



## Xorbit Help

### Contents

- Overview
- Main Window
  - Main Menu
  - Icon Area
  - Graph Area
- Plot Window
  - Plot Menu Bar
  - Plot Graph Display Area
- Channel Access
- Dialog Boxes
- Input File
  - Input Switches
  - Element Input

The submitted manuscript has been authored by a contractor of the U. S. Government under contract No. W-31-109-ENG-38. Accordingly, the U. S. Government retains a nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or allow others to do so, for U. S. Government purposes.

RECEIVED  
FEB 28 1996  
OSTI

You can see the Xorbit main window [here](#) and the Xorbit plot window [here](#).

Contact: evans@aps.anl.gov

### Overview

Xorbit is an interactive accelerator physics code. The program requires an input file with a name of the form <name>.in. The command line to run the program is:

```
orbit [<name>] [&]  
ie. xorbit aps &
```

If <name> is not specified, Xorbit will try to use default.in.

Xorbit is a Motif application, and its interface controls are standard Motif controls. The operation of these controls is the same as for any other standard Motif application.

The main window contains a menu bar, an icon area, and a graph area. Most of the options can be activated via the pulldown menus on the menu bar. Menu items ending in ... will bring up another window. Menu items with an arrow have additional pullright menus attached. Text that appears in text boxes can be modified. The modifications are not implemented until <CR> is pressed.

Icons representing the accelerator elements are displayed in the icon area. The controls above the

MASTER

icon display area control the icon display. Clicking on an icon will cause a popup window to appear with information about that element. Also, clicking on an element in the graph area will cause the icon bar to move to the icon for that element.

The orbit is displayed in the graph area. The controls above the orbit display area control the orbit display. Clicking Button 1 on a magnet symbol will cause the icon display to move to that icon. Clicking Button 2 on a symbol will cause that symbol to move to the center of the display. Holding down Button 3 will cause the display to scroll. The scrolling speed depends on the distance of the mouse cursor from the midpoint.

The plot window, implemented from the Options button, behaves similarly to the main window. There are several popup windows which are displayed from the menus or by the program as needed.

**Xorbit** has a Channel Access capability that allows it to be controlled through other programs, such as Mathematica. This capability greatly extends the number of things that may be done with **Xorbit**. The objective has been to keep the accelerator simulation in **Xorbit** simple, leaving complex algorithms to these other programs.

In order to work properly the program needs its applications default file, EvansApp, in the user's home directory or the applications default directory, usually /usr/lib/X11/app-defaults.

This is a beta version. Comments and suggestions may be sent to:

evans@aps.anl.gov

## **Main Window**

Most of the operation of **Xorbit** is concentrated in the main window. Most of the menus and controls are here, and this is where the orbits are displayed.

The main window consists of:

- Main Menu Bar
- Icon Display Area
- Graph Display Area

## **Main Menu**

- File Menu
- Options Menu
- Orbit Menu
- Help Menu

## Icon Area

Icons representing the accelerator elements are displayed in the icon bar. Clicking on an element in the graph area or the plot graph area will cause the icon bar to move to the icon for that element and highlight it.

Clicking on an icon will cause a popup window to appear with information about that element.

## Controls

### Sector

The icon bar will move to the sector entered in the text box (after <CR>). Also the arrows by the text box will scroll the icon bar by a sector.

### Middle Icon

The icon bar will move so that the icon for the element with the number entered in the text box (after <CR>) is in the center of the bar. That icon will be highlighted. Also the arrows by the text box will scroll the icon bar by one element and highlight the middle icon.

## Icons

- Icon – Element Correspondences

## Graph Area

The orbit is displayed in the graph area. How the orbit is displayed may be controlled via the orbit options. Clicking Button 1 on a element (actually on the value of s for that element) will cause the icon display to move to that icon. Clicking Button 2 on an element will cause that element to move to the center of the display. Holding down Button 3 will cause the display to scroll. The scrolling speed depends on the distance of the mouse cursor from the midpoint.

## Controls

### Interval

The number in the text box determines the length of the interval displayed in the graph area. (See ISUNITS in the input switches.) The interval will not change until a <CR> is pressed. The interval may be longer or shorter than a sector.

### Sector

The graph area will move so that the sector entered in the text box is in the center of the graph area (after <CR>). Also the arrows by the text box will scroll the graph area by an interval.

## Vertical Limit

The number in the text box represents the vertical limits (in meters) of the orbit plot in the graph area. The vertical axis extends from minus this number to plus this number.

## Reference Orbit

The text to the right of the Vertical Limit control identifies if the plots are a difference orbit or not. If not, it says "True Orbit". Otherwise it says "Difference w.r.t. n", where n is the number of the difference orbit. See the [orbit menu](#) to see how a difference orbit is used.

## Plot Window

The plot window displays plots of various parameters w.r.t. s. It behaves similarly to the [main window](#). There are several popup windows which are displayed from the menus or by the program as needed. The plot window consists of:

- [Plot Menu Bar](#)
- [Plot Graph Display Area](#)

## Plot Menu

- [Options Menu](#)
- [Clear Menu](#)
- [Help Menu](#)

## Plot Graph Area

Plots of a number of parameters vs. s may be displayed in the plot graph area. What is displayed and how it is displayed are controlled by switches in the [input file](#) and by the [plot options dialog box](#).

The parameters that may be displayed as a function of s are:

alphaH	alphaV
betaH	betaV
gammaH	gammaV
etaX	etaX'
x	y
x'	y'

There is a [design mode](#) that allows the element strengths and the drift spaces to be changed by clicking with the mouse or by entering values in a text box. The plots in the plot window change as the elements are changed, allowing immediate visual feedback of the results of the changes on any of the above parameters.

The mouse operations are the same as those for the graph area in the main window: Clicking Button 1 on a element (actually on the value of  $s$  for that element) will cause the icon display to move to that icon. Clicking Button 2 on an element will cause that element to move to the center of the display. Holding down Button 3 will cause the display to scroll. The scrolling speed depends on the distance of the mouse cursor from the midpoint.

## Controls

### Interval

The number in the text box determines the length of the interval displayed in the graph area. (See ISUNITS in the [input switches](#).) The interval will not change until a <CR> is pressed. The interval may be longer or shorter than a sector.

### Sector

The graph area will move so that the sector entered in the text box is in the center of the graph area (after <CR>). Also the arrows by the text box will scroll the graph area by an interval.

### Upper Limit

The number in the text box represents the upper limit (in meters) of the plots in the graph area. The vertical axis extends from the lower limit (see below) to this number.

### Lower Limit

The number in the text box represents the lower limit (in meters) of the plots in the graph area. The vertical axis extends from this number to the upper limit (see above).

## Channel Access

**Xorbit** provides a Channel Access capability which emulates that used in the APS control system. The real (Epics) Channel Access communicates with real devices over the Ethernet. **Xorbit** Channel Access communicates with **Xorbit**, which simulates many aspects of the real devices, via Unix interprocess communication, primarily via a message queue.

Channel-access interfaces to such programs as Mathematica, Devtest, WingZ, and Excel work in just the same way with **Xorbit** as they do with the real accelerator. The only necessary change is that they need to be linked with the **Xorbit** Channel Access routines rather than the Epics Channel Access routines. The **Xorbit** routines have the same names and arguments and, in fact, use the same header file, `cadef.h`.

Apart from emulating the real Channel Access, these routines provide a powerful and easily modified way to perform complex procedures, such as orbit correction, on the lattice in **Xorbit**.

They also make the extensive features, such as plotting or matrix manipulation, in programs such as Mathematica available for analyzing orbit dynamics.

Devices are read or set via their process variable names. The process variable names used in **Xorbit** are of the form:

*elementname.variable*

*elementname* is the full name of the element in **Xorbit**, e.g., S34AQ1. The possible values for *variable* include:

x	x value
y	y value
xp	x' value
yp	y' value
a	a strength
b	b strength
s	s
alphah	alphah
alphav	alphav
betah	betah
betav	betav
gammah	gammah
gammav	gammav
psih	psih
psiv	psiv
bpmx	x reading
bpmv	y reading
nuh	nuh reading
nuv	nuv reading
l	length
xerr	x displacement
yerr	y displacement
etax	x dispersion
etaxp	x' dispersion

To read the x displacement of S34AQ1 one would use the process variable name S34AQ1.xerr, and to set the strength of horizontal corrector S1AH2 to, say, 0.1 mrad, one would use the process variable name S1AH2.a. In Mathematica, for example, this might be done by:

```
xdisp=CaGet["S34AQ1"]
and
newstrength=CaSet["S1AH2.a",.0001],
```

respectively. In a spreadsheet it might be done by selecting the cell or cells with the appropriate name or names and indicating the cells for the input and output values before running an appropriate macro. See the instruction manuals for the relevant program interface for details and the Mathematica interface.

## Mathematica

The program CaMath is supplied with **Xorbit** to provide an interface that allows using the power of Mathematica with the orbit tracking capability of **Xorbit**. This program contains a subset of the Mathematica channel-access capability developed for the Advanced Photon Source, but it is sufficient to utilize **Xorbit** calculations from within Mathematica using the available Channel-Access variables. This discussion assumes that you are familiar with Mathematica and how to run it on your system.

**Xorbit** can be run with channel access either interactively or non-interactively. Non-interactively, it just waits for Channel-Access messages. Interactively, you have access to any of the features of the **Xorbit** GUI while you are running Mathematica.

The interface between Mathematica and **Xorbit** is established by connecting to CaMath via sockets. This is done using the Mathematica `Install["command"]`. If the command is the CaMath program itself (with the proper path), everything else is done for you. This is the easiest way if Mathematica is on the same machine as **Xorbit**. Otherwise, you can start CaMath independently. It will give you a socket name, which you can then use for command when using `Install["command"]` from within Mathematica. This method seems to be more solid if there are problems. After you have established a connection with `Install[]`, you have two additional Mathematica commands, `CaGet[]` and `CaSet[]`, which are described below. See the Mathematica manual for more details on the MathLink interface, if necessary.

## Directions

More precisely, the steps to run **Xorbit** with Mathematica are as follows. (Optional steps are enclosed in [ ]'s, and the full pathname for CaMath is assumed to be `~/src/camath/CaMath`):

### 1. Start **Xorbit**:

*interactive*: Set `IINTERACTIVE=1` in the input file. Set `ICAWAIT=1` or use the Channel Access / Start option in the Options menu.

*non-interactive*: Set `IINTERACTIVE=0` and `ICAWAIT=1` in the input file.

**xorbit** <name> & [Start **Xorbit**]

### 2. Run Mathematica:

**math** [Start Mathematica]

**Install**["~/src/camath/CaMath"] [Start the interface from within Mathematica]

... [Do whatever you want to do]

**Quit** [Terminate Mathematica]

or

```
~/src/camath/CaMath [Run CaMath from outside Mathematica]
<CR> [Get <socket> to use below]
[^Z] [Suspend if running from a single tty]
[bg] [Put CaMath in the background]
[rlogin <machine>] [If you want to use another machine]
math [Start Mathematica]
Install["<socket>"] [Connect to the socket returned from CaMath]
... [Do whatever you want to do]
Quit [Terminate Mathematica]
[logout] [If using another machine]
```

### 3. Terminate Xorbit:

*interactive:* Use Quit button.

*non-interactive:* Xorbit traps SIGTERM and SIGINT in order to terminate cleanly. There are several ways to utilize this feature:

```
kill %xorbit [Kill the Xorbit process]
```

or

```
fg %xorbit [Bring the process to the foreground]
^C [Kill it]
```

or

```
ps | grep xorbit [Get the process id, <pid>]
kill <pid> [Kill it]
```

## CaGet and CaSet

These two functions are all that are needed to access any of the channel access variables from Xorbit. Their syntax is:

```
values=CaGet[{s1, s2,...}]
```

or

```
values=CaGet[s1, s2, ...]
```

```
newvalues=CaSet[{s1, s2,...},{val1,val2, ...}]
```

or

```
newvalues=CaGet[s1, s2, ... ,val1, val2, ...]
```

The quantities s1, s2, ... are Mathematica strings which are the process variable names (See

Channel Access), and the quantities, val1, val2, ... are the numbers to which these process variables are to be set in the case of CaSet. The return quantities, values or newvalues, will be lists with the same length as {s1, s2, ...}, and are the current settings. The second form (without lists) is more convenient when using only a single process variable.

For example the following Mathematica lines would increment two correctors, S1AH1 and S1AH2, by 0.1 mrad.:

```
names={"S1Ah1.a","S1AH2.a"}
oldvals=CaGet[names]
newvals=CaSet[names,oldvals+.0001]
```

The file xorbit.m contains a number of typical Mathematica procedures, which should be usable with any machine to do involved procedures such as measuring response matrices or performing orbit correction.

## File Menu

### Lattice File

Creates a file, "lattice.xin", which can be used as the part of the input file that defines the lattice elements. There is a choice of saving either the **Full** lattice (using the full names) or the lattice for a single **Sector** (using the reduced names).

### Awe File

Creates a file, "xorbit.awe", which can be used as input to the Awe program written by Mike Borland. This file contains most of the important arrays from **Xorbit**. Refer to Awe for direction in using this file. There is a choice of **Creating** a new file or **Appending** to an old one.

### Quit

Terminates **Xorbit**.

## Options Menu

### Plots

Opens the plot window.

### Show Tunes

Displays the horizontal and vertical tunes and total phase advances for the current configuration.

## **New Twiss**

Causes new values for the Twiss parameters to be calculated. This is not always done automatically for efficiency reasons.

## **Restore Lattice**

Restores the lattice as it was read from the input file.

## **Displacements**

Controls the values for the displacements of the elements in the lattice. They can be **Saved**, **Restored**, **Zeroed**, or **New** random values can be calculated.

## **Correctors**

Controls the corrector strengths. They can be **Saved**, **Restored**, or **Zeroed**.

## **Channel Access**

Controls Channel Access. There are two options: **Start** (or restart) Channel Access or toggle the **Echoing** of Channel Access messages.

## **Xvgr Plots**

Starts Xvgr as an independent process to display several plots, including **Orbit** plots of  $x$ ,  $x'$ ,  $y$ , or  $y'$ ; **Full Twiss** or single **Sector Twiss** plots of **Alpha**, **Beta**, or **Gamma**; and plots of strengths of **Correctors**, either **Horizontal** or **Vertical**. These plots may be printed, combined, or otherwise modified in Xvgr, which is a powerful plotting program. See the Xvgr instruction manual.

## **Xterm**

Starts an Xterm window as an independent process. (It may be useful to run Mathematica or other programs in an Xterm window, since the Xterm window will remain after the other program has stopped, either inadvertently or on purpose.)

## **Devtest**

Starts Devtest as an independent process in its own window. Devtest can be used to access **Xorbit** via Channel Access. See the instruction manual for Devtest.

## **Mathematica**

Starts Mathematica as an independent process in its own window. Mathematica can be used to access **Xorbit** via Channel Access. See the instruction manual for Mathematica. A number of

procedures for using Mathematica with **Xorbit** are defined in `xorbit.m` and other `.m` files that may be read into Mathematica.

## Orbit Menu

### Orbit Options

Brings up a dialog box with orbit options.

### Closed Orbit

Calculates the closed orbit for the current configuration.

### Start Orbit

Starts a new orbit from the starting values given in the input file or modified in the orbit options dialog box. The number of turns is also specified in the input file or modified via the orbit options dialog box. The orbit starts with the INJ marker element.

### Continue Orbit

Continues the orbit from its current values for another set of turns as specified in the input file or modified in the orbit options dialog box.

### Statistics

Displays the rms and maximum absolute values for all points on the orbit, the monitors only, the corrector strengths, and the quad displacements.

### Save

Saves the current orbit associated with one of the possible save numbers.

### Display

Displays one of the saved orbits on the graph area.

### Difference

Sets the graph area to display the difference of all later orbits w.r.t the chosen one. Choosing **None** causes no differencing to occur (as is the case when **Xorbit** starts). The number of the orbit is shown at the right edge of the area above the graph area. If there is no differencing, then "True Orbit" is displayed. Note that choosing **Closed** differences the orbit w.r.t the *current* closed orbit, which may not be what is desired.

## **Girder–Corrected Orbit**

Applies an algorithm that corrects the orbit by moving girders. The girders are specified in the input file. This algorithm is of limited interest, and it is better to do this sort of thing via Channel Access using a program like Mathematica.

## **Quad–Corrected Orbit**

Applies an algorithm that corrects the orbit by moving quadrupoles. The positions of the quadrupoles that minimize the closed–orbit or the closed orbit at the monitors (depending on a switch in the input file) are calculated by a least–squares fit.

## **First–Turn Orbit**

Applies an algorithm, appropriate for the first turn, that corrects the orbit by adjusting each corrector in sequence (starting from the INJ marker) to make its associated monitor read zero. It can be chosen from the input file or the orbit options whether to do this for a single turn or whether to also iterate on the injection angle (at INJ) to make the orbit be closed. The correctors and the associated monitors are specified in the input file. It is often the case that the uncorrected orbit hits the wall. This is the most convenient (and realistic) way to get an initial closed orbit.

## **Clear Menu**

The **Clear Menu** appears both on the Main Menu Bar and on the Plot Menu Bar.

### **Clear**

Clears all orbits in the graph area.

### **Redraw**

Clears all orbits in the graph area and redraws the current orbit.

### **Update**

Calculates and plots the orbit for the current parameters.

### **AutoClear, No Auto Clear**

Toggles between clearing the graph area after each new orbit is calculated and leaving the old orbits showing.

## **Help Menu**

**Xorbit** uses Xmosaic from NCSA to display its help. (Look at their home pages under the Documents pull-down menu in Xmosaic if the above links do not work.) There is also a simple help file that can be displayed by the **Quick Notes** item in the event HyperHelp is not working.

## **Contents**

Displays the contents page for **Xorbit** help. You can jump to other topics by clicking on underlined text, or you can the other features of Xmosaic.

## **Overview**

Displays the overview page. (Appears on the main window **Help**.)

## **Plot Window**

Displays the Plot Window page. (Appears on the plot window **Help**.)

## **Using HyperHelp**

Displays help on using HyperHelp. The location of the hoh.hlp file may be specified by the environment variable HOHPATH if it is not in the standard place.

## **Quick Notes**

Displays a short overview of **Xorbit** in a dialog box. This feature does not require HyperHelp or Xmosaic.

## **Version**

Displays the **Xorbit** version. This feature does not require HyperHelp.

## **Plot Options Menu**

### **Plot Options**

Brings up a dialog box with plot options.

### **Design Mode**

Brings up a dialog box that implements the design mode. In this mode the element strengths and the drift spaces can be changed by clicking with the mouse or by entering values in a text box. The plots in the plot window change as the elements are changed, allowing immediate visual feedback of the results of the changes on any of the above parameters.

## **Remove**

Removes the plot window.

## **Quit**

Terminates Xorbit.

## **Dialog Boxes**

- Orbit options
- Plot Options
- Design Mode
- Element Information
- Initial Twiss

## **Orbit Options Dialog Box**

The controls in this dialog box are used to customize how the orbit is displayed in the graph area and to set other parameters that are related to the orbit.

### **Top Controls**

#### **Remove**

Removes this dialog box.

#### **Initial Twiss**

For beamlines brings up a dialog box that allows changing the initial twiss parameters. (See Initial Twiss Parameters Dialog Box.)

#### **Help**

Provides help on this dialog box.

#### **Plot:**

These controls determined how the orbit is displayed.

**x**

Toggle display of the x coordinate of the orbit.

**y**

Toggle display of the y coordinate of the orbit.

**x-BPM**

Toggle display of markers at x positions of the orbit where there are monitors.

**y-BPM**

Toggle display of markers at the y positions of the orbit where there are monitors.

**x-err**

Toggle display of markers at the x positions of non-zero element displacements.

**y-err**

Toggle display of markers at the y positions of non-zero element displacements.

**Symbols**

Toggle display of symbols representing the magnet elements on the s axis. Bends are one unit high, and edges are a half unit higher. Quads are two units high and are above or below the axis depending on whether they are focussing or defocussing. Multipoles of order n are n units high and are above or below the axis depending on the sign of the larger of a or b. The injection point is three units high. Monitors are small, square boxes.

**BPM-only**

Toggle display of the orbit at all points or only at monitor points. Showing the orbit with **BPM Only** and **x-BPM** and/or **y-BPM** represents what is known in a real device.

**BPM Only:**

These controls determine options for when **BPM Only** above is selected.

**True Values**

Shows the actual coordinates.

**Values as Read**

Shows the coordinates as actually read by the monitors. The offsets (errors or displacements) are subtracted from the actual values.

**Start Orbit From:**

These controls determine characteristics of the orbit that is started or continued via the Orbit Menu.

### **Specified**

Use the  $x_0$ ,  $x_0'$ ,  $y_0$ , and  $y_0'$  values specified below as initial values at the injection point when starting the orbit.

### **FTO**

Use the  $x_0$ ,  $x_0'$ ,  $y_0$ , and  $y_0'$  values from the last-calculated first-turn orbit as initial values at the injection point when starting the orbit.

### **CLO**

Use the  $x_0$ ,  $x_0'$ ,  $y_0$ , and  $y_0'$  values from the last-calculated closed orbit as initial values at the injection point when starting the orbit.

### **$x_0$ :, $x_0'$ :, $y_0$ :, and $y_0'$ :**

These text boxes set the initial values at the injection point which are used when starting the orbit if **Specified** above is selected. See Orbit Menu for how to start or continue the orbit.

### **Orbit Turns:**

This text box sets the number of turns to take when the orbit is started or continued. See Orbit Menu for how to start or continue the orbit.

### **Xvgr Plot:**

These controls determine if Xvgr plot files are created. These files can be read into Xvgr, then viewed, printed, and modified. Xvgr plots can be also displayed from Xorbit via the Options Menu.

### **ORB**

Make Xvgr plot files when the orbit is calculated. These files have names starting with the prefix of the input file name and ending with the extensions: .porbx, .porby, .porbyp, and .porbyp.

### **FTO**

Make Xvgr plot files when the first-turn orbit is calculated. These files have names starting with the prefix of the input file name and ending with the extensions: .pftox, .pftoy, .pftoxp, and .pftop.

### **CLO**

Make Xvgr plot files when the closed orbit is calculated. These files have names starting with the prefix of the input file name and ending with the extensions: .pclox, .pcloy, .pcloxp, and .pcloyp.

## **First Turn:**

These controls determine which of two ways the first–turn orbit is calculated. See First Turn.

### **Correct Only**

Correct the orbit for one turn only.

### **Correct and Close**

Correct the orbit for one turn and iterate on the injection angle until the orbit is closed.

## **Plot Options Dialog Box**

### **Top Controls**

#### **Remove**

Removes this dialog box.

#### **Redraw**

Clears all orbits in the plot graph area and redraws the current orbit.

#### **Update**

Calculates and plots the orbit for the current parameters.

#### **Help**

Provides help on this dialog box.

## **Options:**

### **Immediate Twiss**

Toggles whether the Twiss parameters are calculated immediately after a change in the element parameters or not. If not, they may be calculated via the Options Menu when desired. For visual feedback in the design mode, it is usually better to set this option on.

### **Symbols**

Toggle display of symbols representing the magnet elements on the s axis. Bends are one unit high, and edges are a half unit higher. Quads are two units high and are above or below the axis depending on whether they are focussing or defocussing. Multipoles of order n are n units high and

are above or below the axis depending on the sign of the larger of a or b. The injection point is three units high. Monitors are small, square boxes.

## Table Controls

For each parameter in the table, these controls determine:

- Column 1: Whether this parameter is plotted in the plot graph area or not. There are also input switches, `IPLOTxxx`, that control whether a variable is plotted or not.
- Column 2: The scale factor by which the values for this parameter is multiplied before plotting. This is one of two ways to control the relative sizes of the curves in the plot graph area. The other is the upper and lower limits control above the plot graph area. There are also input switches, `SCALExxx`, that specify the scale for a variable.
- Column 3: The color of the curve for this parameter. Any name which can be used as a color resource name may be used. These include colors in the color database (usually `/usr/lib/X11/rgb.txt`) and names in the forms `#RGB`, `#RRGGBB`, `#RRRGGBBB`, and `#RRRRGGGBBBB`, where R, G, and B are single hexadecimal digits representing the amounts of red, green, and blue. (For a 256-color system two digits for each color are appropriate and Red is `#FF000`, for example.)

## Design Mode Dialog Box

This dialog box allows the easy changing of elements with immediate feedback in the plot graph area. Typically this mode would be used for designing a lattice by changing element strengths and positions while watching their effect on the beta functions and dispersion.

The current element is chosen by clicking on its s value in either the graph area or the plot graph area or by clicking its icon in the icon bar.

## Top Controls

### Remove

Removes this dialog box.

### Redraw

Clears all orbits in the plot graph area and redraws the current orbit.

### Update

Calculates and plots the orbit for the current parameters.

## **Restore Lattice**

Restores the lattice as it was read from the input file.

## **Help**

Provides help on this dialog box.

## **Options:**

### **All Sectors**

Specifies whether the changes apply to the same element in all sectors or just to this particular element.

## **Attach To:**

Specifies which strength is associated with the value that is changing. See Element Input.

### **A**

The value that changes is the A strength.

### **B**

The value that changes is the B strength.

## **Position Change:**

Specifies the increment for position changes in meters. The modifications are not implemented until <CR> is pressed.

## **Value Change:**

Specifies the increment for strength changes in the units in which the strengths are specified. The modifications are not implemented until <CR> is pressed.

## **Current Value:**

Displays the current strength of the selected element. The strength can be changed via the text box. The modifications are not implemented until <CR> is pressed.

## **Current Element:**

Displays the name and index in the lattice of the current element. This text box is for display only and cannot be edited.

## Arrow Keys

The left and right arrows move the element sideways. It cannot cross the adjacent element, however. The up and down arrows increment the strength.

## Element Information Dialog Box

This dialog box displays information about an individual element and allows its strengths and displacements to be changed (if that is appropriate).

The information is current at the time the dialog box is displayed. Except for values that are changed via the text boxes, this information is not updated as the orbit changes. One can keep track of what is happening at an element by displaying new boxes as changes are made to the orbit.

The information includes the type of the element, its name and index in the lattice, the current  $x$ ,  $x'$ ,  $y$ , and  $y'$  coordinates of the orbit; the  $s$  coordinate of the element (in the current units); the horizontal and vertical Twiss parameters,  $\alpha_H$ ,  $\beta_H$ ,  $\gamma_H$ ,  $\alpha_V$ ,  $\beta_V$ , and  $\gamma_V$ ; the horizontal and vertical phases,  $\phi_H$  and  $\phi_V$  (relative to the injection point in units of  $\pi$ ); and the horizontal dispersion,  $\eta_X$ , and its derivative,  $\eta_X'$ . Both  $X$  and  $H$  refer to horizontal, and both  $Y$  and  $V$  refer to vertical.

There is a plot of the phase ellipses, red for horizontal and blue for vertical.

When appropriate there are text boxes that can be used to change the  $A$  and  $B$  strengths and the  $X$  and  $Y$  displacements,  $X$  Error and  $Y$  Error. (See [Element Input](#).) The modifications are not implemented until  $\text{Enter}$  is pressed.

There is a text box at the bottom for writing notes. (Nothing is done with what is written.) This facilitates keeping track of changes in the element parameters as the orbit changes.

## Top Controls

### Remove

Removes this dialog box.

### Remove All

Removes all the element dialog boxes that are currently displayed.

### Help

Provides help on this dialog box.

## Initial Twiss Parameters Dialog Box

This dialog box allows initial values,  $\alpha_{H0}$ ,  $\beta_{H0}$ ,  $\alpha_{V0}$ , and  $\beta_{V0}$  for the Twiss parameters and  $\eta_{X0}$  and  $\eta_{X0}'$  for the dispersion, to be set when there is a beamline. These are the values at the injection point. The modifications are not implemented until <CR> is pressed.

### Top Controls

#### Remove

Removes this dialog box.

#### Help

Provides help on this dialog box.

### Input File

The input file consists of:

- One Title line (Any text)
- Input NAMELIST (See Input Switches, End with " \$")
- Two alignment guide lines (For entering the elements)
- List of elements and their parameters (End with " \$")
- Zero or more Option Blocks (Start with "*\*<option>*", End with " \$")
  - First Turn
  - Girder Corrected
  - Quad Movement
- Ending line ("*\*End*")

### Sample Input File

An example of an input file is [aps.in](#).

### Input Switches

There are many switches, which control options in **Xorbit**. These are read in the NAMELIST part of the input file. Switches which are not explicitly entered are set to the **Xorbit** defaults.

#### List of Switches

### Element Input

An element in **Xorbit** is defined by the quantities: **name**, **type**, **a**, **b**, **slen**, **xbar**, **ybar**, and **ysig**. For

a multipole there is an additional quantity, **npole**, which is the multipole order. In most cases **a** is the normal strength of the element and **b** is the skew strength, but these quantities are different for a bend and for bend edges. **slen** is the length of the element, **xbar** is the mean displacement in the x direction, and **xsig** is the standard deviation of the displacement. Similarly, for **ybar** and **ysig**. If **xsig** or **ysig** is non-zero, the corresponding displacement of that element is determined randomly from a Gaussian distribution. The input quantities are read in the following order with the following format:

```
name,type,npole,a,b,slen,xbar,xsig,ybar,ysig
FORMAT(A,1X,A1,A1,1X,7F10.0)
```

The length of the names, **namesize**, is determined by the input switch **NAMESIZE**.

If a name is repeated, then the rest of its parameters are overwritten by the parameters for the first occurrence of that name. Consequently they do not need to be typed. Ignored, zero, or blank parameters also do not need to be typed.

The lines containing the element definitions are preceded by two lines which are ignored (and which can be used for alignment). The element definitions must be followed by a line containing "\$" to terminate the element input. See the sample input file.

The possible elements and their inputs in detail are:

- **Injection Point**

type: i  
npole: ignored and set to blank  
a: ignored and set to 0  
b: ignored and set to 0  
slen: ignored and set to 0

(There *must* be one injection point. It need not be at the start of the elements. Initial values for the orbits refer to this location in the lattice. For the first-turn algorithm, this point must be just before a monitor.)

- **Monitor or Marker**

type: m  
npole: ignored and set to blank  
a: ignored and set to 0  
b: ignored and set to 0  
slen: ignored and set to 0

- **Drift Length or Straight**

type: s  
npole: ignored and set to blank

a: ignored and set to 0  
b: ignored and set to 0  
slen: l

(l is the length of the drift space.)

- **Bend**

type: b  
npole: ignored and set to blank  
a: thetab / pi  
b: rho (Use 0 for no bend)  
slen: ignored and set to (rho thetab)

(thetab is the bend angle and rho is the radius.)

- **Edge**

type: e  
npole: ignored and set to blank  
a: thetae / pi  
b: rho (Use 0 for no bend)  
slen: ignored and set to 0

(thetae is angle between the xaxis and the magnet face. thetae = 0 for a sector magnet, and thetae = thetab for a parallel-face magnet. thetae is the same as E1 or E2 for an SBEND in Mad. rho is the radius.)

- **Nonlinear or Multipole**

type: n  
npole: 1-9 (Dipole is 1)  
a: An  
b: Bn  
slen: ignored and set to 0

(An and Bn are the coefficients in the expansion of the field:

$$B_y + i B_x = (B \rho / l) \text{Sum} [ (A_n + i B_n) (x + i y)^{(n-1)} ]$$

Correctors are specified as n1 multipoles. Note that for horizontal correctors An is the corrector strength in radians and that for vertical correctors Bn is (minus) the corrector strength in radians. For normal sextupoles (n3 multipoles), An is (B'' 1) / (2 B0 rho0). (There are no finite-length, nonlinear elements in Xorbit.)

- **Quadrupole or Combined Function**

type: q  
npole: ignored and set to blank  
a: K  
b: rho (Use 0 for a pure quadrupole with no bend)  
slen: l

(K is the quadrupole strength,  $B' / (B0 \text{ rho}0)$ ; rho is the radius for a combined-function element; and l is the length.)

- **Thin**

type: t  
npole: ignored and set to blank  
a: K l  
b: ignored and set to 0  
slen: ignored and set to 0

(K is the quadrupole strength and l is the length)

## **First-Turn Algorithm**

See the Xorbit Manual. An example is in the sample [input file](#).

## **Girder-Correction Algorithm**

See the Xorbit Manual.

## **Quad-Movement Algorithm**

See the Xorbit Manual.

- [Contents](#)
- [Overview](#)

### **DISCLAIMER**

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.