the above message. The following message is displayed:

```
pick a first polyline vertex [d=delete, a=add, f=1st pt, Enter=done]:
```

Click on the vertex that you wish to assign "first point" status. The solid green circle mark is re-assigned to the selected vertex; the vertex next to it (following the existing order) gets the "second vertex" mark (hollow green circle); the two vertices previously marked as first and second revert to blue stars; if the polyline was previously unclosed, it is closed; and the dialogue box message reverts to:

```
pick a polyline vertex to move [d=delete, a=add, f=1st pt, Enter=done]:
```

thereby exiting from the *1st pt* option. To exit from the *1st pt* option without re-assigning the first point, press ⏎.

2.3.2.5 Terminating the edit polyline Function

When the function is in the *move vertex* mode, i.e., when the message

```
pick a polyline vertex to move [d=delete, a=add, f=1st pt, Enter=done]:
```

is displayed, pressing ↵ or ESC causes the function to terminate; changes made during the editing session are retained, irrespective of how the exit is effected.

2.3.3 Build Polyline + continue polyline

The **continue polyline** function is used to extend an existing polyline (by adding new line segments).

To start, select **Build Polyline + continue polyline**. The most recent polyline is highlighted, and the following message is displayed:

```
Select Point [N,U,E: snaps, c,u,i,e, [NP] ]:
```

From this point, the **continue polyline** function works exactly like the **new polyline** function (Section 2.3.1). Any new vertex added is connected to the most recent one; the c, u, i, and e options work as was described for the **new polyline** function; the snap functions are available, and the total number of vertices in the polyline is given as [*NP*] in the dialogue box.

Press Go to accept the polyline and exit from the function. Pressing ESC causes the entire polyline to be deleted! So, do not exit with ESC unless you want to delete the polyline (see also *Exit Options* in Section 2.3.1.1.)

2.3.4 Build Polyline + polyline->nodeline

The **polyline->nodeline** function is used to convert a polyline to a nodeline. The polyline is left in place, superimposed by the nodeline, after the conversion.

Whereas a polyline defines the geometry of a section, a nodeline is required to define the boundary element discretization. The number of segments on a given polyline depends on the complexity of the shape it represents. On the other hand, the number of segments on the corresponding nodeline is influenced by other factors (such as the expected stress gradient), in addition to the section geometry.

When a polyline is converted to a nodeline, the number of segments on the nodeline is at least equal to the number on the polyline.

To start, select **Build Polyline + polyline->nodeline**. The following message is displayed:

```
Pick Polyline to Convert:
```

Click on the polyline to be converted. It is highlighted briefly to indicate that it has been picked; it reverts to blue color; one of its segments is highlighted, and the following message is displayed:

```
Enter number of elements for highlighted segment [default]:
```

Each segment of a nodeline represents one edge of a boundary element. The above message requests the user to specify the number of boundary element edges to be formed from the highlighted polyline segment.

Press ↵ to accept the [*default*], if it is satisfactory; otherwise type the desired number followed by ↵ .

The same message will be displayed for each polyline segment. Respond accordingly. At the end, a light blue nodeline is formed, superimposed on the darker blue polyline.

2.3.5 Build Polyline + nodeline->polyline

This function is used to convert a nodeline to a polyline. The nodeline will be left in place, superimposed by the polyline. The polyline will have the same number of vertices as the nodeline.

To start, select **Build Polyline + nodeline->polyline**.

```
Pick FACELINE to Convert:
```

Click on the nodeline. The polyline forms, and the dialogue box clears to indicate that EXAMINE^{3D} is ready for the next command.

2.3.6 Build Polyline + open edge polyline

An open edge is a hole in a boundary element mesh. The **open edge polyline** function is used to place a polyline at the edge of such a hole. The hole can then be filled using the **face** function (after converting the polyline to a nodeline); or the mesh can be extended from the polyline, using either the **skin** function (if there are other polylines) or the **extrude** function. The functions **face**, **skin** and **extrude** can be accessed via the **Build Object** menu.

To start, select **Build Polyline + open edge polyline**; if there is a hole in the current mesh, the following message will be displayed, after a short wait:

```
 [Num] Polylines Will Be Created, Continue? (y):
```

where [*Num*] is the number of open edges detected in the mesh.

Press ↵ to proceed.

The open edge polyline(s) are formed. A polyline vertex is placed at each node on the open edge.

2.4 Modeler Menu Item: Build Object

transition skin
blend
face -->
extrude
skin

The **Build Object** menu opens access to a group of five functions, for generating the geometry and boundary element discretization of a structure. The pop-up menu is illustrated at left.

File	Toolbox	Build Polyline	Build Object	Pick	Xform	Object Tools	View	Shade	Analysis Param	Field Points	Return

2.4.1 Build Object + skin

The **skin** function *wraps a skin* of boundary elements around two or more polylines. To be acceptable for **skin**, the polylines must have the same number of vertices. Only consecutive polylines can be connected with **skin**.

To start, select **Build Object + skin**; the [Go] buttons appear, and the following message is displayed:

```
Pick a skin polyline [ [Num] picked, Select Go when done]:
```

A polyline is picked by clicking on it, at which it becomes highlighted (yellow). Clicking on a highlighted polyline causes it to be unpicked, at which its color reverts to blue. Once the first polyline is picked, subsequent ones can be picked only if they have the same number of vertices as the first. All polylines picked must be consecutive.

Pick as many as desired. The parameter [*Num*] is updated each time to indicate the number picked so far. Thereafter,

Click on [Go] to indicate that the required polylines have all been picked. The following message is displayed:

```
Use default discretization (y):
```

2.4.1.1 Default Mesh Generation Using skin

To accept the default discretization scheme, press ↵. The following is displayed:

```
Enter mesh density factor [default]:
```

The mesh density factor is used to control the mesh density in the default scheme. Its default value is 1.0. Values larger than 1 give a finer mesh (more elements); the mesh is coarser for smaller values of mesh density factor.

Press ↵ to accept the [*default*], or type the required value, followed by ↵.

Thereafter, temporary interpolation polylines are placed between the selected ones, illustrating the proposed discretization. The following is displayed:

```
Continue with Element Generation? (y):
```

Either: enter N↵; the interpolation polylines are removed, the original selected polylines are deselected, and the function **Build Object + skin** is terminated;

Or: press ↵ to accept the proposed discretization; the message `Creating Elements...` is displayed to indicate that element generation is in progress; thereafter, the EL and ND counters (lower left corner of screen) are updated, the elements generated are displayed, and the dialogue box clears to indicate that EXAMINE^{3D} is ready.

The default discretization is likely to be satisfactory for most structures, especially with a skillful use of the mesh density factor.

2.4.1.2 Non-Default Mesh Generation Using skin

If, after picking polylines and pressing `Go` , you responded with N↵ to the request

```
Use default discretization (y):
```

the following would be displayed, with the first two polylines highlighted:

```
Enter # of interp sections between red sections [default]
```

The user is being requested to specify the discretization by giving the number of interpolation sections between the polylines highlighted red. The space between the red polylines will be divided into ([*num*] + 1) equal parts, where [*num*] is the number of interpolation sections specified. That is, [*num*] temporary interpolation polylines will be placed between the highlighted two.

Type the required number, followed by ↵, or just press ↵ to accept the [*default*].

The same will be done for all the picked polylines, in sets of two. Thereafter, one set of corresponding segments on all the polylines is highlighted red, and the following is displayed:

```
Enter discretization for red segment [default]
```

The user is being requested to define the discretization around the polylines, segment by segment, by specifying the number of boundary element edges to be placed on each polyline segment (as was explained for **polyline->nodeline** in Section 2.3.4). The minimum is 1.

Type the desired number, followed by ↵; or just press ↵ to accept the [*default*].

When this is done for the last set of segments, the message `Creating Elements...` is displayed to indicate that element generation is proceeding. The elements are displayed at the end, the ND and EL counters are updated, and the dialogue box clears to indicate that EXAMINE^{3D} is ready.

2.4.2 Build Object + extrude

The **extrude** function is used to generate the geometry and boundary element mesh for a structure which has uniform cross-section. The cross-section is first defined using a polyline, which would be extruded along a path to generate the structure. If the longitudinal axis of the structure is a straight line, then no other polyline is required, because the extrusion path can be specified by typing in the components of a vector. On the other hand, the longitudinal axis of the structure need not be a straight line. In general, the extrusion path can be specified as a polyline.

2.4.2.1 Procedure for Linear Extrusion

When the polyline to be extruded is ready, select **Build Object + extrude**; the following is displayed:

```
Select Curve to Extrude:
```

Click on the polyline

```
Enter/pick extrusion dir (default)[N,U,E: snaps ]:
```

The extrusion direction is defined in terms of a unit vector (specified by giving its N-U-E components). The [*default*] is the outward normal from the plane of the polyline. For example, if the polyline lies in the N-U plane, with vertices ordered counterclockwise, the default extrusion direction is (0,0,1).

Type the three components of the extrusion direction vector, separated with comma (,) or space, and followed by ↵

```
Enter length of extrusion [default]
```

Type the length of the structure, followed by ↵

```
Use default discretization (y):
```

<u>Either</u>: press ↵ to accept the default discretization; the mesh is generated with no additional user input;
<u>Or</u>: Enter N↵, for a user-specified discretization; in that case the following is displayed:

```
Enter discretization for red segment [default]:
```

A segment of the polyline is highlighted red, and the user is requested to specify the number of boundary element edges to be placed on the segment (see Section 2.3.4 for more details).

Type the number, followed by ↵, or simply press ↵ to accept the [*default*]. Do the same for the remaining polyline segments, one by one. Then,

```
Enter number of divisions along length [default]:
```

The structure will be divided into [*num*] equal parts along its length, where [*num*] is the number supplied here.

Type the required number, followed by ↵, or simply press ↵ to accept the [*default*]. The mesh will be generated thereafter.

2.4.2.2 Procedure for General Extrusion

Two polylines are required to define a general extrusion: The first defines the cross-section of the structure; it is referred to hereafter as the extrusion contour polyline. The second defines the path of extrusion; it is referred to hereafter as the extrusion path polyline.

For best results, the first vertex of both the extrusion contour and extrusion path polylines should coincide; also, the first vertex and the pivot points of both polylines should coincide. Under this arrangement, the extrusion path polyline represents a trace along the wall surface of the structure, parallel to its axis.

When the two polylines are ready, select **Build Object + extrude**; the following is displayed:

```
Select Curve to Extrude:
```

Click on the extrusion *contour* polyline

```
Enter/pick extrusion dir (default)[N,U,E: snaps ]:
```

Click on the extrusion *path* polyline

```
Use default discretization (y):
```

To use the default discretization:

Press ↵

```
Fix first extrusion contour (y):
```

If the first extrusion contour is fixed (the default), then the structure begins at the location of the extrusion contour polyline. If it is not, then the structure begins at the beginning of the extrusion path polyline. These two options are equivalent if the first vertex of the extrusion contour and extrusion path polylines coincide. If the two first vertices are different, fixing the first extrusion contour may yield unacceptable results. It is strongly recommended that the two vertices coincide, in which case

Press ↵

```
Fix extrusion base (y):
```

Fixing the extrusion base ensures that the inclination of the floor normal to the extrusion path remains the same, irrespective of the twists and turns of the path.

Press ↵ to accept the default, if the base is required to be fixed; otherwise, enter N↵ to reject.

The message `Generating Extrusion, Please Wait...` will be displayed until the meshing is done.

For a user-defined discretization:

After picking the two polylines, respond with N↵ to the question `Use default discretization (y):`

The user will be prompted to specify the discretization (i.e., number of boundary element edges) for each segment of, *first* the extrusion contour polyline, and *then* the extrusion path polyline. Respond to each prompt by entering [*Num*]↵, where [*Num*] is the required number for each segment.

Thereafter, the process proceeds in the same manner as for the default discretization.

2.4.3 Build Object + face→

The **face→** function is the most versatile among the mesh generators described in this section; however, unlike the **skin** and **extrude** functions, **face→** does not generate geometry. It is simply a mesh generator, suitable for discretizing any region bounded by one or more nodelines. Each of the nodelines must be closed. Other than this requirement, there is no restriction on their relative orientations or number of nodes; furthermore, they need not lie in the same plane.

The **face→** function consists of a group of functions which are accessed through the FACES screen. To obtain this screen, select **Build Object + face→**; the following message is displayed:

```
Pick a CLOSED Nodeline:
```

Click on the first nodeline which bounds the region to be discretized.

The nodeline is highlighted to indicate that it has been picked, and the Go buttons appear, along with the following message:

```
Pick interior CLOSED Nodeline [Go=done]:
```

Either: the region to be discretized is completely bounded by the first nodeline picked; in which case click on Go, and the FACES screen (e.g. Figure 2.1) will be displayed

Or: the region is bounded by more than one nodeline; in which case pick the remaining nodelines, one by one; then click on Go to indicate that they have all been picked; the FACES screen (e.g. Figure 2.2) will be displayed thereafter; the order in which the nodelines are picked determines which of the enclosed regions will be discretized (please see Tutorial 4 for an example).

Some of the functions available at the FACES screen require some user-input to generate the mesh, whereas others do not.

2.4.3.1 Default Meshing at the FACES screen

The three functions **Array Mesh**, **Radial Mesh**, and **Automatic Mesh** require no user-input to generate a mesh.

Array Mesh

This function is best suited for rectangular faces, if the nodes on opposing edges of the face can be joined to generate a rectangular pattern. The function is not available if the face is bounded by more than one nodeline. Therefore, it cannot be used to discretize a transition face.

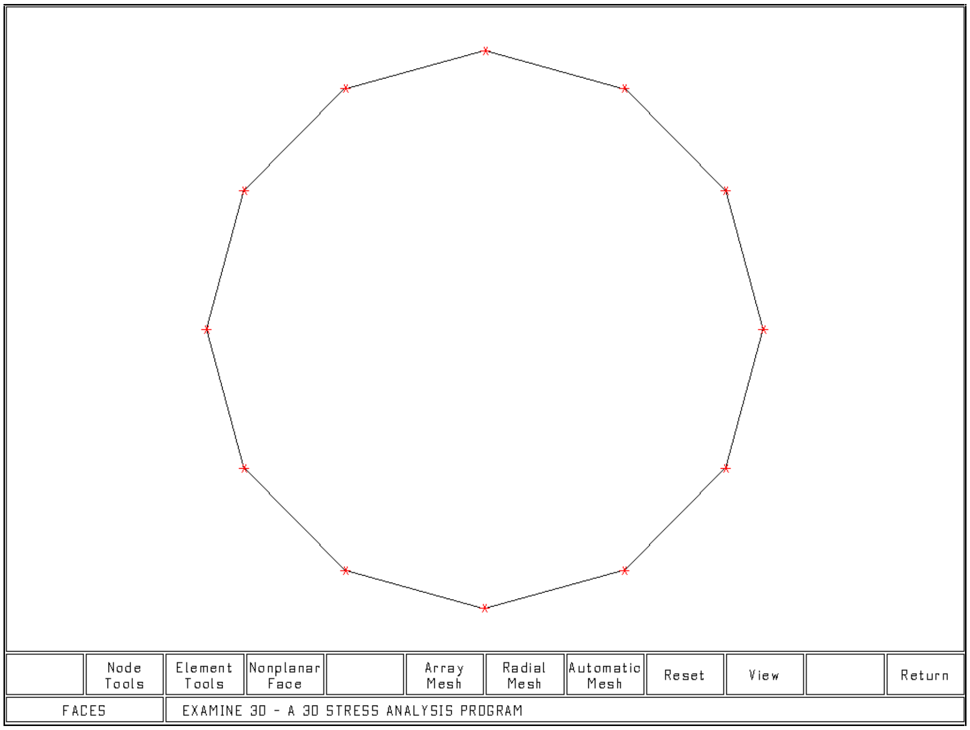


Figure 2.1: The FACES screen, showing a simple face

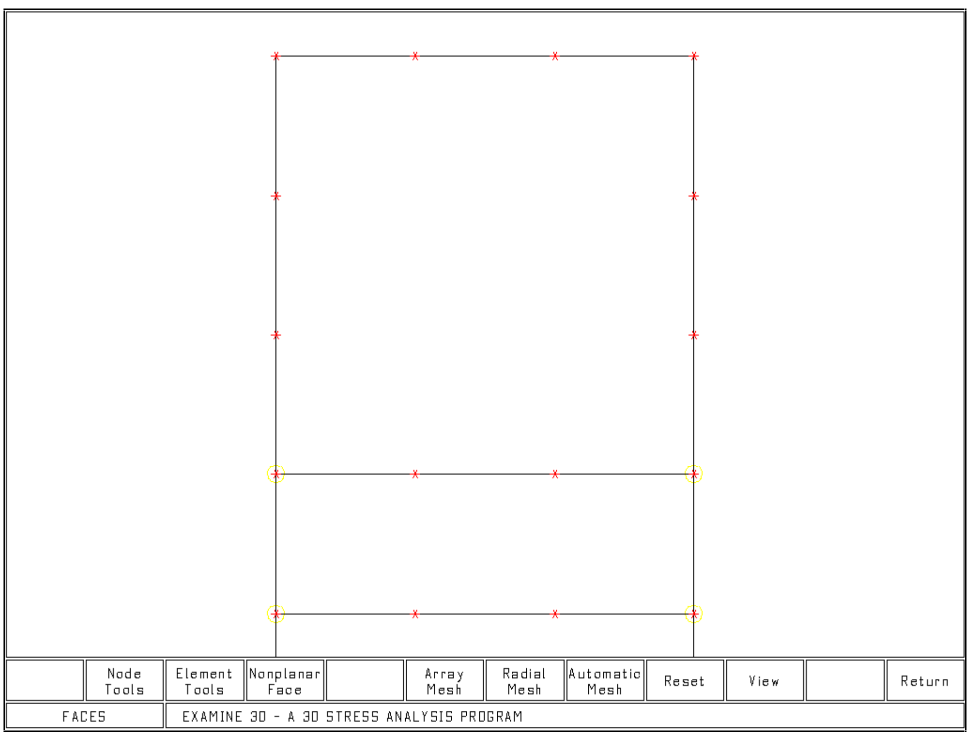


Figure 2.2: The FACES screen, showing a transition face

Radial Mesh

This function is suitable for radially symmetric faces, for which (1) the nodes are roughly equidistant from the center of the face, and (2) their distance from the center is about the same as the length of the line segments (boundary element edges) joining adjacent nodes. The function places a node at the center of the face, and joins each of the other nodes to the central one to generate the mesh.

Automatic Mesh

This function is likely to generate a satisfactory mesh for most faces. The algorithm uses a random process; hence, successive applications of the function to the same face (using **Reset** to clear a mesh before generating a subsequent one) will give slightly different results.

2.4.3.2 Node Tools at the FACES Screen

Selecting **Node Tools** at the FACES screen opens access to a group of three functions for placing nodes in the interior of a face. These functions do not create elements; they simply place the nodes, leaving it up to the user to select a procedure (from **Element Tools**) for generating the elements.

Node Tools + automatic insert

This function places nodes in the interior of a face, without any more user-input. No nodes are generated if the program “thinks” that none are needed. In that case any interior nodes required by the user will have to be placed manually using the next function.

Node Tools + add one node

This function is used to place interior nodes one by one. The following is displayed when this function is selected:

```
Insert node(s) within face [Press Escape when done]:
```

Click on any interior point once to insert a node there. Select each point well before clicking, because no editing is available (except to delete using the next function). To exit from **Node Tools + add one node**, press ESC once.

Node Tools + delete one node

This function allows the user to delete interior nodes, one by one. The following is displayed when the function is selected:

```
Select node(s) to delete [Press Escape when done]:
```

To delete a node, click on it. Delete as many as desired, in the same way. To exit from **Node Tools + delete one node**, press ESC once.

2.4.3.3 Element Tools at the FACES Screen

Selecting **Element Tools** at the FACES screen opens access to three functions, which enable the user to exercise considerable control on the boundary element discretization of a face.

Element Tools + add triangle

This function allows the user to add triangular elements manually, by picking the first, second and third nodes for each element. Any existing node (including the interior nodes inserted using **Node Tools**) may be picked. The following is displayed when the function is selected:

```
Select node 1 making up triangle [Go Counter-clockwise]:
```

A node is selected by clicking on it; the display updates to prompt the user for node 2, and then for node 3. Each node picked is marked with a yellow circle; also, the element edges are drawn as the nodes are picked. When the third node is picked, the element defined by the three nodes is shaded, and the yellow markers are removed. Thereafter, the display is updated, to prompt for node 1 of the next element.

Nodes 1, 2 and 3 for an element should be picked in a counter-clockwise order. To exit from **Element Tools + add triangle**, press ESC once.

Element Tools + delete element

This function allows the user to delete elements selectively, one at a time. This is unlike the **Reset** function, which deletes all elements and interior nodes at once. The following is displayed when **Element Tools + delete element** is selected:

```
Select element(s) to delete [Press Escape when done]:
```

To delete an element, click on its face, or on one of its edges. It is deleted at once. It is better to click on the element face, to ensure that the element deleted is the desired one. To exit from **Element Tools + delete element**, press ESC once.

Element Tools + automatic mesh

This function inserts elements automatically, using all existing nodes (including interior nodes inserted using **Node Tools**). No more user-input is required.

This function cannot be used if there are existing elements on the same face (created using **Element Tools + add triangle**). In that case, the following message is displayed when **Element Tools + automatic mesh** is selected:

```
Error, elements already exist - select Reset
```

To proceed, either delete the existing elements using **Element Tools + delete element**; or delete the existing elements and interior nodes using **Reset**; or continue to add elements manually using **Element Tools + add triangle**.

The **Element Tools + automatic mesh** function should not be confused with the **Automatic Mesh** function. The latter does not accept user-inserted elements or interior nodes. The following message is displayed if **Automatic Mesh** is selected when there are some interior nodes or elements:

```
This will reset all nodes and elements, continue (y)
```

Either: press ↵ to delete all elements and interior nodes, and generate a new mesh;

Or: enter N↵ or press ESC to exit from **Automatic Mesh** without doing anything.

2.4.3.4 Nonplanar Faces at the FACES screen

This function is used to interpolate interior nodes if the bounding polyline defining the face is not planar. To use this option, first create the face using either the automatic or manual methods described above. Then simply choose this option to interpolate the interior nodes. This option should be done just prior to exiting from the faces screen. If the face is planar, do not use this option.

2.4.3.5 Exiting from the FACES Screen

Select **Return**; then, at the exit condition window,

Either: select **Yes** to save the current discretization of the face; it will be *superimposed* on the existing discretization; therefore, any existing discretization of a face should be deleted before invoking **Build Object + face**→ for the same face;

Or: select **No** to discard it.

2.4.4 Build Object + blend

Like the **skin** function, the **blend** function generates a surface of triangular elements around a series of polylines. Unlike the skin function, the polylines need not have the same number of vertices or be ordered consistently. Unfortunately, it is impossible to properly skin all possible sets of contours so the user should take extra care in ensuring that the resulting surface is correct. This method is known to produce erroneous results when the closed polyline loops are long and narrow, as in a cut and fill stope. The **skin** option will always be available as a more rigorous but predictable method.

To start, select **Build Object + blend**; the Go buttons will appear, and the following message is displayed:

```
Pick a polyline [ [Num] picked, Select Go when done]:
```

A polyline is picked by clicking on it, at which it becomes highlighted (yellow). Clicking on a highlighted polyline causes it to be unpicked, at which its color reverts to blue. All polylines picked must be consecutive.

Pick as many as desired. The parameter [*Num*] is updated each time to indicate the number picked so far. Thereafter,

Click on Go to indicate that the required polylines have all been picked. The following message is displayed:

```
Enter density, valid range from 0 to 4 [1.0]:
```

A density value of 1.0 tries to enforce a 3 to 1 element aspect ratio yet create a reasonable number of elements. As a result, the 3 to 1 ratio is not strictly required for each element created and is a function of the discretization of the initial polylines used for blending. As a guide, a density of 3.0 will create about twice as many elements as the default 1.0 density while a density of 0.5 will create about half as many.

Press ↵ to accept the [*default*], or type the required value, followed by ↵.

2.4.5 Build Object + transition skin

The **transition skin** function generates the geometry and boundary element mesh for a structure, between two polylines which have different numbers of vertices. The number of vertices on one polyline must be related to the number on the other polyline by a 2:1 ratio. Therefore, such polylines will often be used to define the transition zone between more detailed and less detailed geometrical cross-sections.

The **transition skin** function prompts the user to identify the two polylines, as follows:

```
Select first transition section:
```

Click on the first polyline (to define the first end of the transition zone).

```
Select second transition section:
```

Click on the other polyline (other end of the transition zone).

Thereafter, the mesh is generated, without requesting any more user-input.

2.5 Modeler Menu Item: Pick

all
element
component
object
nodeline
polyline
nothing

The **pick** group of functions, shown in the pop-up menu at left, enable the user to select a variety of entities to be operated on by some other function. An entity is highlighted (yellow) when selected. The **pick** function works like a toggle: if an entity is *not selected*, then **pick**ing it causes it to be *selected*; on the other hand, **pick**ing a *selected* entity causes it to be *de-selected*. The word **pick** will be used in this sense throughout Section 2.5.

File	Toolbox	Build Polyline	Build Object	Pick	Xform	Object Tools	View	Shade	Analysis Param	Field Points	Return

2.5.1 Pick + nothing

This function causes every entity currently selected to be de-selected, and hence de-highlighted.

2.5.2 Pick + polyline

This function is used to select or de-select polylines. The following message is displayed when the function is selected:

```
Pick Curve [*=all; b=box; c=cbox; ESC=done]:
```

<u>Either</u>: press $\boxed{*}$ to **Pick** all polylines at once, and exit;

<u>Or</u>: press $\boxed{b}$ or $\boxed{c}$ to **Pick** a polyline group (see below);

<u>Or</u>: click on a polyline to **Pick** it; do the same for as many as desired; then press ESC to exit from **Pick + polyline**.

2.5.2.1 Picking Polyline Groups with the BOX Option

Following the message

```
Pick Curve [*=all; b=box; c=cbox; ESC=done]:
```

press $\boxed{b}$ to activate the BOX option. This option causes all polylines *completely enclosed* by a user-defined box to be **Pick**ed. The user is prompted to specify two corners of the box. If the two corners specified lie in different planes, then the 3-d box is defined *explicitly* using the two corners. On the other hand, if the given points lie in the same plane, then the box is defined *implicitly* by infinitely extruding the 2-d box (plane) formed by the two points.

The following message is displayed when the BOX option is selected:

```
Place box corner #1 [N,U,E: snaps]:
```

Either: type the N-U-E coordinates of the first corner, separated by comma (,) or space, and press ↵;

Or: click on a point to place the first corner there.

Adjust the point as required, either through the keyboard or with the mouse. If using the mouse, recall that the snap functions constrain the category of selectable points; also recall that only two coordinates can be modified through any one (and all three coordinates through any two) of the orthogonal view windows.

When the point is satisfactory, click on [Go] to accept it; then the second corner is requested, as follows:

```
Place box corner #2 [N,U,E: snaps]:
```

Either: type in the coordinates, followed by ↵;

Or: click on a point to place the second corner there.

The outlines of the box (or plane) are shown in yellow; adjust the coordinates of the second corner as desired; then click on [Go] to accept the box. The box is removed and all polylines enclosed by it are highlighted (or de-highlighted if already selected) to show that they have been **Pick**ed. The dialogue box message reverts to:

```
Pick Curve [*=all; b=box; c=cbox; ESC=done]:
```

2.5.2.2 Picking Polyline Groups with the CBOX Option

The CBOX option, activated by pressing [c] in response to the above message, works exactly like the BOX option, except for the difference that all polylines *either crossed or completely enclosed* by the box are **Pick**ed.

Please see the previous section for a description of the procedure.

2.5.3 Pick + nodeline

The function enables the user to select or de-select nodelines. The following is displayed when the function is selected:

```
Pick Nodeline [*=all; b=box; c=cbox; ESC=done]:
```

Either: press [*] to **Pick** all nodelines at once, and exit;

Or: press [b] or [c] to **Pick** a nodeline group (see above discussion for polylines, Section 2.5.2.1, since the process is exactly the same);

Or: click on a nodeline to **Pick** it; do the same for as many as desired; then press ESC to exit.

Nodelines and elements cannot be **Pick**ed independently. If elements are already picked, selecting **Pick + nodeline** results in the following error message:

```
Error, cannot pick a nodeline if element is already picked
```

2.5.4 Pick + object

An object, in EXAMINE^{3D}, consists of a block of inter-connected boundary elements. For example, the boundary element mesh for two neighboring parallel tunnels consists of two objects; whereas the one for two intersecting tunnels consists of one object.

Polylines are not objects; on the other hand, because nodelines are integral parts of elements, they and the elements constitute parts of the same object. Hence, nodelines are **Pick**ed along with objects, but polylines are not.

The following message is displayed when **Pick + object** is selected:

```
Pick object [*=all; ESC=done]:
```

Either: press * to **Pick** every object at once, and exit;

Or: click anywhere on an object to **Pick** it; then, when all required objects are **Pick**ed, press ESC to exit.

Objects and nodelines cannot be **Pick**ed independently. Selecting **Pick + object** when nodelines are already picked causes an error message.

2.5.5 Pick + component

A component is a group of elements generated by a single application of any one of the five mesh generation functions, viz., **skin**, **extrude**, **face**, **blend**, or **transition face**. For example, the mesh for a circular tunnel may be generated by one application of **extrude** and two applications of **face**; in which case it consists of three components: the elements on each tunnel face constitute one component, and those on the cylindrical surface constitute one component.

Polylines are not components; on the other hand, because nodelines are integral parts of elements, they and the elements belong to the same component.

The following message is displayed when **Pick + component** is selected:

```
Pick Component [*=all; ESC=done]:
```

Either: press * to **Pick** all components at once, and exit;

Or: click anywhere on a component to **Pick** it; when all required components are **Pick**ed, press ESC to exit.

2.5.6 Pick + element

Elements can be **Pick**ed individually or in groups. The following message is displayed when **Pick + element** is selected:

```
Pick Element [*=all; b=box; c=cbox; r=ratio; ESC=done]:
```

2.5.6.1 Picking Elements Individually

Elements are **Pick**ed individually by clicking on their *edges*. When an edge is shared by two elements, clicking on it repeatedly causes it and the neighboring edges to be highlighted or de-highlighted in turn, thereby indicating which element is selected or de-selected by each click. Therefore, to **Pick** an individual element, click on one of its edges a few times, observing which edges are highlighted or de-highlighted, until it is **Pick**ed.

2.5.6.2 Picking Element Groups with the BOX Option

Following the message

```
Pick Element [*=all; b=box; c=cbox; r=ratio; ESC=done]:
```

press b to activate the BOX option. This option causes all elements *completely enclosed* by a user-defined box to be **Pick**ed. The user is prompted to specify two corners of the box. If the two corners specified lie in different planes, then the box is defined *explicitly* using the two corners. On the other hand, if the given points lie in the same plane, then the box is defined *implicitly* by infinitely extruding the plane formed by the two points.

The following message is displayed when the BOX option is selected:

```
Place box corner #1 [N,U,E: snaps]:
```

Either: type the N-U-E coordinates of the first corner, separated by comma (,) or space, and press ↵;

Or: click on a point to place the first corner there.

Adjust the point as required, either through the keyboard or with the mouse. If using the mouse, recall that the snap functions constrain the category of selectable points; also recall that only two coordinates can be modified through any one (and all three coordinates through any two) of the orthogonal view windows.

When the point is satisfactory, click on Go to accept it; then the second corner is requested, as follows:

```
Place box corner #2 [N,U,E: snaps]:
```

Either: type in the coordinates, followed by ↵;

Or: click on a point to place the second corner there.

The outlines of the box (or plane) are shown in yellow; adjust the coordinates of the second corner as desired; then click on Go to accept the box. The box is removed and all elements enclosed by it are highlighted (or de-highlighted if already selected) to show that they have been **Pick**ed. The dialogue box message reverts to:

```
Pick Element [*=all; b=box; c=cbox; r=ratio; ESC=done]:
```

2.5.6.3 Picking Element Groups with the CBOX Option

The CBOX option, activated by pressing c in response to the above message, works exactly like the BOX option, except for the difference that all elements *either crossed or completely enclosed* by the box are **Pick**ed.

Please see the fore-going paragraphs for a description of the procedure.

2.5.6.4 Picking Element Groups with the RATIO Option

This option enables the user to **Pick** all elements which have aspect ratio larger than a specified value. Having selected **Pick + element**, the user is prompted as follows:

```
Pick Element [*=all; b=box; c=cbox; r=ratio; ESC=done]:
```

Press r to activate the **Pick**-by-RATIO option.

```
Enter element ratio:
```

Type the required value of aspect ratio and press ⮠. A message is displayed indicating that the process is working; thereafter, the following is displayed:

```
[Num] Elements Picked, Press Enter to Continue
```

where [*Num*] is the number of elements which have aspect ratio larger than the specified value. All of them are **Pick**ed: those which were not previously selected are highlighted, indicating that they have been selected; those which were previously selected are de-highlighted, indicating that they have been de-selected.

Press ⮠, and the following is re-displayed, to signal exit from the **Pick**-by-RATIO option:

```
Pick Element [*=all; b=box; c=cbox; r=ratio; ESC=done]:
```

2.5.6.5 More on Picking Elements

In addition to **Pick**ing elements individually or in groups using the BOX, CBOX or RATIO options, all elements can be **Pick**ed at once by pressing *; also, groups of elements can be **Pick**ed as components or objects.

To exit from **Pick + element**, press ESC.

2.5.7 Pick + all

This function is used to select all objects and polylines at once, including those already selected using another function. It is the opposite of **Pick + nothing**.

2.6 Modeler Menu Item: Xform

set pivot
nonp scale
scale
rotate
copy
move

The **Xform** menu opens access to a group of transformation functions, for modifying the shape, size or location of objects or polylines. The functions cannot be applied to nodelines or individual elements or components. The object or polyline to be transformed has to be picked first, using **Pick + object** or **Pick + polyline**, at which it becomes highlighted and its pivot (a large green star) appears. All transformations of the object or polyline are defined with respect to the pivot point.

File	Toolbox	Build Polyline	Build Object	Pick	Xform	Object Tools	View	Shade	Analysis Param	Field Points	Return

2.6.1 Xform + set pivot

This function enables the user to specify a point in space to which all transformations will be referenced. The point is marked by a large green star, which appears when a polyline or object is selected using **Pick**.

The following message is displayed (along with the Go buttons) when **Xform + set pivot** is selected:

```
Enter Pivot Point Loc [N,U,E: snaps]:
```

<u>Either</u>: type the coordinates of the new location, followed by ↵;

<u>Or</u>: click on a point to locate the pivot there, using any two orthogonal view windows (and any of the snap functions) to adjust the coordinates of the point.

Then click on Go to accept the new location, or press ESC to abort the function without effecting any change.

The location of a pivot affects the results of **rotate**, **scale** and **nonp scale**; therefore, ensure that it is appropriately located before applying any of these functions.

2.6.2 Xform + nonp scale

The **nonp scale** function permits an object or polyline to be scaled by a different scale factor in each of the three coordinate directions, thus changing both its shape and size. The following message is displayed when this function is selected:

```
Enter N,U,E Scale Factor [default]:
```

Enter [*fN fU fE*] ⟻, where *fN*, *fU* and *fE* are the values of scale factor to be applied to the N, U and E coordinate directions, respectively.

The transformation is applied immediately, causing an exit from **Xform**; also, the object or polyline is de-selected. It is irreversible, except by re-applying **Xform + nonp scale** with values of scale factor equal to the reciprocals of the previous ones.

2.6.3 Xform + scale

The **scale** function works exactly like the **nonp scale** function, except that the same value of scale factor is applied in all three coordinate directions. Therefore, only one value of scale factor is requested. The following message is displayed when the function is selected:

```
Enter Scale Factor [default]:
```

Enter [*f*] ⟻, where [*f*] is the value of scale factor to be applied.

The transformation is applied immediately, causing an exit from **Xform**; also, the object or polyline is de-selected. It is irreversible, except by re-applying **Xform + scale** with a value of scale factor equal to the reciprocal of the previous one.

2.6.4 Xform + rotate

This function enables an object or polyline to be rotated about its pivot point. The rotation is applied counter-clockwise, by a value specified in terms of three N-U-E components. Each component gives the magnitude of rotation in degrees, about a line through the pivot point and parallel to the coordinate axis.

The following is displayed (along with the [Go] buttons) when **Xform + rotate** is selected:

```
Enter CCW Rot [deg NUE, Go=done, Esc=abort]:
```

Enter [*rN rU rE*] ⟻,

where *rN*, *rU* and *rE* are the magnitudes of rotation in degrees, about lines parallel to the N, U and E coordinate axes, respectively. For example, to rotate 90° about the N-axis, enter [90 0 0]⟻; or to rotate 90° about both the N and U axes enter [90 90 0]⟻.

The result of the rotation is displayed, but is not yet accepted, and the above dialogue box request is kept active. At this point,

Either: enter additional rotations; each subsequent rotation is applied starting from the current position of the object or polyline;

Or: click on [Go] to accept the current position of the object or polyline, thereby exiting from **Xform**;

Or: press ESC to reject all rotations applied, thereby restoring the object or polyline to its position before **Xform + rotate** was invoked; this also causes an exit.

Each of the above exit options (i.e., Go or ESC) also causes the object or polyline to be de-selected.

2.6.5 Xform + copy

The **copy** function makes one or more copies of the selected object or polyline, and places each at a user-specified location. The following message is displayed (along with the Go buttons) when this function is selected:

```
Enter Rel Displ of Copy [N,U,E: snaps]:
```

For each required copy,

Either: enter [*coordinates*] ↵, where [*coordinates*] are the N-U-E components of displacement, which define the location of the pivot of the copy relative to the pivot of the original;

Or: click on a point to select it as the pivot point of the copy.

When all required copies have been placed:

Either: click on Go to accept the current positions of all copies, and thereby exit from **Xform**; this causes the object or polyline to be de-selected.

Or: press ESC to abort, thereby discarding all copies and exiting from **Xform**; the object or polyline remains selected.

2.6.6 Xform + move

This function performs a translation of the object or polyline, relative to the pivot. The following message is displayed (along with the Go buttons) when it is selected:

```
Enter Rel Translation [N,U,E: snaps]:
```

Either: enter [*coordinates*] ↵, where [*coordinates*] are the N-U-E components of the required translation;

Or: click on any point to select it as the new location of the pivot point.

Modify the selected location, either through the keyboard or with the mouse. If using the mouse, recall that the snap functions constrain the category of selectable points; also recall that only two coordinates can be modified through any one (and all three coordinates through any two) of the orthogonal view windows. Then

Either: click on Go to accept the new location, thereby exiting from **Xform**;

Or: press ESC to abort, thereby restoring the object or polyline to its position before **Xform + move** was invoked; this also causes an exit from **Xform**.

Any of the above exit options causes the object or polyline to be de-selected.

2.7 Modeler Menu Item: Object Tools

delete all
delete picked
apply traction
locate surface
relocate node
reorder nodes/vert
subdiv.elem/poly
visible
invisible

The **Object Tools** menu opens access to a group of functions for editing a model. The object, component, element, polyline or nodeline to be operated on by these functions must first be selected using **Pick**.

File	Toolbox	Build Polyline	Build Object	Pick	Xform	Object Tools	View	Shade	Analysis Param	Field Points	Return

2.7.1 Object Tools + invisible

This function is used to hide selected entities from view. Any such hidden entity is essentially temporarily removed. Hence, it cannot be picked or operated on by any function (except **Object Tools + visible**).

The function is useful for gaining access (with the mouse) to elements or components behind the immediate view plane. For example, the elements on the floor of a tunnel can be accessed through the top view window if the roof area of the tunnel is made **invisible**.

To make geometry invisible, first select it using **Pick**, then select **Object Tools + invisible**.

2.7.2 Object Tools + visible

This function is used to restore access to all **invisible** entities, by bringing them back into view.

2.7.3 Object Tools + subdiv elem/poly

This function is used to refine a discretization by subdividing selected elements or polylines.

To subdivide a polyline, first select it using **Pick**, then select **Object Tools + subdiv elem/poly**. The following message is displayed:

```
This is an irreversible change, are You Sure (n)?
```

Either: press ↵ to abort;

Or: enter Y↵; a new vertex is placed at the midpoint of each existing segment of the polyline, thereby dividing the segment into two equal segments; the polyline is de-selected thereafter; this also causes an exit from **Object Tools**.

The process is irreversible, except by using **Build Polyline + edit polyline**.

To subdivide one or more elements, select the elements using **Pick**; then select **Object Tools + subdiv elem/poly**; the same warning query is displayed, as follows:

```
This is an irreversible change, are You Sure (n)?
```

Either: press ↵ to abort;

Or: enter Y↵; the number of elements within the boundaries of the selected group is quadrupled by the subdivision process; furthermore, some transition elements are formed in the immediate neighborhood of the selected elements; the aspect ratio of the transition elements may be bad, depending on the difference between the sizes of elements in the refined and unrefined mesh zones.

The only way to reverse the element subdivision is to delete all elements within the affected zone, and remesh the zone using an appropriate element generation function.

2.7.4 Object Tools + reorder/vert

This function is used to change the order of vertices for a polyline, or nodes for a nodeline. The order of vertices and nodes affect the direction of the normals for the elements generated from such nodelines/polylines.

Polyline vertices are independent of the nodes on superimposed nodelines. Therefore, to reorder both the vertices and nodes, both the polyline and nodeline should be selected and operated on (either together or individually). Operating on the nodeline alone will not affect the polyline, and vice versa.

2.7.5 Object Tools + relocate node

This function permits the user to modify a boundary element mesh by relocating some of the nodes; it also provides for the removal of elements which have a large aspect ratio. To start select **Object Tools + relocate node**; the user is prompted to select the nodes, as follows:

```
Pick Nodes [b=box; f=filter; Go=done; ESC=abort]:
```

2.7.5.1 Relocating Nodes

The dialogue box message indicates that the nodes to be relocated can be selected either individually or in groups. The box option permits the selection of node groups; the user is prompted to specify two corners of the box; if the two corners are in different planes, then they define the box *explicitly*; if they are in the same plane, the box is defined *implicitly* by an infinite extrusion of the plane.

To select an individual node, click on it; it is marked with a red star to indicate that it has been selected. To de-select a marked node, click on it; the red star is cleared, indicating that the node has been de-selected.

To select a group of nodes using the box option, press $\boxed{\text{b}}$; the following is displayed:

```
Place box corner #1 [N,U,E: snaps]:
```

Either: type the N-U-E coordinates of the first corner, separated by comma (,) or space, and press ↵;

Or: click on a point to place the first corner there.

Adjust the point as required, either through the keyboard or with the mouse. If using the mouse, recall that the snap functions constrain the category of selectable points; also recall that only two coordinates can be modified through any one (and all three coordinates through any two) of the orthogonal view windows.

When the point is satisfactory, click on $\boxed{\text{Go}}$ to accept it; then the second corner is requested, as follows:

```
Place box corner #2 [N,U,E: snaps]:
```

Either: type in the coordinates, followed by ↵;

Or: click on a point to place the second corner there.

The outlines of the box (or plane) are shown in yellow; adjust the coordinates of the second corner as desired; then click on $\boxed{\text{Go}}$ to accept the box.

The box is removed and all nodes enclosed by it are marked with red stars, indicating that they have been selected.

Having picked all required nodes, or if you should change your mind about proceeding with node relocation:

Either: press ESC to de-select all nodes, and exit from the function;

Or: click on $\boxed{\text{Go}}$ to accept the nodes; the following message is displayed:

```
Enter Rel Translation [N,U,E: snaps]:
```

The new location of each of the selected nodes is defined in terms of a translation, relative to the current location of its (red) marker. The user is being prompted to specify the translation vector.

Either: type the N-U-E components of the required translation, separated by comma (,) or space, and press ↵;

Or: click on a point to select it as the new location of the *base node*; the translation vector is the vector joining the base node marker to the clicked point.

The translation vector for a group of nodes would be best specified through the keyboard; otherwise, the user would have to experiment with the mouse, in order to identify the base node.

As soon as a translation is specified the red markers are relocated, showing what the effect of the translation would be if it is effected; all subsequent translations are referred to the current location of the markers. Adjust the location of the markers as required, either with the mouse or through the keyboard. When the new locations are satisfactory, or if you should change your mind about proceeding:

Either: press ESC to de-select all nodes, and exit from the function;

Or: click on Go to proceed; the following warning query is displayed:

```
This is an irreversible change, are You Sure (n)?
```

Either: press ↵ to exit, without effecting any change;

Or: enter Y↵; each selected node is relocated at the current position of its marker, and the function terminates.

The process is irreversible by itself; on the other hand, it may be possible to reverse it by applying **Object Tools + relocate node** to the exact same node set, with an equal and opposite translation.

2.7.5.2 Using the Filter Option

The filter option of the **relocate node** function is used to remove very thin elements. Having selected **Object Tools + relocate node**, press f to activate the filter option. The user is prompted to specify the maximum aspect ratio, as follows:

```
Enter element filter ratio [default]:
```

Either press ↵ to accept the default, or type the required value and press ↵. All elements which have larger values of aspect ratio will be removed, depending on how the user responds to the following warning query:

```
This is an irreversible change, are You Sure (n)?
```

Either press ↵ to abort, or enter Y↵ to proceed.

It should be noted that this is a single-pass process on all elements: The aspect ratio of some elements may increase beyond the specified limit, as a result of the removal of high aspect ratio elements, and the program does not check for this possibility. Therefore, the option should be run again to re-check the aspect ratio of all elements.

2.7.6 Object Tools + locate surface

To properly define a ground surface you must first **pick** the object representing the ground surface with the **Pick + object** (Section 2.5.4) function and then define it using the **Object Tools + locate surface** option.

The object representing the ground surface must extend a distance such that its outer boundary will not affect the region of interest. As a rule of thumb, imagine a bounding volume around the zone of interest, then expand this volume by a factor of three or more. The outer edges of your surface should lie outside this volume. After you have located the surface object, the elements composing the object will be drawn in green to indicate that they are part of a free surface.

2.7.7 Object Tools + apply traction

This function is used to apply pressure or traction boundary conditions to excavations or ground surfaces. To do this, within the modeler **pick** the element, component or object that you wish to apply the traction/pressure to, then choose **Object Tools + apply traction**. You will be asked to give the type of boundary condition, pressure or traction. In the case of a pressure load, you will be asked to give its magnitude. A zero pressure value has the affect of removing the boundary condition for the picked elements. Positive pressures are applied to each picked element in a direction *into* the rock mass. If you apply a traction, you will be asked to enter its magnitude *and* direction. The region that you selected will be shaded light blue to indicate that there is a pressure or traction boundary condition on these elements. You may also see the direction of the pressure by turning ON the element normals in the **shade options** (Section 2.9.2) menu, then shading the model using the **quickshade** (Section 2.9.4) option.

To apply a pressure or traction boundary condition, first select the element, component or object using **Pick**, then select **Object Tools + apply traction**. The following message is displayed:

```
Enter Traction Type (1=normal pressure, 2=traction vector) [1]:
```

2.7.7.1 Input Option 1: Normal Pressure

This option allows the user to define the applied traction boundary condition as a normal pressure on the surface of any picked elements. To select this option, enter 1 ⏎ in response to the above message. The following is displayed:

```
Enter Pressure To Apply [0]:
```

Enter the magnitude of the pressure to apply remembering that a positive value has an orientation convention of being *into* the rock mass while a zero value removes any previously defined traction boundary condition.

2.7.7.2 Input Option 2: Traction Vector

This option allows the user to define a traction in any user defined direction, on the surface of any picked elements. To select this option, enter 2 ⏎ in response to the above message. The following is displayed:

```
Enter Traction Vector Magnitude [0]:
```

Enter the magnitude of the traction to apply remembering that a zero value removes any previously defined traction boundary condition. After you have entered a non-zero value the following is displayed:

```
Enter Traction Vector Direction [NUE=(0,-1,0)]:
```

Enter the direction vector, which need not be a normal vector, in terms of the global coordinate system. Note the default direction of (0,-1,0) indicates a traction in the “down” direction.

2.7.8 Object Tools + delete picked

This function deletes all selected (highlighted) entities. The user is not prompted for more input, and the result is irreversible; therefore, be sure you want to delete every selected entity before invoking the function. If it is invoked when there is no selected entity, the following error message is displayed:

```
ERROR, nothing picked
```

2.7.9 Object Tools + delete all

This is the only **Object Tools** function which does not require that entities be selected first. It operates on everything. The following warning query is displayed when **Object Tools + delete all** is selected:

```
This will delete everything, Are You Sure (n)?
```

Either: press ↵ to abort;

Or: enter Y↵ to delete everything.

Another way to delete everything is to exit from EXAMINE^{3D} without saving anything.

2.8 Modeler Menu Item: View

loc & dist
eye + target
pan
zoom in
auto box mode
autoscale

The **View** menu opens access to a group of functions which enable the user to resize or relocate the view windows relative to the model.

File	Toolbox	Build Polyline	Build Object	Pick	Xform	Object Tools	View	Shade	Analysis Param	Field Points	Return

2.8.1 View + autoscale

This function resizes and relocates all four windows to *just fit* and *center* the model, or a portion of the model enclosed by a user-defined box. It works like the $\boxed{A}$-button at the top of each window, except that each $\boxed{A}$-button operates on only one window, whereas **View + autoscale** operates on all four windows.

2.8.2 View + auto box mode

This function enables the user to control the **autoscale** function by selecting a portion of the model to be brought into focus whenever **autoscale** is invoked. The following message is displayed when **View + auto box mode** is selected:

```
Choose Mode (0=ALL GEOMETRY, 1=STRESS POINTS, 2=USER DEFINED)?
```

Option 0, activated by entering $\boxed{0}$↵, causes each window to be resized and relocated to just fit and center the entire model. The option remains in effect, and determines the result of the **autoscale** function each time, until another **auto box mode** is chosen.

Option 1, activated by entering $\boxed{1}$↵, causes each window to be resized and relocated to just fit and center all *user-specified stress points* (see **Field Points** menu). The option remains in effect, and determines the result of the **autoscale** function each time, until another **auto box mode** is chosen.

Option 2, activated by entering $\boxed{2}$↵, causes each window to be resized and relocated to just fit and center a *user-defined box*. The user is prompted to locate two corners of the box as follows:

```
Locate box corner #1 [N,U,E: snaps]:
```

Either: type the N-U-E coordinates of the first corner, followed by ↵;

Or: click on a point to select it as the first corner of the box.

Adjust the coordinates of the point, either by typing or with the mouse, then click on Go to accept it. The dialogue box updates to request the second corner, as follows:

```
Locate box corner #2 [N,U,E: snaps]:
```

Either: type the N-U-E coordinates of the second corner, followed by ⏎;

Or: click on a point to select it as the second corner of the box.

Once a second corner is entered, the outline of the box is shown in yellow. Adjust the box by adjusting the second corner. When it is satisfactory, click on Go to accept it.

While using the mouse to enter coordinates, recall that only two coordinates can be modified through any one of the orthogonal view windows, and that all three coordinates can be modified through any two windows (see Section 1.4.1.3 for more details).

This option remains in effect, and determines the result of the autoscale function each time, until another **auto box mode** is chosen.

2.8.3 View + zoom in

The **zoom in** function enlarges a selected fraction of a window to the full size of the window, thereby exposing more details of the part of the model visible through the enlarged fraction. The user is prompted to select the part of a window to be enlarged, by picking two (diagonally opposite) corners, as follows:

```
Pick region to ZOOM IN on, Select first corner of region:
```

Click on a point to select it as the first corner. Choose the point carefully before clicking; it cannot be changed once picked, except by pressing ESC to abort the function. Once a point is clicked, the dialogue box message updates to request the second point, as follows:

```
Pick opposite corner of region:
```

A rectangle, with one corner anchored at the first point, grows as the mouse pointer is moved away. When the rectangle correctly marks the region to **zoom in** on, click one more time. The rectangle is enlarged to occupy the full size of the window. Because **zoom in** maintains the aspect ratio of the model, the enlarged rectangle may be slightly different from the one selected.

To restore the full window, click on the A-button; the result obtained depends on the current **auto box mode**.

2.8.4 View + pan

This function moves a window relative to the model, by an amount and in a direction specified by the user. It is similar to the arrow buttons at the top of each window. The user is prompted to specify the **pan** vector as follows:

```
Select Base Point:
```

Click anywhere in the window of interest.

```
To Point:
```

A line, anchored at the first point, grows as the mouse pointer is moved away. Grow the line in the direction the model is required to "move" (relative to the window). When the length and direction of the line correctly represent the required relative displacement, click a second time.

The window is moved in the opposite direction of the specified vector, which gives the appearance of the model being moved in the specified direction. To restore the original position of the window:

Either: use the **pan** function again, with an equal and opposite vector;

Or: use the arrow buttons at the top of the window;

Or: click on the A-button; the result obtained depends on the current **auto box mode**.

2.8.5 View + eye+target

The **eye+target** function enables the user to *walk around* the model and examine it from any conceivable viewpoint. It works with all four view windows: the location of the eye relative to the model is displayed through the orthogonal view windows, whereas the model, as would be seen from the current viewpoint, is displayed in the perspective window.

The following message is displayed when **View + eye+target** is selected:

```
Shade Objects in Perspective Window (n):
```

The process is slower if objects are shaded. On the other hand, because shading blocks off the backplanes, a clearer view is presented when objects are shaded.

Either: press ↵ to accept "no shading",

Or: enter Y↵ to activate shading.

```
Please Wait...
```

Each of the orthogonal windows is expanded, thereby shrinking the model down to a "target size"; the current eye position is marked with a "blue camera", connected to the target by a greenish yellow "line of sight"; the picture of the model, as seen from the current eye position, is displayed in the perspective window. To modify the viewpoint:

Either: type the coordinates of the required new eye location, followed by ↵;

Or: click on a point to relocate the eye there. Recall that only the N-U coordinates can be modified through the right view window, E-U coordinates through the front view window, and E-N coordinates through the top view window.

Having selected a desired viewpoint, it can be saved using **Toolbox + store status**; hence, at any subsequent stage in the model analysis, the saved viewpoint can be recalled using **Toolbox + retrieve status**.

To exit from **View + eye+target**, click on [Go]. The current viewpoint remains effective until modified using either **View + eye+target** or **Shade + animate**, or until the current session of EXAMINE^{3D} is terminated.

2.8.6 View + loc & dist

This function is used to both locate a point in your three-dimensional modeling space and to determine a distance to some other user-defined point. This function starts by displaying the coordinates of point#1 in the dialogue box, and locates the same point in all four view windows. The user is prompted to identify the required point, as follows:

```
P1 ( [current point] )[N,U,E: snaps]:
```

Either: enter [*coordinates*]↵, and the point corresponding to the [*coordinates*] will be marked in all four windows;

Or: click on any point, in any one window; the coordinates of the point will be given as the [*current point*] in the dialogue box, and the point will be marked in every window, labeled with its coordinates; recall that the snap functions constrain the category of points selectable with the mouse (see Section 1.5.1 for a description of the snap functions); for example, only model vertices can be picked if [vn=on] .

After you have defined the first point, click on [Go]. You are then prompted to enter the second point as follows:

```
P2 ( [current point]:[distance] )[N,U,E: snaps]:
```

As above, define the second point. You will notice that a line is drawn from the first point defined to the second point. The distance from the first point to the second point is written after the coordinates of the second point in the dialogue box.

To exit from **View + loc & dist**, press ESC at any point during the above procedure.

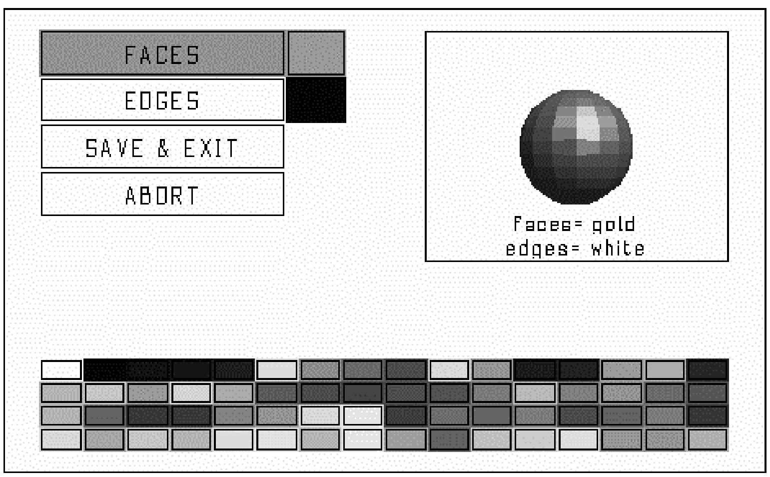
2.9 Modeler Menu Item: Shade

set color
shade options
animate
quickshade

The **Shade** menu opens access to a set of functions for visualizing the over-all model, either shaded or in wireframe mode. Two of these functions, **set color** and **shade options**, are used to set values of parameters for the visualization.

File	Toolbox	Build Polyline	Build Object	Pick	Xform	Object Tools	View	Shade	Analysis Param	Field Points	Return

2.9.1 Shade + set color



The **set color** function opens a sub-menu which enables the user to assign different colors to objects. The function does not operate on individual elements or components, which implies that all elements belonging to one object must be assigned the same color. To assign a color to an object, first select it using **Pick**; then select **Shade + set color** to obtain the sub-menu shown at left.

To assign a color to the element faces, click on **FACES**, then click on the desired color in the color palette. The selected color is displayed in the box on the right of **FACES**; at the same time the sample box at the extreme right is updated to show the effect of the selection.

Click on **EDGES**, and follow the same procedure to assign a color to the element edges.

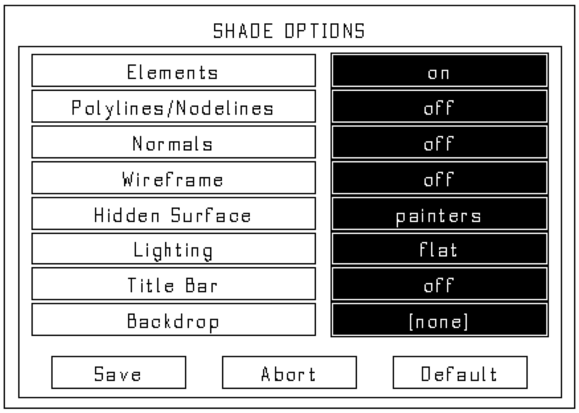
To exit from **Shade + set colors**:

<u>Either</u>: click on **SAVE & EXIT**, to effect the changes and exit;

<u>Or</u>: click on **ABORT** to exit without effecting any changes; the color settings in effect before **Shade + set color** was invoked are restored.

Element faces and edges should be assigned different colors, if they are required to be distinguishable. In the **Interpreter**, **transparency** can also be defined for the picked entity.

2.9.2 Shade + shade options



The SHADE OPTIONS sub-menu, obtained by selecting **Shade + shade options**, enables the user to control the appearance of, and the amount of information included in, shaded models. The default values for the parameters are shown in the sketch on the left.

Click on **Elements** to toggle elements on (default) or off. When elements are off, shaded models display only the geometry of the models; both geometry and boundary element discretization are displayed when elements are on.

Click on **Polylines/Nodelines** to toggle polylines and nodelines off (default) or on.

Set **Normals** to on, and arrows showing the directions of surface normals will be included in shaded models; set **Normals** to off (default), to turn off the arrows. If **Normals** is set to pressure, only the normals associated with elements having a traction boundary condition are displayed.

A shaded view cannot be obtained if **Wireframe** is set to on. The setting is used to force the **animate** function to work in the wireframe mode; otherwise, it works in the shaded mode.

The **Hidden Surface** parameter toggles between painters (default), software, zsort, and hidden line. Hidden surface removal with the zsort option is faster than, but not as accurate as, the other two options. The software hidden surface algorithm uses a software z-buffer to render the shaded image. This produces a more visually pleasing image but requires considerably more memory. The software option is also used for the animation of the model (**Shade + animate**) since it supports paging/double-buffering of the display. The painters algorithm provides the best balance between speed, memory usage and image quality. The hidden line option performs hidden line removal.

The **Lighting** parameter toggles between flat (default) and off in the modeler. Shading is obviously faster when **Lighting** is set to off but much less interesting. In the Interpreter, more advanced rendering options are available which allow for the smooth shading (Gouraud, Phong) of the geometry and data visualization entities. If you wish to capture images of your geometry, it's best to use the facilities in the **Interpreter** which produce more visually pleasing results. Without results, use the **Interpreter + General** option and when asked for a file name just press ↵ , then retrieve the file using the **File** menu.

If **Title Bar** is set to on, you will obtain an expanded title bar (which replaces the entire menu and information bars) when **quickshade** is invoked, or when the screen dump function is invoked (by pressing Ctrl-O). If **Title Bar** is set to off (default), the title bar is limited to the dialogue box.

When you choose to shade your model, the **Backdrop** by default is the background color. If you wish to add a tiled image as the background, create a CompuServe GIF image file and place it in the textures directory in your default EXAMINE^{3D} installation directory. Select the **Backdrop** option and choose the image file you wish to use as the background image and shade your model using the **quickshade** or **animate** option**.** To turn off the backdrop, answer no when prompted as to whether you want a textured background. You will notice that EXAMINE^{3D} comes with a variety of image files that you can use.

To exit from **Shade + shade options**:

Either: click on **Save** to effect the current settings and exit;

Or: click on **Abort**; the settings prior to invoking **Shade + shade options** will be restored.

2.9.3 Shade + animate

The **animate** function tumbles and rotates the perspective view window relative to the model, which has the appearance of tumbling and rotating the model (in the perspective window). The model is shaded, unless the **Wireframe** option in the SHADE OPTIONS sub-menu (Section 2.9.2) is set to on.

The tumbling and rotations proceed faster in *wireframe* mode than in *shaded* mode. The following information is displayed while the **animate** function is in progress:

```
h=shade, arrows/Ins/Del=change dir, Enter=stop, Esc=abort
```

2.9.3.1 Changing Between Shade and Wireframe Modes

Pressing $\boxed{\texttt{h}}$ causes the model to be shaded if the wireframe mode is on; or to return to wireframe mode if it is already shaded.

2.9.3.2 Direction Control

The Ins, Del, and arrow buttons on the keyboard enable the user to control the direction of tumbling and rotation; or to restart the tumbling and rotation if temporarily stopped (see below).

2.9.3.3 Temporary Stop

The tumbling and rotation can be stopped temporarily (i.e., without exiting from **animate**) by pressing ↵. To restart, press any of the Ins, Del, or arrow buttons on the keyboard.

To exit from **animate**, press ESC. This freezes the current viewpoint (i.e., eye location relative to model). The viewpoint may be saved using **Toolbox + save status**; the **View + eye+target** function may be used to re-set the viewpoint.

2.9.3.4 Full Screen Mode

By pressing Ctrl-F, the menus are removed with just the animating model remaining. To return to the previous view containing the menu system, press Ctrl-F again.

2.9.4 Shade + quickshade

This function presents a shaded view of the model. The shaded view is presented in only one window at a time, starting with the perspective window (by default); click anywhere in a given window to obtain a shaded view through it.

The **Wireframe** parameter (see Section 2.9.2) must be set to off in order to obtain a shaded view using **quickshade**.

To capture the screen to an image file, press Ctrl-I or to a postscript file, press Ctrl-O. If you wish to toggle the model to full screen mode (removes the menu system), use the Ctrl-F function.

To exit from **Shade + quickshade**, press ESC.

2.10 Modeler Menu Item: Analysis Param

compute3d stats
analysis options
stress block off
job description
enter parameters

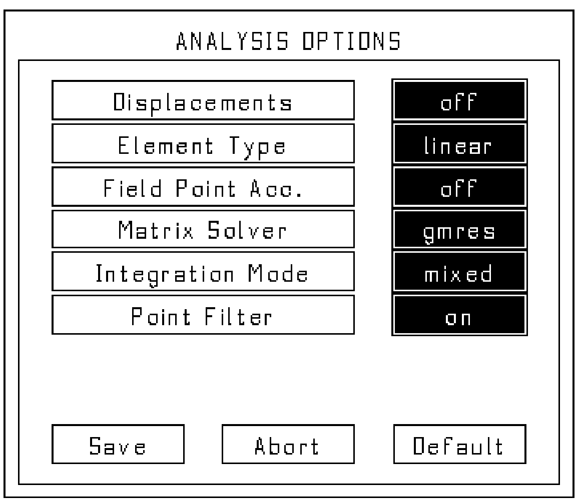
The **Analysis Param** menu opens access to functions used to set values of material parameters and in situ stress, which are required to define the analysis task for COMPUTE$^{3D\text{-}BEM}$. The functions also enable the user to choose between a series of analysis options.

File	Toolbox	Build Polyline	Build Object	Pick	Xform	Object Tools	View	Shade	Analysis Param	Field Points	Return

2.10.1 Analysis Param + compute3d stats

This function estimates and displays the amount of memory required by COMPUTE$^{3D\text{-}BEM}$ to analyze the model using the existing boundary element discretization.

2.10.2 Analysis Param + analysis options



The **analysis options** function opens access to the ANALYSIS OPTIONS sub-menu, from which the user can select a series of analysis options. The default options are shown in the sketch on the left. Other options are explained below.

2.10.2.1 Displacements

Toggle displacements on (or off), to include (or exclude) the computation of displacements. If it is set to off, then COMPUTE$^{3D\text{-}BEM}$ will not compute displacements, and no displacement data will be available in the results (.RES) file.

2.10.2.2 Element Type

Elements can be set to constant, linear, or quadratic, to obtain boundary elements in which displacement is constant, varies linearly, or varies quadratically, respectively, over the element surface. All elements are triangular.

Constant elements have one node each, located at the element centroid. Therefore, the number of nodes is equal to the number of elements. Because displacements are computed at the interior point, compatibility of element edges need not be enforced.

Linear elements have three nodes each, located at the corners of the triangle. Because nodes are shared by neighboring elements, the total number of nodes in an object is a lot less than the number of elements. Therefore, the number of equations for a linear element mesh is less than the number for the corresponding constant element mesh.

Quadratic elements have mid-side nodes in addition to the corner nodes of linear elements. The addition of mid-side nodes may cause the total number of nodes to increase by as much as the number of elements. Therefore, although quadratic elements may be more accurate than linear elements, the increased computational cost may likely outweigh the benefit in accuracy.

2.10.2.3 Field Point Acc.

This option toggles **Field Point Acc**eleration on or off. When it is on, field point calculations are performed for *every other plane* of a grid box, and for *every other line* of a cutting plane; results at the omitted planes and lines are obtained by interpolation. When it is off (the default), field point computations are performed for *every plane* of a grid box, and *every line* of a cutting plane.

If **Field Point Acc**. is set to on, the number of divisions along each axis of a grid box or cutting plane must be even.

2.10.2.4 Matrix Solver

The options for matrix solver are:

j/bi-cg (the default)	Jacobi-biconjugate gradient
gmres	Generalized minimum residual
conj grad	Conjugate gradient
gaus seid	Gauss-Seidel iteration
gaus elim	Gaussian elimination

The option is set by clicking repeatedly on **Matrix Solver** until the rectangular box on its right displays the required method.

2.10.2.5 Integration Mode

This toggles between mixed (the default), in which both numerical and exact integration are used, and exact, in which only exact integration is used.

2.10.2.6 Point Filter

The Point Filter parameter has three possible values: ON, SURFACE, or OFF. When it is set to ON, points which fall on the inside of excavations are ignored by COMPUTE$^{3D\text{-}BEM}$ and **Interpret**. When it is set to SURFACE, points which fall on the outside of free surfaces are also ignored. If it is set to OFF, no point is filtered; that is, every point would be included in data analysis and interpretation, irrespective of its location relative to a free surface or an excavation.

2.10.2.7 More on the ANALYSIS OPTIONS Sub-Menu

The **Default** button resets every parameter to its default value. To exit from **Analysis Param + analysis options**:

<u>Either</u>: select **Save**, to effect the current settings and exit;

<u>Or</u>: select **Abort**, to exit without effecting any change; the parameter settings before the function was invoked are restored.

Settings made at the ANALYSIS OPTIONS sub-menu may be over-ridden using COMPUTE$^{3D\text{-}BEM}$ command line options.

2.10.3 Analysis Param + stress block off/on

This function has two settings: on or off. When it is on, a cross showing the directions and relative magnitudes of the principal in situ stress components is displayed in each of the orthogonal view windows. The red arm of the cross indicates the direction of the maximum principal in situ stress, green arm indicates the intermediate, and the blue arm indicates the minimum principal stress.

2.10.4 Analysis Param + job description

This function prompts the user to assign a title to the model, as follows:

```
Enter Job Description:
```

Enter [*job title*] ↵

The [*job title*] is displayed in the title bar when **Shade + quickshade** is invoked, or when the screen dump function is invoked (by pressing Ctrl-O). The default job title is “EXAMINE 3D -- A 3D BOUNDARY ELEMENT STRESS ANALYSIS PROGRAM”. An expanded title bar (which occupies the entire position of the menu and information bars) can be obtained by setting **Title Bar** to on (via **Shade + shade options**, described in Section 2.9.2).

The **Interpret** version of this function, described in Section 3.7.4, allows the user to specify the character size for the job title.

2.10.5 Analysis Param + enter parameters

This function enables the user to set the values of elastic constants, field stress, and strength parameters. A sketch of the MODEL PARAMETERS sub-menu, obtained when **Analysis Param + enter parameters** is selected, is shown in Figure 2.3. Each menu item is represented by a *name box* (green) and a *value box* (white).

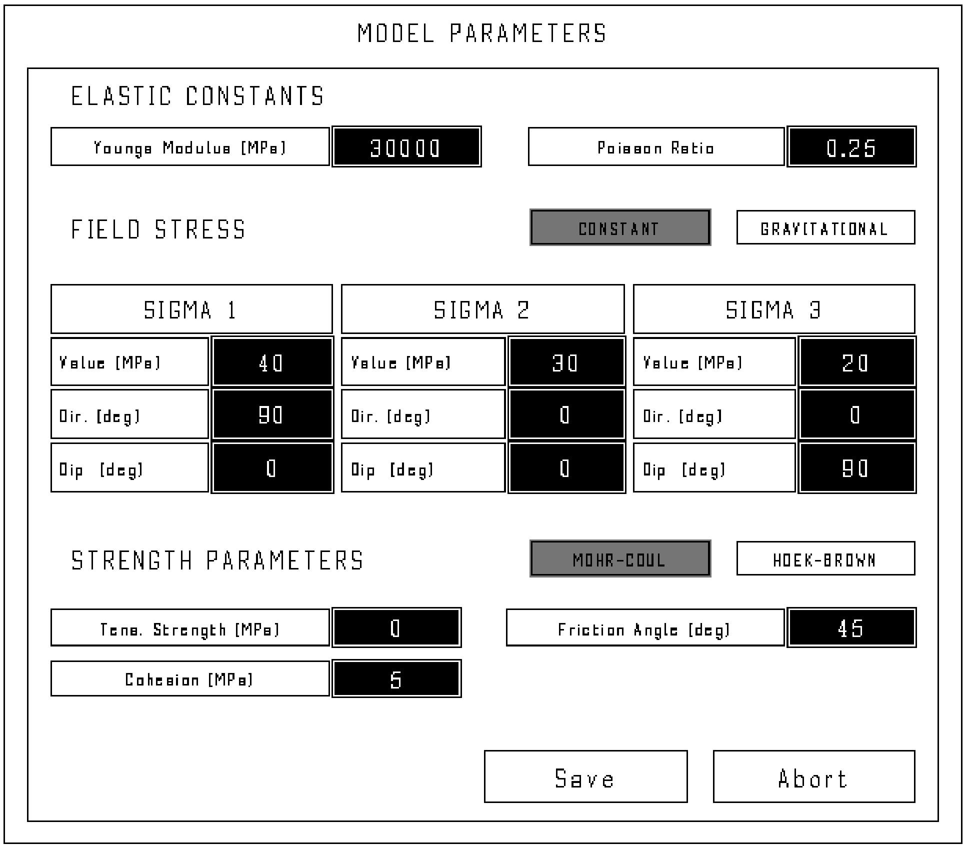


Figure 2.3: The Model Parameters sub-menu

2.10.5.1 Elastic Constants

Elastic constants are defined in terms of Young's modulus (in MPa) and Poisson's ratio. To change the value of Young's modulus, click on the white box (value box) beside **Young's Modulus (MPa)**, type the required value, and press ↵. To restore the existing value after clicking on the value box, press ↵ without typing a new value.

The value of Poisson's ratio is set in a similar way.

2.10.5.2 Field Stress: Constant Option

The values of field stress can be set either in terms of constant values of the principal stress components (CONSTANT option), or in terms of the values of stress gradient components (GRAVITATIONAL option). The constant option, obtained by clicking on CONSTANT, is the default.

When the CONSTANT option is in effect, data entry fields are displayed for the three principal stress components, named SIGMA 1, SIGMA 2 and SIGMA 3. Each stress component is defined in terms of three parameters, as follows:

1. its magnitude (value) in MPa;

2. its direction (Dir) in degrees, measured clockwise from North (0°) and ranging in value from 0°-360°;

3. its Dip, i.e., inclination from the horizontal, measured in degrees, and ranging from 0°-90°.

Each of the parameters has a name box and a value box. To change the value of a parameter, click on its value box, type the required value, and press ↵. To restore the existing value after clicking on the value box, press ↵ without typing a new value.

2.10.5.3 Field Stress: Gravitational Option

The gravitational option is selected by clicking on GRAVITATIONAL. It presents data entry fields for the following parameters:

1. surface elevation in metres, i.e., the value of U-coordinate at which the magnitude of vertical stress is zero;

2. unit weight in MN/m^3;

3. the direction (Dir), in degrees, of three stress components named SIGMA V, SIGMA H1 and SIGMA H2; directions are measured clockwise from North (0°), and range in value from 0°-360°;

4. the dip (Dip) of each of the stress components; dip is measured from the horizontal, and it ranges from 0°-90°;

5. the stress ratios K1 (equal to the ratio of SIGMA H1 to SIGMA V) and K2 (equal to the ratio of SIGMA H2 to SIGMA V).

6. the locked-in far field stresses in the horizontal H1 and H2 directions. These stresses correspond to the magnitude of the horizontal stresses at the ground surface elevation.

In equation form, the gravitational field stresses are thus defined as:

$$\sigma_v = \gamma h$$
$$\sigma_h = \sigma_o + K\gamma h$$

where,

σ_v = vertical far field stress
σ_h = horizontal far field stress
σ_o = locked - in horizontal far field stress
γ = unit weight of the rock mass
h = depth below surface
K = horizontal stress ratio

Each of the parameters has a name box and a value box. To change the value of a parameter, click on its value box, type the required value, and press ↵. To restore the existing value after clicking on the value box, press ↵ without typing a new value.

2.10.5.4 Strength Parameters: Mohr-Coulomb Option

Rock strength is defined in terms of either the Mohr-Coulomb or Hoek-Brown strength criteria. The Mohr-Coulomb option, obtained by clicking on MOHR-COUL, presents data entry fields for:

1. tensile strength in MPa;
2. friction angle in degrees; and
3. cohesion in MPa.

Each parameter has a name box and a value box. To change the value of a parameter, click on its value box, type the required value, and press ↵. To restore the existing value after clicking on the value box, press ↵ without typing a new value.

2.10.5.5 Strength Parameters: Hoek-Brown Option

The Hoek-Brown option, obtained by clicking on HOEK-BROWN, presents data entry fields for:

1. unconfined compressive strength of intact rock, in MPa;
2. parameter *m*; and
3. parameter *s*.

Each parameter has a name box and a value box. To change the value of a parameter, click on its value box, type the required value, and press ↵. To restore the existing value after clicking on the value box, press ↵ without typing a new value.

2.10.5.6 Exiting from the Model Parameters Sub-Menu

<u>Either</u>: select **Save**, to effect the current settings and exit;

Or: select **Abort** to exit without effecting any changes; the values of all parameters before **Analysis Param + enter parameters** was invoked are restored.

2.11 Modeler Menu Item: Field Points

intern vis off
edge vis on
edit points
delete points
add points

The **Field Points** menu opens access to functions which enable the user to specify the points at which analysis results should be computed. Boundary element analysis does not automatically compute material response in the medium around the structure of interest. Such computation is performed only if specifically requested, and only at the points requested.

File	Toolbox	Build Polyline	Build Object	Pick	Xform	Object Tools	View	Shade	Analysis Param	Field Points	Return

2.11.1 Field Points + add points

EXAMINE^{3D} provides six methods for defining field points; when **Field Points + add points** is selected, the user is prompted to select one of the methods, as follows:

```
Enter Input Method (1=plane,2=3pt plane,3=grid,4=file,5=pt,6=line) [default]:
```

2.11.1.1 Input Option 1: Cutting Plane

This option allows the user to define the field points in terms of a plane, which is specified by giving the coordinates of two diagonally opposite corners. To select this option, enter 1 ↵ in response to the above message. The following is displayed, along with the Go buttons:

```
Input plane point #1 [N,U,E: snaps]:
```

Either: type the coordinates of the first corner, followed by ↵;

Or: click on any point to place the first corner there.

The selected point is marked with a red star; its location can be modified, either by entering alternative coordinates through the keyboard, or by clicking with the mouse. If using the mouse, notice that the coordinate snap functions are available; as was explained in Section 1.5.1, they constrain the category of points selectable with the mouse. Furthermore, recall that only two coordinates can be modified through any one (and all three coordinates can be modified through any two) of the orthogonal view windows.

When the selected point is satisfactory, click on Go to accept it. The dialogue box updates to request the second corner, as follows:

```
Input plane point #2 [N,U,E: snaps]:
```

Enter the coordinates of the second corner, either from the keyboard, or with the mouse. The rectangle formed by the two points (corners) is displayed; it may be adjusted by modifying the coordinates of the second point, following the procedure described for the first point.

When the rectangle is satisfactory, click on [Go] to accept it. Its color changes to red; then one of its edges is highlighted yellow, and the following message is displayed:

```
Enter number of divisions in u direction [default]:
```

The rectangle is oriented with respect to a local coordinate system; one of its edges corresponds to the local u-axis, and a second perpendicular one corresponds to the local v-axis. The local coordinate axes are not assumed to be coincident with the global (N-U-E) system. Field points will be generated by discretizing the rectangle to form a network of grid points. The above dialogue box message requests the user to specify the number of grid divisions for the highlighted (local u-axis) direction.

Enter [*nu*] ↵, where *nu* is the required number of divisions.

```
Enter number of divisions in v direction [default]:
```

Enter [*nv*] ↵, where *nv* is the required number of divisions. The number of field points so defined is therefore (nu + 1) * (nv + 1).

The outline of the cutting plane is displayed in red. The function **Field Points + add points** is terminated. It will have to be selected again, if it is required to define more field points.

2.11.1.2 Input Option 2: Cutting Plane (3 points)

This option allows the user to define the field points in terms of a plane, which is specified by giving the coordinates of any three different points. To start, select **Field Points + add points**, which causes the following message to be displayed:

```
Enter Input Method (1=plane,2=3pt plane,3=grid,4=file,5=pt,6=line) [default]:
```

Then enter [2] ↵; the following message appears, along with the [Go] buttons:

```
Input plane point #1 [N,U,E: vn=off, s=off, o=off]:
```

Either: type the coordinates of the first corner, followed by ↵;

Or: click on any point to place the first corner there.

The location of the point can be adjusted, either by entering alternative coordinates from the keyboard, or by clicking with the mouse (taking advantage of the snap functions and multi-view windows, as was explained above). When the selected point is satisfactory, click on [Go] to accept it. The dialogue box updates to request the second point, as follows:

```
Input plane point #2 [N,U,E: vn=off, s=off, o=off]:
```

Enter the coordinates of the second point, either from the keyboard, or with the mouse. A yellow line is displayed, joining the second point to the first one. The location of the second point can be adjusted, using the same procedure described above for the first one. When the point is satisfactory, click on [Go] to accept it. The dialogue box updates to request the third point, as follows:

```
Input plane point #3 [N,U,E: vn=off, s=off, o=off]:
```

Enter the coordinates of the third point. The rectangle defined by the three points is displayed. Its size and orientation can be adjusted by relocating the third point. When the rectangle is satisfactory, click on [Go] to accept it. The following message is displayed:

```
Enter number of divisions in u direction [default]:
```

The discretization of the cutting plane into a grid network follows the procedure described in the previous section. Respond by giving the required number of grid divisions in the u and v directions of the plane. After the v direction grid is accepted, the outline of the plane is displayed and the function **Field Points + add points** is terminated.

2.11.1.3 Input Option 3: Grid Box

This option provides for the field points to be defined in terms of a box, specified by giving the coordinates of two diagonally opposite corners. The field points are generated by discretizing the box into a grid network.

Start by selecting **Field Points + add points**, to obtain the following message:

```
Enter Input Method (1=plane,2=3pt plane,3=grid,4=file,5=pt,6=line) [default]:
```

Then enter [3] ↵; the following message appears, along with the [Go] buttons:

```
Locate box corner #1 [N,U,E: vn=off, s=off, o=off]:
```

Either: type the coordinates of the first corner, followed by ↵;

Or: click on any point to place the first corner there.

The location of the point can be adjusted by entering alternative coordinates, either through the keyboard or with the mouse. When it is satisfactory, click on [Go] to accept it. The dialogue box updates to request the second corner, as follows:

```
Locate box corner #2 [N,U,E: vn=off, s=off, o=off]:
```

Enter the coordinates of the second corner, either through the keyboard or with the mouse. The outline of the rectangular box formed by the two points is displayed. The size and orientation of the box can be adjusted by entering alternative coordinates for the second corner. There is no restriction on its size or orientation, except that its volume must be non-zero. The box can also be rotated using the right mouse button.

When the box is satisfactory, click on [Go] to accept it. The user is then prompted to discretize the box into grids, as follows:

```
Enter number of divisions in u direction [default]:
```

Enter the required number of divisions, followed by ↵; respond the same way for the v and w directions. After the w value is accepted, the outline of the box is displayed and the function **Field Points + add points** is terminated.

2.11.1.4 Input Option 4: Points File

This option provides for the field points to be read in as free points, from an ASCII file, named *filename*.PTS. The points file consists of sets of three numbers, separated by spaces, each set occupying a line. Each set of three numbers locate a point in space, according to the coordinate system used in the file. If the coordinate system differs from the EXAMINE^{3D} system, it must be identified using **File + coord transform** before invoking this input option (see Section 2.1.4).

Start by selecting **Field Points + add points**; the following message is displayed:

```
Enter Input Method (1=plane,2=3pt plane,3=grid,4=file,5=pt,6=line) [default]:
```

Then enter 4 ↵; Use the dialog to navigate to the folder containing the points file. Then select it or type in the name. All the points read are displayed as red points, spheres, boxes, diamonds, or strips, depending on the current setting for markers (Section 2.11.3.1).

2.11.1.5 Input Option 5: Point

This option provides for the field points to be defined individually as discrete points in space. Each point is entered, one at a time. This option is useful for a few points where stresses and/or displacements are wanted, but is not intended for large numbers of points. Use the above points file method for this case.

Start by selecting **Field Points + add points**, to obtain the following message:

```
Enter Input Method (1=plane,2=3pt plane,3=grid,4=file,5=pt,6=line) [default]:
```

Then enter 5 ↵; the following message appears, along with the Go buttons:

```
Select/Enter Point Location [N,U,E: vn=off, s=off, o=off]:
```

<u>Either</u>: type the coordinates of the point, followed by ↵;

<u>Or</u>: click on any point to place the point there.

The location of the point can be adjusted by entering alternative coordinates, either through the keyboard or with the mouse. When it is satisfactory, click on Go to accept it.

2.11.1.6 Input Option 6: Line

This option provides for the field points to be defined in terms of a line, specified by giving the coordinates of two points in space. The field points are generated by discretizing the line into a user defined set of equally spaced intervals. This option may be useful for comparing model results with extensometer results.

Start by selecting **Field Points + add points**, to obtain the following message:

```
Enter Input Method (1=plane,2=3pt plane,3=grid,4=file,5=pt,6=line) [default]:
```

Then enter 6 ↵; the following message appears, along with the Go buttons:

```
P1 ( [current point] ) [N,U,E: snaps]:
```

Either: enter [*coordinates*]⟵, and the point corresponding to the [*coordinates*] will be marked in all four windows;

Or: click on any point, in any one window; the coordinates of the point will be given as the [*current point*] in the dialogue box, and the point will be marked in every window, labeled with its coordinates; recall that the snap functions constrain the category of points selectable with the mouse (see Section 1.5.1 for a description of the snap functions); for example, only model vertices can be picked if `vn=on` .

After defining the first point, click on `Go`. You are then prompted to enter the second point as follows:

```
P2 ( [current point]:[distance] ) [N,U,E: snaps]:
```

As above, define the second point. You will notice that a line is drawn from the first point to the second point. The distance from the first point to the second point is written after the coordinates of the second point in the dialogue box. After you have defined the second point, click on `Go`. You are then prompted to enter the discretization as follows:

```
Enter number of divisions along line [default]:
```

Enter the required number of divisions, followed by ⟵; after the value is accepted, the points along the line at the discretization intervals are displayed and the function **Field Points + add points** is terminated.

2.11.1.7 Terminating Field Points + add points

The function **Field Points + add points** can be terminated either by completing a selected input task (at which the function terminates normally), or by pressing ESC (at which the uncompleted task is discarded).

2.11.2 Field Points + delete points

This function permits the user to delete any or all of the following:

1. all the points read in from a .PTS file;
2. individual planes, from among those defined using one of the cutting plane options;
3. individual grid boxes;
4. individual points; or
5. points defined using a line

The following message is displayed when the function is selected:

```
Select plane/grid/pts to delete [*=all, ESC=abort]:
```

Either: press `*` to delete all grid boxes, cutting planes and free points;

Or: click on any grid box, cutting plane, or free point to delete it; clicking on one free point causes all free points to be deleted

When all the required points, planes and boxes have been deleted, press ESC to exit.

2.11.3 Field Points + edit points

This function enables the user to modify the grid discretization of cutting planes and grid boxes, or to change the type and size of symbols used to mark the free points (those read from .PTS files).

The following is displayed when the function is selected:

```
Select plane/grid/pts to edit [Esc=abort]:
```

2.11.3.1 Editing Free Point Markers

To change the type and size of free point markers, click on one of the free points (in response to the above message). The following is displayed:

```
Enter Marker Type (0=point, 1=diamond, 2=box, 3=sphere, 4=strip) [default]:
```

Type the required number (0,1,2,3 or 4), followed by ↵;

```
Enter scale factor [default]:
```

Enter $\boxed{[SF]}$ ↵, where [*SF*] is the factor by which each marker should be scaled. Values larger than 1 enlarge, and those smaller than 1 reduce, the size of each marker. Thereafter, the markers are modified according to the specification, and the function **Field Points + edit points** is terminated.

2.11.3.2 Re-discretization of Cutting Planes and Grid Boxes

Start by selecting **Field Points + edit points**, to obtain the following message:

```
Select plane/grid/pts to edit [Esc=abort]:
```

Click on the required cutting plane or grid box. One of its edges will be highlighted, and the following message is displayed:

```
Enter number of divisions in u direction [current value]:
```

<u>Either</u>: type the new value and press ↵

<u>Or</u>: press ↵ to retain the current value

The same prompt appears for the v direction, and (for grid boxes) the w direction. Respond in the same way. The function is terminated thereafter.

2.11.4 Field Points + edge vis on/off

This function toggles on or off when selected, to turn on or off the edges of all cutting planes and grid boxes. The default setting is on.

2.11.5 Field Points + intern vis off/on

This function toggles on or off when selected, to turn on or off the internal grid lines of all cutting planes and grid boxes. It is set to off by default.

2.12 Modeler Menu Item: Return

File	Toolbox	Build Polyline	Build Object	Pick	Xform	Object Tools	View	Shade	Analysis Param	Field Points	Return

The **Return** function returns control to the welcome screen (Figure 1.1). At the welcome screen:

Either: select **Modeler** to open the modeler screen;

Or: select **Interpret** to open the interpret screen;

Or: select **Exit**; then select **Yes** to exit from EXAMINE^{3D}.

Ensure that your work is saved, if necessary, before exiting.

3. Interpret

The EXAMINE^{3D} group of functions for the interpretation of analysis results are accessed by selecting **Interpret** from the WELCOME screen (Figure 1.1). This causes one of two things to happen, depending on the status of the parameter “general” (last line in the configuration file e3.cfg, which resides in the EXAM3D directory).

If general=on (the default), then **Interpret** opens the DATASETS screen, which presents three possible datasets: **Stress**, **Seismic** and **General**, as illustrated in the menu bar below.

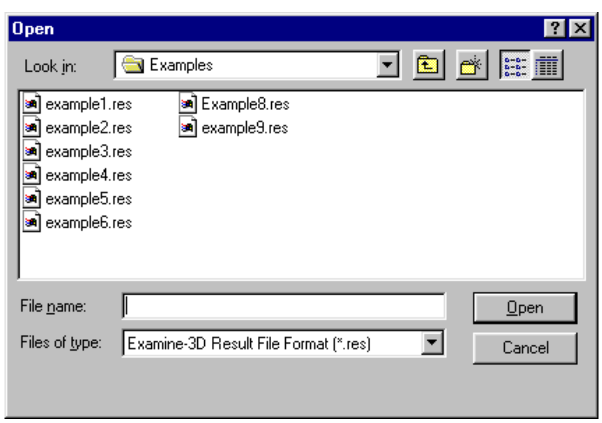
				Stress →	Seismic →	General →					Return

Either: select **Stress**, to interpret COMPUTE$^{3D\text{-}BEM}$ analysis results, stored in a .RES file;

Or: select **Seismic** to interpret seismic data;

Or: select **General** to interpret any other kind of data, stored in a .DAT file, following the general format described in Chapter 5.

If general=off, then only .RES files can be interpreted. In that case the DATASETS screen is not presented; instead, selecting **Interpret** at the WELCOME screen causes the following dialog to be displayed:



Use the dialog to navigate to the folder containing the EXAMINE^{3D} data file. Then select it or type in the name. For a *filename* to be acceptable, the files *filename*.EX3 and *filename*.RES must both reside in the same subdirectory.

A message is displayed indicating that the file is being read. Thereafter, the **Interpret** DATA SELECTION screen (Figure1.4) is presented. The menu bar from this screen is illustrated below.

Analysis Param	Sigma 1 →	Sigma 2 →	Sigma 3 →	Sig. 1-3 →	1+2+3 /3 →	Stress Tensor	Str. Fac. →	Ubiq. Joints	Displmnt	User Defined	Return

The DATA SELECTION screen presents a choice of the following results variables to be interpreted:

Sigma 1	Maximum principal compressive stress σ_1
Sigma 2	Intermediate principal stress σ_2
Sigma 3	Minimum principal compressive stress σ_3
Sig. 1-3	Maximum shear stress $\sigma_1 - \sigma_3$
(1+2+3)/3	Mean principal stress
Str. Fac.	Strength/Load factor, a measure of overstress and failure

To select any of these results variables, click on its name; a message is displayed indicating that the data is being read (or calculated). Thereafter, the **Interpret** DATA INTERPRETATION screen (Figure 1.5) is presented, thereby opening access to the EXAMINE^{3D} interpretation group of functions. Each of the menu items will be described subsequently.

The **Interpret** DATA SELECTION screen also offers access to five menu items, viz., **Analysis Param**, **Stress Tensor, Ubiq. Joints, Displmnts**, and **User Defined** each of which opens a pop-up menu. The use of these menu items will be described first.

3.1 Interpret Menu Item: Analysis Param

analysis options
job description
enter parameters

This menu item is available at the DATA SELECTION screen. It opens access to three functions, shown in the pop-up menu on the left. The functions are similar to those offered by the **Modeler** version of **Analysis Param**; but the **Interpret** version has a more limited scope.

Analysis Param	Sigma 1 →	Sigma 2 →	Sigma 3 →	Sig. 1-3 →	1+2+3 /3 →	Stress Tensor	Str. Fac. →	Ubiq. Joints	Displmnt	User Defined	Return

3.1.1 Analysis Param + analysis options

This function opens the same sub-menu as was shown in Section 2.10.2 for the **Modeler** version. It permits an examination of the current settings; but **no change** is permitted.

Use **Abort** to exit from the sub-menu.

3.1.2 Analysis Param + job description

This function works exactly as was described in Section 2.10.4

3.1.3 Analysis Param + enter parameters

This function opens the MODEL PARAMETERS sub-menu (Figure 2.3). It permits the user to modify the values of strength parameters (which affect the **Str Fac** data), but **no other change** is permitted. Please see Section 2.10.5.4 and Section 2.10.5.5 for a description of how to enter values for the strength parameters.

3.2 Interpret Menu Item: Ubiq. Joints

joint properties
normal stress -->
shear stress -->
joint F.S. -->
combined F.S. -->

This menu item is available at the DATA SELECTION screen. It opens access to functions which enable the user to interpret stress data with respect to the properties of an omnipresent joint. One of the pop-up menu items (**joint properties**) opens a sub-menu; the others (each of which has a right-pointing arrow by its name) represent results variables. The DATA INTERPRETATION screen (Figure 1.5) is opened when any of the results variables is selected.

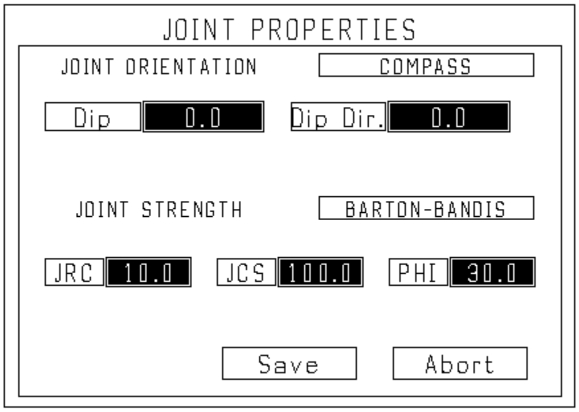
Analysis Param	Sigma 1 →	Sigma 2 →	Sigma 3 →	Sig. 1-3 →	1+2+3 /3 →	Stress Tensor	Str. Fac. →	Ubiq. Joints	Displmnt	User Defined	Return

The following results variables are available:

normal stress	normal stress on the ubiquitous joint
shear stress	resultant shear stress on the ubiquitous joint
joint F.S.	factor of safety against slip on the ubiquitous joint
combined F.S.	the smaller of **joint F.S**. and **Str Fac**

Any of these variables can be interpreted using functions accessed through the DATA INTERPRETATION screen, which will be described subsequently.

3.2.1 Ubiq. Joints + joint properties



This function opens the JOINT PROPERTIES sub-menu, at which the user can specify the values of orientation and strength parameters for the ubiquitous joint.

3.2.1.1 Joint Orientation: Compass Option

This is the default option for describing the orientation of joints. It is obtained by clicking on CARTESIAN (if the Cartesian option is on). Two parameters, **Dip** and **Dip Dir**, need to be specified. The parameter **Dip** is the inclination of the joint plane from the horizontal. Its values range from 0° to 90°. The parameter **Dip Dir** is the dip direction, measured in degrees from North. Its values range from 0° to 360°.

To change the value of a parameter, click on its value box (white), type the required value, and press ↵.

3.2.1.2 Joint Orientation: Cartesian Option

The Cartesian option is obtained by clicking on COMPASS (if the Compass option is on). In this option, the orientation of a joint is described in terms of the components of its normal vector. The components are named X, Y, and Z. The X-component coincides with the N-axis, the Y-component with the U-axis, and the Z-component with the E-axis.

To change the value of a component, click on its value box, type the required value, and press ↵.

3.2.1.3 Joint Strength: Barton-Bandis Option

This is the default option for specifying the strength of a joint. It is obtained by clicking on MOHR-COUL (if the Mohr-Coulomb option is on). The following parameters (Barton's indices) need to be specified:

JRC	Joint roughness coefficient
JCS	Joint compressive strength (MPa)
PHI	Basic friction angle of rock surface (degrees)

To change the value of a parameter, click on its value box, type the required value, and press ↵.

3.2.1.4 Joint Strength: Mohr-Coulomb Option

This option is obtained by clicking on BARTON-BANDIS (if the Barton-Bandis option is on). It requires the strength of a joint to be specified in terms of the Mohr-Coulomb parameters:

C	the cohesion intercept (MPa)
PHI	friction angle (degrees)

To change the value of a parameter, click on its value box, type the required value, and press ↵.

3.2.1.5 Exiting from Ubiq. Joints + joint properties

To exit from the JOINT PROPERTIES sub-menu

Either:	click on **Save** to effect all the changes and exit;
Or:	click on **Abort** to discard all changes and exit; the values of all parameters prior to invoking the function are restored.

3.3 Interpret Menu Item: Stress Tensor

sigma nn(xx) -->
sigma uu(yy) -->
sigma ee(zz) -->
sigma nu(xy) -->
sigma ne(xz) -->
sigma ue(yz) -->

The **Stress Tensor** menu is available at the DATA SELECTION screen. It opens access to the six independent three-dimensional stress tensor components calculated by COMPUTE$^{3D\text{-}BEM}$.

Analysis Param	Sigma 1 →	Sigma 2 →	Sigma 3 →	Sig. 1-3 →	1+2+3 /3 →	Stress Tensor	Str. Fac. →	Ubiq. Joints	Displmnt	User Defined	Return

The available tensor variables are:

sigma nn(xx)	stress in the north (compass), or x (cartesian) direction
sigma uu(yy)	stress in the up (compass), or y (cartesian) direction
sigma ee(zz)	stress in the east (compass), or z (cartesian) direction
sigma nu(xy)	shear stress on a plane ⊥ to north in the up direction
sigma ne(xz)	shear stress on a plane ⊥ to north in the east direction
sigma ue(yz)	shear stress on a plane ⊥ to up in the east direction

The DATA INTERPRETATION screen is opened when any of the tensor variables is selected.

3.4 Interpret Menu Item: Displmnts

east (z) -->
up (y) -->
north (x) -->
total -->

The **Displmnts** menu is available at the DATA SELECTION screen. It opens access to four displacement variables, if the parameter **Displacements** in the ANALYSIS OPTIONS sub-menu (Section 2.10.2) is set to on (or the computation of displacements was requested via the command line options for COMPUTE$^{\text{3D-BEM}}$).

Analysis Param	Sigma 1 →	Sigma 2 →	Sigma 3 →	Sig. 1-3 →	1+2+3 /3 →	Stress Tensor	Str. Fac. →	Ubiq. Joints	Displmnt	User Defined	Return

The available displacement variables are:

east (z)	E-component of displacement
up (y)	U-component of displacement
north (x)	N-component of displacement
total	Resultant displacement vector

The DATA INTERPRETATION screen is opened when any of the variables is selected.

3.5 Interpret Menu Item: User Defined

Analysis Param	Sigma 1 →	Sigma 2 →	Sigma 3 →	Sig. 1-3 →	1+2+3 /3 →	Stress Tensor	Str. Fac. →	Ubiq. Joints	Displmnt	User Defined	Return

With the **User Defined** option, the user is no longer limited to viewing principal stresses and displacements only. With this option, the user can create contour plots of:

1. any stress tensor component (also available in the **Stress Tensor** menu)
2. any displacement component (also available in the **Displmnts** menu)
3. write a mathematical expression containing these components, and view the results of this function

The **User Defined** option is customized through the EXAMINE$^{\text{3D}}$ configuration file **e3.cfg**, which is supplied for this purpose.

You will find the e3.cfg file in your installation directory. Take a moment to view this file using any standard ASCII text editor. You will find that it already contains information which will create a customized user menu. Read on for further information.

3.5.1 EXAMINE$^{\text{3D}}$ Configuration File – e3.cfg

The format of the e3.cfg file is simple. If there is an asterisk (*) in the first column of a line, then the line is a **comment**. Otherwise the line contains configuration **information**.

The first few sections of the file contain configuration settings for default colors and default EXAMINE^{3D} operating characteristics. The user is referred to the comments within the file for complete descriptions of these parameters. At the end of the file you will find the configuration settings for **User Defined** data.

3.5.2 User Defined Functions

The following is a list of **variables** available to the user:

```
s1  = major principle stress
s2  = intermediate principle stress
s3  = minor principle stress
snn = stress in north direction (or cartesian x direction)
suu = stress in up direction (or cartesian y direction)
see = stress in east direction (or cartesian z direction)
snu = shear stress on a north face in the up direction
sne = shear stress on a north face in the east direction
sue = shear stress on a up face in the east direction
sn  = normal stress on a ubiquitous joint, orientaion of the joint
      is defined in the joint properties menu in the ubiquitous
      joint popup in the interpreter.
ss  = maximum shear stress on a ubiquitous joint, orientaion of the joint
      is defined in the joint properties menu in the ubiquitous
      joint popup in the interpreter.
dn  = displacement in north direction
du  = displacement in up direction
de  = displacement in east direction
ym  = young’s modulus
g   = shear modulus = ym/(2*(1+pr))
pr  = poisson ratio
m   = hoek-brown failure criterion m
s   = hoek-brown failure criterion s
ucs = hoek-brown failure criterion unconfined compressive strength
pi  = 3.1415926
t   = mohr-coulomb failure criterion tensile strength
phi = mohr-coulomb failure criterion tensile friction angle in radians
c   = mohr-coulomb failure criterion cohesive strength
```

The following is a list of the basic mathematical **operators** supported:

```
+ = addition
- = subtraction
/ = division
* = multiplication
^ = to the power of (ie. 2^3 = 8)
```

The following **intrinsic functions** are available for the calculation of trigonometric and other quantities:

```
acos(x)  = arc cosine of expression x
asin(x)  = arc sine of expression x
atan(x)  = arc tangent of expression x
cosh(x)  = hyperbolic cosine of expression x
sinh(x)  = hyperbolic sine of expression x
tanh(x)  = hyperbolic tangent of expression x
cos(x)   = cosine of expression x
sin(x)   = sine of expression x
tan(x)   = tangent of expression x
log10(x) = log base 10 of expression x
exp(x)   = exponential e to the x
log(x)   = natural logarithm of expression x
sqrt(x)  = square root of expression x
abs(x)   = integer absolute of expression x
fabs(x)  = floating point absolute of expression x

Note:  all angles must be expressed in radians
```

The user is allowed to specify a maximum of 6 functions, using the above **variables**, **operators** and **intrinsic functions**. Each function will be represented by a “button” in the **User Defined** menu – 5 functions have been defined in the e3.cfg file supplied with EXAMINE^{3D}.

To define a function, simply define the “button label” and then define the function, in the following format:

```
option_?_title = “<button label>“
option_? = <mathematical expression>
```

where the ? = the number of the button from 1–6.

For example, the **User Defined** functions supplied in your e3.cfg file when you installed EXAMINE^{3D}, were created by including the following lines in the e3.cfg file:

```
option_1_title = “Max. Shear”
option_1 = (s1-s3)/2.0
option_2_title = “Strain Energy”
option_2 = (s1^2 + s2^2 + s3^2 - 2.0*pr*(s1*s2+s2*s3+s1*s3))/(2.0*ym)
option_3_title = “Major Strain”
option_3 = (((1.0-pr^2)*s1) - (pr*(1.0+pr)*s3))/ym
option_4_title = “Minor Strain”
option_4 = (((1.0-pr^2)*s3) - (pr*(1.0+pr)*s1))/ym
option_5_title = “Excess Shear”
option_5 = (1.0+(sn*tan(0.523598776)))-fabs(ss)
```

NOTE:
1. the lines preceded by an **asterisk** (*) are comment lines
2. the “**button labels**” must be enclosed by quotation marks, as shown
3. the **functions** are NOT enclosed by quotation marks

3.6 Interpret Menu Item: File

print
export image
coord transform
append to model
save file

This menu opens access to a group of five file management functions. They work in the same way as the **Modeler File** group of functions, which are described in Section 2.1.

File	Toolbox	Volume Data	Cutting Plane	Contour Tools	Marker Tools	Pick	Object Tools	View	Shade	Field Points	Return

3.6.1 File + print

Please see Section 2.1.6 for a description of this function.

3.6.2 File + export image

Please see Section 2.1.5 for a description of this function.

3.6.3 File + coord transform

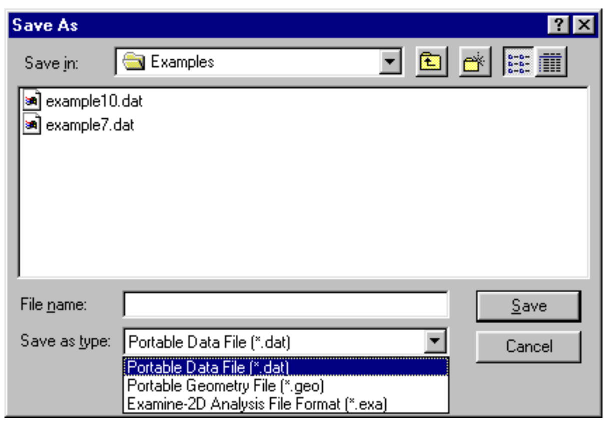
Please see Section 2.1.4 for a description of this function.

3.6.4 File + append to model

This function works exactly as was described in Section 2.1.3. It permits the user to append data from a EXAMINE^{3D} .EX3, Portable Geometry .GEO or Autocad .DXF file to the current database. The structure described in the appended files would usually not be a part of (or the same as) the one already in the dataset being interpreted. It is important to note that in the Interpreter you can directly append the geometry in an Autocad DXF file. This may be useful in visualizing the results since the appended geometry can be used as a reference to known structures that were not modeled.

3.6.5 File + save file

The following dialog is displayed when **File + save file** is selected:



As shown in the figure you may store a file in one of three file formats:

If you choose to store a Portable Data File the program will extract and store a .DAT (general data) file. Each .DAT file lists C1-C2-C3-s values; where C1-C2-C3 are the coordinates of a point (according to the prevailing coordinate system) and s is the value of the variable at the C1-C2-C3 point. The *current variable*, i.e., the one named at the bottom left corner of the screen, is the one stored. General data files can be interpreted by selecting the **General** option at the DATASETS screen (this will only be available if the parameter general=on in the e3.cfg file – see the beginning of this chapter for details).

If you select a portable geometry file, an ascii file containing just the geometry is written. The purpose of this file is to allow the transfer of geometry from/to other programs. The format of the file is described in Section 5.6.

You may also write an EXAMINE^{2D} file. You will be prompted to select a cutting plane containing the data you wish to store to the .EXA EXAMINE^{2D} file. EXAMINE^{2D} may then be used to view the data. As well, you may use the model to perform a two-dimensional plane strain analysis. You can use this information to determine whether a two-dimensional analysis will correctly model the stress distribution around your structure.

Use the dialog to navigate to the folder to store the data file. Then select it or type in the name.

3.7 Interpret Menu Item: Toolbox

setup options
insert text
delete text
job description
store status
retrieve status

The **Interpret** version of **Toolbox** offers access to six miscellaneous functions. Three of these, viz., **setup options**, **store status**, and **retrieve status**, work in much the same way as was described for the **Modeler** version in Section 2.2 with a few exceptions listed below.

File	Toolbox	Volume Data	Cutting Plane	Contour Tools	Marker Tools	Pick	Object Tools	View	Shade	Field Points	Return

3.7.1 Toolbox + setup options

The **Interpret** version of **setup options** provides all the functions described in Section 2.2.3; in addition, it also allows the user to modify the name of the *current results variable* (displayed at the bottom left corner of the DATA INTERPRETATION screen, Figure 1.5).

The **Interpret** version of **setup options** also provides the user with the ability to embed the **contour legend** for planes and surface contours into the upper left hand corner of any of the four view windows. During full-screen mode (enabled by pressing Ctrl-F during **quickshade** or **animate**), the contour legend will still be visible in the upper corner of the window.

3.7.2 Toolbox + insert text

The **insert text** function is used to insert text strings.

Start by selecting **Toolbox** + **insert text**, which causes the following message to be displayed:

```
Enter text string [default]:
```

Type the text string, followed by ↵

```
Enter Text Size [default]:
```

The text size is given as a fraction of the height of the screen. For example, to obtain text about 2% of the screen height, enter `0.02` ↵.

```
Place Text on Black Backgound (y)?
```

A black background is recommended, for improved readability. Accept “yes” by pressing ↵

```
Select Text Location: [N,U,E: vn=off, s=off, o=off ]:
```

The text location can be defined by typing the coordinates, followed by ⏎, or directly with the mouse. For the latter, click on the required point; the text can be relocated by clicking on a different point, or by dragging the mouse with the left button held down. The text is centered at the point selected; when its location is satisfactory, click on Go to accept it.

Once it is accepted by clicking Go, it can no longer be changed in any way, except by deleting it.

3.7.3 Toolbox + delete text

This function is used to delete text strings placed using the **insert text** function. The user is prompted as follows to pick the text to be deleted (having selected **Toolbox + delete text**):

```
Select text to delete [*=all, ESC=abort]:
```

Either: enter * ⏎ to delete all text at once;

Or: click on a specific text string to delete it;

Or: press ESC to terminate the function without deleting more text.

3.7.4 Toolbox + job description

This function offers more flexibility than its **Modeler** version, described in Section 2.10.4, in that it allows the user to specify the character size for the job title.

The user is prompted to supply the character size, as follows:

```
Enter Title Bar Text Size [default]:
```

The character size is given as a fraction of the height of the screen. For example, to obtain characters about 2% of the screen height, enter 0.02 ⏎.

Thereafter, the function proceeds as was described in Section 2.10.4

3.7.5 Toolbox + store status

The **Interpret** version of **store status** does a little more than its **Modeler** version, in that it also stores any text entered using the **insert text** function (described above).

The magnitudes of any defined isosurfaces are also stored. As a result, the isosurfaces corresponding to these magnitudes are automatically created for any model in which this status file is retrieved.

3.8 Interpret Menu Item: Volume Data

surf. contours
traj. ribbons
store isosurf
retrieve isosurf
create isosurf

This menu opens access to five functions for full three-dimensional examination of data. Three of the functions operate on isosurfaces, one displays the directions of vector variables, whereas the other displays contours on the surfaces of structures.

File	Toolbox	Volume Data	Cutting Plane	Contour Tools	Marker Tools	Pick	Object Tools	View	Shade	Field Points	Return

3.8.1 Volume Data + create isosurface

An isosurface is a surface joining all points in space (within the region of available data) at which a given variable has the same value (referred to as the *isovalue*). For example, the σ_1 = 60 MPa isosurface is the surface joining all points at which the variable σ_1 has a value of 60 MPa.

This function enables the user to set up one isosurface at a time. The following is displayed when **Volume Data + create isosurface** is selected:

```
Enter isovalue:
```

Type the required isovalue, and press ↵ (e.g., `60.0` ↵)

In wireframe mode, an isosurface is displayed as interconnected polygons. Each polygon is displayed as it is generated; at the end the following message is displayed:

```
isosurface successfully created [#polygons = [nump] ]
```

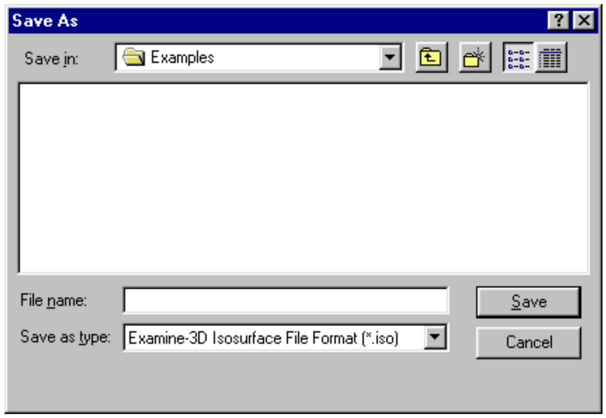
At the same time the Isosurface Legend (right end of the screen, above the Contour Legend) is updated to include information for the current isosurface. Like contours, many isosurfaces can be displayed simultaneously. An isosurface can be **Pick**ed (using **Pick + isosurface**, described later); the edges and faces of the polygons can be assigned colors (just like elements), using **Shade + set color**; furthermore, an isosurface can be shaded, and the data describing it can be stored in, and retrieved from, a file.

3.8.2 Volume Data + store isosurface

This function writes isosurface data to a file named *filename*.ISO. The following is displayed when the function is selected:

```
Pick isosurface to store:
```

Click anywhere on the isosurface. It is highlighted (yellow), indicating that is has been selected; thereafter the user is prompted to enter the *filename* as follows:

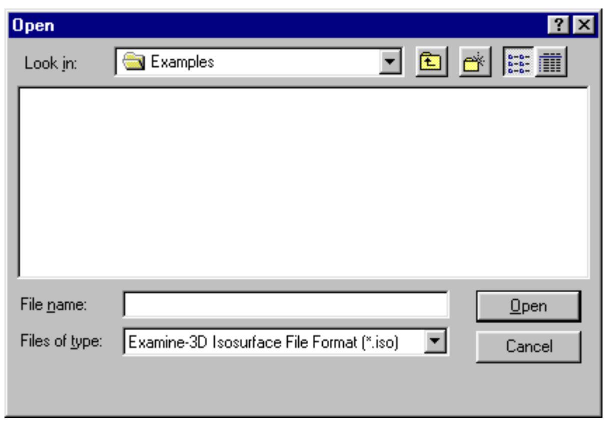


In choosing a *filename*, bear in mind that each isosurface contains data for only one value of one variable.

3.8.3 Volume Data + retrieve isosurface

This function reads data from an .ISO file, and incorporates the data into the current database; the isosurface is then displayed as part of the existing data.

The following is dialog is displayed when the function is selected:



Enter the *filename*, either through the keyboard or with the mouse.

3.8.4 Volume Data + traj ribbons

A *trajectory ribbon* is a tool for displaying the spatial variation of the three principal stress directions, all at the same time. It consists of a "ribbon" passed through selected points, such that its orientation represents the principal stress directions, as follows:

The length dimension of the ribbon coincides with the direction of the maximum principal compressive stress (σ_1); the width dimension coincides with the direction of the intermediate principal stress (σ_2); and the normal to the ribbon coincides with the direction of the minimum principal compressive stress (σ_3).

The **traj ribbons** function draws either trajectory ribbons or *trajectory polylines*. Unlike ribbons, polylines have no width; therefore, trajectory polylines represent only the direction of the maximum principal stress.

In addition to showing the principal stress directions, trajectory ribbons and polylines are colored to represent the magnitudes of the current stress component, according to the Contour Legend. They are treated like contours, and are thus operated on by any function that operates on contours (see **Contour Tools**).

Each ribbon or polyline is constructed starting from a *tracer input point*. When **Volume Data + traj ribbons** is selected, the user is prompted to specify, first the tracer input method, and next the density of input points, as follows:

```
Enter Tracer Input Method (1,2=line; 3,4=plane)[default]:
```

3.8.4.1 Tracer Input Option 1

In this option **ribbons** are grown **from** input points distributed along **a line**. Having selected this option, the user is prompted to locate the line and specify the input point density, as follows:

```
Enter first point [N,U,E: vn=off, s=off, o=off]:
```

Enter the first end of the line by

Either: typing the N-U-E coordinates, followed by ⏎,

Or: clicking with the mouse.

Adjust the coordinates as necessary, using the above procedure. If using the mouse, recall that the coordinate snaps (Section 1.5.1) constrain the category of points selectable with the mouse; recall also that only two coordinates can be modified through any one (and all three coordinates through any two) of the orthogonal view windows. When the location of the point is satisfactory, click on [Go] to accept it.

```
Enter second point [N,U,E: vn=off, s=off, o=off]:
```

Enter the second end of the line, following the procedure described for the first. The line is displayed as soon as a second point is entered. Adjust the point as necessary; then click on [Go] to accept the line.

```
Enter number of intervals [default]:
```

Enter [Num] ⏎, where [*Num*] + 1 is the required number of tracer input points.

3.8.4.2 Tracer Input Option 2

Input Option 2 works exactly like Option 1, except for the difference that Option 2 generates trajectory polylines, whereas Option 1 generates ribbons.

3.8.4.3 Tracer Input Option 3

In this case, trajectory **ribbons** are grown **from** input points distributed on **a plane**. Having selected this option, the user is prompted to locate the plane and specify the input point density, as follows:

```
Input plane point #1[N,U,E: vn=off, s=off, o=off]:
```

The plane is defined in terms of two diagonally opposite corners. Enter the coordinates of the first corner, either with the mouse or through the keyboard, following the procedure described under *Input Option 1*. When the location of the point is satisfactory, click on Go to accept it.

```
Input plane point #2[N,U,E: vn=off, s=off, o=off]:
```

Enter the coordinates of the second corner; the outline of the plane is displayed as soon as a second corner is entered. Adjust the point as necessary; when the plane is satisfactory, click on Go to accept it.

```
Enter number of divisions in u direction [default]:
```

Enter *[n1]* ↵

where *[n1]* + 1 is the required number of input points per line parallel to the highlighted edge of the plane.

```
Enter number of divisions in v direction [default]:
```

Enter *[n2]* ↵

The total number of input points is therefore ($n1$ + 1) * ($n2$ + 1). Because the ribbons "grow" in the direction of σ_1, tracer input planes *normal* to the general σ_1 direction will display more information.

3.8.4.4 Tracer Input Option 4

Input Option 4 works exactly like Option 3, except for the difference that Option 4 generates trajectory polylines, whereas Option 3 generates ribbons.

3.8.5 Volume Data + surface contours

This function displays contours of the current variable on the surface of the structure. For example, it would display contours on the roof, floor, and sidewall surfaces, and on the two end faces, of a tunnel.

The function does not require any user-input.

3.9 Interpret Menu Item: Cutting Plane

visible
invisible
trajectories off
cutting plane

This menu opens access to four functions used to set up and manipulate cutting planes.

File	Toolbox	Volume Data	Cutting Plane	Contour Tools	Marker Tools	Pick	Object Tools	View	Shade	Field Points	Return

3.9.1 Cutting Plane + cutting plane

This function is used to set up cutting planes through a grid box. Unlike those set up using the **Modeler** function **Field Points + add points**, these cutting planes are not permitted to extend beyond the boundaries of the grid box; furthermore, their orientation is limited to parallel and perpendicular to the grid box boundary planes. Please see Section 2.11.1 for more discussion on cutting planes and grid boxes.

To start, select **Cutting Plane + cutting plane**. If there are two or more grid boxes, the user is prompted to select the required one, as follows:

```
Select grid or cutting plane:
```

Click on the required grid box to select it. Thereafter, the following query is displayed (the function begins with this query if there is only one grid box):

```
Autoscale Grid Box (n)?
```

<u>Either</u>: enter Y↵, and each window will be re-sized and relocated to just fit and center the grid box;

<u>Or</u>: just press ↵, and the current **auto box mode** (see Section 2.8.2) will remain in effect.

Thereafter, the intrinsic coordinate axes of the grid box are displayed, highlighted yellow and named u, v and w. These axes need not coincide with the global axes; w-axis parallels the long dimension of the box, u-axis the width dimension, and v-axis parallels the third dimension. The outline of a proposed plane is also displayed (green dashed line); along with the following message, indicating the available options:

```
snap=s, sweep=w, shade=h, Go=done; [c1:c2:c3]
```

The numbers *[c1:c2:c3]* in the message are the global (N-U-E) coordinates of the point of intersection of the current cutting plane with one of the local coordinate axes (yellow) of the grid box; there is only one such point for every cutting plane (among those formed using the **cutting plane** function); the *[c1:c2:c3]* information is updated each time the cutting plane is relocated.

The h -option toggles the structure between the wireframe and shaded modes; this option is recommended. Ensure that **Wireframe** is set to off, in the SHADE OPTIONS sub-menu (Section 2.9.2).

The s -option toggles on and off; when it is on, only planes passing through actual data points can be selected; when it is off, any plane can be selected, interpolation being used to obtain data on planes which do not pass through actual data points. Please note that this snap option is not the same as that described in Section 1.5.1.

The w -option toggles the sweep mode on or off; when it is on the cutting plane is always contoured to show the data variation at its current location; when it is off, only the outline of the cutting plane is displayed. Irrespective of the status of the sweep option, a cutting plane is contoured after its location is confirmed by clicking on Go.

To set up a cutting plane, click anywhere on one of the local grid box axes (yellow); the cutting plane normal to and intersecting the clicked axis at the clicked point is displayed. If the sweep mode is on, the cutting plane is contoured; otherwise, only its outline is displayed. The location and orientation of the cutting plane may be changed by clicking on the required axis at the required point. It is always normal to the clicked axis.

To exit from **Cutting Plane + cutting plane**:

Either: click on Go to retain the current cutting plane and exit; the cutting plane is contoured, irrespective of the status of "sweep";

Or: press ESC to discard the cutting plane and exit.

If the first exit option is used, the cutting plane so-formed remains in place until it is deleted using **Contour Tools + delete all**. It may be used to examine other results variables (i.e., by using **Return** to re-open the DATA SELECTION screen, and selecting another variable). Data describing the plane may be saved for EXAMINE^{2d}, using **File + store 2D**.

Only one cutting plane can be formed each time **Cutting Plane + cutting plane** is invoked; but several cutting planes formed at different times can coexist.

3.9.2 Cutting Plane + trajectories off/on

This function toggles on or off. When it is off, cutting planes display contours of magnitudes of the current variable; no direction-related information is included. When it is on, cutting planes display direction information for vector variables; also, the direction indicators are colored to represent magnitudes, according to the Contour Legend.

3.9.2.1 Trajectories and Principal Stresses

If the current variable is a principal stress component, the default direction indicators are *trajectory plates* oriented as described in Table 3.1. As the table shows, trajectory plates are always oriented with their long dimension in the direction of the current principal stress component (current variable). This is unlike trajectory ribbons (Section 3.8.4), which are always oriented with their long dimension in the direction of the maximum principal stress component.

Table 3.1: Relationship between the orientation of trajectory plates and the directions of the principal stress components

Current Variable	Long direction of plate	Width direction of plate	Normal to plate surface
σ_1	σ_1	σ_2	σ_3
σ_2	σ_2	σ_1	σ_3
σ_3	σ_3	σ_1	σ_2

The shape of the stress trajectories may be modified using **Contour Tools + traj edit**, which allows a choice of one of three shapes, viz., strips (i.e., trajectory plates), arrows, or sticks. The arrows option gives a series of short arrow-plates oriented according to Table 3.1. The sticks option converts the trajectories to lines oriented in the direction of the current principal stress component.

3.9.2.2 Trajectories and Displacements

If the current variable is the displacement vector, the direction indicators consist of a series of short arrows pointing in the direction of the vector. If it is one of the displacement components, the arrows point in the direction of the component.

3.9.2.3 Trajectories and Ubiquitous Joints

If the current variable is either the shear stress or normal stress on the joint, the direction indicators consist of short arrow-plates pointing in the potential direction of shear; the normal to the arrow surface coincides with the normal stress direction.

3.9.3 Cutting Plane + invisible

This function hides selected cutting planes. The user is prompted to select the cutting planes, as follows:

```
Select plane to hide [*=all, Esc=abort]:
```

Either: enter $\boxed{*}$ ↵ to hide all cutting planes;

Or: click on a cutting plane to hide it; hide as many as desired, then press ESC to exit from the function.

Pressing the $\boxed{*}$ button makes all the planes invisible but does not exit this option. By selecting the red outline of any plane, it will toggle the plane between visible and invisible. If the plane is invisible, selecting it will make it visible, while if it is visible, selecting it will make it invisible. Using this method, planes can be toggled off and on very easily.

3.9.4 Cutting Plane + visible

This function restores all hidden cutting planes at once, without prompting for any user-input.

3.10 Interpret Menu Item: Contour Tools

lights off
delete all
traj. edit
alter color
alter range

This menu opens access to a group of functions for modifying the appearance of contours and trajectories.

File	Toolbox	Volume Data	Cutting Plane	Contour Tools	Marker Tools	Pick	Object Tools	View	Shade	Field Points	Return

3.10.1 Contour Tools + alter range

The contour range is defined by the minimum value *[vmin]* and the maximum value *[vmax]*. These are the values at the top and bottom, respectively, of the Contour Legend. The difference (*[vmax]* **-** *[vmin]*) is divided by 7 to obtain the size of each contour interval.

Usually *[vmin]* and *[vmax]* should be chosen to just include the full range of values for the current variable; but at times they are chosen to focus on a selected range of data values. The smaller the range, the more detailed the contour representation of the data.

The user is prompted for *[vmin]* and *[vmax]*, as follows:

```
Enter min,max range, [default]:
```

Either enter `[vmin] , [vmax]` ↵;

Or: press ↵ to accept the [*default*].

All contours and the Contour Legend are updated to reflect the new range.

3.10.2 Contour Tools + alter color

This function permits the user to select a desired color for each contour interval. The following is displayed when the function is selected:

```
Select Legend Color Box To Alter [d=default]:
```

To alter the color of a contour interval, click repeatedly on its color box in the Contour Legend, until the desired color is displayed in the box. Modify as many as desired.

To restore the default color for all contour intervals, press `d` .

To exit from the function:

Either: click on Go; all contours will be updated to reflect the new color selections;

Or: press ESC, to exit without effecting any change; the color settings prior to invoking the function are restored.

3.10.3 Contour Tools + traj. edit

This function allows the user to choose from three possible shapes for stress trajectories. The choice made here affects the stress trajectories obtained using the function **Cutting Plane + trajectories off/on**. It has no effect on the direction indicators for displacement or ubiquitous joint stress vectors (see Section 3.9.2.2 and Section 3.9.2.3). It also has no effect on trajectory ribbons (see Section 3.8.4).

The following is displayed when **Contour Tools + traj. edit** is selected:

```
Enter Marker Type (1=strip, 2=arrow, 3=stick) [default]:
```

Option 1 (strip) gives trajectory plates, oriented according to Table 3.1, in Section 3.9.2.

Option 2 (arrow) represents each trajectory with a series of arrow-plates, oriented according to Table 3.1.

Option 3 (stick) represents each trajectory as a line oriented in the direction of the current stress component.

To select an option, type the corresponding number and press ↵; the following warning query is displayed:

```
This will change all markers, Are You Sure (n)?
```

Either: enter Y↵, and all trajectories will be modified to reflect the choice;

Or: press ↵ to discard the choice; the type of trajectory in effect prior to invoking the function will be restored.

```
Enter trajectory scale factor [current value]:
```

Type the required value of scale factor, and press ↵. Values larger than 1 cause enlargement of the trajectories. Note that the space between the trajectory elements is not affected by the scaling. For example, if a large enough scale factor is applied with the arrow option, the arrow-plates join up to form trajectory plates; as the factor gets larger, the space between the plates disappears, giving the appearance of contours (with rough boundaries).

The **trajectories off/on** toggle (Section 3.9.2) must be set to on, in order to observe the effects of any changes made with **Contour Tools + traj. edit**.

3.10.4 Contour Tools + delete all

This function deletes all contours, at once. No user-input is requested; so be sure that you want to delete every contour before invoking the function.

3.10.5 Contour Tools + lights off/on

The lighting option is used for determining whether cutting planes, isosurfaces, trajectory ribbons, and surface contours are shaded using the current light source.

3.11 Interpret Menu Item: Marker Tools

marker edit
retrieve markers
enter point

The use of diamonds, boxes, or spheres to mark free data points was described earlier (Section 2.11.3.1). Points attached to structures can be marked as well. This menu opens access to a group of functions used to set and manipulate markers (for both free and attached points).

File	Toolbox	Volume Data	Cutting Plane	Contour Tools	Marker Tools	Pick	Object Tools	View	Shade	Field Points	Return

3.11.1 Marker Tools + enter point

This function permits the user to locate and mark individual points. The user is prompted for the coordinates of the point, as follows:

```
Select/Enter Point Location [N,U,E: vn=off, s=off, o=off]:
```

Either: type the coordinates of the required point, and press ↵;

Or: click on a point, to select it.

The selected point is marked with a red star. Adjust its coordinates as required, either through the keyboard or with the mouse. If using the mouse, recall that the snap functions (Section 1.5.1) constrain the category of points selectable with the mouse; also, recall that only two coordinates can be modified through any one (and all three coordinates can be modified through any two) of the orthogonal view windows.

When the location of the point is satisfactory, or if you should decide to discard it,

Either: click on Go to accept the point; a marker (of the current type and size) is placed at the location;

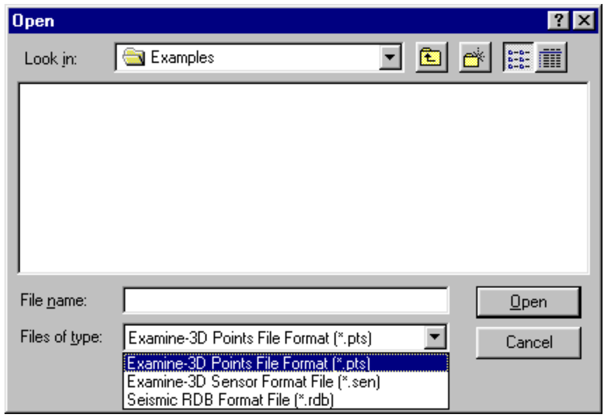
Or: press ESC to discard the point.

3.11.2 Marker Tools + retrieve markers

This function reads the coordinates of a group of points from a file, and places a marker (of the current type and size) at each of the points. Any of the following three file types may be read:

.RDB file	special seismic data file
.SEN file	special seismic data file
.PTS file	free points file; please see Section 2.11.1.4

The user is asked to select the file type and the filename through the following dialog:



Enter the *filename*, either through the keyboard or with the mouse. The points will be displayed as they are read in.

3.11.3 Marker Tools + marker edit

This function, like the **Modeler** version described in Section 2.11.3.1, permits the user to choose the type and size of symbols for marking points. Any change made here is applied to every marked point, including those located subsequently (using either **Marker Tools + enter point** or **Marker Tools + retrieve markers**).

The user is prompted to specify the type and size of marker, as follows:

```
Enter Marker Type (1=diamond, 2=box, 3=sphere)[default]:
```

To select a symbol, type its number and press ↵.

```
This will change all markers, Are You Sure (n)?
```

Either: enter Y↵; all markers will be changed to the selected type *at the current size*, before proceeding;

Or: press ↵; the marker type in effect prior to invoking the function will be restored; then

```
Enter scale factor [1]:
```

The current size of the symbols will be multiplied by the scale factor to set the new size. Type the required value of scale factor, and press ↵; if the value entered differs from 1, the following warning query is displayed:

```
This will change all markers, Are You Sure (n)?
```

Either: enter Y↵ to effect the size change and exit;

Or: press ↵ to exit without changing the size of the markers.

3.12 Interpret Menu Item: Pick

polyline
component
object
marker
isosurface
nothing

The **Interpret** version of **Pick** opens access to six functions, which enable the user to select (or de-select) entities to be operated on by other functions. Four of these functions have been described in Section 2.5. The remaining two, i.e., **Pick + marker** and **Pick + isosurface**, are described in this section.
Recall that **Pick** refers to the action of selecting *or* de-selecting an entity: **Pick**ing an entity causes it to be selected (highlighted) if it is currently not selected; otherwise, if it is currently selected, **Pick**ing an entity causes it to be de-selected (de-highlighted).

File	Toolbox	Volume Data	Cutting Plane	Contour Tools	Marker Tools	Pick	Object Tools	View	Shade	Field Points	Return

3.12.1 Pick + marker

Markers can be **Pick**ed in file-groups. All markers read from one file are assigned to a file-group; each group can be **Pick**ed through one application of **Pick + marker**. Stand-alone markers (those placed using **Marker Tools + enter point**) can be **Pick**ed individually.

When you select **Pick + marker,** the following is displayed:

```
Pick Markers [*=all, ESC=done]:
```

Either: press [*] to **Pick** all markers (including stand-alone markers and group members);

Or: click on a marker to **Pick** it; all members of its group (if any) are also **Pick**ed.

When every required marker has been **Pick**ed, press ESC to exit from **Pick + marker**.

3.12.2 Pick + isosurface

This function is used to select or de-select isosurfaces. The user is prompted for the isosurface, as follows:

```
Pick isosurface [*=all; ESC=done]:
```

Either: press [*] to **Pick** all isosurfaces, and exit;

Or: click on an isosurface to **Pick** it; **Pick** as many as required; then press ESC to exit.

3.13 Interpret Menu Item: Object Tools

delete picked
visible
invisible

The **Interpret** version of **Object Tools** opens access to three functions, which operate on selected entities. Each entity, having been selected using **Pick**, may be deleted or temporarily hidden from view, using **delete picked** or **invisible**, respectively. The function **visible** restores all hidden entities back into view. Please see Section 2.7 for a detailed description of these functions.

File	Toolbox	Volume Data	Cutting Plane	Contour Tools	Marker Tools	Pick	Object Tools	View	Shade	Field Points	Return

3.14 Interpret Menu Item: View

The functions accessed through the **View** menu are the same for both **Interpret** and **Modeler**. Please see Section 2.8 for their descriptions.

File	Toolbox	Volume Data	Cutting Plane	Contour Tools	Marker Tools	Pick	Object Tools	View	Shade	Field Points	Return

3.15 Interpret Menu Item: Shade

The functions accessed through the **Shade** menu are the same for both **Interpret** and **Modeler**. Please see Section 2.9 for their descriptions.

File	Toolbox	Volume Data	Cutting Plane	Contour Tools	Marker Tools	Pick	Object Tools	View	Shade	Field Points	Return

3.16 Interpret Menu Item: Field Points

intern vis off
edge vis on
tgl stress blk
write pts data
edit points

The **Interpret** version of the **Field Points** menu offers five functions. Two of these, **intern vis off/on** and **edge vis on/off** have been described in Section 2.11.5 and Section 2.11.4, respectively. The third, **tgl stress blk** is equivalent to the function **Analysis Param + stress block off/on**, which was described in Section 2.10.3. The fourth and fifth functions are described below.

File	Toolbox	Volume Data	Cutting Plane	Contour Tools	Marker Tools	Pick	Object Tools	View	Shade	Field Points	Return

3.16.1 Field Points + write pts data

This function is used to output results data for free points into an external ASCII file. For the function to be available, the free points must have been incorporated into the .EX3 file prior to submitting the analysis task to COMPUTE$^{\text{3D-BEM}}$. In that case the free points are displayed in the **Interpret** DATA INTERPRETATION screen, each colored according to the color settings for the Contour Legend.

When **Field Points + write pts data** is selected, the user is prompted for the output file name, as follows:

```
enter output filename:
```

Type a *filename* (including an extension), and press ↵. No default extension is assigned. If the current variable is a ubiquitous joint stress component, the output file data consists of three lines of data per free point: The first line gives the free point number and its N-U-E coordinates; the second gives the magnitude, dip, and dip direction, for the maximum, intermediate, and minimum principal stress components; and the third gives the magnitude, dip, and dip direction for the joint normal and shear stresses, followed by the values of safety factor for the joint and for the intact rock. If the current variable is a principal stress component, the output file data consists of two lines per free point, the two lines being exactly the same as the first two lines described above; ubiquitous joint data is not included.

3.16.2 Field Points + edit points

This function, like the **Modeler** version described in Section 2.11.3, enables the user to modify the type and size of free point markers. Unlike the **Modeler** version, the **Interpret** version permits different sizes to be assigned to different points.

Each free point marker is colored to reflect the magnitude of the current results variable at the point, in accordance with the Contour Legend. In addition, markers of the same color can be assigned different sizes to reflect the relative magnitudes of the variable. Those which represent the minimum magnitude are assigned the smallest size, whereas those which represent the maximum magnitude are assigned the largest size; the rest are assigned sizes between the smallest and the largest, depending on the magnitude they represent.

The user is prompted as follows, having selected **Field Points + edit points**:

```
Enter Marker Type (1=diamond, 2=box, 3=sphere, 4=strip) [default]:
```

Type the required number (1,2,3 or 4), and press ↵.

```
Enter minimum point scalefactor [default]:
```

Enter the value of scale factor for the smallest size of markers.

```
Enter maximum point scalefactor [default]:
```

Enter the value of scale factor for the largest size of markers.

3.17 Interpret Menu Item: Return

File	Toolbox	Volume Data	Cutting Plane	Contour Tools	Marker Tools	Pick	Object Tools	View	Shade	Field Points	Return

The **Return** option of the DATA INTERPRETATION screen (Figure 1.5) returns you to the DATA SELECTION screen (Figure 1.4), allowing you to select a different variable for viewing if you wish.

4. Boundary Element Analysis Using COMPUTE $^{3D\text{-}BEM}$

The program COMPUTE$^{3D\text{-}BEM}$ is supplied as part of the EXAMINE^{3D} package. It is a three-dimensional boundary element stress analysis program, for linear elastic, isotropic and homogeneous media. Several of the functions in EXAMINE^{3D} are designed to prepare the input data for, and interpret the output from, COMPUTE$^{3D\text{-}BEM}$. Specific details of the data preparation process depend on the nature of the problem being modeled. Generally, the following steps are involved:

1. Define the three dimensional geometry of the structure to be analyzed, using the polyline building functions in Section 2.3.

2. Generate the boundary element mesh, using one or more of the mesh generation functions in Section 2.4.

3. Assign values to the analysis parameters and options, using the functions in Section 2.10.

4. Specify the field points at which analysis results should be computed, using the functions in Section 2.11.

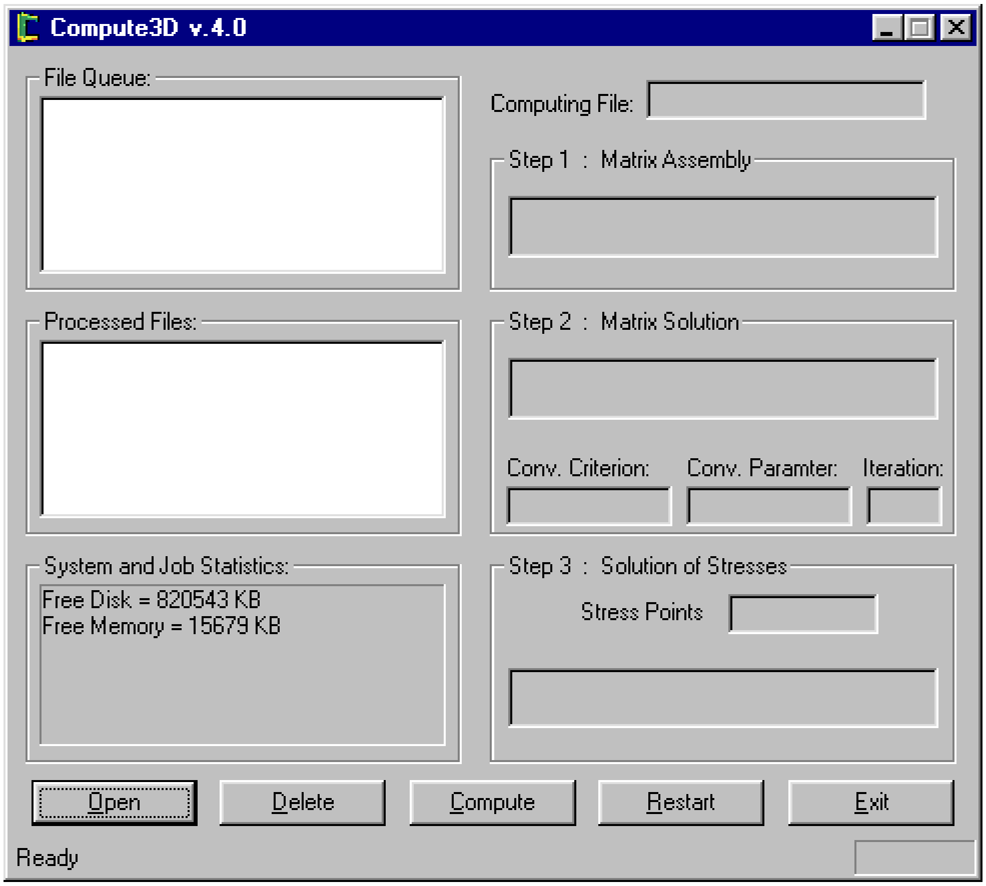
5. Write the data to a file, assigned the name *filename*.EX3 (where *filename* is user-specified), using **File + save file** (Section 2.1.2).

6. Submit the analysis task to COMPUTE$^{3D\text{-}BEM}$, using the procedure described subsequently. The analysis results will be stored in a file named *filename*.RES.

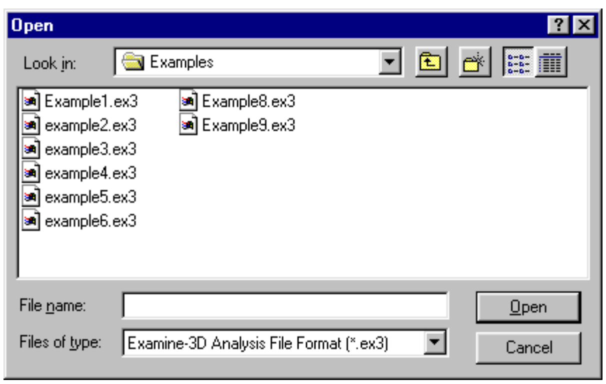
7. Interpret the analysis results using the **Interpret** group of functions.

4.1 Submitting a Task to COMPUTE $^{3D\text{-}BEM}$

Select the COMPUTE^{3D} icon in the Start→Rocscience→Examine3D menu. The following dialog will appear:



To submit a task, select the **Open** button then select the EXAMINE^{3D} .EX3 file through the following dialog:



You may select multiple files by holding down the CTRL key and selecting files with the mouse or by holding down the left mouse button and rubberbanding a selection window. Once you have selected the files press the **Open** button. You will return to the main dialog window and the files you have selected will have been placed into the **File Queue**. If you have made a mistake, files may be removed from the **File Queue** by using the mouse to select the file in the **File Queue** window then pressing the **Delete** button.

Once you have the files that you wish to analyze in the file queue, you may start the analysis procedure by selecting the **Compute** button. The analysis proceeds with the progress of each file computation displayed in the right hand portion of the main dialog. The analysis procedure of a single file can be divided into three major components: matrix assembly, matrix solution and solution of the far field points. At any point during the analysis cycle, the user may abort the process by selecting the **Abort** button. This will halt computation of the current file and suspend computation of all files in the **File Queue.**

When the computation of a file has been successfully completed, the file is moved into the **Processed Files** list box.

4.2 COMPUTE $^{3D\text{-}BEM}$ System and Job Statistics

During the computation of a data file, system and job resource utilization statistics are displayed in the **System and Job Statistics** window. The amount of **Free Disk** space on the drive containing the c3.exe executable, along with the amount of **Free Memory** (physical) available to the operating system is displayed. When the computation of a data file starts, statistics on the amount of **Matrix Memory, Memory Used, Number of Swaps,** and **Swap Space** are displayed.

Matrix Memory is the total amount of physical memory that the boundary element matrix would require if the whole matrix were to be stored in memory.

Memory Used is the amount of physical memory being used to store the matrix. If the matrix is very large, the matrix is actually stored as a disk file and portions of the matrix are read into memory for iterative solution. In this case the **Memory Used** can be a small proportion of the **Matrix Memory**.

The **Number of Swaps** is equal to the **Matrix Memory/Memory Used** if the **Memory Used** is smaller than the **Matrix Memory**. If the complete matrix can be stored in memory (e.g. small number of elements) then disk swapping is not required and the **Number of Swaps** is 0.

The **Swap Space** is either equal to the amount of **Matrix Memory** if disk swapping is occurring or equal to 0 if the complete matrix is being stored in physical memory.

4.3 Disk Swapping with COMPUTE $^{3D\text{-}BEM}$

COMPUTE$^{3D\text{-}BEM}$ does disk swapping on problems which exceed the memory capabilities of the computer being used. Disk swapping is done automatically without any user intervention. By default, COMPUTE$^{3D\text{-}BEM}$ will attempt to use 75% of the total amount of available physical memory (memory not being used by the operating system). This ensures that Windows virtual memory will not be used resulting in faster solution times. If the amount of physical memory is less than 4MB then the program will use a quarter of the total physical memory. This may result in Windows virtual memory swapping slowing down your process.

To override the amount of memory used for disk swapping, set the environment variable SWAP to the amount of memory (in KB) that you wish to use for swapping e.g.,

SET SWAP=5000

would use 5MB (5000KB) of swap space.

4.4 Restarting a Previous Analysis

COMPUTE$^{3D\text{-}BEM}$ permits the user to restart an analysis if for some reason the program is aborted during the calculation of field point stresses. This option may also be used when the user wishes to define a different region for field stress calculation. The user may define new sets of field points, using any of the functions in Section 2.11, and perform a new analysis without re-assembling the system matrices, provided that the geometry and discretization are unchanged. For large problems, the assembly and solution of the boundary element equations can be a major component of the time taken to perform the analysis.

4.5 Restarting with New Sets of Field Points

Read the existing .EX3 file into EXAMINE^{3D}, and modify the field points definition as desired. Save the new .EX3 file, and use the restart3 utility program provided in the UTILITIES/MISC directory to create a restart .RES file. Thereafter, run COMPUTE$^{3D\text{-}BEM}$, open the file and place it in the file queue and press the **Restart** button.

4.6 Restarting with No Change in Field Points

It is also possible to restart a file if COMPUTE$^{3D\text{-}BEM}$ was aborted (say by pressing **Abort** to free the computer for another task) during the calculation of the plane/grid field points. The program will sense how far the analysis has progressed and continue from that point. Just re-open the file and place it in the file queue and press the **Restart** button.

5. Utilities and File Formats

5.1 Conversion of AUTOCAD .DXF Files to EXAMINE^{3D} .GEO Files

Although EXAMINE^{3D} cannot read .DXF files created by Autocad™ directly, it is possible to use the DXFGEO.exe utility to convert a .DXF file to a .GEO file. DXFGEO is a bi-directional conversion utility, allowing the transformation of .DXF files to .GEO files, and .GEO files to .DXF files. The file, DXFGEO.C is also provided. It contains the C language source code, to enable the user to run the utility on a different platform, with any necessary changes to the code. The requirements for this version of DXFGEO is a 386/486 computer with a minimum of 2MB of extended memory. The program can be found in the UTILITIES/DXFGEO subdirectory of your EXAMINE^{3D} installation directory.

To run the program, type

DXFGEO [options] [inputfile] [outputfile] ↵

[options]:
-q Do not convert quadrilaterals to triangles
-l Do not convert lines to polylines

[inputfile]
Either: *filename*.GEO to convert .GEO file to .DXF file

Or: *filename*.DXF to convert .DXF file to .GEO file

[outputfile] is optional; if it is omitted, the results would be stored in *filename*.GEO (for *filename*.DXF input), or in *filename*.DXF (for *filename*.GEO input).

For example, to convert .DXF TO .GEO:

Either: DXFGEO junk.dxf ↵

Or: DXFGEO junk.dxf morejunk.geo ↵

To convert .GEO TO .DXF:

Either: DXFGEO junk.geo ↵

Or: DXFGEO junk.geo morejunk.dxf ↵

Typing DXFGEO ↵ (no arguments) will display a brief syntax usage screen for reference purposes.

EXAMINE^{3D} supports the following Autocad™ entities during .DXF to .GEO conversion: LINE, PLINE, 3DPOLY, 3DFACE, 3DMESH and PFACE (3 and 4 nodes per face only).

When building a model in Autocad™ please ensure that the model has the proper connectivity for a boundary element mesh.

5.2 Interpolation of Scattered Data using the EDEN3 utility program

EXAMINE^{3D} now has bundled with it a utility called eden3. Eden3 (not to be confused with eden or eden2) is a utility that can take a set of discrete points in space, each point having an associated scalar value, and interpolate these values to some spatial function. It then maps this function to a regular grid, and generates a DAT file that can be viewed within the **Interpret + General** menu of EXAMINE^{3D}. The utility can also calculate event and energy densities. The utility can read in a variety of seismic format files (i.e. MP250, RDB etc.), along with a free format file that can be used by people with unsupported data files. Converters between different seismic data formats and EXAMINE^{3D} formats can be found in the UTILITIES/CONVERT directory. Eden3 is completely menu driven and easy to use. The program can be found in the UTILITIES/EDEN3 subdirectory of your EXAMINE^{3D} installation directory.

Note: Earlier versions of this program were funded by the Mining Research Directorate under the CRRP.

5.3 Conversion of Seismic Event Files to EXAMINE^{3D} .PTS Files

EXAMINE^{3D} provides utilities for the conversion of many of the more popular file formats for storing seismic events. The conversion process takes a seismic event file and creates an EXAMINE^{3D} PTS file which can easily be imported into EXAMINE^{3D} (Section 3.11). The conversion utilities, source code, and sample files can be found in the UTILITIES/CONVERT subdirectory of your EXAMINE^{3D} installation directory.

5.3.1 Conversion of .RDB Files to .PTS Files

An .RDB file is a file produced by the Queen's University Rock Physics Laboratory, which contains data regarding the location and time of a microseismic event. In order for EXAMINE^{3D} to use this data, the .RDB file must be converted into a readable .PTS file. This is accomplished using the program RDBTOPTS.

The command line is:

rdbtopts [-t DN DU DE] [*filename* (.rdb)] [*filename* (.pts)]

[options]:

-t translates points in N,U,E directions by given amounts DN, DU, DE
DN,DU,DE the relative displacements in N,U,E directions

Data from any coordinate system can be read into the .PTS file, which consists of a columnar listing of xyz coordinates based on the NORTH, UP, EAST default format. These coordinate settings can be altered later using the EXAMINE^{3D} function **File + coord transform**.

.PTS files serve two purposes in EXAMINE^{3D}:

1. provide stress point locations

2. provide marker locations

5.3.2 Other Seismic Conversion Utilities

Utilities with the same execution syntax as the above RDB conversion utility exist for MP250, CR1, SPA files, and FRE format files. The conversion utilities, source code, and sample files can be found in the UTILITIES/CONVERT subdirectory of your EXAMINE^{3D} installation directory.

5.4 Miscellaneous Utilities

The UTILITIES/MISC subdirectory of your EXAMINE^{3D} installation directory contains utilities and source code for:

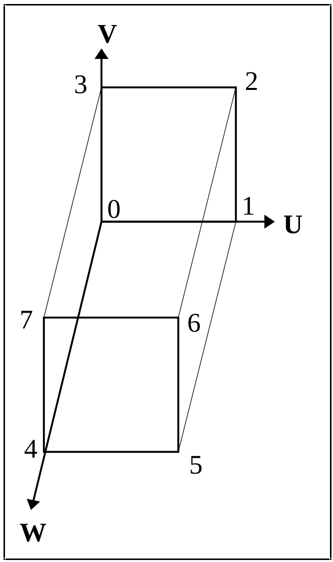
1. Subtracting the results of one analysis from another, useful for looking at differential stresses due to staged excavation (dstress utility).
2. Subtracting the far-field stresses from the results for looking at induced stresses (dfield utility).
3. Restarting an analysis with a different set of field points (restart3 utility).
4. Calculating event density using the obsolete eden utility, see the readme.txt file.

5.5 The .DAT File Format

EXAMINE^{3D} includes the capability to read or write general data files (assigned the .DAT extension). To interpret a .DAT file, ensure that the parameter *general* is set to *on* in the file e3.cfg; in that case **Interpret** opens the DATASETS screen, from which **General** should be selected. There are at present five types of .DAT file, viz., GRID, RESOLUTION_GRID, PLANE, TENSOR_POINT, and POINTS data. Examples of .DAT files can be found in the EXAMPLES subdirectory of your EXAMINE^{3D} installation directory. The files are contained within a self extracting archive named examples.exe. To unarchive the files, type *examples* at the DOS prompt.

5.5.1 Grid and Resolution_Grid Data Files

For both GRID and RESOLUTION_GRID data files, each file describes the spatial distribution of one variable within a volume, in terms of values given at regularly spaced grid points. In addition, RESOLUTION_GRID data files include confidence values at each grid point. Generally, each file provides three sets of information: (1) environment set up information for EXAMINE^{3D} (2) a description of the boundaries within which data is provided, and (3) a listing of grid-point coordinates and the values of the variable at such points.



A typical GRID or RESOLUTION_GRID data file is illustrated in Table 5.1. The first input block of the file begins with the keyword SETUP. It provides the dataset name, the settings for the orthogonal coordinate system used in the file, the name (if any) of a .GEO or .EX3 file containing the geometry data, a choice of compass or Cartesian coordinate convention, and the name of the .INF file associated with this dataset (see Section 5.5.5). The input block is terminated with the keyword ENDSETUP.
The next input block, which begins with the keyword GRID, consists of two sections: The first section provides the global coordinates of the grid-box corners, followed by the number of grid subdivisions in each local coordinate direction of the grid box. The grid box coordinate system is arbitrarily oriented relative to the global system; furthermore, the grid-box boundaries need not be parallel. The coordinates of the corners must be given in the order P0, P1 ... P7, as illustrated in the sketch on the left.

The second section of the second input block gives the global coordinates and values of the variable at the grid intersections. It terminates with the keyword ENDGRID.

5.5.2 Plane Data Files

The format of .DAT files for the PLANE option is the same as for the GRID option, except for the difference that PLANE gives data for a plane, whereas GRID gives data for a box. A second difference is that the beginning and end of the second input block are marked by the keywords PLANE and ENDPLANE, respectively (in the place of GRID and ENDGRID, which are used for grid data).

The plane is described with respect to its local U-V coordinates (the W coordinate value is fixed). The local coordinate system is arbitrarily oriented with respect to the global coordinate system; furthermore, the edges of the plane need not be parallel. A typical .DAT file for a plane is given in Table 5.2. Points 0, 1, 2 and 3 are the same as in the figure in Section 5.5.1.

5.5.3 Points Data Files

This data file consists of two input blocks: the SETUP block and the data block. The first is the same as was described for GRIDS and PLANES; the second block begins with the keyword POINTS and ends with the keyword ENDPOINTS. Each input line in this block gives a number (line number), the coordinates of a point, and the value of the variable at the point described by the coordinates. An example is given in Table 5.3.

```
**********************************************************
SETUP
name = VELOCITY
transform = north,up,east
read = strath.geo
coordinate = compass
info = strath.inf
ENDSETUP
**********************************************************
GRID
*
* GRID DEFINITION
* numu = number of grid cells in the U direction (P0->P1)
* numv = number of grid cells in the V direction (P0->P3)
* numw = number of grid cells in the W direction (P0->P4)
*|------------------------|-----|-----|-----|
*| P0 P1 P2 P3 P4 P5 P6 P7 |numu |numv |numw |
*|------------------------|-----|-----|-----|
  9950 -2162.5 20550
  9950 -2162.5 21250
  9950 -2537.5 21250
  9950 -2537.5 20550
  10750 -2162.5 20550
  10750 -2162.5 21250
  10750 -2537.5 21250
  10750 -2537.5 20550
  7 8 5
**********************************************************
* Grid Data is written such that grid cell values in the U
* direction increment the quickest, then V, then W.
*
* GRID DATA
*|---|------------|------------|------------|---------|
*| # |  North (X) |     Up (Y) |   East (Z) | value   |
*|---|------------|------------|----------- |---------|
1  9950.000000 -2162.500000 20550.000000 20.364357
2  9950.000000 -2162.500000 20650.000000 20.361856
3  9950.000000 -2162.500000 20750.000000 20.477411
4  9950.000000 -2162.500000 20850.000000 20.711025
5  9950.000000 -2162.500000 20950.000000 20.829741
etc etc
431 10750.000000 -2537.500000 21150.000000 20.949574
432 10750.000000 -2537.500000 21250.000000 20.759338
ENDGRID
```

Table 5.1: A .DAT file example, for GRID data

```
*********************************************************
SETUP
name = SIGMA 1
transform = north,up,east
read = neutrino.ex3
coordinate = compass
ENDSETUP
PLANE
*********************************************************
* numu = number of plane cells in the U direction (P0->P1)
* numv = number of plane cells in the V direction (P0->P3)
* PLANE DEFINITION
*|-------------|-----|-----|
*| P0 P1 P2 P3 |numu |numv |
*|-------------|-----|-----|
  -45.7921 -12.6502 0.141164
  45.9988 -12.6502 0.141164
  45.9988 54.5573 -0.367933
  -45.7921 54.5573 -0.367933
  40  30
*********************************************************
* Plane Data
* Plane Data is written such that values in the U direction
* increment the quickest, then V, then W.
*********************************************************
* PLANE DATA
*|---|------------|------------|------------|---------|
*| # |  North (X) |     Up (Y) |   East (Z) | value   |
*|---|------------|------------|----------- |---------|
1     -45.7921     -12.6502     0.141164     95.7532
2     -43.4973     -12.6502     0.141164     95.845
3     -41.2026     -12.6502     0.141164     95.9501
4     -38.9078     -12.6502     0.141164     96.0702
5     -36.613     -12.6502     0.141164     96.2066
etc etc
1270     43.704     54.5573     -0.367933     95.7924
1271     45.9988     54.5573     -0.367933     95.709
ENDPLANE
```

Table 5.2: A .DAT file example, for PLANE data

```
*******************************************************
SETUP
name = SIGMA 1
transform = north,up,east
read = strathpt.ex3
coordinate = compass
ENDSETUP
POINTS
********************************************************
* POINTS DATA
* coordinates in NORTH,UP,EAST (X,Y,Z) format
*|---|-----------|----------|----------|---------|
*| # | North (X) |   Up (Y) | East (Z) | value   |
*|---|-----------|----------|--------- |---------|
********************************************************
1     10542.6     -2271.08     21172     60.6535
2     10769       -2258.52     21246     35.3479
3     10689.5     -2272.17     21106.1   35.9265
4     10422.8     -2273.19     20538.2   35.2349
5     10508.6     -2283.31     20645.1   35.3146
etc etc
51    10912.4     -2875.21     22274.4   35.7745
ENDPOINTS
```

Table 5.3: A .DAT file example, for POINTS data

```
******************************************************************
SETUP
name = MOMENT TENSOR
transform = east,north,down
read = aeclmt.ex3
coordinate = compass
eigenvalue = 1
ENDSETUP
TENSOR_POINTS
******************************************************************
* coordinates according to the current transform.
* Tensor is written in the form:
* T11 , T12 , T13 , T22 , T23 , T33
* where 1,2,3 can be either NORTH,SOUTH,EAST,WEST,UP or DOWN.
* ie. if transform = north,up,east then coordinates are written as
*     NORTHING,UP,EASTING and the tensor is written as
*     TNN , TNU , TNE , TUU , TUE , TEE
* NOTE: Tensors are obviously symmetric!
******************************************************************
* TENSOR DATA AT RANDOM POINTS
*|---|------------|-----------------------------------|
*| # |  X | Y | Z | T11 | T12 | T13 | T22 | T23 | T33 |
*|---|------------|-----------------------------------|
1  761.0 438.3 123.3 -3624.5  1042.5 1532.7 -1660.3 -735.6  -3677.6
2  761.3 438.0 123.1 -2043.9  958.5  464.1  -1695.4 -1190.0  1131.3
3  760.8 437.9 123.5 -1347.2 -218.4  216.3  -378.7   424.5  -102.9
etc etc
37 759.5 436.5 123.7 -1086.6  627.4  23.8   -1464.6 -88.4    694.8
ENDTENSOR_POINTS
```

Table 5.4: A .DAT file example, for TENSOR_POINT data

5.5.4 Tensor_Point Data Files

TENSOR_POINT data files provide (stress) tensor data at random points; only six components are read at each point, which is enough to accommodate symmetric stress tensors. Like the other two options, the file begins with a setup section, which provides environment setup information for EXAMINE^{3D}. An example is given in Table 5.4.

5.5.5 Additional Information File

The .INF file is an information file that contains the data that appears in the Interpret Data Window. A typical example is given below:

```
Strathcona
Main Sill

FIELD STRESS

constant
s1 = 36 : 288/18
s2 = 32 : 194/13
s3 = 22 : 72/66

MODEL PARAM.

elements = 1378
nodes = 697
planes = 0
grids = 1
```

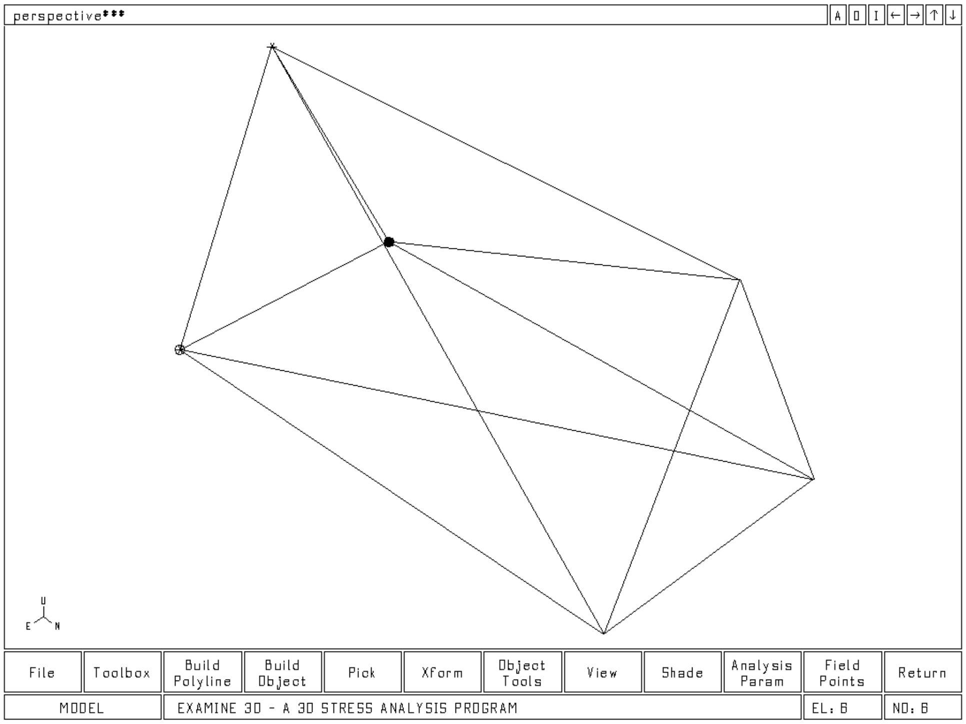
Table 5.5: A .INF file example

5.6 The .GEO File Format

A .GEO file provides a numeric description of the object under consideration by specifying the vertices (and nodes) of polygons and polylines. These vertices exist within the 3 principal planes. For example, Table 5.6 describes the figure in Fig. 5.1.

The first line of integers provides a general summary of the object. The first integer gives the total number of vertices (10) used to define the object and polylines; the second and third integers give the total number of polygons (6) and polylines (1) respectively. Polyline vertices are counted separately even if they coincide with other vertices.

The next input block, consisting of 3 columns of real numbers, provides the xyz coordinates for the vertices. Each input line within this block defines a vertex; the order of input lines is used to define the connectivity of the polygons in the next input block.



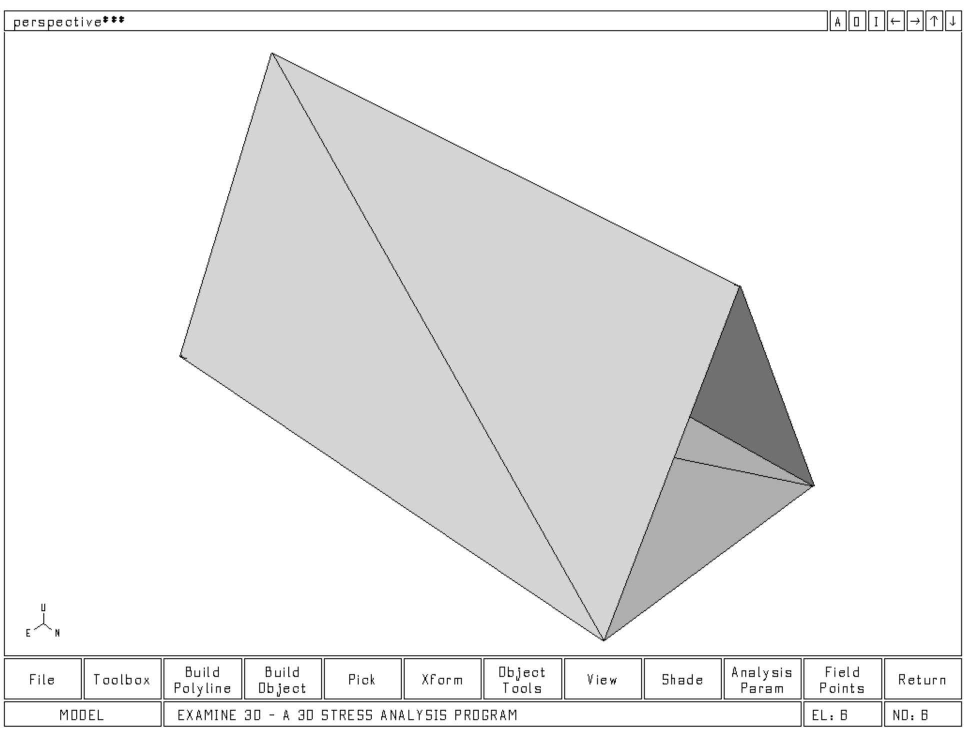


Figure 5.1: Triangular prism described by data in Table 5.5

```
10 6 1
0 -1 -1
0 -1  1
0  1  0
4 -1 -1
4 -1  1
4  1  0
0 -1 -1
0 -1  1
0  1  0
0 -1 -1
3 1  4  2
3 2  4  5
3 2  5  3
3 3  5  6
3 3  6  1
3 1  6  4
4 7 8 9 10
```

Table 5.6: A .GEO file example

The next input block, following the coordinates, provides data to describe the connectivity of the polygons. Each line of input describes one polygon. The first integer gives the number of vertices used to form the polygon; the remaining integers are the identification numbers for the vertices. Each identification number represents a line of xyz coordinates. For example, the first line of this input block in Table 5.6 describes the first polygon in Fig 5.1 as

3 1 4 2

This implies that the polygon has three vertices; the 1st, 2nd and 3rd vertices correspond to the 1st, 4th and 2nd input lines, respectively, of the coordinates block. That is, the vertices of the polygon (a triangle) are:

1^{st}: 0 -1 -1
2^{nd}: 4 -1 -1
3^{rd}: 0 -1 1

The final input block provides data to describe the polylines. Each line of input describes one polyline. The first integer gives the number of vertices on the polyline; the remaining integers, as in the case of the polygons, are the identification numbers for the vertices. For example, the only input line for this block in Table 5.6 describes the polyline in Figure 5.1 as:

4 7 8 9 10

This implies that the polyline has four vertices; the 1st, 2nd, 3rd and 4th vertices correspond to the 7th, 8th, 9th and 10th input lines, respectively, of the coordinates block. That is, the vertices of the polyline are:

1^{st}: 0 -1 -1
2^{nd}: 0 -1 1
3^{rd}: 0 1 0
4^{th}: 0 -1 -1

6. Tutorial 1: Rectangular Tunnel

Note: Chapter 1 of this manual contains important introductory information on EXAMINE^{3D}; please read it before proceeding with the tutorials. The typographical conventions used in both the reference and tutorial chapters are explained in Section 1.8.

> In the tutorial chapters, indented text (like this paragraph) gives more detailed explanations of a given exercise. They may be ignored during the first run through a tutorial, in which case the user should go through the tutorial at least one more time.

In this tutorial, the EXAMINE^{3D} MODELER will be used to generate a boundary element mesh for a rectangular tunnel. The tunnel has a square section of side 4 m, in the U-E plane, and a length of 12 m in the N-coordinate direction.

Startup EXAMINE^{3D} by selecting the EXAMINE^{3D} icon in the Start→Rocscience→Examine3D menu.

Select **Modeler**.

Select **View + auto box mode**.

The following is displayed in the dialogue box:

```
Choose Mode (0=ALL GEOMETRY, 1=STRESS POINTS, 2=USER DEFINED)?
```

> The **auto box mode** function enables the user to specify a size and location for the view windows. Let's use this function to set starting dimensions for the windows.

Enter `2` ↵

```
Locate box corner #1 [N,U,E: vn=off, s=off, o=off]:
```

Enter `-5,-5,-5` ↵, and click on `Go` .

```
Locate box corner #2 [N,U,E: vn=off,s=off,o=off]:
```

Enter `5,5,5` ↵, and click on `Go` .

> A set of numbers, like (5 5 5), can either be separated by commas (,) or white spaces.

Select **Toolbox + setup options**, to obtain the SETUP OPTIONS menu.

Click on **Grid Spacing**; then enter 2 ↵

Click on **Grids**, to toggle it on.

Select **Save**, to effect these settings and exit from the menu. Note that grid lines can also be toggled on or off by pressing the F7 keyboard function key.

Select **Build Polyline + new polyline**. The following appears in the dialogue box:

```
Select Point [N,U,E: vn=off, s=off, o=off, c,u,i,e,0]:
```

Press s .

Notice that the s-option in the dialogue box has changed to s=on .

Move the mouse pointer to anywhere in the lower left quadrant of the *front view* (E-U coordinate) window and click the left mouse button.

Notice the red star formed at a grid intersection.

Move the pointer approximately horizontally from, and about two grid spaces to the right of the red star, and click the left mouse button. Another red star is formed.

Move the pointer approximately vertically from, and about two grid spaces above the second red star, and click the left mouse button. A third red star is formed.

Move the pointer approximately horizontally from, and about two grid spaces to the left of, the third red star, and click the left mouse button. A fourth red star is formed. Notice the counter at the end of the dialogue box prompt is updated as the points are entered. It should now read 4.

Press c, to close the polyline and exit from the **new polyline** function.

The blue square formed is the required polyline; it represents an outline of the rectangular tunnel section. The grid lines can now be removed, as follows:

Select **Toolbox + setup options**. At the SETUP OPTIONS menu, toggle **Grids** off, and Select **Save**.

> A polyline consists of a series of line segments in space, connected at *vertices*. The default color of a polyline in EXAMINE^{3D} is blue, and the vertices are marked with blue stars. The first vertex is marked with a solid green circle, and the second with a hollow green circle, thus defining the order of the vertices.

In order to appreciate the difference between *polylines* and *nodelines*, let's convert the polyline to a nodeline, as follows:

Select **Build Polyline + polyline→nodeline**.

```
Pick Polyline to Convert:
```

Click anywhere on the polyline. The first segment of the polyline is highlighted in red, and the dialogue box requests:

```
Enter number of elements for highlighted segment[1]:
```

A polyline represents a trace of the intersection of a structure (underground opening, in this case) with a plane containing the polyline. It defines the geometry of the structure on that plane. The number of segments in the polyline depends on the number of segments required to describe this geometry satisfactorily. For example, four segments are good enough for a rectangular opening. On the other hand, the number of boundary elements required on a given section through a structure depends on other factors, such as the expected stress gradient, as well as on the geometry. It is either equal to or more than the number of segments required to define the geometry satisfactorily.
Therefore, whereas polylines are used to define geometry, nodelines are used to define boundary element discretization. The number of segments on a nodeline is equal to or more than the number of segments on the corresponding polyline. For this tunnel cross-section, let each of the sidewalls be represented with two boundary elements, and the roof and floor with three elements each (per vertical section).

The highlighted segment is a trace of the tunnel floor. Therefore,

Enter $\boxed{3}$ ↵

The next highlighted segment is a trace of one sidewall. So,

Enter $\boxed{2}$ ↵

The next highlighted segment represents the roof.

Enter $\boxed{3}$ ↵

The next is the other sidewall.

Enter $\boxed{2}$ ↵

Notice that the color of the square has changed to light blue, the default color of nodelines in EXAMINE^{3D}. Light blue stars mark the positions of the boundary element nodes. Each vertex of the original polyline has been converted to a boundary element node; in addition, nodes have also been created between the polyline vertices, depending on the number of elements requested for each polyline segment.
Notice the element EL and node ND counters at the lower right corner of the screen. ND has been set to 10, because 10 nodes have been created. On the other hand, EL is still zero because no element has been created. Surface elements (triangles and/or quadrilaterals) are required in 3D boundary element analysis. Although a nodeline is used to define the number of elements intersecting a cross-section, other nodelines (out-of-plane from the section) are required in order to actually form the boundary elements.
One way of doing this is to use the **extrude** function in the **Build Object** menu. The mesh for the rectangular tunnel will be generated by extruding the original polyline. First, to illustrate that polylines remain in place after they have been used to create nodelines, let's delete the nodeline.

The nodeline should now be deleted, as follows:

Select **Pick + nodeline**.

```
Pick Nodeline [*=all; b=box; c=cbox; ESC=done]:
```

Click on the nodeline. Notice that it is now highlighted in yellow, to indicate that it has been **pick**ed.

Select **Object Tools + delete picked**.

> The nodeline disappears, leaving the polyline. Notice that the ND counter is reset to zero.

Select **Build Object + extrude**.

```
Select Curve to Extrude:
```

Click on the polyline. Notice that it is highlighted in yellow. Respond at the dialogue box as follows:

```
Enter/Pick extrusion dir ([default]) [N,U,E: vn=off, s=off, o=off]:
```

Enter `1 0 0` ↵; or Press ↵, if the [*default*] is (1,0,0)

```
Enter length of extrusion [default]:
```

Enter `12` ↵

```
Use default discretization (y):
```

> The default discretization may at times be satisfactory. Accept it for now to see what it gives.

Press ↵

When meshing is done, Select **View + auto box mode**; then Enter `0` ↵, to autoscale to ALL GEOMETRY.

> A boundary element mesh of the rectangular tunnel has been formed by a northwise extrusion of the original rectangular section. Two nodelines, light blue, define the end sections of the tunnel. The original polyline (darker blue) is still in place. The floor, roof, and two sidewalls are each represented by four triangular elements. The end faces have not yet been discretized. Notice that the counters EL and ND have been set to 16 and 12 respectively.
> This discretization may be satisfactory for some problems, but do not accept it yet. So delete the elements and nodelines, in order to repeat the extrusion in a different manner.

Select **Pick + object**. The following request appears in the dialogue box:

```
Pick Object [*=all; ESC=done]
```

Click anywhere on the tunnel; then, when the above message is displayed again, press ESC to indicate that all required objects have been picked.

Select **Object Tools + delete picked**.

> The elements are gone. Notice the changes in the EL and ND counters.

Select **Build Object + extrude**. Respond at the dialogue box as follows:

```
Select Curve to Extrude:
```

Click on the polyline.

```
Enter/Pick extrusion dir ([default]) [N,U,E:  vn=off,  s=off, o=off]:
```

Enter `1 0 0` ⏎

```
Enter length of extrusion [default]:
```

Enter `12` ⏎

```
Use default discretization (y):
```

Enter N ⏎

> The default discretization has been rejected. Therefore, it is necessary to supply the required discretization.

Respond to the dialogue box as follows (the floor segment is currently highlighted):

```
Enter discretization for red segment[1]:
```

Enter `3` ⏎

Next, the eastern sidewall segment is highlighted:

Enter `2` ⏎

The roof segment is next highlighted:

Enter `3` ⏎

The western sidewall is now highlighted:

Enter `2` ⏎

Thereafter the following request appears:

```
Enter number of divisions along length [2]:
```

Enter `6` ⏎

> The boundary element discretization of the roof, floor and sidewalls is now complete. Notice that the counters EL and ND are now set to 120 and 70, respectively.

The end faces will now be discretized, to complete the mesh generation.

Select **Build Object + face**→

```
Pick a CLOSED Nodeline:
```

Click on one of the nodelines (light blue). Notice that the `Go` buttons appear at the bottom right corner of each window, in addition to the following dialogue box request:

```
Pick interior CLOSED Nodeline [Go=done]:
```

Click on any one Go button. A new screen is displayed, showing the nodeline, with red stars marking the positions of the nodes.

The **Array Mesh** function will be used to discretize this face. But first, explore the other options to see what they do.

Select **Automatic Mesh**.

That's a good mesh actually. The **Automatic Mesh** function here does a good job most of the time. However, the mesh on this face should conform with those on the sidewalls, roof and floor. Therefore,

Select **Reset** to discard the discretization.

Select **Radial Mesh**. Not bad either. But,

Select **Reset** to discard this one as well. Next,

Select **Array Mesh**.

Select **Return**. Then Select **Yes**.

One of the end faces is now discretized. Do the same for the other one, as follows:

Select **Build Object + face**→

```
Pick a CLOSED Nodeline:
```

Click on the other nodeline.

```
Pick interior CLOSED Nodeline [Go=done]:
```

Click on Go .

Select **Array Mesh**.

Select **Return**. Then Select **Yes**.

The boundary element mesh for the tunnel is now done. The **object check** function in the **Toolbox** menu should now be used to ensure that the mesh satisfies some basic rules. Proceed as follows:

Select **Toolbox + object check**. Then continue according to the following dialogue:

```
Check object, element, and node numbering (y):
```

Press ↵

```
Check for zero area elements (y):
```

Press ↵

```
Check triangular element base/height ratios (y):
```

Press ⏎

```
Maximum ratio = [rmax]; element number [Nel] -- Press Enter to Continue
```

Press ⏎

```
Check for invalid overlapping elements (y):
```

Press ⏎

```
Check for invalid intersecting elements (y):
```

Press ⏎

```
Check for leaky objects (y):
```

Press ⏎

```
Geometry is NOT LEAKY
```

Splendid!

The mesh is now satisfactorily complete. In order to formulate a problem which can be submitted to COMPUTE^{3D-BEM} for solution, two more things need to be done: define the in situ stresses and material properties, and define locations within the rock mass at which stresses and displacements should be computed. These tasks will be treated in another tutorial. However, this mesh may be needed later; therefore, save it as a .GEO file. The polylines and nodelines are no longer needed. So they should be discarded before saving the mesh. Proceed as follows:

Select **Pick + polyline**.

```
Pick Curve [*=all;b=box;c=cbox;ESC=done]:
```

Press `*` .

Select **Pick + nodeline**.

```
Pick Nodeline [*=all;b=box;c=cbox;ESC=done]:
```

Press `*` .

Select **Object Tools + delete picked**. Only the elements are left.

Select **File + save file**.

Select the Portable Geometry File Format (*.geo) from the Files of type drop down list. Then in the File name edit box:

Enter `TUT01` ⏎. Wait.

Select **Return**.

Select **Exit**.

Select **Yes**.

Congratulations!

7. Tutorial 2: Circular Tunnel

In this tutorial, a boundary element mesh will be generated for a horizontal 4 m diameter tunnel, 20 m long in the North-South direction. The mesh density along the circumference will be the same everywhere; on the other hand, the axial density of the mesh will be finer within 2 m of each tunnel end, than in the middle 16 m.

Startup EXAMINE^{3D} by selecting the EXAMINE^{3D} icon in the Start→Rocscience→Examine3D menu.

Select **Modeler**.

Set **Grid Spacing** to 2, and toggle **Grids** on, via **Toolbox + setup options**.

Select **Build polyline + new polyline**.

```
Select Point [N,U,E: vn=off, s=off, o=off, c,u,i,e,0]:
```

Press I to enter circle-drawing mode.

```
Enter circle radius [default]:
```

Enter 2 ↵

```
Enter number of line segments/circle [8]:
```

Press ↵

Select **View + autoscale** to see the results.

A blue circle appears. It is drawn in the N-U plane, centered at (0,0,0). This is the default position for circles in EXAMINE^{3D}. The desired position of the circle can be obtained by rotating and/or moving this one. First, refine the geometry of the circle by increasing the number of segments on the polyline.

Select **Object Tools + delete all**.

```
This will delete everything, Are You Sure ?
```

Only one polyline has been created, and you do want to delete it. Therefore,

Select **Yes**

Select **Build Polyline + new polyline**.

```
Select Point [N,U,E: vn=off,  s=off,  o=off,  c,u,i,e,0]:
```

Press I .

```
Enter circle radius [default]:
```

Enter 2 ↵

```
Enter number of line segments/circle [8]:
```

Enter 12 ↵

The circle looks better now. Notice the solid and hollow green circles, which indicate the counter-clockwise order for the vertices. Before rotating the circle to its desired position, use the **View + loc & dist** function to explore its current position.

Select **View + loc & dist**

```
P1(0,0,0)[N,U,E:  vn=off, s=off, o=off]:
```

Move the mouse pointer to anywhere near the center of the circle in the N-U plane (right view) window and click the left mouse button.

Note the coordinates of the point, printed on the prompt line.

Now press S to toggle *grid snap* on.

Again, click at any point near the center of the circle in the N-U plane (right view) window.

Notice that the coordinates are now given as (0,0,0) in the prompt line.

The grid snap function returns the coordinates of the grid intersection closest to the clicked point (the position of the mouse pointer when the left mouse button is clicked).

Press ESC to exit from the **View** menu.

Select **Pick + polyline**.

```
Pick Curve [*=all;b=box;c=cbox;ESC=done]
```

Click on the polyline and press ESC .

Notice the large green star formed at the center of the circle. It is the pivot of the polyline. All translations and rotations specified for the polyline will be measured relative to the pivot point.

Select **Xform + rotate**.

```
Enter CCW Rot. [deg NUE, Go=done, ESC=abort]:
```

The prompt line informs that:

(i) the rotation will be applied counter-clockwise (CCW);
(ii) the rotation is defined in terms of three numbers, which represent the magnitudes of rotation in degrees about the N-coordinate, U-coordinate, and E-coordinate axes, respectively.
(iii) The rotation applied (so far) can either be accepted by clicking on `Go`, or every rotation can be canceled by pressing `ESC`.

First rotate the circle 90 degrees counter-clockwise about the U-axis, to see what happens.

Enter `0 90 0` ↵

The circle is rotated to the correct position, i.e., it is now in the E-U plane (normal to the proposed tunnel axis). But notice that the polyline vertices are ordered clockwise (the first vertex is at the western-most point of the circle). A counter-clockwise ordering of polyline vertices is preferred. Therefore, cancel this rotation and start again.

Press `ESC` .

Select **Pick + polyline**.

Click on the polyline and press `ESC` .

Select **Xform + rotate**.

Enter `0 -90 0` ↵

Click on `Go` to accept the current position.

Make three copies of this polyline, centered at (2,0,0), (18,0,0) and (20,0,0), respectively, to locate the two tunnel ends (N=0 and N=20 m) and the positions at which the axial mesh density will change (at N=2 and N=18 m).

Select **Pick + polyline** and click on the polyline.

Select **Xform + copy**.

```
Enter Rel. Displ. of Copy [N,U,E: vn=off,s=off,o=off]:
```

Notice that the pivot is highlighted (red center), indicating the point relative to which all translations defined here are measured. The dialogue box informs that a copy of the polyline will be made and placed at a distance from the pivot, defined by the N,U,E translation components. All copies made can be accepted by clicking on `Go`, or all copies canceled by pressing `ESC`.

Enter `2 0 0` ↵

Enter `18 0 0` ↵

Enter `20 0 0` ↵

Click on `Go` .

The copies have been made, but two of them are outside of the current view windows. EXAMINE^{3D} automatically extends the coordinate axes to accommodate things as they are defined. But the current view window is not automatically extended. Use the **autoscale** function to bring everything into view.

Select **View + autoscale**.

Toggle **Grids** off, via **Toolbox + setup options** (or by pressing F7).

In Tutorial 1 the geometry and boundary element discretization of the tunnel surface were defined by extruding a square polyline. That procedure works best when both the geometry and boundary element discretization do not vary along the extrusion axis. For the current tunnel, although the geometry of the tunnel section is the same everywhere, the boundary element mesh is desired to be denser (axially) near the tunnel ends than in the middle part. Polylines have been placed at the points along the tunnel axis where the mesh density will change. The mesh will be generated by *wrapping a skin* around the polylines, beginning at one end and proceeding section by section towards the other end. This is accomplished using the **skin** function in the **Build Object** menu.
It will be better now to maximize (i.e., fill the screen with) one view window at a time, in order to see the objects more clearly. Explore the possibilities first.

Move the mouse pointer to anywhere within the greenish bar at the top of the perspective (upper right) window. Click the left mouse button.

The perspective view window is maximized. In this window N-coordinates increase approximately diagonally, from the top left to the bottom right corners of the screen. Therefore, the southern end of the tunnel is defined by the polyline closest to the top left corner, whereas the northern end is defined by the one closest to the bottom right corner.

Click on the greenish bar at the top of the window to return to *multi-view* display.

This works for all windows. When all four windows are in view, any one can be maximized by clicking on its top greenish bar. Also, when only one window is in view, clicking on the top greenish bar restores multi-view display.
You can now choose any of the windows to work in, except that the front view (i.e., the E-U coordinate) window is not a good choice for the following exercise. The perspective window is recommended, and it will be assumed that the user is working in this window.

Click on the greenish bar at the top of the perspective window to maximize it once more.

Select **Build Object + skin**.

```
Pick a skin polyline [0 picked, Select Go when done]:
```

Click on the southern-most polyline (the one closest to the top left corner of the screen).

```
Pick a skin polyline [1 picked, Select Go when done]:
```

Notice that the information in the dialogue box has changed from `...0 picked...` to `...1 picked...`

Click on the next polyline (closest to the southern-most one).

```
Pick a skin polyline [2 picked, Select Go when done]:
```

Click on Go .

```
Use default discretization (y):
```

Enter n ⏎

```
Enter # of interp. sections between red sections [default]:
```

The number of interpolation sections is 1 less than the number of transverse layers of boundary elements between the highlighted polylines. The two polylines are 2 m apart. Let the axial dimension of the transverse layers (of elements) be 0.5 m in this region, which implies 4 layers. Therefore the number of interpolation sections is 1 less than 4, that is 3.

Enter 3 ⏎

Three new polylines are placed between the selected two, dividing the distance between them into 4 equal parts. The dialogue box requests as follows:

```
Enter discretization for red segment [default]:
```

The circumferential discretization for this region is now requested. Recall that there are 12 segments in each polyline. The default discretization is 1 axial strip of boundary elements per polyline segment, which gives 12 strips around the tunnel perimeter. Accept this default. The above dialogue box request will appear 12 times, one for each polyline segment.

Press ⏎ in response to each of the 12 requests.

Notice that the EL and ND counters (bottom right corner of the screen) have been set to 96 and 60, respectively, indicating that 96 elements and 60 nodes have been created.

Select **Build Object + skin** to continue the mesh generation.

```
Pick a skin polyline [0 picked, Select Go when done]:
```

Click on the polyline at 2 m from the south tunnel face (that is, the one which marks the boundary between the meshed and unmeshed regions).

```
Pick a skin polyline [1 picked, Select Go when done]:
```

Click on the next polyline (the one at 18 m from the south face).

```
Pick a skin polyline [2 picked, Select Go when done]:
```

Click on Go .

```
Use default discretization (y):
```

Enter n ⏎

```
Enter # of interp. sections between red sections [default]:
```

The two selected polylines are 16 m apart. Let the transverse layers (of elements) be 2 m thick, that is 8 layers, in this region. Then the required number of interpolation sections is 7.

Enter `7` ↵

Notice that you are no longer asked to specify the circumferential discretization. It has to be the same as in the first region. It is valuable to pause again to examine the elements formed so far. Notice that EL and ND have been set to 288 and 156, respectively.

Maximize the perspective view window if it is not already maximized. Then,

Select **Build Object + skin** to continue the mesh generation.

```
Pick a skin polyline [0 picked, Select Go when done]:
```

Click on the polyline at 18 m from the south tunnel face (that is, the one which marks the boundary between the meshed and unmeshed regions).

```
Pick a skin polyline [1 picked, Select Go when done]:
```

Click on the polyline at the north face.

```
Pick a skin polyline [2 picked, Select Go when done]:
```

Click on `Go` .

```
Use default discretization (y):
```

Enter `n` ↵

```
Enter # of interp. sections between red sections [default]:
```

Enter `3` ↵

The cylindrical surface is now fully discretized. Notice that the four original polylines are still in place, along with superimposed nodelines (light blue) formed during the discretization process. On the other hand, the interpolation polylines were not left in place. Now use the **face** function to discretize the tunnel faces.

Select **Build Object + face**→

```
Pick a CLOSED nodeline
```

Click on the nodeline at the south tunnel face.

```
Pick interior CLOSED nodeline [Go=done]:
```

Click on `Go`, and the FACES screen appears.

Select **Automatic Mesh**.

Select **Return**; then select **Yes**.

Select **Build Object + face**→

```
Pick a CLOSED nodeline
```

Click on the nodeline at the north tunnel face.

```
Pick interior CLOSED nodeline [Go=done]:
```

Click on `Go` .

Select **Automatic Mesh**.

Select **Return**; then select **Yes**.

Return to multi-view display.

> This is a good point to take a break if you so desire. First save the mesh in a .GEO file, and retrieve it when you are ready to continue with the tutorial. The retrieval process is not presented here.

Select **File + save file**.

Select the Portable Geometry File Format (*.geo) from the Files of type drop down list. Then in the File name edit box:

Enter `TUT02` ↵

> It is a good idea to always run the **object check** function on a newly generated mesh, before proceeding to do anything with it.

Select **Toolbox + object check**. Then continue according to the following dialogue:

```
Check object, element, and node numbering (y):
```

Press ↵

```
Check for zero area elements (y):
```

Press ↵

```
Check triangular element base/height ratios (y):
```

Press ↵

```
Maximum ratio = [rmax]; element number [Nel] -- Press Enter to Continue
```

Press ↵

```
Check for invalid overlapping elements (y):
```

Press ↵

```
Check for invalid intersecting elements (y):
```

Press ↵

```
Check for leaky objects (y):
```

Press ↵

```
Geometry is NOT LEAKY
```

Splendid!

> The polylines and nodelines are no longer needed. You can delete them now, as follows.

Select **Pick + polyline**.

```
Pick Curve [*=all; ESC=done]:
```

Press [*] .

Select **Pick + nodeline**.

```
Pick Nodeline [*=all; ESC=done]:
```

Press [*] .

Select **Object Tools + delete picked**. Only the elements are left.

Select **File + save file**.

Select the Portable Geometry File Format (*.geo) from the Files of type drop down list. Then in the File name edit box:

Enter [TUT02] ↵

A dialog box will appear asking you whether you want to overwrite the existing tut02.geo file. Select the Yes button.

> The boundary element mesh generation is now complete. However, in order to fully describe the problem to be submitted to COMPUTE$^{3D\text{-}BEM}$, it is necessary to define the field stress and material properties.

Select **Analysis Param + compute3d stats**. The following self-explanatory message appears in the dialogue box:

```
Memory required for analysis = 3 MB
```

Select **Analysis Param + enter parameters**.

> The MODEL PARAMETERS screen appears.
> The top two fields are for the elastic parameters, showing default values of 30000 MPa for Young's modulus and 0.25 for Poisson's ratio. You can change any of these values by clicking on it, typing the new value, followed by ↵. If you click on a value and press ↵ without typing any new value, the existing value reappears.

The next lower field allows the stress field to be specified either in terms of constant magnitudes of stress components (CONSTANT option) or in terms of products of unit weight, depth below the surface, and horizontal to vertical stress ratios (GRAVITATIONAL option). The CONSTANT option is the default.

Click on `GRAVITATIONAL` to see what happens. Then click on `CONSTANT` to accept the default.

The next three lower fields describe the principal compressive stress components SIGMA 1, SIGMA 2 and SIGMA 3. For each stress component, the value in MPa, direction in degrees (from North) and the dip in degrees (inclination from horizontal) must be entered.

Click on the value of SIGMA 1, then

Enter `45` ↵

Click on the value of SIGMA 2

Enter `34` ↵

Click on the value of SIGMA 3

Enter `15` ↵

The values of the in situ principal stress components are thus set to 45, 34 and 15 MPa, respectively.
The remaining fields are really not required for the analysis; they are only needed for certain interpretations of the computed stresses. Leave this *as is* for now.

Click on **Save**.

The mesh may at this stage be submitted to COMPUTE$^{3D\text{-}BEM}$, but there will be no results to look at. It is necessary to define explicitly a series of field points at which stresses should be computed. The grid lines will be useful for this exercise.

Return to multi-view display if any one view is currently maximized.

Select **Toolbox + setup options**; set **Grid Spacing** to 2; toggle **Grids** on; then Select **Save**.

Select **Field Points + add points**.

```
Enter Input Method (1=plane,2=3pt plane,3=grid,4=file,5=pt,6=line) [default]:
```

The dialogue box informs that the field points can be described as follows: By defining a plane which can either be *rubber-banded* in on the screen (option 1), or defined by giving the coordinates of three corners (option 2). A grid box can also be defined (option 3); also, the coordinates of a series of field points can be given in an ASCII file---the .PTS file (option 4). A single point (option 5) or a series of points along a line (option 6) can also be defined. The grid box option is demonstrated in this tutorial.

Enter `3` ↵

```
Locate box corner #1 [N,U,E: vn=off,s=off,o=off]:
```

The grid box can be defined by explicitly giving the N-U-E coordinates of any two diagonally opposite corners, or by entering the two points on the screen with the mouse. The second method is demonstrated here; it helps illustrate one power of the multiple view windows.
First, define a box with the tunnel centered inside it, extending 6 m beyond the tunnel in the E-U plane, and 4 m beyond the tunnel ends (in the N direction).

Press [S] to toggle grid snap on.

In the front view (E-U coordinate) window, click on a point 8 m west of, and 8 m below, the tunnel centerline.

A red star is formed at the point. Notice the position of the red star in the right view (N-U coordinate) window. Its U-position is correct, but the N-position needs adjustment.

In the right view window, click on a point 4 m south of the tunnel, and on the same horizontal line as the current position of the star in this window.

Notice that the position of the star in the right view window has changed to the new point clicked; on the other hand, its position in the front view remains unchanged. That is because only the N-position has changed; the E- and U- positions have not.

The N-U-E coordinates for one extreme corner of the box has now been correctly defined. Check to see that this is so; otherwise, adjust the point as desired, using the front view and right view windows as necessary. Then

Click on [Go] to accept this point.

Notice that it turns yellow. The following message appears in the dialogue box:

```
Locate box corner #2 [N,U,E: vn=off, s=on, o=off]:
```

In the front view (E-U coordinate) window, click on a point 8 m east of, and 8 m above, the tunnel centerline.

A yellow square forms in the E-U plane, showing the outline of the box in this plane.

In the right view (N-U coordinate) window, click on a point 4 m north of, and 6 m above, the tunnel. A yellow rectangle also forms in this plane, showing the outline of the box in the N-U plane.

Check to see that the dimensions of the box are as desired. Then click on [Go] to accept the point.

The box turns red, and one of the sides is highlighted yellow. Respond as follows to the dialogue box requests:

```
Enter number of divisions in u direction [default]:
```

Press ↵ to accept the default.

```
Enter number of divisions in v direction [default]:
```

Press ↵ to accept the default.

```
Enter number of divisions in w direction [default]:
```

Press ↵ to accept the default.

Toggle **Grids** off, via **Toolbox + setup options**.

Select **View + autoscale** to bring everything into view in every window.

Select **Field Points + intern vis on** for a full display of the grid box.

Select **Field Points + intern vis off** to turn off the internal grid of the grid box.

Select **Field Points + edge vis off** to hide the outline of the grid box.

Select **Field Points + edge vis on** to bring it back on.

> The computation of field point responses consumes a lot of time in boundary element analysis. On the other hand, if the field points are too far apart, the computed results may lead to an incorrect assessment of the rock mass response. Therefore, it is necessary that the grid box be as small as possible and the grid lines as closely spaced as necessary. In order to achieve this, the grid box should be located only where it is needed. For example, advantage should be taken of any symmetries which exist in a structure in locating the grid box.
> A grid box which exploits all the planes of symmetry in the present tunnel will now be defined. The box will enclose only the upper NE octant of the tunnel, extending 6 m to the east beyond the tunnel, 6 m above the tunnel, and 4 m to the north beyond the tunnel. First discard the current grid box.

Select **Field Points + delete points**.

```
Select plane/grid/pts to delete [*=all, ESC=abort]:
```

Enter [*] ↵

Select **View + autoscale**.

Toggle **Grids** on again, using **Toolbox + setup options**.

Select **Field Points + add points**.

```
Enter Input Method (1=plane,2=3pt plane,3=grid,4=file,5=pt,6=line) [default]:
```

The default option is now 3. So press ↵ to accept the default.

```
Locate box corner #1 [N,U,E: vn=off,s=off,o=off]:
```

Press [S] to toggle grid snap on.

In the front view window, click on a point 6 m above the tunnel, on the vertical line passing through the center of the tunnel section.

> Notice that the red star is properly located in both the right view and front view windows.

Click on [Go] to accept the point.

```
Locate box corner #2 [N,U,E: vn=off, s=on, o=off]:
```

In the front view window, click on a point 6 m east of the tunnel, on the horizontal line passing through the center of the tunnel section. A yellow square forms in the E-U plane.

In the right view window, click on a point 4 m north of the tunnel end, on the horizontal tunnel axis. A yellow rectangle also forms in the N-U plane.

> Check that the dimensions and locations of the square in the E-U plane and the rectangle in the N-U plane are consistent with the dimensions and location of the required grid box.

Click on `Go` .

```
Enter number of divisions in u direction [11]:
```

Press ↵ to get 0.5 m spacing in the highlighted direction.

```
Enter number of divisions in v direction [12]:
```

Press ↵

```
Enter number of divisions in w direction [20]:
```

Press ↵

Toggle **Grids** off.

Select **View + autoscale**.

> The mesh is now ready to be submitted to COMPUTE$^{3D\text{-}BEM}$

Select **File + save file**.

Select the EXAMINE-3D Analysis File Format (*.ex3) from the Files of type drop down list. Then in the File name edit box type tut02.ex3 as the filename.

Wait for the message `Done Storing` to appear; then

Select **Return**.

Select **Exit**.

Select **Yes**.

Congratulations! You have successfully completed Tutorial 2, during which you generated a boundary element mesh for a 20 m long circular tunnel, defined the stress analysis problem for the tunnel (for the boundary element program COMPUTE$^{3D\text{-}BEM}$), and learnt some of the basic tools in the **Modeler** part of EXAMINE^{3D}. You can now submit the problem to COMPUTE$^{3D\text{-}BEM}$ by selecting the COMPUTE^{3D} icon in the Start→Rocscience→Examine3D menu. Follow the directions in section 4.1 for running the analysis. It will take some time, so you should submit it when you are sure that you will not need your computer for a while. The analysis results will be stored in a file tut02.res, which will be required in a subsequent tutorial.

8. Tutorial 3: Excavation with Irregular Geometry

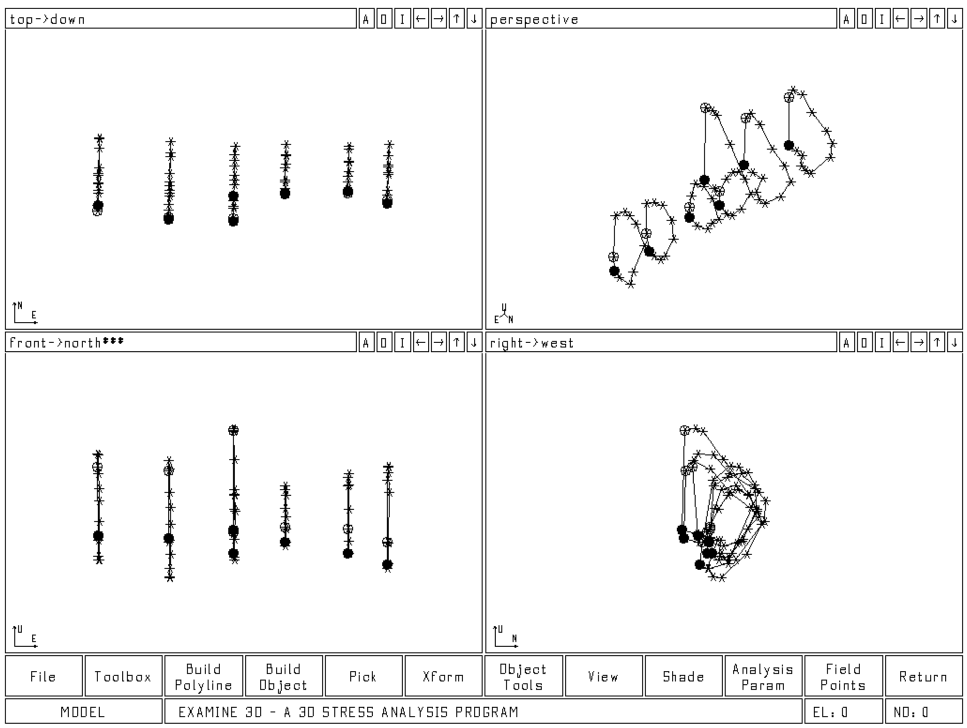


Figure 8.1: Starting polylines for Tutorial 3

Mesh generation for an irregular opening will be illustrated in this tutorial. The geometry of such an opening is usually defined starting from a series of “plans” or “blue prints”, which describe individual sections through the opening. Each section can be converted to a polyline, by defining a series of segments which describe the geometry of the section as closely as possible. EXAMINE^{3D} places certain restrictions on the number of segments on each polyline. The nature of these restrictions depends on how the polylines will be connected to define the three-dimensional geometry of the opening. These restrictions are best explained through examples, the first of which is the subject of this tutorial.

Startup EXAMINE^{3D} by selecting the EXAMINE^{3D} icon in the Start→Rocscience→Examine3D menu.

Select **Modeler**.

Select **File + append to model**.

Select the Portable Geometry File Format (*.geo) from the Files of type drop down list. Then in the File name edit box:

Enter `TUT03` ↵

> Alternatively (instead of typing TUT03 ↵) you can click on TUT03 to select it and then select the Open button.

A group of polylines is displayed (compare with Figure 8.1). The polylines describe an opening with an approximately E-W longitudinal axis.

Click on the greenish bar at the top of the perspective view window, to maximize the window.

Recall that the E-coordinate increases approximately diagonally from the top right to the bottom left corners of the screen.

There are two groups of polylines: the western-most three, which define three cross-sections of a larger opening (the *main* excavation), and the other four, which define four cross-sections of a smaller opening (the *adit*).

Notice that the three western-most polylines have 13 vertices each, whereas the other four have 10 vertices each.

Notice that the third polyline (counting from the west end) of the first group, and the first polyline of the second group overlap (i.e., they share some vertices).

Select **Pick + polyline**.

```
Pick Curve [*=all;b=box;c=cbox;ESC=done]
```

Click anywhere in the upper end of the third polyline of the first group. Notice which polyline is highlighted. Click on it again to deselect it.

Next, click anywhere in the lower end of the same polyline. Which one is highlighted? Click at the same point over and over, observing which polyline is highlighted. Now deselect everything, by clicking until none is highlighted.

> When two polylines overlap, the only way to select a particular one is to click on one of its *exclusive segments* (i.e., those not shared with the other polyline). In this example, the exclusive segments for the larger polyline are those which lie above the smaller polyline; on the other hand, for the smaller polyline, the exclusive segments are those inside the larger polyline.

Now click on one of the exclusive segments of the larger polyline. Click on it three more times and observe what happens each time.

Next, click on one of the exclusive segments of the smaller polyline. Click on it three more times and observe what happens each time.

Ensure that nothing is highlighted, then press `ESC` .

> The two overlapping polylines in this example define a *transition face*. It marks a sudden change in the geometry of an opening. In this example, an adit is extended ahead of a larger (main) excavation.
> The polylines which define sections of the adit have 10 vertices each; so they can be **skin**ned together to define the geometry of the adit. Similarly, the polylines which define sections of the main excavation have 13 vertices each, and they can be **skin**ned together to define the geometry of the main excavation.
> On the other hand, the two groups of polylines cannot be **skin**ned, because the number of vertices per polyline is different for each group. Therefore, the connection between the two arms of the opening is defined by the transition face.

Select **Build Object + skin**.

```
Pick a skin polyline [0 picked, Select Go when done]:
```

Click on the eastern-most polyline (east end of the adit).

```
Pick a skin polyline [1 picked, Select Go when done]:
```

Click on the second polyline (counting from the east end).

```
Pick a skin polyline [2 picked, Select Go when done]:
```

Click on the third polyline.

```
Pick a skin polyline [3 picked, Select Go when done]:
```

> There are two overlapping polylines at the next location. The only one that can now be selected is the one consistent with those already selected; that is, the smaller of the two. It will be selected by clicking on one of its exclusive segments. Clicking anywhere else will produce no response.

Click on one exclusive segment of the smaller one of the two overlapping polylines.

```
Pick a skin polyline [4 picked, Select Go when done]:
```

Click on `Go` .

```
Use default discretization (y):
```

> The default discretization is recommended for irregular openings. Therefore,

Press ⏎

```
Enter mesh density factor [1]:
```

The fineness of a mesh (in the default discretization) can be controlled using the mesh density factor. Its default value is 1. Values larger than 1 give finer mesh (more elements), whereas values smaller than 1 give coarser mesh (fewer elements). Accept the default for this example.

Press ↵

```
Continue with Element Generation? (y):
```

Notice that the interval between the first two (eastern-most) polylines has been divided into two, the next interval is divided into three, and the last one is also divided into three. The original polylines are highlighted (yellow); the blue ones between them are temporary interpolation polylines.

Press ↵

A message appears indicating that element generation is in progress. When the generation is done, the message disappears, the interpolation polylines are removed, and the elements are shown. Notice that the EL and ND counters have now been updated.
The surface of the adit has been discretized. Proceed with the main excavation, as follows:

Select **Build Object + skin**.

```
Pick a skin polyline [0 picked, Select Go when done]:
```

Click on an exclusive segment of the larger polyline at the transition face; that is, the eastern-most end of the main excavation.

```
Pick a skin polyline [1 picked, Select Go when done]:
```

Click on the next polyline (going westwards).

```
Pick a skin polyline [2 picked, Select Go when done]:
```

Click on the last polyline (west end of the main excavation).

```
Pick a skin polyline [3 picked, Select Go when done]:
```

Click on Go .

```
Use default discretization (y):
```

Press ↵

```
Enter mesh density factor [1]:
```

Press ↵

Notice that five interpolation polylines appear, which, with the three selected polylines, give a total of 7 interpolation sections.

```
Continue with Element Generation? (y):
```

Press ⏎

The excavation surface has now been discretized, except for the end faces. There are three faces to be discretized: (1) the east end face of the adit, (2) the west end face of the main excavation, and (3) the portion of the transition face above the adit.

Select **Build Object + face**→

```
Pick a CLOSED Nodeline
```

Click on the nodeline at the east end of the adit.

```
Pick interior CLOSED Nodeline [Go = done]:
```

Click on Go. The FACES screen appears.

Select **Automatic Mesh**. Wait. When the face is shaded, showing the elements generated,

Select **Return**; then Select **Yes** to accept the discretization. A message appears indicating that the element data is being saved. Thereafter the perspective window returns.

Select **Build Object + face**→

```
Pick a CLOSED Nodeline
```

Click on the nodeline at the west end of the main excavation.

```
Pick interior CLOSED Nodeline [Go = done]:
```

Click on Go to obtain the FACES screen.

Select **Automatic Mesh**. When meshing is done,

Select **Return**; then Select **Yes**.

It is time to discretize the transition face. The process is basically the same as for the other faces, but there is an important difference.

Select **Build Object + face**→

```
Pick a CLOSED Nodeline
```

Click on the exclusive part of the larger nodeline at the transition face.

```
Pick interior CLOSED Nodeline [Go = done]:
```

Notice that the selected nodeline is highlighted. In contrast to the faces discretized previously, the transition face is delineated by two overlapping nodelines (superimposed on the two overlapping polylines). Therefore, it is necessary to identify (pick) the second nodeline which completes the boundary of the face.

Click on the exclusive part of the smaller nodeline at the transition face. Notice that it also gets highlighted (after clicking on it).

```
Pick interior CLOSED Nodeline [Go = done]:
```

Click on `Go` to obtain the FACES screen.

> The two overlapping nodelines are shown. Those nodes which bound the face to be discretized are marked with red stars. The rest are marked with yellow circles (yellow outline, red interior).

Select **Automatic Mesh**. When the discretization is done,

Select **Return**; then Select **Yes**.

> The mesh generation is now complete. Shade the excavation surface and ends, to view its three-dimensional picture. It is faster if the elements are hidden.

Select **Shade + shade options**.

At the SHADE OPTIONS menu, toggle **Elements** off; then Select **Save**. Thereafter,

Select **Shade + quickshade**, and wait.

The glory of your work shines on the screen. You can now see why it was necessary to use at least three polylines to define the geometry of the main excavation. You can also see the shape of the intersection of the adit with the main excavation.

Press `ESC` .

Click on the greenish bar at the top of the screen to bring all four view windows back (we have been working in the maximized perspective window).

Select **Shade + quickshade** again.

Only one window is shaded at a time. Click in any one window to get a shaded view through it.

> The E-U view window clearly shows the vertical nature of the transition face. The use of the two overlapping polylines demonstrated here is most appropriate for this type of junction. The geometry of the junction would have been missed if the two arms of the excavation were connected with mutually **skin**nable polylines.

This mesh will not be used for analysis. But run an **Object Check** on it, following the procedure described in the first two tutorials. Thereafter, discard it using **Object tools + delete all**. Then proceed to Tutorial 4, for another example on transition faces.

9. Tutorial 4: En Echelon Stopes

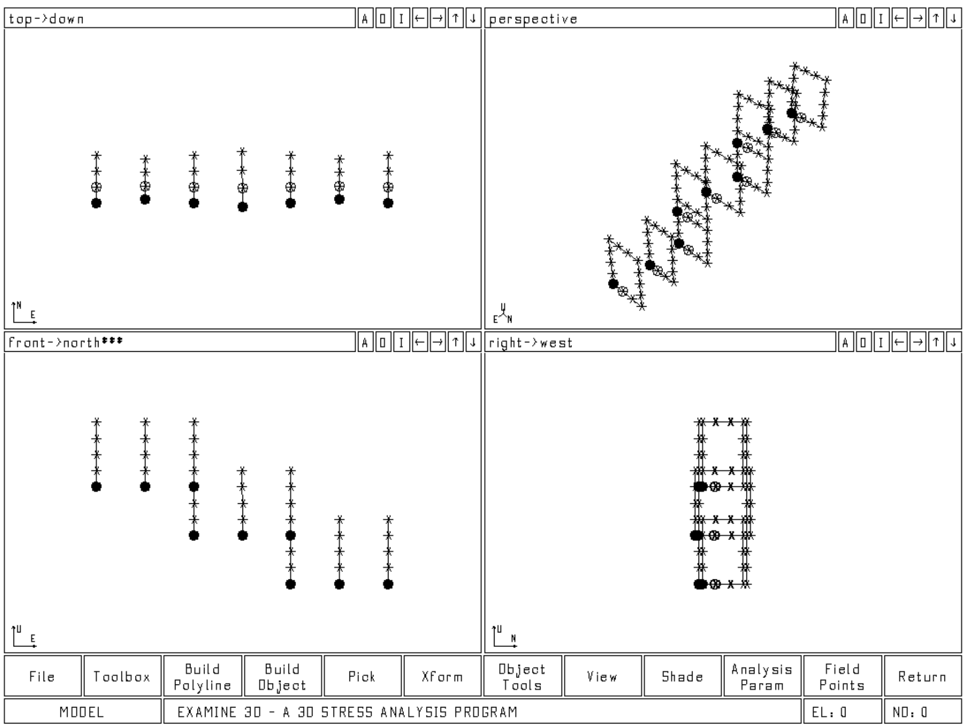


Figure 9.1: Starting polylines for Tutorial 4

In this tutorial, a boundary element mesh will be generated for a series of stopes, arranged in *en echelon* pattern, following an eastward dipping ore body. Only three of the stopes are shown in the example. The contact between adjacent stopes illustrates another category of transition faces.

Startup EXAMINE^{3D} by selecting the EXAMINE^{3D} icon in the Start→Rocscience→Examine3D menu.

Select **Modeler**.

Select **File + append to model**.

Select the Portable Geometry File Format (*.geo) from the Files of type drop down list. Click on TUT04 to select it and then press the Open button.

A group of rectangular polylines is displayed (compare with Figure 9.1). Each set of three consecutive polylines, beginning from one end, represents a stope; each stope being at a lower elevation than the one immediately to its west. There is a 50 m overlap (in the U direction) between the polyline representing the east end of a stope and the one representing the west end of the next stope.

Click on the greenish bar at the top of the front view (E-U coordinate) window, to maximize it.

Select **Toolbox + setup options**; set **Grid Spacing** to 100; toggle **Grids** on; then Select **Save**.

The relative elevations of the three stopes, and the nature of their overlaps are well illustrated in the resulting display.

Toggle **Grids** off, via **Toolbox + setup options**.

Click on the greenish bar at the top of the screen to return to multi-view display.

Then, maximize the perspective view window.

Select **Build Object + skin**.

```
Pick a skin polyline [0 picked, Select Go when done]:
```

Click on the eastern-most polyline.

```
Pick a skin polyline [1 picked, Select Go when done]:
```

Click on the next polyline (second from the east end).

```
Pick a skin polyline [2 picked, Select Go when done]:
```

There are two overlapping polylines at the next location. Click on (an exclusive segment of) the lower polyline.

```
Pick a skin polyline [3 picked, Select Go when done]:
```

Ensure that the three highlighted polylines are consecutive; then,

Click on Go .

```
Use default discretization (y):
```

Press ⟵

```
Enter mesh density factor [1]:
```

Press ⟵ to accept the default mesh density factor of 1.

```
Continue with Element Generation? (y):
```

> Notice that 4 interpolation sections have been formed, using two temporary interpolation polylines and the three selected polylines. If you entered `n` ⏎ in response to the dialogue box request, the interpolation polylines would be removed, the original selected polylines would be deselected, and the program would exit from the **Build Object** menu. In that case, you would have to repeat the process, starting from "Select **Build Object + skin**".

The proposed discretization will be accepted. Therefore,

Press ⏎

Select **Build Object + skin**.

```
Pick a skin polyline [0 picked, Select Go when done]:
```

Two polylines overlap at the junction of the meshed and unmeshed zones; click on the upper one.

```
Pick a skin polyline [1 picked, Select Go when done]:
```

Click on the next polyline to the west.

```
Pick a skin polyline [2 picked, Select Go when done]:
```

There are two overlapping polylines at the next location. Click on the lower one.

```
Pick a skin polyline [3 picked, Select Go when done]:
```

Ensure that the three highlighted polylines are consecutive; then,

Click on `Go` .

```
Use default discretization (y):
```

Press ⏎

```
Enter mesh density factor [1]:
```

Press ⏎ to accept the default mesh density factor of 1.

```
Continue with Element Generation? (y):
```

> Notice that 4 interpolation sections have been formed, using two temporary interpolation polylines and the three selected polylines.

Press ⏎

Select **Build Object + skin**.

```
Pick a skin polyline [0 picked, Select Go when done]:
```

Click on the upper polyline, at the junction of the meshed and unmeshed zones.

```
Pick a skin polyline [1 picked, Select Go when done]:
```

Click on the next polyline to the west (second polyline from the west end).

```
Pick a skin polyline [2 picked, Select Go when done]:
```

Click on the polyline at the west end.

```
Pick a skin polyline [3 picked, Select Go when done]:
```

Click on Go .

```
Use default discretization (y):
```

Press ↵

```
Enter mesh density factor [1]:
```

Press ↵ to accept the default mesh density factor of 1.

```
Continue with Element Generation? (y):
```

There should be 4 interpolation sections, formed from two temporary interpolation polylines and the three selected polylines.

Press ↵

The E-N and E-U surfaces of the stopes have been discretized. Proceed with the faces as follows:

Select **Build Object + face**→

```
Pick a CLOSED nodeline
```

Click on the nodeline at the east face of the eastern-most stope.

```
Pick interior CLOSED nodeline [Go = done]:
```

This face is defined by one nodeline. Therefore,

Click on Go to obtain the FACES screen.

This face is a good candidate for the **Array Mesh** option, which is recommended when the nodes on a face can be joined to form a rectangular pattern.

Select **Array Mesh**.

When meshing is done, Select **Return**; then Select **Yes**.

Next, discretize the other end face.

Select **Build Object + face**→

```
Pick a CLOSED nodeline
```

Click on the nodeline at the west face of the western-most stope.

```
Pick interior CLOSED nodeline [Go = done]:
```

This face is defined by one nodeline. Therefore,

Click on [Go] to obtain the FACES screen.

Select **Array Mesh**.

When meshing is done, Select **Return**; then Select **Yes**.

The transition faces will now be discretized. Each transition face consists of an upper face and a lower face separated by a rectangular opening.
To discretize the upper face, pick the upper nodeline first, then pick the lower nodeline as the *interior* nodeline.
To discretize the lower face, pick the lower nodeline first, then pick the upper nodeline as the *interior* nodeline.
Proceed as follows:

Select **Build Object + face**→

```
Pick a CLOSED Nodeline
```

Click on the upper nodeline, at the junction of the eastern-most and middle stopes.

```
Pick interior CLOSED Nodeline [Go = done]:
```

Click on the lower nodeline at the same junction.

```
Pick interior CLOSED Nodeline [Go = done]:
```

Ensure that both nodelines at this transition face are highlighted; then

Click on [Go] to obtain the FACES screen.

The **Array Mesh** option is not permitted on transition faces. Therefore,

Select **Automatic Mesh**.

When meshing is done, examine the mesh to see if the elements are of the same general size, more or less. If not, exploit the randomness of the **Automatic Mesh**ing process as follows: first Select **Reset** to discard the mesh; then Select **Automatic Mesh** again; repeat the process until the mesh appears satisfactory. Thereafter,

Select **Return**; then Select **Yes**.

Next, discretize the lower face.

Select **Build Object + face**→

```
Pick a CLOSED Nodeline
```

Click on the lower nodeline, at the junction of the eastern-most and middle stopes.

```
Pick interior CLOSED Nodeline [Go = done]:
```

Click on the upper nodeline at the same junction.

```
Pick interior CLOSED Nodeline [Go = done]:
```

Ensure that both nodelines at this transition face are highlighted; then

Click on Go to obtain the FACES screen.

Select **Automatic Mesh**.

Adjust the mesh if necessary, using the **Reset** and **Automatic Mesh** functions, as was explained above. When meshing is done,

Select **Return**; then Select **Yes**.

Now discretize the upper face of the remaining transition face .

Select **Build Object + face**→

```
Pick a CLOSED Nodeline
```

Click on the upper nodeline, at the junction of the western-most and middle stopes.

```
Pick interior CLOSED Nodeline [Go = done]:
```

Click on the lower nodeline at the same junction.

```
Pick interior CLOSED Nodeline [Go = done]:
```

Ensure that both nodelines at this transition face are highlighted; then

Click on Go to obtain the FACES screen.

Select **Automatic Mesh**.

Adjust the mesh if necessary; then

Select **Return**; then Select **Yes**.

For the lower face,

Select **Build Object + face**→

```
Pick a CLOSED Nodeline
```

Click on the lower nodeline, at the junction of the western-most and middle stopes.

```
Pick interior CLOSED Nodeline [Go = done]:
```

Click on the upper nodeline at the same junction.

```
Pick interior CLOSED Nodeline [Go = done]:
```

Ensure that both nodelines at this transition face are highlighted; then

Click on `Go` to obtain the FACES screen.

Select **Automatic Mesh**.

Adjust the mesh if necessary; then

Select **Return**; then Select **Yes**.

The discretization is complete. It is useful to shade the structure, to obtain a clearer display of the geometry.

Select **Shade + shade options**.

At the SHADE OPTIONS menu, toggle **Elements** off; then Select **Save**. Thereafter,

Select **Shade + quickshade**. Wait.

After viewing the shaded three-dimensional picture of the stopes, press `ESC` .

Return to multi-view display, by clicking on the greenish bar at the top of the screen.

The mesh generated in this tutorial will not be used for analysis. Therefore, it will now be discarded. This is a good opportunity to explore a few more of the EXAMINE^{3D} functions.

Select **Pick + object**.

```
Pick Object [*=all; ESC=done]:
```

Click anywhere in the mesh just generated.

Notice that all the elements are selected. Any group of inter-connected elements constitute an object. There is only one object in this boundary element mesh.

Click again anywhere in the mesh, to deselect everything. Then press `ESC` .

Select **Pick + component**.

```
Pick Component [*=all; ESC=done]:
```

A component is a group of elements formed through one application of the **skin**, **extrude**, **face**, **transition skin** or **blend** functions. Let’s delete the elements, component by component, in order to illustrate this definition.

Click on the top surface of the eastern-most stope.

Notice that all the elements on the four E-N and E-U surfaces of the stope are selected. These elements were formed through one application of the **skin** function. Hence they constitute one component.

Select **Object Tools + delete picked** to delete the selected component.

Select **Pick + component**.

```
Pick Component [*=all; ESC=done]:
```

Click on the top surface of the middle stope, to select all elements formed through the second application of **skin**.

Click on the top surface of the western-most stope, to also select all elements formed through the third application of **skin**. Thereafter,

Select **Object Tools + delete picked** to delete the two selected components.

> All elements on the E-U and E-N surfaces of the stopes have now been deleted. Only the face elements are left, revealing the positions of the faces and the *connecting doorways* between the stopes.

Maximize the perspective view window once more.

Select **Shade + quickshade**.

> This gives a clearer display of the 6 faces of the stopes. The four middle faces are actually two transition faces; as was explained before, each transition face consists of an upper face and a lower face, separated by a *doorway* (which connects the two adjacent stopes).

Press `ESC` .

Select **Shade + shade options**; toggle **Normals** on; then Select **Save**.

Select **Shade + quickshade**.

Normals point outward from properly constituted elements. Therefore, notice that the normals point eastward on the east end face and on the upper face of the transition faces; on the other hand, they point westwards on the west end face and on the lower face of the transition faces.

Press `ESC` .

You can now select the faces one by one, using **Pick + component**, to satisfy yourself that they are indeed different components; delete them thereafter.

Only the original polylines are left. You can delete them now, using **Object Tools + delete all**, and proceed to Tutorial 5.

10. Tutorial 5: Extending An Existing Mesh

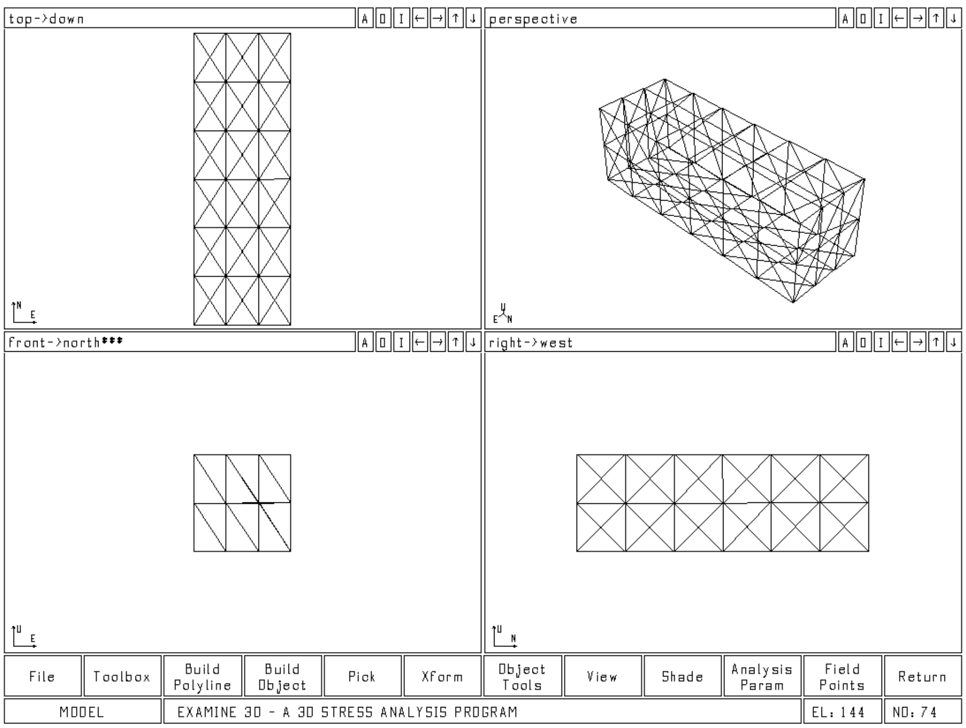


Figure 10.1: Starting boundary element mesh for Tutorial 5

It is required to model a shaft extended from the bottom of an existing tunnel. The boundary element mesh for the tunnel has previously been generated, and it is desired to model the shaft by extending the existing mesh. This is a very simple example of a potentially complicated problem, but the example illustrates the basic idea.

Startup EXAMINE^{3D} by selecting the EXAMINE^{3D} icon in the Start→Rocscience→Examine3D menu.

Select **Modeler** and retrieve TUT05.GEO using **File + append to model**.

The boundary element mesh is displayed, for a 12 m long tunnel with a square (4 by 4 m) section (compare with Figure 10.1). It is required to model a 6 m deep shaft with a 2 by 2 m section. This example is particularly convenient, because the existing mesh consists of 2 by 2 m squares divided diagonally to form triangular elements. The corners of the top end of the shaft have to coincide with the corners of one such square at the bottom of the tunnel.

The shaft will be attached as follows: (1) delete two triangular elements to form a (2 by 2 m) square hole at the bottom surface of the tunnel; (2) replace the edges of the hole with a polyline; (3) extrude the polyline, thereby discretizing the vertical surfaces of the shaft; and (4) discretize the bottom face of the shaft.

It is necessary to hide the elements on the top surface of the tunnel, using the **invisible** function, in order to gain access to its bottom surface (through the top view window).

Set **Grid Spacing** to 4 m, and toggle **Grids** on, using **Toolbox + setup options**.

Select **Pick + element**.

```
Pick Element [*=all; b=box; c=cbox; r=ratio; ESC=done]:
```

> The dialogue box gives four options for picking arbitrary element groups:
> (1) All elements can be picked by pressing $\boxed{*}$
> (2) Press $\boxed{\text{B}}$ and you would be prompted to define a box which completely encloses all the elements to be picked
> (3) Press $\boxed{\text{C}}$ and you would be prompted to define a box, similar to the $\boxed{\text{B}}$-option, but all elements *crossed or completely enclosed* by the box will be picked
> (4) Press $\boxed{\text{R}}$ and you would be prompted to specify a value for aspect ratio; all elements which have aspect ratio larger than the specified value would be picked.
> Of course, single elements can also be picked, by clicking on each individual element (as will be illustrated later).

Press $\boxed{\text{B}}$ to activate the box option.

```
Place box corner #1 [N,U,E: vn=off,s=off,o=off]:
```

> A pick box can be defined either explicitly (by defining its three-dimensional boundaries) or implicitly (by defining a plane, which is then extruded infinitely along its normal to obtain a box). In either case, two opposite corners need to be given, either by clicking on the screen or by typing in their coordinates. Let's use the implicit approach, with points defined by clicking on the screen.

In the front view (E-U coordinate) window, click on a point about 1 m below the tunnel roof, and a short distance to the west of the tunnel.

Click on $\boxed{\text{Go}}$. Notice that the red star turns colour, indicating that this point has been chosen. The dialogue box will then request:

```
Place box corner #2 [N,U,E: vn=off,s=off,o=off]:
```

In the front view window, click on a point about 1 m above the tunnel roof, and a short distance to the east of the tunnel.

Ensure that the yellow rectangle so formed completely encloses the roof line, in the E-U window.

Click on `Go` .

Notice that all, (and only), the elements on the tunnel roof are picked (highlighted).

Select **Object Tools + invisible**.

Now only the tunnel floor elements are visible through the top view (E-N coordinate) window.

Assume that the top edge of the shaft corresponds to the middle square in the third row of squares, counting from the north (in the top view window). Therefore, the two triangular elements which make up the square need to be removed.

Select **Pick + element**.

```
Pick Element [*=all; b=box; c=cbox; r=ratio; ESC=done]:
```

Click on the common (diagonal) edge of the two elements to be removed.

Notice that it is highlighted, indicating that one of the elements has been picked; but which one?

Click again on the same edge.

The element is deselected (no longer highlighted).

Click on it a third time.

Now two other edges are highlighted, clearly indicating which element has been picked (the one formed by the clicked (diagonal) edge and the two highlighted edges).

Click on the same diagonal a fourth time.

The diagonal is also highlighted, indicating that the other element which shares this edge has been picked.

Select **Object Tools + delete picked**.

> The two elements are deleted, leaving a square hole at the tunnel floor. A shaded view will illustrate this more clearly.

Select **Shade + quickshade**.

> The hole is not visible in the current shaded perspective view, because the tunnel floor is blocked by the eastern sidewall.

Click anywhere in the top view window – the square hole is clearly visible through this window.

Press `ESC` to exit from the **Shade** menu.

Select **Build Polyline + open edge polyline**.

```
1 Polyline Will Be Created, Continue?
```

Press ↵

The open square is replaced with a polyline (blue), which will now be extruded downwards to form and discretize the vertical surfaces of the shaft.

Select **Build Object + extrude**.

```
Select Curve to Extrude:
```

Click on the polyline.

The polyline is highlighted (yellow), and the following request appears:

```
Enter/pick extrusion dir ([default]) [N,U,E: vn=off,s=off,o=off]:
```

The extrusion direction is defined in terms of the N,U,E components of a unit vector. The default is the outward normal vector to the existing surface, which in this case implies a vector directed downwards (0, -1, 0) from the tunnel floor.

If the [*default*] is (0, -1, 0) press ⏎ to accept it. Otherwise, enter `0 -1 0` ⏎

```
Enter length of extrusion [default]:
```

Enter `6` ⏎

```
Use default discretization (y):
```

Press ⏎

Select **View + autoscale** to see what has been done.

Select **Pick + component**.

```
Pick Component [*=all; ESC=done]:
```

Click anywhere on the shaft.

Select **Object Tools + delete picked**.

The shaft elements are deleted, leaving nodelines and polylines. Repeat the extrusion process.

Select **Build Object + extrude**.

```
Select Curve to Extrude:
```

Click on the polyline (there is only one).

```
Enter/pick extrusion dir ([default]) [N,U,E: vn=off,s=off,o=off]:
```

Press ⏎ to accept (0, -1, 0)

```
Enter length of extrusion [default]:
```

Enter `6` ⏎

```
Use default discretization (y):
```

Enter n ⮠

```
Enter number of divisions along length [2]:
```

Enter 3 ⮠

Better! Let's discretize the shaft floor.

Select **Build Object + face**→

```
Pick a CLOSED nodeline
```

Click on the nodeline at the lower end of the shaft.

```
Pick interior CLOSED nodeline [Go = done]:
```

Click on Go to obtain the FACES screen.

Select **Array Mesh**.

The square is divided along one diagonal, giving two triangular elements.

Select **Return**; then Select **Yes**.

Select **Object Tools + visible** to bring the hidden tunnel roof elements back into view. Then delete the nodelines and polylines, as follows:

Select **Pick + polyline** and Press *

Select **Pick + nodeline** and Press *

Select **Object Tools + delete picked**.

The discretization of the shaft is complete. It would be advisable to run an **object check** on the mesh at this stage. However, for the purpose of this tutorial, let's reverse the process of attaching and discretizing the shaft.

Select **Pick + component**.

```
Pick Component [*=all, ESC=done]
```

Click on the bottom of the shaft to pick the bottom face elements.

Click anywhere else on the shaft to also pick all the other shaft elements.

Select **Object Tools + delete picked** to delete the shaft.

Hide the tunnel roof elements once more.

Select **Pick + element**.

```
Pick Element [*=all; b=box; c=cbox; r=ratio; ESC=done]:
```

Press C .

> Earlier in this tutorial, the pick box was defined implicitly (using an infinitely extruded plane). Let's define it explicitly now. The cbox option has been selected.

In the front view window, click on a point just below the roof line of the tunnel, and a short distance to the west of the tunnel.

In the right view window, click on a point just south of the tunnel, at about the same elevation as the red star.

Click on Go .

In the front view window, click on a point above the tunnel, and a short distance to the east of the tunnel.

In the right view window, click on a point north of, and above the tunnel.

Examine both the front and right view windows, to ensure that the rectangles formed in both windows completely enclose the roof line, without completely enclosing any elements below the roof line.

Then click on Go .

> Because the cross-box (cbox) option was selected, all the roof elements (which were completely enclosed by the box) and some sidewall elements (which were crossed, but not enclosed, by the box) have been picked (highlighted).

Select **Object Tools + invisible** to hide the highlighted elements, thus exposing the hole at the tunnel floor.

> Let's fill the hole using the **face** function. A nodeline is required.

Select **Build Polyline + open edge polyline**.

```
1 Polyline Will Be Created, Continue?
```

Press ↵

Select **Build Polyline + polyline→nodeline**.

```
Pick Polyline to Convert:
```

Click on the polyline.

Select **Build Object + face→**

```
Pick a CLOSED nodeline
```

Click on the nodeline; then Click on Go to obtain the FACES screen.

Select **Array Mesh**. Thereafter,

Select **Return**; then Select **Yes**.

Select **Object Tools + visible** to bring hidden elements back into view.

Delete the polyline and nodeline, using **Pick + polyline**, followed by **Pick + nodeline**, and **Object Tools + delete picked**.

The use of the **open edge polyline** function to attach a structure to an existing mesh, or to close a hole caused by the detachment of a branch from an existing mesh, was illustrated in this tutorial. You can delete the mesh now, and proceed to Tutorial 6.

11. Tutorial 6: Ramped Underground Opening

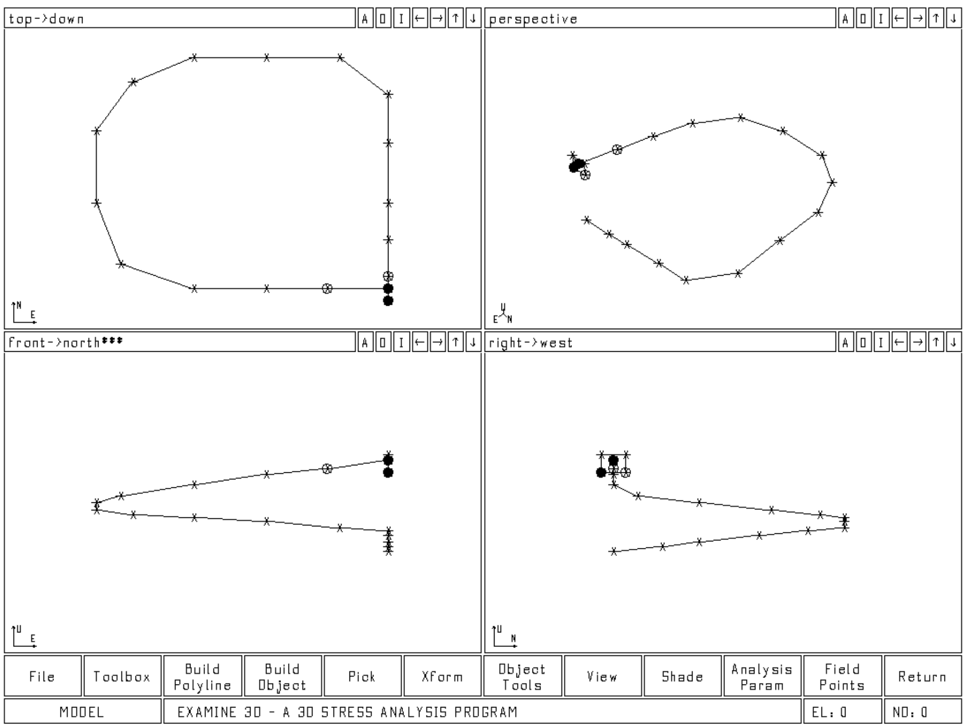


Figure 11.1: Starting polylines for Tutorial 6

The purpose of this tutorial is to illustrate (1) how to read in a .GEO file containing data in an orthogonal coordinate system that is different from the EXAMINE^{3D} coordinate system; (2) how to edit a polyline; and (3) the use of the **extrude** function to define the geometry and discretization for an excavation with nonlinear longitudinal axis.

Startup EXAMINE^{3D} by selecting the EXAMINE^{3D} icon in the Start→Rocscience→Examine3D menu.

Select **Modeler** and retrieve TUT06.GEO using **File + append to model**.

> The picture displayed on your screen is not the same as shown in Figure 11.1. This is because the data in TUT06.GEO has not been interpreted correctly; it is given in a coordinate system different from the EXAMINE^{3D} system. The data is given in the East-North-Down system (1st Coordinate = East; 2nd Coordinate = North; 3rd Coordinate = Down).

Select **Object Tools + delete all**; then Enter Y ↵ to delete everything.

Select **File + coord transform**, to obtain the COORDINATE TRANSFORM sub-menu.

Click repeatedly on **Coordinate 1**, until its value is set to **east**.

Set **Coordinate 2** equal to **north**, using the same procedure.

Set **Coordinate 3** equal to **down**, in the same way.

Select **Save** to effect these settings and exit from the COORDINATE TRANSFORM sub-menu.

Select **File + append to model**, choose the Portable Geometry File type, then select the TUT06 file and open it.

Two polylines are displayed, exactly as shown in Figure 11.1.

Click on the greenish bar at the top of the perspective view window, to maximize it.

> The longer (non-closed) polyline traces one bottom corner of a ramped excavation. The shorter (closed) rectangular one traces the cross-section of the higher elevation end of the excavation. The second polyline will be edited later to modify the cross-sectional geometry of the ramp.

Return to multi-view display, by clicking on the top greenish bar.

Select **Toolbox + setup options**; toggle **Polyline Markers** off; then Select **Save**.

> The polyline markers (blue stars and greenish-yellow discs and circles) are turned off, thereby presenting a cleaner picture of the polylines.

Maximize the right view window.

Select **Toolbox + setup options**; toggle **Polyline Markers** on; set **Grid Spacing** to 0.75; toggle **Grids** on; then Select **Save**.

A very dense grid is displayed.

Select **View + zoom in**.

```
Pick region to ZOOM IN on, Select first corner of region
```

Click on a point about three grid spacings above and about four to the south (left) of the rectangular polyline.

```
Pick opposite corner of region
```

Move the mouse downwards and to the north (right); notice that a rectangle is formed, which expands downwards and northwards as the mouse pointer moves in this direction.

Click on a point about three grid spacings below and about four to the north of the rectangular polyline.

> The rectangular region defined by the two points is enlarged to fill the screen, thereby exposing more details of the rectangular polyline. This polyline will now be edited to properly define the cross-section of the ramp.
> There are four vertices on the cross-sectional polyline. Recall that the 1st and 2nd vertices are marked with a greenish-yellow disc and circle, respectively. Two more vertices will be added to the top edge of the polyline (between the current 3rd and 4th vertices).

Select **Build Polyline + edit polyline**.

```
pick a polyline vertex to move [d=delete, a=add, f=1st pt, Enter=done]:
```

Press `A` to indicate that you wish to add a vertex.

```
pick a polyline edge to insert a vertex on [d=delete, a=add, f=1st pt, Enter=done]:
```

Click on the top edge (between the 3rd and 4th vertices) of the rectangular polyline.

A new vertex is formed at the midpoint of the top edge, dividing it into two segments. Then, the dialogue box request reverts to:

```
pick a polyline vertex to move [d=delete, a=add, f=1st pt, Enter=done]:
```

Click on the newly formed vertex. It becomes highlighted, and the following message appears:

```
New N, U, E Location (Enter=done): [N,U,E: vn=off, s=off, o=off]:
```

Press `S` to toggle grid snap on. Notice that this option changes to `...s=on...` in the dialogue box.

Click on the grid intersection one space to the south (left) and one space above the current position of the picked vertex.

> The vertex is moved to the new location. The two top segments of the polyline, which used to be horizontal, have been changed to sloping segments, one plunging north and the other plunging south.

```
New N, U, E Location (Enter=done): [N,U,E: vn=off, s=off, o=off]:
```

Press ↵ to accept the change.

```
pick a polyline vertex to move [d=delete, a=add, f=1st pt, Enter=done]:
```

Press `A` to add another vertex.

```
pick a polyline edge to insert a vertex on [d=delete, a=add, f=1st pt, Enter=done]:
```

Click on the northward plunging segment formed as a result of the last changes made.

A new vertex is formed at the midpoint of the segment; and the dialogue box message reverts to:

```
pick a polyline vertex to move [d=delete, a=add, f=1st pt, Enter=done]:
```

Click on the new vertex.

```
New N, U, E Location (Enter=done): [N,U,E: vn=off, s=off, o=off]:
```

Press S to toggle grid snap on.

Click on the grid intersection two spaces to the north (right) of the top vertex, i.e., just north of, and above the current position of the picked vertex.

```
New N, U, E Location (Enter=done): [N,U,E: vn=off, s=off, o=off]:
```

Press ↵ to accept the change.

```
pick a polyline vertex to move [d=delete, a=add, f=1st pt, Enter=done]:
```

Press ↵ once more to end the editing session.

Select **Toolbox + setup options**. Then, reset **Grid Spacing** to 3.0; toggle **Grids** off; and Select **Save**.

The picture on your screen should look like Figure 11.2, if everything has been done correctly.

Click on the A near the top right corner of the current window. This gives an **autoscaled** view through the current window.

Return to multi-view display, by clicking on the greenish bar at the top of the screen.

> The polyline which has just been edited describes the cross-sectional geometry of the higher end of the ramped excavation. It will now be extruded along the other polyline, to define the geometry and boundary element discretization for the ramp.

Maximize the perspective view window.

Select **Build Object + extrude**.

```
Select Curve to Extrude
```

Click on the cross-sectional polyline (the one which was edited).

```
Enter/pick extrusion dir ([default]) [N,U,E: vn=off, s=off, o=off]:
```

> In the extrusions performed in previous tutorials, the extrusion direction was always constant and coincident with one coordinate axis. Generally, the extrusion direction *can vary along an arbitrary path*, defined using a polyline. In this example, the polyline which traces one bottom corner of the proposed ramp is used as the extrusion path.

Click on the longer (non-closed) polyline.

```
Use default discretization (y):
```

Press ↵

```
Fix first extrusion contour (y):
```

Press ↵

```
Fix extrusion base (y):
```

Press ↵

```
Creating Extrusion, Please Wait ...
```

Wait!

A boundary element mesh is generated for the surface of the ramped excavation. You may use the **face** function (**Build Object** menu) to discretize the two faces. Use either the **Automatic Mesh** or the **Radial Mesh** functions at the FACES screen. Both would give essentially the same result. (The **Array Mesh** option is not suitable for any of the faces.) The procedure has been explained in previous tutorials.

You may use **Shade + quickshade** to shade the structure. Thereafter, delete it using **Object Tools + delete all**. Then proceed to Tutorial 7.

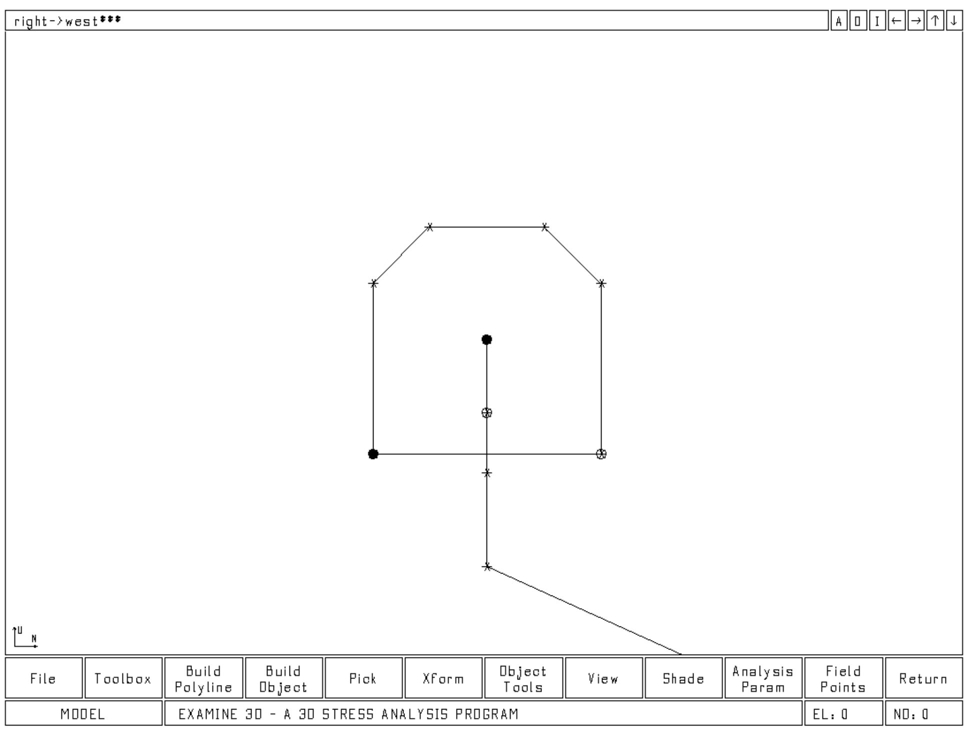


Figure 11.2: Polyline representation of ramp cross-section

12. Tutorial 7: Tunnel Close to a Free Surface

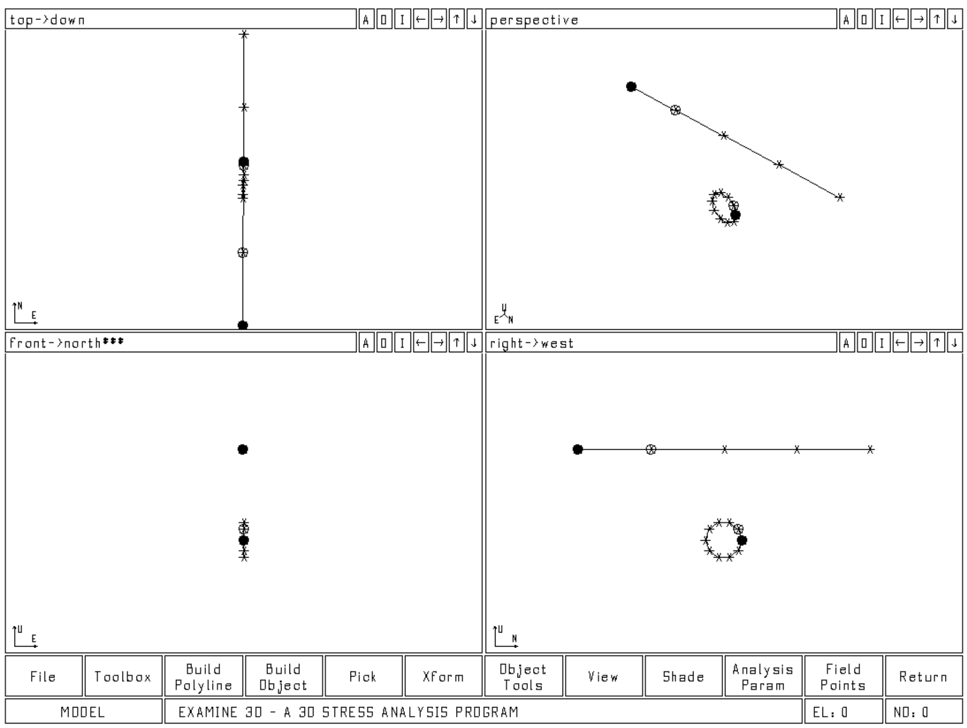


Figure 12.1: Starting polylines for Tutorial 7

The geometry and boundary element discretization for a tunnel close to a free surface will be generated in this tutorial. The use of the element subdivision function (**Object Tools** menu) to refine a mesh will also be illustrated.

Startup EXAMINE^{3D} by selecting the EXAMINE^{3D} icon in the Start→Rocscience→Examine3D menu.

Select **Modeler** and retrieve TUT07.GEO using **File + append to model**.

The picture on your screen should be the same as in Figure 12.1. Two polylines are displayed: a circular polyline, which represents a cross-section of an east-trending tunnel; and a second polyline (horizontal, non-closed), which represents a cross-section of the ground surface.

Maximize the right view window (by clicking on its top greenish bar).

Select **Toolbox + setup options**; set **Grid Spacing** to 5; toggle **Grids** on; then Select **Save**.

The display shows that the tunnel diameter is 5 m, and its distance from the free surface is 10 m. Generally, for a linear elastic medium, a free surface is likely to influence the mechanical behavior of a structure if it is less than about 3 diameters away. COMPUTE$^{3D\text{-}BEM}$ requires that such free surfaces be discretized and incorporated in the analysis.

Turn off the grid lines using **Toolbox + setup options**

Return to multi-view display (by clicking on the top greenish bar).

Select **Build Object + extrude**.

```
Select Curve to Extrude:
```

Click on the non-closed polyline.

```
Enter/pick extrusion dir ([default]) [N,U,E: vn=off,s=off,o=off]:
```

Enter `0 0 1` ↵ to extrude eastwards.

```
Enter length of extrusion [default]:
```

Enter `50` ↵

```
Use default discretization (y):
```

Press ↵

When meshing is done Select **View + autoscale**.

A region of the ground surface, defined by sweeping the polyline 50 m east, is divided into 12 rectangles, each of which is divided diagonally into two triangular elements. Notice that EL = 24 and ND = 20 (bottom right corner of screen).
Let’s refine the north end of the mesh by subdividing the 6 elements there.

Maximize the top view (E-N coordinate) window.

If necessary, adjust the size of the window relative to the object, by clicking repeatedly on the `O` (zoom out) or on the `I` (zoom in) buttons near the top right corner of the window.

Let’s hide all elements south of the north end row of rectangles. Recall that, in this window, N coordinates increase upwards, and E coordinates increase towards the right.

Select **Pick + element**.

```
Pick Element [*=all; b=box; c=cbox; r=ratio; ESC=done]:
```

Press B to activate the box mode.

```
Place box corner #1 [N,U,E: vn=off, s=off, o=off]:
```

Click on a point just above the second horizontal line from the top, and to the left of the mesh.

Click on Go .

```
Place box corner #2 [N,U,E: vn=off, s=off, o=off]:
```

Click on a point to the right of and below the mesh.

Ensure that the yellow rectangle so-formed completely encloses all elements, except those in the top row of rectangles. Then,

Click on Go. All but the 6 north end elements are highlighted.

Press ESC to exit from the **Pick** menu.

Select **Object Tools + invisible** to hide the picked elements.

> The north end elements will now be picked, in order to apply the element subdivision function on them.

Select **Pick + element**.

```
Pick Element [*=all; b=box; c=cbox; r=ratio; ESC=done]:
```

Press C to activate the cross-box mode.

```
Place box corner #1 [N,U,E: vn=off, s=off, o=off]:
```

Click on a point between the two visible horizontal lines, and to the left of the mesh.

Click on Go .

```
Place box corner #2 [N,U,E: vn=off, s=off, o=off]:
```

Click on a point to the right of and above the mesh.

Ensure that the yellow rectangle crosses all, but only, the 6 elements at the north end. Then,

Click on Go .

Press ESC to exit the **Pick** menu.

Select **Object Tools + subdiv elem/poly**

```
This is an irreversible change, are You Sure?
```

Press ⏎

The original 3 rectangles have been subdivided into 12 new ones, each of which is divided diagonally into two triangular elements, giving a total of 24 elements in the refined zone. The subdivision also affected the immediate neighbors of the refined zone. Bring all elements back into view to examine the results.

Select **Object Tools + visible**.

In the original mesh (before the subdivision), 3 elements outside the refined zone shared common boundaries with the zone. These 3 elements have also been subdivided, to form the transition between the refined and non-refined mesh zones. Notice that some of the *transition elements* so-formed have larger values of aspect (base/height) ratio than all other elements.
Let's subdivide the north end elements once more, to further emphasize this point.

Select **Pick + element**.

```
Pick Element [*=all; b=box; c=cbox; r=ratio; ESC=done]:
```

Press B to activate the box mode.

```
Place box corner #1 [N,U,E: vn=off, s=off, o=off]:
```

Define a box which completely encloses all, but only, the elements in the top two rows of rectangles, as follows:

Click on a point between the 3rd and 4th horizontal lines (counting from the top) and to the left of the mesh.

Click on Go .

```
Place box corner #2 [N,U,E: vn=off, s=off, o=off]:
```

Click on a point above and to the right of the mesh.

Ensure that the yellow rectangle so-formed completely encloses all, but only, the 24 top elements. Then,

Click on Go .

Select **Object Tools + subdiv elem/poly**

```
This is an irreversible change, are you sure?
```

Press ↵ to select **Yes**

The 24 north end elements have been subdivided into 96.
Note the shape of the transition elements.
Note that the number of elements in the refined zone (excluding the transition zone) was quadrupled by each application of the subdivision function (first from 6 to 24, and next from 24 to 96).
If an application of the element subdivision function gives rise to a bad mesh (like this one), the only remedy is to delete all the affected elements, including the transition zone (at least), and rebuild the mesh. Let's do that now.

Select **Pick + object**; click anywhere on the mesh, and press the `ESC` button.

Select **Object Tools + delete picked**.

Return to multi-view display (by clicking on the top greenish bar).

Select **Build Object + extrude**, and click on the non-closed polyline.

```
Enter/pick extrusion dir ([default]) [N,U,E: vn=off,s=off,o=off]:
```

Enter `0 0 1` ⏎ to extrude eastwards.

```
Enter length of extrusion [default]:
```

Enter `50` ⏎

```
Use default discretization (y):
```

Enter `n` ⏎

```
Enter discretization for red segment [1]:
```

There are 4 segments in the polyline, one of which is currently highlighted red. Therefore, the above request will appear 4 times. Press ⏎ to accept the default [1] each time. The following request appears after the fourth one:

```
Enter number of divisions along length [default]:
```

Enter `4` ⏎

When meshing is done, notice that EL = 32 and ND = 25.

Select **Shade + shade options**; toggle **Normals** on; then Select **Save**.

Select **Shade + quickshade**.

The display shows that the normals are pointing upwards. They should point downwards, i.e., into the rock (outwards from the open space bounded by the boundary elements). We'll use the **Object Tools + locate surface** option to tell EXAMINE^{3D} that an open boundary exists that defines a ground surface. This is also required for correct computation of results in COMPUTE$^{3D\text{-}BEM}$.

Press `ESC` .

Select **Pick + object**; click anywhere on the mesh defining the surface, and press the `ESC` button.

Select **Object Tools + locate surface**.

```
Choose Mode (0=remove,1=add) [default]:
```

Enter `1` ⏎

The surface now changes color to green to signify that it is a ground surface. Let's look at the element normals to see how they have changed.

Select **Shade + quickshade**.

The normals are now correctly defined and pointing down into the rock mass.

Press `ESC` .

Let's use the **extrude** function to generate the geometry and boundary element mesh for the tunnel itself.
First, **move** the circular polyline 10 m east, then **extrude** it 30 m east to generate a 30 m long tunnel. This leaves the free surface extending 10 m beyond the east and west ends of the tunnel.

Select **Pick + polyline**, and click on the circular polyline.

Select **Xform + move**.

```
Enter Rel Translation [N,U,E: vn=off, s=off, o=off]:
```

Enter `0 0 10` ↵

Then click on `Go` to accept the change.

Select **Build Object + extrude**, and click on the circular polyline.

```
Enter/pick extrusion dir (0,0,1)[N,U,E: vn=off, s=off, o=off]:
```

Press ↵ to accept (0,0,1), i.e., eastward extrusion.

```
Enter length of extrusion [default]:
```

Enter `30` ↵

```
Use default discretization (y):
```

Press ↵

When meshing is done, notice that the EL and ND counters are updated to 232 and 135, respectively.

Mesh the two faces of the tunnel, using **Build Object + face**→ to obtain the FACES screen; select **Automatic Mesh**, then **Return** and **Yes**. The procedure has been explained in previous tutorials.

The next task is to set values for the material parameters and in situ stress components. The elevation (U-coordinate) of the ground surface is required.

Select **View + loc & dist**.

```
P1(current position)[N,U,E: vn=off, s=off, o=off]:
```

Press `V` to toggle vertex snap on.

Click anywhere on the ground surface mesh. The values in parenthesis in the dialogue box are updated to give the coordinates of the vertex closest to the point clicked. The second value is the U-coordinate; it is equal to -5 m for this example.

Press `ESC` .

Select **Analysis Param + enter parameters** to obtain the MODEL PARAMETERS menu.

The top field gives default values of 30000 MPa and 0.25 for Young's modulus and Poisson's ratio. Change the value of Young's modulus to 40000 MPa by clicking on the 30000 and typing `40000` ↵ .

> The next field gives two options for specifying field stress. The CONSTANT option is the default, but free surface analysis requires the GRAVITATIONAL option.

Click on GRAVITATIONAL to select it.

Click on the white box to the right of **Surface Elev (m);** then type `-5` ↵ .

The default value of 0.025 MN/m^3 for unit weight is okay.

> The next fields define the directions of the principal stress components, and the values of the horizontal to vertical stress ratio. The trend (**Dir**) is in degrees from North, and the inclination (**Dip**) is in degrees below the horizontal.
> Accept the default values.
> Let's use the **eye+target** function to walk around the structure. Save your work before proceeding.

Select **Save** to exit from the MODEL PARAMETERS menu.

Select **File + save file**.

Select the EXAMINE-3D Analysis File Format (*.ex3) from the Files of type drop down list. Then in the File name edit box type tut07.ex3 as the filename.

Wait for the message `Done Storing` to appear; then

Select **Toolbox + setup options**; set **Grid Spacing** equal to 50; toggle **Grids** on; then Select **Save** to exit from the menu.

Select **Shade + shade options**; toggle **Elements** off; toggle **Normals** off; then Select **Save**.

Select **View + eye+target**.

```
Shade Objects in Perspective Window?
```

> The **eye+target** function works by using the top, front and right view windows to position the eye, while the three-dimensional picture seen from the current viewpoint is shown in the perspective window.
> The perspective window is redrawn each time the eye position is modified. Therefore, shading the object at the same time may slow the process down, depending on the degree of complexity of the object. Let's request shading anyway, to see what happens.

Press ↵

> The current eye position is about 45 degrees above the ground surface and approximately northeast of the structure; therefore, the line of sight is directed southwest, at a plunge of about 45 degrees.
> Let's move the eye vertically down.

In the front or right view window, click on a point vertically below the current eye position.

> The U-position of the eye changes to the point clicked (its E-N position is essentially unchanged), and the perspective view is updated accordingly.

Following the same procedure, move the eye vertically down, about half a grid spacing (25 m) at a time. Examine the perspective window carefully at each eye position.

When the eye is at about half a grid spacing below the tunnel floor, transfer the mouse pointer to the top view (E-N coordinate) window.

Move the eye southwards in 50 m increments (keeping its E-position fixed); then move it westwards the same way; then northwards, and finally eastwards back to its starting position. Examine the result of each 50 m increment before proceeding to the next.

> As this exercise may have shown, the **eye+target** function lets one *walk around* a structure, to examine it from any conceivable angle. Explore the function as much as you wish. Then,

Return the eye to its original position (approximately northeast of the tunnel, at an angle of about 45 degrees above the horizontal).

Press ESC to exit from the **View** menu.

> Let's run an **object check** on the mesh. Usually, **object check** should not be run on a free surface, because of the risk of the normals being reversed to point in the wrong direction. It is advisable to discretize the underground excavations first, run **object check** on the mesh, and add the free surface afterwards. However, let's make the mistake this one time, to see what happens.

Select **Toolbox + object check**. Respond with ↵ to all the requests. At the end, the following message is displayed:

```
Geometry is LEAKY and invalid for analysis -- Press Enter
```

> Notice that red lines have been drawn around the free surface. The **object check** function draws red lines around *open edges* to indicate where the geometry is *leaky*. The open edges in this case are however okay, because the ground surface is not a closed surface.
> Open edges in the tunnel mesh would have been a problem, but none were detected. Therefore, contrary to the above message, the geometry is NOT LEAKY.
> The free surface is the only case for which the LEAKY verdict of the **object check** function can be disregarded.

Press ↵. Proceed with the clockwise elements check. When it is done,

Maximize the front view window.

Select **Shade + shade options**; toggle **Elements** off and **Normals** on; and exit from the menu.

Select **Shade + quickshade**.

Ensure that all normals point into the rock mass, i.e., outwards from the open space enclosed by the boundary elements. If the free surface normals point upwards, reverse them using **Object Tools + reorder nodes/vert** (having first **Pick**ed the free surface).

> The next task is to define a grid box (to specify points at which analysis results should be computed).

Press ESC to exit from **Shade**.

Return to multi-view display.

Select **Toolbox + setup options**; set **Grid Spacing** to 5; ensure that **Grids** is on; then exit from the menu.

Select **Field Points + add points**.

```
Enter Input Method (1=plane,2=3pt plane,3=grid,4=file,5=pt,6=line) [default]
```

Enter 3 ↵ to activate the grid box mode.

```
Locate box corner #1 [N,U,E: vn=off, s=off, o=off]:
```

Press S to toggle grid snap on.

In the right view window, click on the midpoint of the ground surface line.

Click on Go to accept this as the first corner.

```
Locate box corner #2 [N,U,E: vn=off, s=on, o=off]:
```

In the right view window, click on a point 10 m north of the free surface, and 10 m below the tunnel floor.

In the front view window, click on a point 10 m east of the free surface, and 10 m below the tunnel floor.

Ensure that the yellow rectangle formed in the front and right view windows encloses half of the tunnel in each window, and that its top edge coincides with the ground surface. Then,

Click on Go .

Respond with ↵ to the requests that follow, thereby accepting the default discretization for the grid box.

Select **Analysis Param + analysis options** to obtain the ANALYSIS OPTIONS menu. Click a few times on **Point Filter**, until its value is set to "surface"; then exit from the menu.

> The **Point Filter** function is used to filter out field points that might lie above the ground surface or on the inside of underground openings. If it is set to "on", only points on the inside of openings are filtered out; if it is set to "surface", (as in this example), then points above (i.e., on the outside of) the ground surface are also filtered out. Points filtering is turned off by setting **Point Filter** to "off ".

Toggle grid lines off (F7 or **Toolbox + setup options**).

Select **View + autoscale**.

Delete the polylines and nodelines, using **Pick + polyline**, $\boxed{*}$, **Pick + nodeline**, $\boxed{*}$, and **Object Tools + delete picked**.

Select **File + save file**.

Select the EXAMINE-3D Analysis File Format (*.ex3) from the Files of type drop down list. Then in the File name edit box type tut07.ex3 as the filename. Respond YES to overwriting the existing version of the file.

Wait for the message `Done Storing` to appear; then

Select **Return**; Select **Exit**; then Select **Yes**, to exit from EXAMINE^{3D}.

Submit the TUT07.EX3 to COMPUTE$^{3D\text{-}BEM}$ sometime, when you do not need your computer for a while.

13. Tutorial 8: Transition Skin

The **transition skin** function allows one to vary the level of detail admitted in the geometrical description of adjacent sections of a structure. Its use will be illustrated using a 30 m long, 4 m diameter, east-west trending tunnel. The geometric description of the cross-section will be more detailed within 2 m long sections at both ends, but less detailed elsewhere. There will be a 1 m long transition zone between the end and middle sections.

The model will be built from scratch.

Startup EXAMINE^{3D} by selecting the EXAMINE^{3D} icon in the Start→Rocscience→Examine3D menu.

Select **Modeler**.

Select **Build Polyline + new polyline**; then Press `I` to activate the circle mode.

```
Enter circle radius [default]:
```

Enter `2` ↵

```
Enter number of line segments/circle [default]:
```

Enter `16` ↵

Select **View + autoscale**.

> The circle is oriented normal to the E-axis, and its center is at (0,0,0). This default orientation is in agreement with the desired east-west trend of the tunnel. Let's assume that the *length* of the tunnel is also to be centered at (0,0,0). In that case the current polyline, which represents an end section, needs to be moved 15m westwards.

Select **Pick + polyline**; and click on the circle.

Select **Xform + move**.

```
Enter Rel Translation [N,U,E: vn=off, s=off, o=off]
```

Enter `0 0 -15` ↵; then click on `Go` .

Select **View + autoscale**.

This polyline will form the west end of the tunnel. Let's place copies of it at 2, 28 and 30 m from the west end, to define the boundaries of the end sections.

Select **Pick + polyline**; then click on the polyline.

Select **Xform + copy**.

```
Enter Rel Displ of Copy [N,U,E: vn=off, s=off, o=off]:
```

Enter `0 0 2` ⏎

Enter `0 0 28` ⏎

Enter `0 0 30` ⏎

Then click on `Go` .

Select **View + autoscale**.

Boundaries of the end sections have been defined using circular 16-vertex polylines. Boundaries of the middle section will be defined using circular 8-vertex polylines. The number of vertices on adjacent polylines have to be related by a 2:1 ratio if the **transition skin** function will be used to connect such polylines.

Select **Build Polyline + new polyline**; then Press `I` to activate the circle mode.

```
Enter circle radius [1]:
```

Enter `2` ⏎

```
Enter number of line segments [default]:
```

Enter `8` ⏎, or Press ⏎ if the [*default*] is 8.

The new circle formed is centered at (0,0,0). It is best viewed through the top, front and perspective view windows. Let's move it 12 m west, and then place a copy of it at 3 m from the east end, to define the boundaries of the middle section (thereby creating 1 m long transition zones at each end).

Select **Pick + polyline**; then click on the middle polyline (in the front, top, or perspective view window).

Select **Xform + move**.

```
Enter Rel Translation [N,U,E: vn=off, s=off, o=off]:
```

Enter `0 0 -12` ⏎; then click on `Go` .

Pick the same polyline again. Then,

Select **Xform + copy**.

```
Enter Rel Displ of Copy [N,U,E: vn=off, s=off, o=off]:
```

Enter `0 0 24` ⏎; then click on `Go` .

Maximize the front view window.

Select **View + zoom in**.

```
Pick region to ZOOM IN on, Select first corner of region:
```

Click on a point just above and to the west of the west end polyline.

```
Pick opposite corner of region
```

Move the mouse pointer to the east and down. Notice that a rectangle is formed, which grows eastwards and downwards as the mouse pointer moves in this direction. When the three west end polylines are just completely enclosed by the rectangle, click the mouse button.

> Let's generate the mesh, using the **skin** function for the end and middle sections, and the **transition skin** function for the transition sections.

Select **Build Object + skin**.

```
Pick a skin polyline [0 picked, Select Go when done]:
```

Click on the west end polyline.

```
Pick a skin polyline [1 picked, Select Go when done]:
```

Click on the next polyline.

```
Pick a skin polyline [2 picked, Select Go when done]:
```

Click on `Go` .

```
Use default discretization (y):
```

Enter `n` ⏎

```
Enter # of interp sections between red sections [default]:
```

Enter `3` ⏎

Three interpolation polylines are inserted between the **pick**ed two; one segment of the polylines is highlighted red, and:

```
Enter discretization for red segment [1]:
```

Press ⏎. The request will appear 16 times. Respond with ⏎ each time. The elements will be created after your 16th response. Notice that EL=128 and ND=80.

> The next tunnel section is bounded by a 16-vertex polyline on the west and an 8-vertex polyline on the east. The only way to connect them is through the **transition skin** (or **blend**) functions. In this case, it is appropriate to use the **transition skin** function.

Select **Build Object + transition skin**.

```
Select first transition section:
```

Click on the boundary between the meshed and unmeshed zones.

```
Select second transition section:
```

Click on the next polyline, towards the east.

> Examine the mesh to see how the transition elements are formed. Notice that EL=152 and ND=88.

Click on the [A] near the top right corner to autoscale the window.

Select **Build Object + skin**, to discretize the middle section.

```
Pick a skin polyline [0 picked, Select Go when done]:
```

Click on the boundary between the meshed and unmeshed zones.

Click on the next polyline, to the east.

Click on [Go] .

```
Use default discretization (y):
```

Enter [n] ↵

The two polylines which bound the middle section are highlighted; then:

```
Enter # of interp sections between red sections [default]:
```

Enter [7] ↵ to obtain 3 m long sections.

> Generate the east end transition zone and the east end section. You may **zoom in** on this region if you want.

Select **Build Object + transition skin**.

Click on the boundary between the meshed and unmeshed zones.

Then click on the next polyline to the east.

When the transition zone meshing is done,

Select **Build Object + skin**; click on the boundary between the meshed and unmeshed zones; click on the east end polyline; then click on [Go] .

```
Use default discretization (y):
```

Enter [n] ↵

```
Enter # of interp sections between red sections [default]:
```

Enter $\boxed{3}$ ↵

When meshing is done, return to multi-view display.

You can explore the mesh as you wish, using the **Shade** and/or **View** menus. Thereafter, delete everything, and proceed to Tutorial 9.

14. Tutorial 9: Using Interpret

The analysis results in TUT02.RES will be used in this tutorial to illustrate some of the **Interpret** group of functions. It is assumed that the user has completed Tutorial 2 and the associated boundary element analysis (using COMPUTE$^{3D\text{-}BEM}$).

Startup EXAMINE^{3D} by selecting the EXAMINE^{3D} icon in the Start→Rocscience→Examine3D menu.

Select **Interpret**.

Select **Stress** (from the DATASETS screen). This will be available if the parameter “general=on” in your e3.cfg file. See the beginning of Chapter 3 for details. If the parameter “general=off”, then you will be prompted to enter a filename immediately after selecting **Interpret**.

When The Open File dialog appears, click on TUT02 to select it. The DATA SELECTION screen appears, showing the variables that are available to be examined (Figure 1.4).

Select **Sigma 1**, to examine values of the maximum principal stress, σ_1

The **Interpret** DATA INTERPRETATION screen (Figure 1.5) appears. The values of the elastic properties, field stress, strength parameters, and model parameters are summarized in the **Interpret Data** column, at the right edge of the screen. The legends for the isosurfaces and contours are given at the bottom right of the screen. The rest of the screen gives the usual four view windows, viz., the right, front, top and perspective view windows.

14.1 Contoured Planes

The **cutting plane** function allows for the examination of three-dimensional data on different planes through the structure (provided that results have been computed on the planes).

Select **Cutting Plane + cutting plane**.

```
Autoscale Grid Box (n)?
```

Responding with *yes* to this question causes the grid box to be maximized in each window. Otherwise, each window would be left *as is*. Let's choose *yes*.

Enter Y ⏎

> A view of the grid box is shown in each window, with two orthogonal axes (three in the perspective window) highlighted yellow. The orthogonal axes are local coordinate axes for the grid box, labeled u-v-w. These axes are not necessarily coincident with the global N-U-E axes, but they are in this example: the w-axis corresponds to the global N-axis, the u-axis to the global E, and the v-axis to the global U.
> The following options appear in the dialogue box:

```
snap=s, sweep=w, shade=h, Go=done [u:v:w]:
```

Press W to activate the sweep mode.

Press H and wait; when shading is done,

Click anywhere on the w-axis (yellow horizontal line in the right view window). A contoured cutting plane is displayed, which is normal to the w-axis, and passes through the point clicked.

> The w-position of the cutting plane can be changed by clicking another point on the w-axis.
> Vertical cutting planes normal to the tunnel axis are obtained by clicking on the w-axis (either in the perspective, right, or top view windows).
> Vertical cutting planes longitudinal to the tunnel are obtained by clicking on the u-axis (either in the perspective, front or top view windows).
> Horizontal cutting planes are obtained by clicking on the v-axis.

Examine 5 different vertical cutting planes normal to the tunnel axis, by clicking on 5 different points, along the w-axis.

Do the same on the u-axis, obtaining vertical longitudinal planes.

Examine the same number of horizontal planes, by clicking on the v-axis.

Press ESC to exit from **Cutting Planes**.

Select **View + autoscale**.

> Exiting from the **cutting plane** function with ESC, as in the above example, causes the cutting plane(s) to be discarded. To retain a cutting plane, exit from the function by clicking on Go .

Select **Cutting Plane + cutting plane**.

```
Autoscale Grid Box (n)?
```

Press ⏎

Click on the w-axis, at a location near the south end of the grid box (middle of tunnel), preferably in the right view window. Then click on Go .

Select **Cutting Plane + cutting plane**; then Press ⏎ .

Click on the w-axis again, this time at a location near the fine/coarse mesh boundary. Then click on Go .

Using the same procedure, create another cutting plane normal to the tunnel axis, at the north end of the tunnel.

Now create a horizontal cutting plane close to the tunnel roof, using the same procedure (except that the v-axis should be clicked this time).

Maximize the perspective window to examine the contours more closely. Then return to multi-view display.

Select **View + eye+target**.

```
Shade Objects in Perspective Window (n):
```

Enter `y` ↵

> The default eye position is approximately north-east of, and about 45 degrees above, the tunnel. Let's leave the eye at the default elevation, but modify its N-E position. The top view window is best for this.

In the top view window, click on a point south of the tunnel and on the same vertical line (E-coordinate) as the current eye position. Then examine the results through the perspective window.

In the top view window, click on a point west of the tunnel and on the same horizontal line (N-coordinate) as the current eye position. Examine the results through the perspective window.

Using the same procedure, move the eye north of the tunnel, without changing either its east position or elevation. Then examine the results.

Next move the eye east of the tunnel, keeping its north position and elevation unchanged (this should bring the eye back to its default position, more or less).

Repeat the above exercise with the eye at an elevation below the tunnel. Thereafter, move the eye back to its default position. Then,

Press `ESC` to exit from **View**.

Delete the cutting planes, using **Contour Tools + delete all**.

14.2 Full Three-Dimensional Visualization

The **Volume Data** menu provides access to a number of functions used for full three-dimensional visualization.

Select **Volume Data + surf contours**.

> The displayed contours describe the variation of σ_1 on the tunnel wall surface. Back planes are not visible when objects are colored. Therefore, the top view describes the variation in the roof area, the right view gives the variation above and below the spring line on the eastern sidewall, whereas the front view describes the variation on the south end face.

Select **Contour Tools + delete all** to delete the contours.

Select **Volume Data + create isosurf**.

```
Enter isovalue:
```

An isosurface is a surface joining all points in space (within the bounds of the grid box) at which a selected variable (in this case σ_1) has the same value. The isovalue is the value of the variable on the isosurface. Let's request the σ_1 = 60 MPa isosurface.

Enter `60` ↵

The σ_1 = 60 MPa isosurface is created, and its color key appears in the Isosurface Legend (bottom right of screen).

When the message `isosurface successfully created` appears,

Select **Pick + object**; then Press `*` to select all elements.

Select **Object Tools + invisible**.

The entire tunnel is hidden, leaving only the isosurface and grid box. Recall that the grid box encloses only the upper north-east octant of the tunnel; therefore, only the corresponding portion of the isosurface is available. You can maximize any of the windows to view the isosurface more closely. You can also shade it using **Shade + quickshade**; then exit from **Shade** by pressing `ESC` .

Select **Object Tools + visible**.

Select **Shade + shade options**; toggle **Elements** off (if they are on); then Select **Save & Exit**.

Select **Shade + quickshade**.

quickshade shades only one window at a time. Click anywhere in a given window to obtain a shaded view through it. Obey the `Please Wait ...` prompt after each click. Then press `ESC` to exit from **quickshade**.

Select **Pick + isosurface**.

```
Pick isosurface [*=all; ESC=done]:
```

Click anywhere on the isosurface, and notice that it is highlighted to indicate that it has been **pick**ed.

Press `ESC` to indicate that all required isosurfaces have been **pick**ed (there could have been more than one).

Select **Shade + set color**.

The **set color** sub-menu appears. Any changes made here will apply to all the **pick**ed isosurfaces. Therefore, if it is required to change the colors of two or more isosurfaces, this function should be applied to them individually (to assign a different color to each). There is only one in this example.
An isosurface consists of a series of connected polygons; and the **set color** function provides for a choice of colors for the faces and edges of such polygons. Both are assigned the same color by default.

FACES is currently highlighted. Click on any color in the color bar to select it for the faces.

Then click on EDGES. Choose a color for it by clicking on the required one in the color bar. For this exercise, choose a different color for the edges than for the faces.

Select **Save & Exit**. Notice that the isosurface legend is updated to reflect the new colors.

Select **Shade + quickshade**, to view the effect of the color changes.

Press ESC to exit from **Shade**.

Select **Pick + isosurface**; Press *; then Select **Object Tools + delete picked** to delete the isosurface.

> Contours and isosurfaces present the *magnitudes* of the principal stress components. Their *directions* can be visualized using the **traj ribbons** function.

Select **Volume Data + traj ribbons**.

```
Enter Tracer Input Method (1,2=line; 3,4=plane) [default]:
```

Enter 1 ⏎; or Press ⏎ if the [*default*] equals 1.

```
Enter first point [N,U,E: vn=off, s=off, o=off]:
```

In the top view window, click on a point about midway between the two vertical red lines (grid box edges), and just above the lower horizontal red line. Then click on Go .

```
Enter second point [N,U,E: vn=off, s=off, o=off]:
```

In the top view window, click on a point vertically above the first point and close to the top horizontal red line. Then click on Go .

```
Enter number of intervals [default]:
```

Enter 4 ⏎

Wait for the 5 ribbons to be plotted.

> The trajectory ribbon is like a stress flow path. The long dimension of the ribbon coincides with the direction of the maximum principal stress; the width direction coincides with the direction of the intermediate principal stress; the normal to the ribbon surface coincides with the direction of the minimum principal stress.
> Delete the ribbons and repeat the process, using a different tracer input option, as follows:

Select **Contour Tools + delete all**.

Select **Volume Data + traj ribbons**.

```
Enter Tracer Input Method (1,2=line; 3,4=plane) [default]:
```

Enter 2 ⏎

Define a line in the top view window, as was done for the first set of ribbons. When the line is done, the following message appears:

```
Enter number of intervals [default]:
```

Enter 4 ↵

> A set of 5 *polylines* are drawn through the grid box, each directed along the direction of the maximum principal stress.
> The **traj ribbons** function draws ribbons *or* polylines, which indicate the directions of the principal stress components. Each ribbon (or polyline) originates from a point, referred to as the *tracer input point*. There are four options for specifying the tracer input points: (1) Ribbons are grown from tracer points, distributed along a line (2) Trajectory polylines are grown from tracer points distributed along a line (3) Ribbons are grown from tracer points distributed on a plane (4) Trajectory polylines are grown from tracer points distributed on a plane.
> Let's explore option 3 to see what happens.

Select **Contour Tools + delete all** to delete the trajectory polylines.

Select **Volume Data + traj ribbons**.

Enter 3 ↵

```
Input plane point #1 [N,U,E: vn=off, s=off, o=off]:
```

In the right view window, click on a point close to the lower left corner of the grid box, but still within the box. Then click on Go .

```
Input plane point #2 [N,U,E: vn=off, s=off, o=off]:
```

In the same window, click on a point close to the upper right corner of the grid box, and still within the box. Then click on Go .

```
Enter number of divisions in u direction [default]:
```

Enter 3 ↵

```
Enter number of divisions in v direction [default]:
```

Enter 2 ↵

When the 12 ribbons are done, maximize the perspective window. The ribbons show how the directions of the principal stress components change within the grid box.

Return to multi-view display.

Select **Contour Tools + delete all** to delete the trajectory ribbons.

Select **Return**.

This concludes our examination of σ_1 data. Either σ_2 or σ_3 can be examined in exactly the same way, by selecting **Sigma 2** or **Sigma 3**, respectively. The parameter **Str Fac** can also be examined in the same way, except that there are no trajectories for it (being a scalar quantity).

Select **Str. Fac**. Wait.

Select **Volume Data + create isosurf**.

```
Enter isovalue:
```

Enter `1` ↵

When the message `isosurface successfully created [#polygons =  [NUM]]` appears,

Select **Shade + quickshade**.

Recall that you click anywhere in a window to obtain a shaded view through it.
Values of **Str Fac** ≤ 1 indicate over-stressed rock (relative to the current strength criterion).

Press `ESC` to exit from **Shade**.

Press **Ctrl-X** to quick exit from EXAMINE^{3D}, or exit from the main menu by:

Select **Return** (to return from the DATA INTERPRETATION screen);

Select **Return** (to return from the DATA SELECTION screen);

Select **Return** (to return from the DATASETS screen);

Select **Exit**; then Select **Yes** to end the EXAMINE^{3D} session.

15. Tutorial 10: Intersection of Two Openings

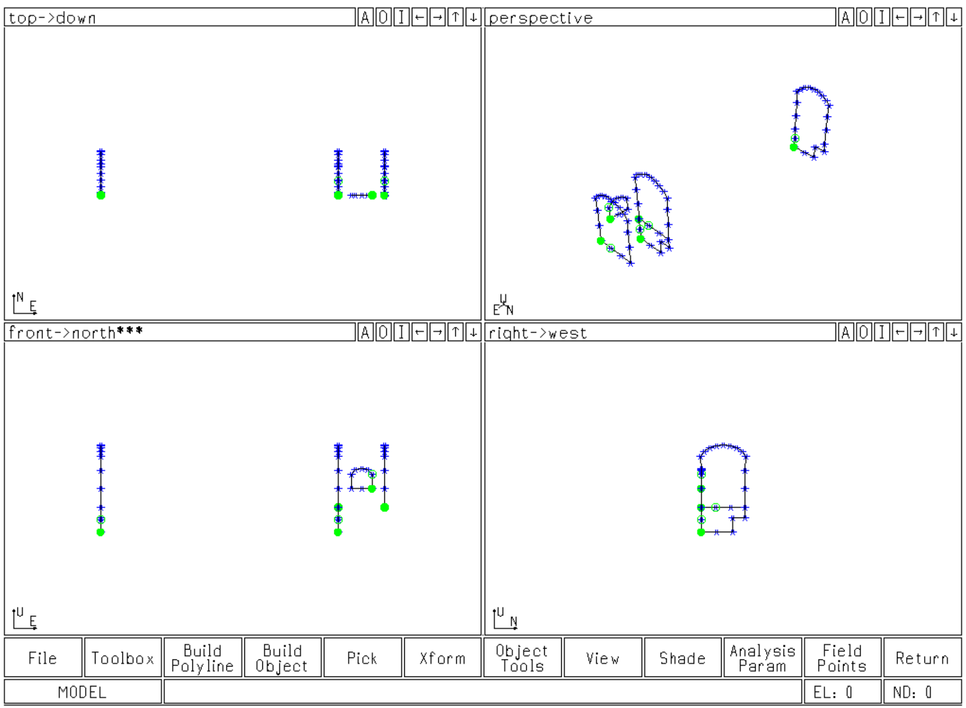


Figure 15.1: Starting polylines for Tutorial 10

In this tutorial, a boundary element mesh will be generated for two intersecting openings. The first opening represents an underground powerhouse while the second opening is an access tunnel which intersects the power house as shown in figure 15.2.

Startup EXAMINE^{3D} by selecting the EXAMINE^{3D} icon in the Start→Rocscience→Examine3D menu.

Select **Modeler**.

Select **File + append to model**.

Select the Portable Geometry File Format (*.geo) from the Files of type drop down list. Click on TUT10 to select it and then press the Open button.

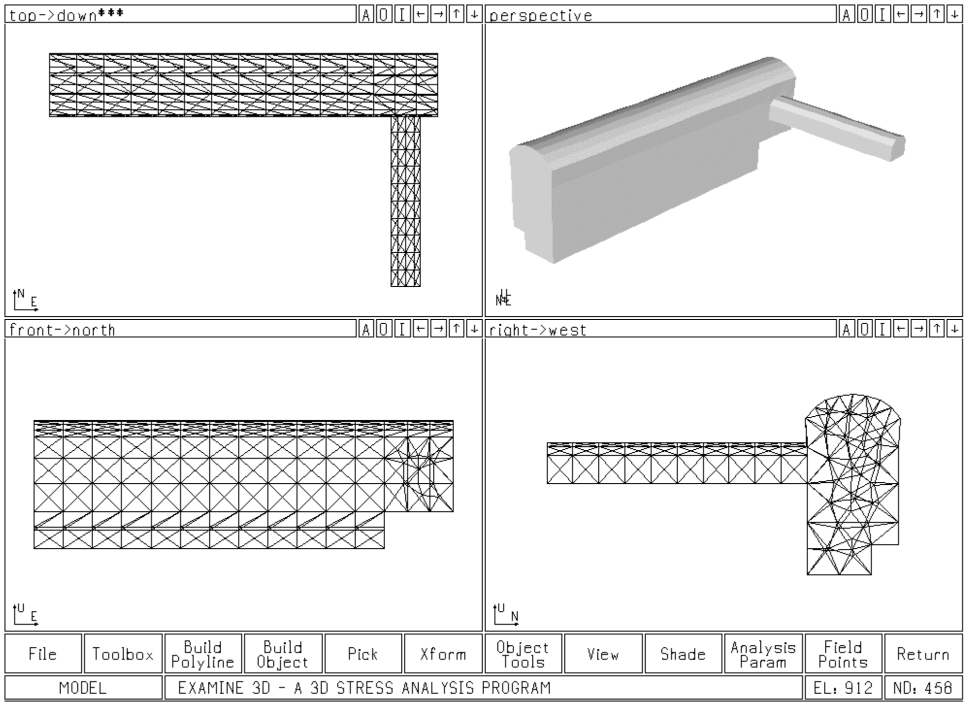


Figure 15.2: Model Geometry for Tutorial 10 after Construction

Click on the greenish bar at the top of the front view (E-U coordinate) window, to maximize it. Press the F6 function key to turn on the coordinates.

Select **Build Object + skin**.

```
Pick a skin polyline [0 picked, Select Go when done]:
```

Click on the western-most polyline. This polyline is at an easting of 900. The easting coordinate of the mouse can be viewed in the greenish bar at the top of the view. The first coordinate represents the horizontal direction (easting in this case) while the vertical direction is the second coordinate (up direction for the front view).

```
Pick a skin polyline [1 picked, Select Go when done]:
```

The next polyline is the next western most polyline at an easting of 994.5. Notice that there are two overlapping polylines at this location. EXAMINE^{3D} is smart enough to recognize that the polyline that you want to select for the skinning process must have the same number of vertices as the first polyline selected. As a result you may select anywhere on the polyline. Also notice that where the two polylines overlap or cross, they have ***coincident*** vertices. This is ***strictly*** required by EXAMINE^{3D}.

```
Pick a skin polyline [2 picked, Select Go when done]:
```

Ensure that the two polylines are chosen; then,

Click on Go .

```
Use default discretization (y):
```

Press ↵

```
Enter mesh density factor [1]:
```

Press ↵ to accept the default mesh density factor of 1.

```
Continue with Element Generation? (y):
```

> Notice that 12 interpolation sections have been formed, using 11 temporary interpolation polylines and the two selected polylines. If you entered n ↵ in response to the dialogue box request, the interpolation polylines would be removed, the original selected polylines would be deselected, and the program would exit from the **Build Object** menu. In that case, you would have to repeat the process, starting from “Select **Build Object + skin**”.

The proposed discretization will be accepted. Therefore,

Press ↵

Select **Build Object + skin**.

```
Pick a skin polyline [0 picked, Select Go when done]:
```

Click on the eastern-most polyline. This polyline is at an easting of 1013.

```
Pick a skin polyline [1 picked, Select Go when done]:
```

Once again, there are two overlapping polylines at the next location. Click on the upper most, smaller polyline at an easting of 994.5 remembering that EXAMINE^{3D} will automatically choose the polyline with the same number of vertices.

```
Pick a skin polyline [2 picked, Select Go when done]:
```

Click on Go .

```
Use default discretization (y):
```

Press ↵

```
Enter mesh density factor [1]:
```

Press ↵ to accept the default mesh density factor of 1.

```
Continue with Element Generation? (y):
```

> Notice that 3 interpolation sections have been formed, using two temporary interpolation polylines and the two selected polylines.

Press ↵

Click on the greenish bar at the top of the front view (E-U coordinate) window, to return to all four views. Press the F6 function key to turn off the coordinates.

Select **Build Object + face**→

```
Pick a CLOSED nodeline
```

Click on the nodeline at the east face of the eastern-most opening.

```
Pick interior CLOSED nodeline [Go = done]:
```

This face is defined by one nodeline. Therefore,

Click on Go to obtain the FACES screen.

Select **Automatic Mesh**.

When meshing is done, Select **Return**; then Select **Yes**.

Next, discretize the other end face on the western most end of the opening.

Select **Build Object + face**→

```
Pick a CLOSED nodeline
```

Click on the nodeline at the west face of the western-most opening.

```
Pick interior CLOSED nodeline [Go = done]:
```

This face is defined by one nodeline. Therefore,

Click on Go to obtain the FACES screen.

Select **Automatic Mesh**.

When meshing is done, Select **Return**; then Select **Yes**.

The transition faces will now be discretized. This is the section defined by easting 994.5 and represents the change in geometry where the floor of the powerhouse changes from an elevation of 928.5 to 938.5. At this location, only the region between the large polyline and small polyline needs to be meshed since this area represents the rock surface.

Proceed as follows:

Select **Build Object + face**→

```
Pick a CLOSED Nodeline
```

Click on the larger nodeline, at the junction of the eastern and western sections of the powerhouse (at easting 994.5). Select it by noting that it does not overlap the smaller nodeline on the floor section. You may want to enlarge the perspective view and zoom into the region around these two nodelines.

```
Pick interior CLOSED Nodeline [Go = done]:
```

Click on the smaller nodeline by selecting its floor segment as well. Ensure that both nodelines at this transition face are highlighted; then

Click on [Go] to obtain the FACES screen.

Select **Automatic Mesh**.

When meshing is done, examine the mesh to see if the elements are of the same general size, more or less. If not, exploit the randomness of the **Automatic Mesh**ing process as follows: first Select **Reset** to discard the mesh; then Select **Automatic Mesh** again; repeat the process until the mesh appears satisfactory. Thereafter,

Select **Return**; then Select **Yes**.

The discretization of the main powerhouse excavation is complete. It is useful to shade the structure, to obtain a clearer display of the geometry.

Select **Shade + shade options**.

At the SHADE OPTIONS menu, toggle **Elements** off; then Select **Save**. Thereafter,

Select **Shade + quickshade**. Wait.

After viewing the shaded three-dimensional picture of the opening, press [ESC] .

The **object check** function in the **Toolbox** menu should now be used to ensure that the mesh satisfies some basic rules. Proceed as follows:

Select **Toolbox + object check**. Then continue according to the following dialogue:

```
Check object, element, and node numbering (y):
```

Press ↵

```
Check for zero area elements (y):
```

Press ↵

```
Check triangular element base/height ratios (y):
```

Press ↵

```
Maximum ratio = [rmax]; element number [Nel] -- Press Enter to Continue
```

Press ↵

```
Check for invalid overlapping elements (y):
```

Press ↵

```
Check for invalid intersecting elements (y):
```

Press ↵

```
Check for leaky objects (y):
```

Press ⏎

```
Geometry is NOT LEAKY
```

Splendid!

The powerhouse object is now a valid mesh that could be analyzed on its own. But we want to add an access tunnel intersecting the eastern end of the structure. Notice that the polyline section of the access was included with the geo file and can easily be seen in the front view at around 1003 easting, 950 up.

The first step in the process of attaching two objects is to cut a hole in the larger structure (powerhouse) that the smaller structure (access tunnel) can fit into. The second step is to mesh between the boundary of the cut out region in the larger structure and the surface of the smaller structure. In order to create a good mesh between the two structures, the region which is cut out should be sufficiently large enough to create well formed elements between the two objects. The following demonstrates this process.

Return to multi-view display if you are not already in it. Now delete the elements in a region around where the access drift is going to intersect the powerhouse.

Select **Pick + element**.

```
Pick Element [*=all;b=box;c=cbox;r=ratio;ESC=done]:
```

Press B to activate the box mode.

```
Place box corner #1 [N,U,E: vn=off, s=off, o=off]:
```

Enter 900 960 990 ⏎

Click on Go .

```
Place box corner #2 [N,U,E: vn=off, s=off, o=off]:
```

Enter 895 935 1015 ⏎

Click on Go .

Select **Object Tools + delete picked** to delete the selected elements.

Let's create a nodeline along the edge of the hole we just cut out. Then we can use the facing facilities to mesh between the edge of this hole and the end of the access tunnel we are connecting.

Select **Build Polyline + open edge polyline**.

```
1 Polyline Will Be Created, Continue?
```

Press ⏎

This created a polyline that follows the edge of the hole created by the deletion of the elements in the powerhouse. Now let’s convert it to a nodeline. Remember that you can only face between nodelines.

Select **Build Polyline + polyline->nodeline.**

```
Pick Polyline to Convert:
```

Pick the polyline that was just created. This is the polyline that follows the boundary of the hole created through the deletion of the elements on the powerhouse.

This creates a nodeline on the surface of the powerhouse. Now extrude the access drift.

Select **Build Object + extrude**.

```
Select Curve to Extrude:
```

In the front view, click on the access tunnel cross-section polyline at around 1003 easting, 950 up. Respond at the prompt line as follows:

```
Enter/Pick extrusion dir ([default]) [N,U,E: vn=off, s=off, o=off]:
```

Press ↵, if the [*default*] is (-1,0,0)

```
Enter length of extrusion [default]:
```

Enter 50 ↵

```
Use default discretization (y):
```

Press ↵

The default discretization is satisfactory.

Now let’s mesh the hole between the access tunnel and the powerhouse.

Select **Build Object + face**→

```
Pick a CLOSED Nodeline
```

Click on the nodeline that was just created on the surface of the powerhouse.

```
Pick interior CLOSED Nodeline [Go = done]:
```

Click on the nodeline on the north end of the access tunnel; then

Click on Go to obtain the FACES screen.

Select **Automatic Mesh**.

When meshing is done, examine the mesh to see if the elements are of the same general size, more or less.

Select **Return**; then Select **Yes**.

You will now notice that the powerhouse and the access tunnel are joined together. All that remains is the closing off of the south end of the access tunnel.

Select **Build Object + face**→

```
Pick a CLOSED Nodeline
```

Click on the nodeline at the south end of the access tunnel.

```
Pick interior CLOSED nodeline [Go = done]:
```

This face is defined by one nodeline. Therefore,

Click on [Go] to obtain the FACES screen.

Select **Automatic Mesh**.

When meshing is done, examine the mesh to see if the elements are of the same general size, more or less.

Select **Return**; then Select **Yes**.

The discretization of the main powerhouse excavation and the access tunnel is complete and they are attached to each other. It is useful to shade the structure, to obtain a clearer display of the geometry.

Select **Shade + quickshade**. Wait.

After viewing the shaded three-dimensional picture of the opening, press [ESC] .

The **object check** function in the **Toolbox** menu should now be used to ensure that the mesh satisfies some basic rules. Proceed as follows:

Select **Toolbox + object check**. Then continue according to the following dialogue:

```
Check object, element, and node numbering (y):
```

Press ↵

```
Check for zero area elements (y):
```

Press ↵

```
Check triangular element base/height ratios (y):
```

Press ↵

```
Maximum ratio = [rmax]; element number [Nel] -- Press Enter to Continue
```

Press ↵

```
Check for invalid overlapping elements (y):
```

Press ↵

```
Check for invalid intersecting elements (y):
```

Press ↵

```
Check for leaky objects (y):
```

Press ↵

```
Geometry is NOT LEAKY
```

Splendid!

The geometry is valid for the analysis. This completes the tutorial, you may now save the file if you wish. Then exit the program by pressing **Ctrl-X** or returning to the main menu and choosing the **Exit** button.

Index

A

B

C

D

E

F

G

H

I

J

L

M

N

O

P

Q

R

S

T

U

V

W

Z