A User's Guide to PEGS

Matt Bishop

September, 1986

Research Institute for Advanced Computer Science
NASA Ames Research Center

RIACS TR 86.18

Research Institute for Advanced Computer Science

A User's Guide to PEGS

Matt Bishop

Research Institute for Advanced Computer Science
NASA Ames Research Center
Moffett Field, CA 94035

ABSTRACT

PEGS is a prototype 2 dimensional editor for grid selection. This version deals with the positioning of boundary and mesh points. The user can:

- mark or erase mesh points
- distribute points over a connected set of lines
- slide points along a line, or across many lines
- expand a portion of the figure for more visual precision
- save a set of mesh points
- load a set of mesh points
- erase the current figure and draw a new one without changing the points
- overlay a new figure without changing the points
- move or spin the figure

Currently, one cannot generate grids on the IRIS, but must send the set of points to another computer. Once the grid has been obtained, the resulting image (and points) can be loaded into PEGS and the points edited.

1. Introduction

A large part of computational fluid dynamics, or CFD, deals with the computation of grids around bodies. These grids are used to study fluid flow on, and close to, the objects. In order to generate these grids, boundary points must be placed on the surface of the body. The current version of PEGS addresses this problem.

CFD problem solving usually proceeds in five steps:

Work reported herein was supported by Cooperative Agreement NCC 2-409 from the National Aeronautics and Space Administration (NASA) to the Universities Space Research Association (USRA).

1. Define a surface. Surfaces usually are provided as sets of lines, which are obtained by examining the real surface. (For example, a digitizer might be used in the case of a 2 dimensional object.)
2. Pick grid points on the surface. These will be used as boundary points for the partial differential equations used to solve the flow about the surface.
3. Generate a computational grid. Then, the grid is examined and if need be adjusted; the computational fluid dynamicist may decide to change, add, or eliminate some boundary points, and go back to step 2.
4. Solve the equations. Once the scientist is satisfied with the computational grid, he will then use it to solve the equations of fluid flow by putting it into a flow solver.
5. Plot the solution. This translates the mathematics of the solution into a form which the sacientist can easily visualize. At this point, the computational fluid dynamicist may decide to narrow his focus to a part of the surface, and go back to step 2.

Eventually, the second and third steps should be done within an editor for grid selection, and in three dimensions. PEGS is a prototype for such an editor. Its main purpose is to provide a useable interface for CFD computations in those steps. Currently, PEGS works only in two dimensions and deals only with selecting boundary points on the surface of the body, although once a set of mesh points has been generated it can be used to move these points around, add new ones, and delete other points.

This manual is a user's guide to PEGS. It begins by describing how to start PEGS, and then provides a quick tutorial using a triangle as the curve about which points are to be distributed. After this, a more formal description of the PEGS environment and commands is given. The next section describes the internals of PEGS, and the final section discusses what more is needed.

2. Starting PEGS

To run PEGS, the IRIS must have a version of the graphics library that is later than 2.3. In other words, PEGS runs on any IRIS 3000 series machine, and on any IRIS 2000 series machine running GL2.4 or later. It will not run on an IRIS 1000 series machine.

The window manager *mex*(1G)† must also be running, since PEGS uses the window managing tools to keep track of various things. In particular, mode and flag indicators and messages to the user appear in the title of the window. PEGS will cooperate with other processes being run in other windows by preserving the color map and writemask that it began with. Whenever the input focus moves out of PEGS, these will be restored; this way, the other windows will be unchanged. Note that when the input focus is in PEGS, however, these other windows will use PEGS' color map and writemask, so they may look very strange!

To start PEGS, create a file containing the description of a figure. Each line of the file contains four floating point numbers; the first two are the x and y co-ordinates of the starting point of a line, and the last two are the x and y co-ordinates of the ending point of that line. The figure should be (roughly) centered at (0, 0). For example, the three-line file in Figure 1 describes a triangle:

```
# draw a triangle
-100      0      100      100
 100    100      100     -100
 100   -100     -100        0
```

Figure 1. Definition of a Triangle.

(Lines beginning with "#" are comments and thus ignored.) Lines need not be in any particular order, but the algorithms used generally work faster if connected lines are

† See [1] and [2].

given in sequence (as with the triangle.)

Now type

pegs *filename*

where *filename* is the name of the file containing the figure. In the case of the triangle, start PEGS by typing

pegs SHAPES/triangle

The cursor will have a new shape resembling an upper left corner; position it where you want the corner of the PEGS window to be, press the leftmost mouse button, and move the cursor to where you want the lower right corner to be. (Don't put the corner in at the top of the monitor; leave some space for the window's title.) PEGS will immediately begin drawing something in the window that you just created. Move your cursor over inside the PEGS window and click the right mouse button down and up. This attaches the keyboard and the mouse to that window.

Look at the top border of the window. It has a line with "PEGS" followed by some space and then an "X". The first part, "PEGS", identifies the program running in this window. Following that is a space for some mode indicators and flags which will be described in a moment; the "X" simply means that co-ordinate axes are currently displayed. To the right of that, where the stripes begin, is a region reserved for prompts and messages. When asked to type something (such as the number of points to distribute, or for a file name) the request is printed here, and what you type will be displayed as you type it.

At this point, you are ready to begin work.

3. A Tutorial

This section is a tutorial for the new user of PEGS. It walks the reader through

several commands and very informally explains what is going on. Those who are interested in a more precise description of the commands should skip to the next section, which is a reference guide containing descriptions of all the modes, flags, and commands of PEGS.

Throughout this section, we will be dealing with two figures: a triangle and a square. The definition of the triangle was given previously, and is contained in the file SHAPES/triangle; the square is defined in figure 2, and is contained in the file SHAPES/square.

```
# draw a square
-100    -100    -100     100
-100     100     100     100
 100     100     100    -100
 100    -100    -100    -100
```

Figure 2. Definition of a Square.

3.1. Creating and Erasing Nodes

Press the middle mouse button and release it. Notice that there is a small red dot (about 2 pixels in diameter) where the cursor pointed to. This is a *node*† or point to be fed to the grid generator. Create some more such points, by moving the cursor to various different places and pressing and releasing the middle mouse button.

Now erase those points. To do this, press the right mouse button, position the cursor over the menu entry *cancel*, and release the button. This sets the *cancel* flag, and now when you move the cursor over a node and press and release the middle mouse button, the node will be erased. The cursor must point *exactly* at the node – if you're only off by one pixel, PEGS will beep and not erase anything. If you try to erase a node and there is no node there, the IRIS will beep.

---

† In this document, a *node* and a *point* are the same thing unless otherwise stated.

Now look up at the modes and flags section of the window title. Note that there is a **C** between the two sets of asterisks. This means the cancel flag is set. To turn it off, press the right mouse button, position the cursor over the menu entry *cancel*, and release the button. This entry just toggles the setting of the cancel flag.

3.2. Attach Mode

When you are in this mode, you can distribute points throughout the PEGS window. Unfortunately, should you want to put a point on the figure, you might not be able to unless you are in attach mode. The cursor provides co-ordinates in screen space, which are then converted to world space, and the point is created at the world space co-ordinates. But because the correspondence is not exact (two points in world space may occupy the same pixel in screen space) a point in the same pixel as part of the a line segment on the figure does not mean that the point lies on the figure. So PEGS must be told to change the world space co-ordinates to get the point precisely on the figure. A mode called attach mode exists to do just that. To see how it works, select the "attach" entry of the menu. You should see an **A** appear in the modes and flags section of the title. Try to put a point anywhere except on the figure. The beep informs you PEGS will not allow this. Now move the cursor to lie directly on the figure, and again press the middle mouse button. A point will appear on the figure. (You may have to try this a couple of times before you have the cursor positioned just right.) The moral of this is, if you want to deal with points on the figure, you must be in attach mode.

Notice that if you set the cancel flag while in attach mode, you can only erase points on the figure. If you want to erase other points, you must leave attach mode; do so by selecting the "attach" entry in the menu. Experiment creating and erasing some points both in and not in attach mode.

3.3. Moving the Figure

Now we are going to see how to move the figure about. Select the "rotate" entry in the menu. You will see a black copy of the figure, the co-ordinate axes, and the points appear directly over the figure. As you move the cursor, this black image will rotate. (Move the cursor carefully; it spins very fast if you move too fast.) When you have rotated it to a new position you like (for this example, spin it 90°), press the left mouse button. A new menu will appear. If you want to redraw the figure in the position of the black "shadow", select the menu entry that says "select this position"; if you decide to leave the figure as it originally was, select "cancel the request"; and if you want to keep rotating, move the cursor to the menu title or off the menu entirely and release the mouse button.

You can also move the location of the figure by selecting the "translate" entry in the main menu. It works just like "rotate" does; you get the same shadow figure, and moving the cursor moves this shadow around. But be careful to keep the mouse inside the PEGS window, or else when you press the right mouse button, you'll get the window manager's menu!

Incidentally, if the presence of co-ordinate axes annoys you, select the "axes" entry in the main menu to erase them. (Should you do so, the **X** in the modes and flags line will disappear.) To redraw the axes, just select "axes" again.

3.4. Where The Points Are

Now that we can create and delete points and move the figure about, it would be nice to see where a point is located. To do this, you must enter display mode. Select the "display" entry in the main menu. (A **D** will appear in the modes and flags part of the title window.) Now move the cursor to some point on the figure and press and

release the middle mouse button. In the title, the world space co-ordinates of that point will appear.

Suppose we wish to move that point. We could erase it, move to the new location, and create a new node. But display mode gives us the ability to slide the point. To do this, put the mouse at that point. Make sure that both display and attach modes are be on. Press the leftmost mouse button. The co-ordinates of the point will appear in the title, and the node will turn black (the shadow color.) Now move the cursor to where you want the point to be. The node will move to the nearest part of the figure. If this is not right, slowly move the cursor some more; the node will follow, remaining on the figure. Notice that the cursor need *not* be on the figure while sliding; it must touch the point to be slid when the left mouse button is pressed, but after that, the cursor can be moved about the window and the point will move to the position on the figure nearest the cursor. When the point is at the desired position, release the left mouse button. A menu requesting confirmation will immediately appear; if you select the "yes, I am; do it" entry, the node will be moved; if you select anything else, the node will return to its initial position.

Now turn off attach mode. You can slide any point around, but as noted earlier, no node will be adjusted to lie on the figure. Also notice that in display mode the cancel flag is ignored – you can't delete a node while in display mode.

3.5. Distributions of Points

For this part of the tutorial, you should be in attach mode and not in display mode. You are going to distribute some nodes onto the figure. Put two points anywhere on the figure. Position the cursor on the figure between them (it does not matter whether the cursor is between them going clockwise or going counterclockwise). Press the left mouse button and release it. A menu will appear asking what distribution of

points you want. Select "uniform". Then look in the title bar; when the prompt "How many points?" appears, enter 10 and hit the return bar. (If you make a mistake, use the backspace key to erase it.) You will see ten points appear on the figure, uniformly distributed between the two points; a menu will appear asking for confirmation. If you select "yes, I am; do it", the new points will be redrawn in red and be added to the figure. If you select anything else, the points will disappear. Notice that the distributions go across line segment endpoints; you can have one point at (-100, 0), the other at (100, -100), and put the cursor at (100, 100), and you will get 10 points uniformly distributed along the segments (-100, 0) – (100, 100) and (100, 100) – (100, -100). These points can be treated as though you had entered them manually, of course.

You can perform this function whether or not you are in attach mode; but the cursor must lie on the figure between the nodes on the path on which the points are distributed. The nodes must lie on the figure; the endpoints are found by converting the cursor position to a point on the figure, and then tracing the figure in either direction until a node is encountered. (Yes, that means you can get away with using only one point on a closed figure.)

3.6. Expanding and Shrinking the Figure

Sometimes you want to get a close up view of part of a figure. PEGS has an "expand" function that allows you to do this. Select the "expand" entry in the main menu. Notice the cursor looks like the upper corner of a rectangle, in black. To anchor the upper right corner, press the left mouse button down and hold it down; move the mouse until the black square surrounds the area you want expanded, and then release the left mouse button. The area within the rectangle will be expanded to fill the screen. Notice the number in the modes and flags part of the window title; this means you are at that level of expansion. You can perform all functions in the expansion (including

expanding it further.) To return to the previous expansion level, select the "reduce" entry in the menu.

3.7. Some Loose Ends

When you are happy with the nodes you have marked, you may save them by selecting the "save pts" entry in the menu. You will be prompted for the name of a file. If you want to load a previously-saved set of points, select the "load pts" entry in the menu. You will be prompted for the name of a file containing the points. This is quite useful when overlaying figures and sets of points.

While in attach mode, put several points on the triangle. Make sure some of these points are on the slanted lines. You are going to erase the triangle, load a square, and erase the points in the middle of the square. (This would be useful, for example, if you generated a grid about an airplane body, and then attached wings to it; you would have to erase those mesh points which fell inside the wings.) Select the "erase drawing" entry of the main menu; the triangle will disappear, but the points will remain. Then select the main menu entry "draw"; you will be prompted for a file name, so type "SHAPES/square". The square will be drawn, and some of the points will lie inside the square. Leave attach mode, set the cancel flag, and erase those points.

Currently, there is no grid generation function, but in anticipation of what will be, PEGS can save the nodes for grid generators. In order to generate a grid correctly, many grid generators require boundary points to be input in clockwise order. So, select the main menu entry "grid". The points on the figure will be saved in clockwise order, and the other points in random order, in a file named "CRAY.DAT". At this point PEGS will suggest it is not appropriate to run a grid solver on the IRIS, and wait for another command.

If the screen gets messed up, you can redraw it by selecting the main menu entry "redraw." This is useful if other graphics processes start changing the color map. Incidentally, PEGS saves the color map that is present when it begins; its color definitions leave the four lower bitplanes alone, so it will not affect other windows which use colors in that range, and when the input is shifted from the PEGS window, PEGS restores the original color map. This way, PEGS can be run with other graphics processes without creating a mess.

3.8. Entering Commands From the Keyboard

Finally, all commands in the main menu may be invoked from the keyboard. Figure 3 gives the keyboard commands that correspond to each main menu entry.

key	*menu entry*	*key*	*menu entry*	*key*	*menu entry*
CTRL-L	redraw	f	draw	s	reduce
a	attach	g	grid	t	translate
c	cancel	l	load pts	w	save pts
d	display	q	exit	x	axes
e	erase drawing	r	rotate	z	expand

Figure 3. Keyboard Commands

4. The PEGS Environment

Many of the commands used by PEGS perform different functions depending on what *mode* the editor is in and what *flags* are set. For example, *attach* mode, the leftmost mouse button is a command to distribute points between two other points. In *display* mode, the leftmost mouse button slides a point around. This section describes the modes.

4.1. Attach Mode

In this mode, when the middle mouse button (which adds or deletes points) or the left mouse button (which distributes points over the figure) is depressed, PEGS checks

that the cursor is exactly† on the figure. If it is not, the bell sounds, indicating that the cursor must be positioned on the figure. Otherwise, PEGS gets the position of the cursor and normalizes the world space co-ordinates to be the point of the figure closest to the cursor (the exact mechanism is described in section 6.4.)

A capital **A** appears in the modes and flags part of the window title when attach mode is on.

Attach mode is necessary due to the difference between world space and screen space. Simply because two objects occupy the same pixel does not mean that they have precisely the same world space co-ordinates. Without attach mode, there would be no way for the user to indicate that the position being indicated indeed has the same world space co-ordinates as the line drawn through that pixel.

4.2. Display Mode

This mode is used to determine positions of points. When the middle mouse button (which asks for the world space co-ordinates of a point) or the left mouse button (which indicates the point to be moved) is depressed, the exact position of the node will be displayed in the title section of the window just after the mode and flag section.

When display mode is on, a capital **D** appears in the modes and flags part of the window title.

This mode is used to position points very precisely. In it, a point may be moved about the PEGS window or along the figure.

---

† "Exactly" means *exactly*. If the cursor is even one pixel off, an error is indicated.

4.3. Cancel Flag

When this flag is set, points can be deleted but not added. This flag is ignored when display mode is on.

When the cancel flag is set, a capital **C** appears in the modes and flags part of the window title.

4.4. Axes Flag

When this flag is set, the co-ordinate axes are displayed. If the figure is rotated or translated, the axes move with the figure.

When the axes flag is set, a capital **X** appears in the modes and flags part of the window title.

5. PEGS Commands

Commands may be entered in two ways: either from the keyboard or from the menu. To get the menu, press the right mouse button down. A yellow menu will appear; as the cursor moves over the entries, the entry to which the cursor points will turn blue. To select a menu entry, move the cursor to the desired entry and release the button.

In addition, pressing mouse buttons and moving the mouse will also enter commands under certain circumstances. This section describes each of the commands, and how to enter them. First, the functions of the mouse buttons are explained, then the commands that can be entered from the menu and the keyboard.

5.1. Confirmation

Many commands require you to confirm them before they are made permanent. When this happens, the menu in figure 4 will appear.

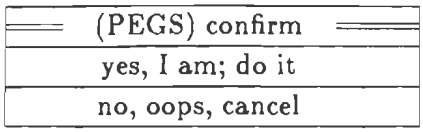

Figure 4. Confirmation Menu

To confirm, move the cursor over the "yes, I am; do it" entry and release (or press and release) the right mouse button. To cancel, select the other menu entry. If you release the right mouse button anywhere else, it is taken to be non-confirmation and the request is cancelled.

5.2. Errors

Errors are indicated by PEGS beeping. An error message will usually appear in the window title indicating exactly what was wrong.

5.3. Right Mouse Button

In general, this button will provide a menu. Usually this menu is the main PEGS menu; sometimes, it is a confirmation menu or a menu for a specific command.

Main menu commands are described below.

5.4. Middle Mouse Button

This button acts on a point at the cursor position. In normal mode, pressing and releasing the middle mouse button creates a node at that position. In attach mode, this creates a point only if the cursor is precisely on the figure. (Being even one pixel off is an error.) If PEGS is in attach mode and the cursor is not on the figure, pressing the middle mouse button is an error. Releasing the button is ignored.

If display mode is on, the cursor must point to a node when the middle mouse button is pressed. In the window title, after the modes and flags section, PEGS will display the co-ordinates of the node pointed to. As before, releasing the middle mouse

button does nothing.

If display mode is off and the cancel flag is set, the cursor must point to a node. The node will be erased when the button is pressed. In attach mode the node must lie on the figure; in normal mode, the point may be anywhere.

5.5. Left Mouse Button

The left mouse button has two functions. It can be used to slide points around the figure or window, and it can be used to distribute points over the figure.

If display mode is off, the cursor must point to the figure. When the left mouse button is pressed, the user will be prompted for a distribution and its parameters. At this point, a menu will appear and the left mouse button may be released. The right mouse button should be pressed and released when the appropriate menu entry has been selected. A distribution of nodes meeting the parameters will be generated between the two nodes closest to the cursor position (one is found by tracing the figure to the right, the other to the left.) PEGS will display the generated nodes in black, and ask the user for confirmation. If confirmed, the nodes will be added and they will change color to red. If not confirmed, the nodes will be deleted.

If display mode is on, the cursor must point to a node. The node will turn black. When the left mouse button is depressed, moving the cursor will move the node. Releasing the mouse button will position the point at the current location. If in attach mode, the node will move along the figure to the point on the figure nearest the cursor position; the cursor need *not* move along the figure. If not in attach mode, the node will follow the cursor position as it moves.

As usual, if in attach mode, the node being moved must lie on the figure; otherwise it can be anywhere in the window.

5.6. Main Menu

The main menu is given in figure 5.

PEGS
rotate
translate
cancel
attach
display
axes
grids
redraw
load pts
save pts
draw
erase drawing
expand
reduce
exit

Figure 5. Main Menu

To get the menu, press the right mouse button. To select a menu entry, move the cursor position to the desired entry and release the right mouse button. To erase the menu, move the cursor position to the title entry (or off the menu entirely) and release the button.

The following sections describe each command.

5.6.1. Rotate

Selecting this entry rotates the figure about the origin. The figure, nodes, and coordinate axes (if present) are superimposed on the window in black, and as the mouse moves, the black image rotates. (The user should move the mouse slowly, because the amount of rotation is a function of the change in position of the mouse; see section 6.6.) To select a rotation, move the black figure into the right position, and depress the left mouse button. The menu in figure 6 will appear.

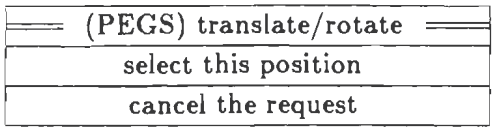

Figure 6. Motion Selection Menu

To select the current position, release the left mouse button at once. (This selects the "select this position" entry.) To erase the menu and continue the rotation, move the cursor position to the title or off the menu and release the mouse button. To cancel the requested rotation, select the "cancel this request" menu entry.

When a rotation is selected, the black figure will assume the colors of the figure and the nodes (and the co-ordinate axes if present), and the figure at the old position will vanish.

5.6.2. Translate

Selecting this entry moves the origin of the figure. The figure, nodes, and co-ordinate axes (if present) are superimposed on the window in black, and as the mouse moves, the black image moves. (The user should move the mouse slowly, because the amount of motion is a function of the change in position of the mouse; see section 6.7.) To select a translation, move the black figure into the right position, and press the left mouse button. The menu in figure 6 will appear.

To select the current position, release the left mouse button at once. (This selects the "select this position" entry.) To erase the menu and continue the translation, move the cursor position to the title or off the menu and release the mouse button. To cancel the requested rotation, select the "cancel this request" menu entry.

When a translation is selected, the black figure will assume the colors of the figure and the nodes (and the co-ordinate axes if present), and the figure at the old position will vanish.

5.6.3. Cancel

Selecting this menu entry toggles the cancel flag. When the cancel flag is on, a **C** will be displayed in the mode and flags section of the window title.

5.6.4. Attach

Selecting this menu entry toggles attach mode. When attach mode is on, an **A** will be displayed in the mode and flags section of the window title.

5.6.5. Display

Selecting this menu entry toggles display mode. When display mode is on, a **D** will be displayed in the mode and flags section of the window title.

5.6.6. Axes

Selecting this menu entry toggles the axes flag. When the axes flag is on, an **X** will be displayed in the mode and flags section of the window title.

5.6.7. Grids

This command saves all nodes in the file "CRAY.DAT" in the current directory. Nodes on the figure are saved in a counterclockwise order; the remaining nodes are appended to this list. In the future, the nodes will be sent (with appropriate job control) to another computer so a grid can be computed and sent back to PEGS for display and manipulation. However, for now, PEGS simply complains that it is not running on a Cray.

5.6.8. Redraw

This command erases the contents of the PEGS window and redraws it. It is quite useful when the IRIS screen is messed up.

5.6.9. Load Pts

This command loads a set of nodes and displays them. PEGS prompts the user for a file containing the node co-ordinates. The format of the file is described in section 6.9.

5.6.10. Save Pts

This command saves the co-ordinates of the nodes currently defined in the window. PEGS prompts the user for a file to write these in.

5.6.11. Draw

This command draws a figure in the window. If a figure is there already, the new figure is appended to the old. PEGS prompts the user for a file containing the figure; the format is given in section 6.10 (also, see figures 1 and 2 for examples.)

5.7. Erase Drawing

This command erases the drawing in the window. The nodes are unchanged.

5.7.1. Expand

When this menu entry is selected, the cursor changes to an upper left corner of a square. Position the cursor by pressing the left mouse button down. Holding that button down, move the cursor until the region to expand is surrounded by the black square. Then release the left mouse button. The area within the black square will expand to fill the window.

PEGS can currently expand the figure up to three levels; the current expansion level will appear as a digit in the modes and flags section of the window's title.

All commands work when expanded.

5.7.2. Reduce

When this menu entry is selected, the last expansion is undone. The original figure cannot be reduced.

5.7.3. Exit

To leave PEGS, select this entry. Confirmation is required.

6. The Internal Mechanisms

This section gives a brief description of how some of the commands were implemented. It explains what choices were made and why.

6.1. Write Masks and Color Maps

Since PEGS is intended to run under the IRIS window system, it must attempt to co-operate with other processes running concurrently under the window manager. Specifically, since the color map and writemask are common to all graphics programs, PEGS cannot simply change these. It must take into account a user's moving from the PEGS window to another window and dealing with a process there.

In order to keep as many windows as possible unchanged, PEGS uses the fifth through ninth bitplanes for its color maps. This way, any other process that writes into the lowest four bitplanes will appear in its normal colors. (In particular, the clock process still shows a clock.) Further, before PEGS sets up its color map, it saves the color map and attributes of the IRIS screen; then it changes its color map and writemask. Whenever input shifts out of the window (that is, the token *INPUTCHANGE* is received in the device queue), the original color map and attributes are restored; when input shifts back into the PEGS window, PEGS' colormap and attributes are restored. When PEGS exits, it again restores the original color map and writemask.

This is why other windows can look very strange when PEGS is running, but return to normal when PEGS exits or when the cursor is attached to another window.

6.2. Menu and Keyboard

Commands can be entered either from the menu or from the keyboard. This is done for two reasons.

The first is that using a menu can be quite annoying to an experienced user who wants to do things quickly; the time involved in drawing the menu and positioning the mouse can be quite aggravating. Such a user can simply use the keyboard.

The second reason is that some commands, such as entering modes, require the cursor to move; and there are times when this interrupts a previous command. For example, a very common error is to forget you have the cancel flag set and position the cursor on the figure to create a node. If you had to use the menu to turn off the cancel flag, you would have to move the mouse. Turning off the cancel flag from the keyboard is very useful in this case.

6.3. Overlaying Black Figures to Indicate Intermediate Results

Clearly the user needs an indication of what is going on with certain commands; the command to distribute points, to rotate or translate the figure, and to move a node around are obvious examples. Given this, there are two choices; either the original colors, or a new color indicating these results are intermediary, can be used.

The latter is less confusing. There is never any question which figure or node is the original and which is the intermediate. With the first choice, it may is difficult to distinguish between the initial and intermediate positions. Other methods, such as dotting lines or putting a box around the changing objects could be used; but these add extraneous information that can confuse an unwary user.

6.4. Adjusting Points to Lie on the Figure

In attach mode the world space coordinates of points are adjusted to lie on the figure. This is done as follows.

First, the world space co-ordinates of the point are computed, and the line that of the figure nearest to it is found by use of the picking mechanism of the IRIS. The perpendicular to this line passing through the point is then computed, and the point of intersection of the line and that perpendicular is found. This is the point used.

Let P be the point and let it have co-ordinates (x_P , y_P). Let the line be L and let its endpoints be (x_0, y_0) and (x_1, y_1). The equation of L in point-slope form is

$$y = m_L x + b_L \tag{1}$$

where

$$m_L = \frac{y_1 - y_0}{x_1 - x_0}$$

and

$$b_L = y_L - m_L x_L$$

The equation of the perpendicular through P is

$$y = -\frac{1}{m_L} x + b_P \tag{2}$$

where

$$b_P = y_P + \frac{x_P}{m_L}$$

At the point of intersection (x_I , y_I), by (1) and (2),

$$m_L x_I + b_L = -\frac{1}{m_L} x_I + b_P \tag{3}$$

Simplifying (3),

$$x_I = \frac{b_P - b_L}{m_L + \frac{1}{m_L}}$$

and

$$y_I = m_L x_I + b_L$$

6.5. Precision of Cursor Pointing

Currently, when the cursor must point at a node or the figure, the cursor position must occupy the same screen pixel that the node or figure does. This is done for consistency. There are times when the IRIS pick mechanism is used (as in putting points on figures or erasing points) but there are times when the IRIS pick mechanism would provide ambiguous results (if, for example, two lines of the figure lay in the same pick window.) While restricting the pick window to one pixel does not eliminate ambiguity, it does reduce the ambiguity of many situations.

6.6. Rotation

The angle of rotation is computed in radians by the formula

$$\tan^{-1}\frac{\Delta y}{\Delta x}$$

where Δy and Δx are the changes in the mouse's y and x positions respectively.

6.7. Translation

If (x_O, y_O) are the current co-ordinates of the origin, the new co-ordinates (x_O', y_O') of the origin are computed by the formulae

$$x_O' = x_O + \Delta x$$
$$y_O' = y_O + \Delta y$$

6.8. Scaling

Once the little black square in the expansion has been fixed, the square is expanded to fill the window as follows. First, the lower left corner of the square is shifted to the origin. Then one side of the square is expanded to fill the window. The window is then shifted to display the region requested. These commands are appended to the object definition.

Reduction is performed by deleting the scaling and translation commands from the object. Each set of scaling commands is surrounded by tags, so this is quite easy.

6.9. Format of the Point Files

The format of the file in which points are saved with the "Save Pts" command is a list of lines with two numbers per line, the first being the x co-ordinate and the second being the y co-ordinate. When the "Load Pts" command reads a file, it expects data to have the same format. Figure 7 contains an example file.

```
10.345278 8.234172
-100.000000 99.234109
76.28492 -83.299299
```

Figure 7. Sample Data Point File

6.10. Format of the Figure Definition File

Each line of this file must contain four floating point numbers; each of these describe a line segment that is part of the figure. The first two numbers are the x and y co-ordinates of the starting point of the line segment and the last two numbers are the x and y co-ordinates of the ending point of the line segment. Any line beginning with a "#" is a comment line and is ignored. Figures 1 and 2 contain sample figure definition files.

7. Future Directions

PEGS is a simple prototype intended to be used as a base for future research. It has several problems, the major ones being it does not yet work on three dimensional figures and it can only manipulate points rather than grids. The latter is a particularly thorny issue, since it is desirable for a scientist to be able to shift a grid line with the mouse and then have the flow recomputed. But whether or not this can be done locally, as opposed to recompute over the entire grid, is another question. Theoretically, the answer is no, since any change changes the solution. But in practise, there may very well be an area in which the changes affect the solution noticeably, and beyond which the change has no appreciable effect.

The next step is to build a version of PEGS that will handle three dimensional figures. Many algorithms will have to be changed; although the ideas are the same, the projection of the third dimension onto the screen requires special handling. To take the most obvious example, it is very easy to go from screen space to a point in world space when the world space has two dimensions. But when the world space has three dimensions, the same transformation produces a line, not a point. It should also be able to move grid elements, and communicate with faster computers and established CFD programs to perform grid generation and distribution of points. (It should not use its own code for this; since the CFD codes have existed for a long time and are trusted, users would believe their results; but new codes would be suspect since they have not been tested as thoroughly as the existing ones.)

The goal of this research is to provide an interface with which computational fluid dynamicists are comfortable, and powerful enough to allow them to work visually with problems which have in the past been dealt with computationally. PEGS is a first step towards this goal.

Acknowledgements: I would like to thank Tim Barth, Scott Eberhart, Dave Kerlick, Tom Lasinski, and Joe Steger for discussing the way in which computational fluid dynamicists approach problems, and for making invaluable suggestions and comments on this program, and Jim Houston for his valuable comments on this manual.

References

[1] –, *UNIX Programmer's Manual, Volume IA: Commands*, Version 2.0, Silicon Graphics, Inc., 2011 Stierlin Road, Mountain View, CA 94043 (1986). Document Number 007-0101-020.

[2] –, *IRIS User's Guide*, Version 2.4, Silicon Graphics, Inc., 2011 Stierlin Road, Mountain View, CA 94043 (1986). Document Number 007-1101-024.

Mail Stop 230-5
NASA Ames Research Center
Moffett Field, CA 94035
(415) 694-6363

The Research Institute for Advanced Computer Science
is operated by
Universities Space Research Association
The American City Building
Suite 311
Columbia, MD 21044
(301) 730-2656