

WinAGILE

WINdows

Alternating Gradient Interactive Lattice dEsign

USERS' GUIDE

Version 4.0

P.J. Bryant

August 2010

Introduction

The original DOS version of AGILE released in 1991 was written as an educational tool that was meant to be simple, intuitive and fast enough to work interactively with the user. Once upgraded to MS Windows, it started to be used professionally and gathered a number of more advanced features, but hopefully much of its original simplicity and its intuitive interface remained. After some years WinAGILE became limited by its 16-bit compiler, which was ageing in the fast-moving world of computing. The present Users' Guide marks a major upgrade to a 32-bit compiler with much enhanced memory handling and improved programming capabilities. The prodigious advance in processor speeds has ensured that the program still responds quickly despite the increasing computational load of the improvements and the larger and more complicated lattices that can be treated. Being able to work interactively allows the user to develop a “feel” for how the lattice and beam behave. There are only a few calculations, such as space charge with direct particle-particle forces, for which the user will want to leave the program to run by itself.

Contents

Chapter 1	Description	1-14
1.1	What can WinAGILE do?	
1.2	Calculation strategy	
1.3	Program structure	
1.3.1	Main Window	
1.3.2	Edit Window	
1.3.3	Line Window	
1.3.4	Ring Window	
1.3.5	Numerical matching Window	
1.3.6	Kick and Misalignment Windows	
1.4	User environment	
1.4.1	Spreadsheet displays	
1.4.2	Graph / plot windows	
1.4.3	Dialogue boxes	
1.5	Getting started	
Chapter 2	Definitions and conventions	15-45
2.1	Lattice database	
2.1.1	Indices, unit names and alias names	
2.1.2	Parameters of the lattice elements	
2.1.3	Element types	
2.1.4	Beam aperture	
2.1.5	Half gaps and radii of inscribed circles	
2.1.6	Dipole fringe-field integral FINT	
2.2	Geometry	
2.2.1	Local beam co-ordinates	
2.2.2	Position and direction in the survey co-ordinate system	
2.2.3	Normal-mode roll angle	
2.2.4	Other conventions	
2.2.5	General comments on sign conventions	
Chapter 3	Main Window	46-81
3.1	Options Menu	
3.1.1	Set/change beam particle and energy...	
3.1.2	Chromaticity formula...	
3.1.3	Quadrupole fringe field correction...	
3.1.4	Quit AGILE (Ctrl+Q)	
3.2	File Menu	
3.2.1	New... (Ctrl+N)...	
3.2.2	Open WinAGILE lattice file... (Ctrl+O)...	
3.2.3	Edit... (Ctrl+E)...	
3.2.4	Save (Ctrl+S)...	
3.2.5	SaveAs... (Ctrl+A)...	
3.2.6	X-purge (Ctrl+X)...	
3.2.7	Edit and enter kicks	
3.2.8	Edit and enter misalignments...	
3.2.9	Open MAD-X file	
3.2.10	Open TRACE-3D file...	
3.2.11	Open TRANSPORT file...	
3.2.12	Recently-used files	
3.3	Calculations Menu	

- 3.3.1 Ring or matched section (Ctrl+R)...
- 3.3.2 Transfer line (Ctrl+L)...
- 3.3.3 Matrices (Ctrl+M)...
- 3.3.4 Geometry... (Ctrl+G)...
- 3.3.5 Fit geometry to an end point...
- 3.3.6 Fit geometry by closing a ring...
- 3.3.7 Optimise synchronous phase at centre of cavity...
- 3.4 Aids Menu
 - 3.4.1 Conventional magnet design aid
 - 3.4.2 Conventional magnet cost aid
 - 3.4.3 Water-cooled coil design aid...
 - 3.4.4 Vacuum design aid...
 - 3.4.5 Eddy current calculator...
 - 3.4.6 RF cavity performance aid...
 - 3.4.7 RFQ period performance aid...
 - 3.4.8 Relativistic calculator...
 - 3.4.9 Analytic matching...
- 3.5 Tables Menu
 - 3.5.1 Lattice elements
 - 3.5.2 Matrices
 - 3.5.3 Geometry
 - 3.5.4 Kicks
 - 3.5.5 Element misalignments
 - 3.5.6 Inventories...
 - 3.5.7 Go to line...
 - 3.5.8 Find and count...
- 3.6 Graphs Menu
 - 3.6.1 2D Geometry (X, Y);(Z, X);(Y, Z) (Ctrl+Y)
 - 3.6.2 3D Geometry
 - 3.6.3 Blank tune diagrams and resonance pattern
 - 3.6.4 Kicks
- 3.7 Output Menu
 - 3.7.1 Print display... (Ctrl+P)
 - 3.7.2 Write export file... (Ctrl+W)
 - 3.7.3 Copy selection to clipboard (Ctrl+C)
 - 3.7.4 Copy selection to notebook (Ctrl+K)
- 3.8 Help (F1)
- 3.9 Other functions

Chapter 4 *Edit Window*

82-102

- 4.1 Options Menu
 - 4.1.1 Back to the Main Window (F9)
 - 4.1.2 Quit AGILE (Ctrl+Q)
- 4.2 Edit Menu
 - 4.2.1 Insert line(s) (Ins)
 - 4.2.2 Delete line(s) (Del)
 - 4.2.3 Undelete lines one by one (F7)
 - 4.2.4 Copy line(s) (Ctrl+C)
 - 4.2.5 Cut line(s) (Ctrl+X)
 - 4.2.6 Paste lines (Ctrl+V)
 - 4.2.7 Inventories...
 - 4.2.8 Go to line...
 - 4.2.9 Find and count
 - 4.2.10 Global renaming of units...
- 4.3 Structures Menu

- 4.3.1 Repeat 'n' times...
- 4.3.2 Invert structure
- 4.3.3 Follow structure with reflection
- 4.3.4 Set vacuum chambers using customised criteria (in whole lattice or selected lines)...
- 4.3.5 Set half gaps using customised criteria (in whole lattice or selected lines)...
- 4.4 Auxiliary Data Menu
 - 4.4.1 Edit/enter data for scatterers...
 - 4.4.2 Edit/enter user-specified matrices...
 - 4.4.3 Edit/enter advanced RF cavity data...
 - 4.4.4 Edit/enter additional data for RFQs...
 - 4.4.5 Edit/enter additional data for helical magnetic dipoles...
 - 4.4.6 Edit/enter additional data for electrostatic bends...
 - 4.4.7 Edit/enter additional data for helical quadrupoles...
 - 4.4.8 Edit/enter additional data for magnetic wigglers/undulators...
- 4.5 Check Data Menu
 - 4.5.1 Check with decompression (if needed) (F2)
 - 4.5.2 Check without decompression (F8)
- 4.6 Goodies Menu
 - 4.6.1 Sub-divide selected unit (F6)
 - 4.6.2 Sub-divide all units (F12)
 - 4.6.3 Combine all divided units
 - 4.6.4 Import lattice... (F10)
- 4.7 Help Menu (F1)
- 4.8 Other functions
 - 4.8.1 Direct data entry (double-click cell)
 - 4.8.2 Collected data entry by element

Chapter 5 *Ring Window*

103-152

- 5.1 Options Menu
 - 5.1.1 Incoherent space charge (coasting 2D) ON...
 - 5.1.2 Incoherent space charge (bunched 3D) ON...
 - 5.1.3 Coherent space charge (coasting 2D) ON...
 - 5.1.4 Space charge OFF...
 - 5.1.5 Eddy currents (linear ramp) ON...
 - 5.1.6 Eddy currents OFF...
 - 5.1.7 Back to Main Window (F9)
 - 5.1.8 Quit AGILE (Ctrl+Q)
- 5.2 File Menu
 - 5.2.1 Save... (Ctrl+S)...
 - 5.2.2 SaveAs... (Ctrl+A)...
 - 5.2.3 Recover original lattice
 - 5.2.4 Edit and enter kicks
 - 5.2.5 Edit and enter misalignments...
 - 5.2.6 Import a measured closed-orbit file...
- 5.3 Calculations I Menu
 - 5.3.1 Point-to-point matrix
 - 5.3.2 Orbit bumps
 - 5.3.3 Tracking single particles...
 - 5.3.4 Envelopes...

- 5.3.5 Tracking transverse distributions (with momentum spread)...
- 5.3.6 Maps (linear lattice condensed)...
- 5.3.7 Closed orbit...
- 5.3.8 Statistics for closed orbit...
- 5.3.9 Correct closed orbit...
- 5.3.10 Statistics for corrected closed orbits...
- 5.3.11 Betatron modulation...
- 5.3.12 Statistics for betatron modulation...
- 5.3.13 Fit tunes/phase advances...
- 5.3.14 Fit chromaticities...
- 5.3.15 Fit tunes and Dx...
- 5.3.16 Fit tunes, Dx and dDx/ds...
- 5.4 Calculations II Menu
 - 5.4.1 Create Twiss on off-axis, closed orbit...
 - 5.4.2 Reset central orbit
 - 5.4.3 Reset Twiss on a stored, off-axis, closed orbit...
 - 5.4.4 Create Twiss on a distorted closed orbit...
 - 5.4.5 Reset undistorted central orbit
 - 5.4.6 Linear coupling, driving terms, single or summed...
 - 5.4.7 Statistics for linear coupling
 - 5.4.8 Resonance driving terms (2nd-5th order), single or summed...
 - 5.4.9 Fit coupling difference and/or sum resonance, single or summed, with or without 'Twiss' tune control...
 - 5.4.10 Decouple single-turn matrix...
 - 5.4.11 Decouple single-turn matrix with Teng tune control...
- 5.5 Tables Menu
 - 5.5.1 Sigmas matrices
 - 5.5.2 Twiss transverse parameters
 - 5.5.3 Twiss longitudinal parameters
 - 5.5.4 Normalised dispersion
 - 5.5.5 Twiss-style transverse bunched-beam parameters
 - 5.5.6 Teng-Edwards transverse and Twiss longitudinal parameters
 - 5.5.7 Trajectories transverse and longitudinal
 - 5.5.8 Losses
 - 5.5.9 Closed orbits
 - 5.5.10 Envelopes
 - 5.5.11 Betatron modulation
 - 5.5.12 Chromatic variables
 - 5.5.13 Linear coupling, diff. and sum resonances, single and summed modes
 - 5.5.14 Resonance driving terms (2nd-5th order), single or summed
 - 5.5.15 Separatrices
 - 5.5.16 Summary of eddy current data
 - 5.5.17 Other derived data
 - 5.5.18 Synchrotron radiation data
 - 5.5.19 Condensed lattice data
- 5.6 Graphs Menu
- 5.7 Output Menu
- 5.8 Help (F1)

Chapter 6 **Line Window** **153**

Chapter 7 **Numerical Matching Window** **154-162**

- 7.1 Options and File Menus
- 7.2 Minimisation Menu

- 7.2.1 Choice of variables...
- 7.2.2 Boundary conditions...
- 7.2.3 Steps...
- 7.2.4 Show variables and links
- 7.2.5 Minimisation
- 7.3 Tables, Graphs and Help

Chapter 8 *Plot Windows* 163-197

- 8.1 Transverse Twiss Plot Window
 - 8.1.1 Uncoupled, unbunched lattices with no scatterers
 - 8.1.2 Twiss-styled, bunched-beam parameters
 - 8.1.3 Twiss parameters with scattering
- 8.2 Longitudinal Twiss Plot Window
- 8.3 Dispersion Plot Window
- 8.4 Geometry Plot Window
 - 8.4.1 2D geometry (X, Y), (Z, X), (Y, Z)
 - 8.4.2 3D geometry
 - 8.4.3 Trajectories in 2D geometry (X, Y), (Z, X), (Y, Z)
- 8.5 Closed-orbit Plot Window
 - 8.5.1 Machine generated closed orbits
 - 8.5.2 Measured closed orbits
 - 8.5.3 Correction of closed orbits
 - 8.5.4 Editing monitor and corrector lists
- 8.6 Transverse Track Plot Window
- 8.7 Longitudinal Track Plot Window
- 8.8 Transverse Envelope Plot Window
- 8.9 Longitudinal Envelope Plot Window
- 8.10 Transverse Distribution Plot Window
- 8.11 Transverse Map Plot Window
- 8.12 Coupling Plot Window
- 8.13 W-vector Plot Window
- 8.14 Tune Diagram
- 8.15 Element icons

Chapter 9 *Kick and Misalignment Windows* 198-204

- 9.1 Kick Window
 - 9.1.1 Dipole error distributions for random errors
 - 9.1.2 Dipole error distributions for systematic errors
 - 9.1.3 Quadrupole error distributions for random errors
 - 9.1.4 Quadrupole error distributions for systematic errors
 - 9.1.5 Skew quadrupole error distributions for random errors
 - 9.1.6 Skew quadrupole error distributions for systematic errors
 - 9.1.7 Momentum kicks
- 9.2 Misalignment Window
 - 9.2.1 Random misalignments
 - 9.2.2 Systematic misalignments

Chapter 10 *How to do...* 205-208

- 10.1 Load a lattice file
- 10.2 Create and run a lattice
- 10.3 Adjust the tune values of a lattice
- 10.4 Export data and graphics
- 10.5 Combining graphs
- 10.6 On-line Help

* * *

Chapter 1 Description

WINAGILE is written for the IBM PC and MS Windows environment in DELPHI™ by Borland. It enables the user to simulate the behaviour of charged particle beams in synchrotron rings and transfer lines. The program is interactive and largely intuitive in its operation. Users of earlier Windows versions will find that the appearance and general usage are largely unchanged. Files from older versions can still be read and created for backwards compatibility.

1.1 What can WinAGILE do?

WinAGILE can handle lattices that bend, focus, accelerate, decelerate, scatter, introduce coupling and excite resonances up to decapole. Lattice elements can be magnetic, electrostatic and electromagnetic. For simulating beams, the program stores over one thousand different stable and quasi-stable beam ions, as well as allowing the user to define ions. In the longitudinal plane, RF cavities can be introduced singly, or as a series of coupled cavities, as in a linac. It is also possible to simulate an RFQ with individually tailored parameters for each cell. The user can also try some more exotic elements such as wigglers and helical dipoles.

In the majority of cases, the beam behaviour is analysed using the well-known Twiss formalism and by tracking either single particles or distributions using 6×6 matrices. However, the alternative tools of sigma matrices and the Teng-Edwards (Twiss-styled) parameters are also available. These formalisms have the advantage of being exact in the presence of transverse coupling. The sigma matrices are particularly useful for finding beam envelopes. Since the user is able to introduce his own matrices, it is possible to create lattices that cannot be treated by any of the above methods, in which case it is still possible to resort to particle tracking. Off-axis, as well as on-axis, calculations can be made and space-charge and eddy-current effects can be included. Lattice parameters and most results can be exported in files with standard spreadsheet formats, or simply printed out. Any user-defined region of the on-screen, spreadsheet displays and the data in the larger dialogue boxes can be copied to the Windows clipboard or a notebook that is stored with the lattice file. Graphs can be customised in a number of ways and superimposed for comparison. They can be plotted directly, or stored for pasting into reports in colour or monochrome versions.

Routines are included for the calculation of closed-orbit bumps, steering, closed-orbit distortion, betatron amplitude modulation, the statistical analysis of machines with randomly generated errors (dipole, quadrupole and skew quadrupole) and the correction of closed orbits. These routines are meant for linear machines and all non-linear elements are automatically switched off.

The tracking of single particles and the tracking of particle distributions work with all linear and non-linear elements switched on. When tracking particle distributions, loss plots can be made along the lattice. There are also expert routines for tasks such as multi-turn injection design, phase-space maps and the design of slow-extraction schemes for third-integer resonances. Diagnostic tools are provided for fitting ellipses to the beam

distributions in phase space. Many of these functions can be used with off-momentum orbits as well as the central orbit.

The beam can be visualised by distributions of ions and/or envelopes and be viewed with the vacuum chamber wall in a dynamic way as the observer travels around a ring or along a transfer line. Some specialised routines such as the design of slow-extraction schemes also use animated displays to describe the progress of the calculation.

Interactive lattice design is supported by fitting and matching routines. Fitting routines optimise geometry, set tunes, set chromaticities and set the driving terms of the second-order sum and difference coupling resonances. Numerical matching can be watched on-line as the graphical representation of the lattice changes. The matching variables, boundary conditions and steps can be edited at any time.

Transverse space charge can be treated in a number of ways. Incoherent and coherent space charge can be incorporated into the Twiss optics, but these two regimes cannot exist together. Incoherent space-charge calculations are also possible on off-axis orbits, but with self-fields only (images are switched off automatically). In general, the program automatically inhibits those calculations that are not appropriate for the type of space charge being used. Numerical matching and fitting can also be carried out with incoherent space charge applied when on central orbit. Another method for calculating transverse space charge is by tracking particle distributions with direct inter-particle forces, which allows all non-linearities, scattering and coupling to be included. This is an experimental routine that might typically be used for self-space charge forces at MeV energies in a multi-turn injection scheme.

One special feature is the calculation of sections of transfer lines that rotate on their axis. This is basically a coupling problem and has been included for the calculation of gantries and rotators in medical facilities.

Dipole, quadrupole, skew-quadrupole and momentum kick files can be generated to simulate typical random and systematic alignment and powering errors. These files can be generated in batches of up to 1000 for statistical studies. The possibility of calculating more exactly with misalignment matrices also exists.

Scatterers can be introduced into transfer lines with all the beam particles available in the program, except electrons, H minus ions and H2 plus ions. The program can treat the scattering data in two ways. The first method modifies the Twiss functions to calculate beam envelopes with the scattering included. The beam envelopes, beam emittances, average momentum loss, increase in momentum spread and transmission efficiency are tabulated along the whole lattice. It is also possible to use the thickness of the scatterers as a matching variable and to match the beam parameters and beam sizes (or emittances) in the numerical matching routine. The second method for treating scattering is to track particle distributions through lattices with scatterers. The principal scattering effect is multiple coulomb scattering, but nuclear elastic scattering can be added for protons when in the tracking mode.

The field distortion due to eddy currents can be evaluated and used to check the design of vacuum chambers, solid yokes, laminations and end plates of conventional magnets as a function of dimensions and the materials used. Calculation aids are also

given for conventional magnets, water-cooled coils, RF cavities, RFQ periods, vacuum systems and relativistic parameters. Finally, WinAGILE can write and read most MAD-X and Trace-3D files.

1.2 Calculation strategy

Apart from the self-contained design aids, the program follows a certain strategy:

Step 1. Obtain a lattice either by direct input from the user or from a file

- Elements can be defined by fields or by normalised parameters. Different elements can be defined in either way within the same lattice, but within one element the method must be uniform. Error checking at this stage may only be partial, e.g. a dipole with 100π radian bending angle will be signalled as unacceptable, but a dipole with a 10 T field will be accepted provisionally until the beam ion and its energy are known.
- Before making calculations, a lattice has to be checked by the program. Lattices read from files that have been previously checked do not require re-checking.
- Although most elements are completely defined within the spread sheet display, there are special elements that require additional parameters referred to as *auxiliary data*. This applies to scatterers, RF cavities, RFQs and some other specialised elements.

Step 2. Determine the synchronous beam ion and its energy along the lattice

- On the first attempt to use a lattice, a dialogue box will request confirmation of the ion and its momentum/energy. The program proposes a proton with 1 GeV/c as default. If the user is sure that this information will not be needed for his particular calculation, he should simply accept the defaults.
- Once the beam ion is confirmed, the program re-checks the lattice to see if dipoles and electrostatic bends that have been defined by fields have reasonable bending angles.
- The program then calculates the energy along the full length of the lattice on the equilibrium and central orbits. In most cases this is trivial, but for RF cavities the program allows two ways of configuring the downstream lattice that are governed by the parameter “*ramp*” (see Figure 1.1):
 - a) *ramp* = 1 In the downstream lattice, the new momentum is treated as the central orbit momentum so that the equilibrium orbit of the beam remains the central orbit of the lattice.
 - b) *ramp* = 0 In the downstream lattice, the new momentum is treated as being a momentum deviation and the central orbit is defined by the upstream momentum.
- The mode can be set independently for each active RF cavity. The result of the energy calculation will be seen in the ‘Lattice elements’ display in columns 39 to 43 inclusive (i.e. central orbit kinetic energy, the central orbit momentum, the relativistic gamma and beta values and the average momentum deviation of the beam). If the momentum deviation is zero, then the central and equilibrium orbits are equivalent and, if not, the two orbits will laterally diverge in the next dispersion region.
- Scatterers also change the momentum and *ramp* can be set to 0 or 1.
- RFQs also change the momentum but in this case only *ramp*=1 is allowed.
- Note that when establishing the synchronous ion, the ion enters the lattice by definition with zero lead or lag ($\Delta s = 0$). However, if there is an element with *ramp* = 0 then finite Δs and $\Delta p/p$ will be created with respect to a hypothetical beam that did not pass through the active element.

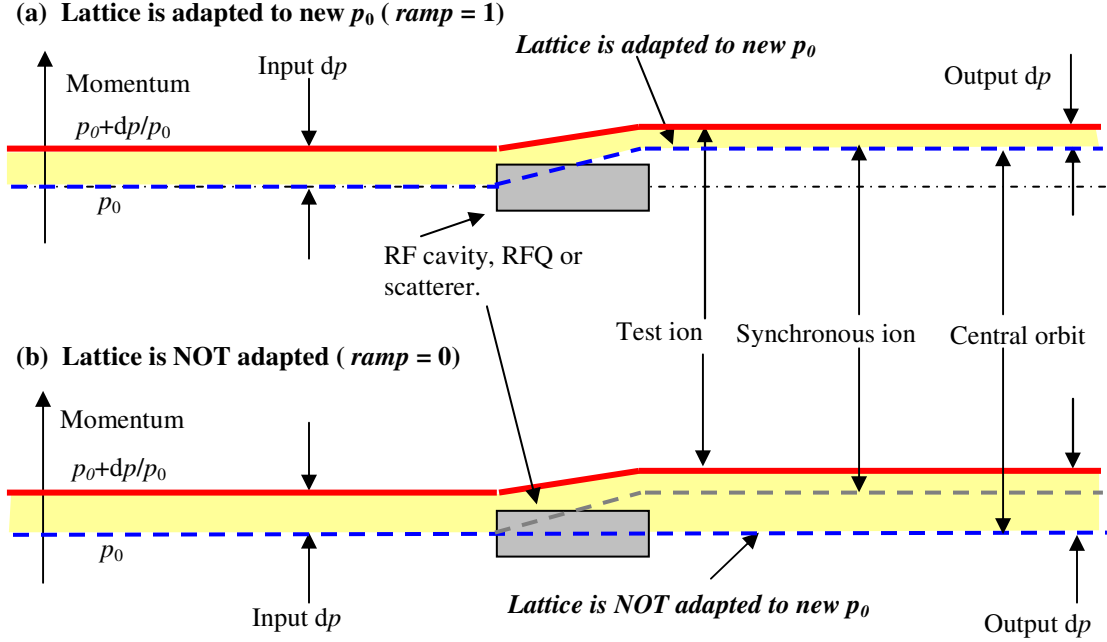


Figure 1.1 Graphical representation of how the program deals with momentum changes in RF cavities, RFQs and scatterers (determined by the *ramp* variable in the lattice data)

Step 3. Determine how to model the beam behaviour

- In the first instance, the program searches for how to model the beam with envelope functions.
 - Non-linear lenses (sextupole and above) are initially ignored to obtain the *linear lattice*.
 - A linear lattice without coupling, is always treated directly by the *Twiss formalism*.
 - A linear lattice containing tilted elements (except specific cases that produce non-coupling elements e.g. $\pi/2$ on a dipole) or rotation planes or user-defined matrices with coupling is always treated directly by *sigma matrices* for beam sizes and *Teng-Edwards parameters* for normal mode calculations.
 - A linear ring lattice with skew quadrupoles and solenoids can be treated, at the user's choice, either by a *perturbation method* or by the sigma matrices and Teng-Edwards method mentioned above. In the perturbation method, the coupling elements are replaced by drift spaces and the underlying uncoupled lattice is calculated with the normal Twiss formalism. The influence of the coupling elements is then added by a perturbation technique*.
 - If the Twiss formalism has been applied, then non-linear lenses (sextupole and above) that were originally removed are re-instated and calculated as perturbations to the linear lattice.

* The perturbation method is reserved for lattices with skew quadrupoles and solenoids, because it is usually true that the underlying linear lattice is dominant and can be found by simply replacing elements by drift spaces, whereas it is not meaningful to replace a tilted dipole by a drift space. To extract the essential features of the underlying linear lattice in this case, one would have to replace the dipole by say two horizontal bends flanking a vertical bend. Similarly, user-defined matrices with coupling that have a geometrical action ascribed to them cannot be simply replaced by a straight drift.

- In all the above cases, the lattice can also be analysed by tracking single particles or distributions of particles.
- In the case of a ring or matched section, it may be that the lattice is unstable transversally so that none of the envelope methods can be made to work. It may also be that exotic matrices introduced as user-defined elements may render the lattice intractable. In this case, the user can still proceed, but only by tracking single particles or distributions of particles.

Step 4. Calculations

- All calculations are separated into two categories according to the boundary conditions applying to the lattice:
 - Those using lattices with periodicity (i.e rings and matched sections).are collected in the Ring Window
 - Those using non-periodic lattices (i.e. transfer lines) that require the user to define the beam parameters at some point in the line are collected in the Line Window.
 - Scattering is not allowed in the Ring Window, as it destroys the periodicity condition.
 - Except for lines with coupling or scattering, the user is free to define the envelope parameters at any point along a line and the program will back-track to get the entry conditions.
- The calculation of off-axis orbits is somewhat unconventional. First the form of the orbit is established by dipole tracking. Using this trajectory the lattice elements are redefined, e.g. when the beam passes off-axis through a quadrupole this is redefined as a dipole with appropriate edge angles and a gradient. In this way, the off-axis orbit becomes the central orbit of the ‘new lattice’ and, in principle, the accuracy with which the focusing can be calculated is the same as for the original central orbit. For small momentum deviations, there is little difference between this method and using second-order matrices, but for large momentum deviations the precision should be progressively better.
- Strongly distorted orbits can be calculated in a similar way to off-axis orbits. A typical strongly distorted orbit would be an extraction trajectory between septum and kicker.

1.3 Program structure

The WinAGILE application is built in the Windows event-driven style. After initialising a data base in global memory that describes the beam and the lattice, the application launches a message loop that poles all input devices and then acts on events such as mouse clicks. There are a number of Principal Windows with their child Windows in which the calculation routines are hosted. The Principal Windows form a star structure with the Main Window at the centre, see Figure 1.2. The program opens in the Main Window and, in response to user input, it activates routines and switches between windows (using the Main Window as a gateway).

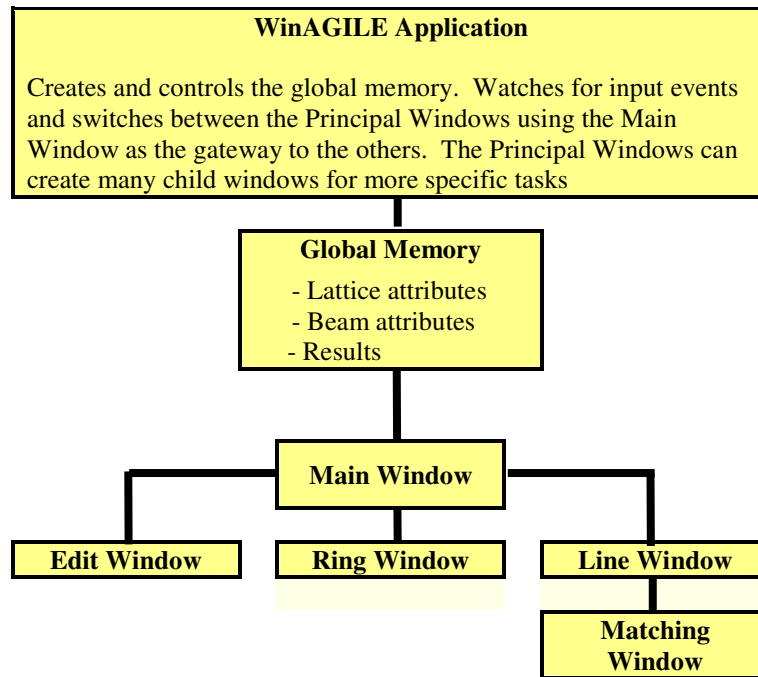


Figure 1.2 Structure of WinAGILE

1.3.1 Main Window

The program opens in the Main Window. This window performs basic functions such as the input and output of files and acts as a gateway to other windows. In addition, it hosts the engineering design-aids for such components as water-cooled coils. The tasks that can be performed in this window are basically those that do not need to distinguish between rings, matched sections and transfer lines.

1.3.2 Edit Window

The Edit Window is used for the creation, editing and checking of lattice files. The Edit Window is only accessible from the Main Window.

1.3.3 Line Window

The Line Window is devoted to the calculation of transfer lines, or more exactly non-periodic lines in which the input conditions do not necessarily equal those of the output. In these cases, the user is asked to define the beam parameters at some point along the line. The Line Window is only accessible from the Main Window.

1.3.4 Ring Window

The Ring Window is devoted to the calculation of closed rings and matched sections, or more exactly periodic sections of line for which the input conditions equal those of the output. The imposed periodicity makes it unnecessary for the user to specify any conditions at any point in the line. The Ring Window is only accessible from the Main Window.

1.3.5 Numerical Matching Window

The matching window is devoted to transverse matching by numerical minimisation. The Matching Window is only accessible from the Line Window, once a working transfer line has been established.

1.3.6 Kick and Misalignment Windows (not shown in Figure 1.2)

The Kick Window is used for creating and editing files of dipole, quadrupole, skew-quadrupole and momentum kicks. The Misalignment Window performs a similar job for lattice misalignments. Both windows are accessible from the Main, Line and Ring Windows.

1.4 User environment

The user will work in three basic environments: spreadsheet displays, graph windows and dialogue boxes.


1.4.1 Spreadsheet displays

The Main, Edit, Line and Ring Windows and a number of descendant windows are all of the spreadsheet type and resemble the Main Window shown in Figure 1.3

Unit no.	Name	Type	Length [m]	Hor. Mbend [rad]	Vert. Mbend [rad]	Edge angle-1 [rad]	Edge angle-2 [rad]	k-Mquad [1/m2]	kk-Msxt [1/m3]
1	FQH1	QUADR	0.5000	0.000000	0.000000	0.00000	0.00000	-0.103513	0.000000
2	ss1	DRIFT	9.0000	0.000000	0.000000	0.00000	0.00000	0.000000	0.000000
3	DQH1	QUADR	0.5000	0.000000	0.000000	0.00000	0.00000	0.103513	0.000000
4	DQH1	QUADR	0.5000	0.000000	0.000000	0.00000	0.00000	0.103513	0.000000
5	ss1	DRIFT	9.0000	0.000000	0.000000	0.00000	0.00000	0.000000	0.000000
6	FQH1	QUADR	0.5000	0.000000	0.000000	0.00000	0.00000	-0.103513	0.000000
7	End		20.0000	0.000000	0.000000	0.00000	0.00000	0.000000	0.000000

Figure 1.3 Main Window

Columns 1-3 give the index number of the element, its name and type, while columns 4-10 show data. Columns 1-3 remain fixed when scrolling horizontally and all columns scroll vertically. There is a maximum of 45 columns and, depending on the program version, 5000 or more lines. The window has the usual ‘System’ and ‘Menu’ bars, sliders for scrolling and a number of dedicated buttons that are listed in Table 1.1. The window can also be resized and will rescale its fonts and adapt the number of lines shown in order to maintain a useful display.

Button	Action
Quit	Also available in the Options menu and via the  in the System bar.
Alias/Nme	Toggles column 2 between the names to be used by the program (call these the ‘optical names’) and a set of alias names. The ‘optical names’ are used by the program to recognise units that are optically equivalent, e.g. all units with the same optical name will be automatically updated if one of them is edited and units with the same optical name will behave as if powered in series. In contrast, the alias name can be unique, it is not used by the program and it can be anything e.g. the number of the engineering drawing, the manufacturer’s series number, etc.
Grid	Toggles between a grid and no grid in the spreadsheet.
Fld/Norm	Toggles between the field and momentum-normalised parameter definitions of elements e.g. field [T] and bending angle [rad]. Only active when displaying the lattice elements.
Notes	Toggles between showing and hiding the Note Book that is stored with the lattice file.
Long/Tran	Toggles the column layout in certain displays to favour the longitudinal or transverse planes e.g. toggles the order in which RF equipment and magnets appear in the ‘Lattice Elements’ spreadsheet. It also affects the default displays and graphs when longitudinal and transverse cases exist. Only active when appropriate.
Unlabelled	Not for normal use.
Unlabelled	Not for normal use.
V#.#	Shows the version number and when held down shows the release number.
#	Shows the number of elements in the current lattice and when held down shows the maximum number of elements that can be handled.
AGILE	Displays the ‘About AGILE’ dialogue box.
▲ / Top	Scrolls display one step up / Moves to start of lattice.
▼ / End	Scrolls display one step down / Moves to end of lattice.
All	Select the whole lattice.
Graph	Shows the most appropriate graph, if any, for the data currently being displayed.

Notes: Buttons are listed anticlockwise around edge of the window starting top left.
Buttons that toggle two states show what will be obtained on clicking and NOT what is currently active.

Table 1.1 Dedicated buttons in the spreadsheet windows

The top panel, directly below the menu bar, comprises a matrix of boxes in 4 columns and 3 rows that are reserved for status information. These boxes may be empty or display the messages in Table 1.2. To the left, there is a ‘zoom box’ that shows what has been selected in the spreadsheet with a higher precision, or with more detail.

Col. 1, row 1	<ul style="list-style-type: none"> • 'No lattice in memory'. • 'Lattice unchecked' [before lattice can be used, it is necessary to go to the Edit Window and check the lattice under Check_data in the Menu bar]. • 'Lattice compressed' [before lattice can be used, it is necessary to go to the Edit Window and decompress the lattice under Check_data in the Menu bar]. • 'Lattice OK' [lattice appears to be OK, but bending angles must still be checked for the magnetic rigidity of the ion]. • 'Lattice OK with ion'.
Col. 1, row 2	<ul style="list-style-type: none"> • Quad fringe-field On' [fringe-field correction for quadrupoles affects ONLY the off-axis orbits]. • Quad fringe-field Off'.
Col. 1, row 3	<ul style="list-style-type: none"> • 'TL chromaticity eqn' [thin-lens chromaticity formula that uses central orbit values]. • 'HJM chromaticity eqn' [Hardt-Jaeger-Möhl chromaticity formula that uses central orbit values].
Col. 2, row 1	'Beam ion' [e.g. proton].
Col. 2, row 2	'xx.xx [GeV/u](entry)' [kinetic energy of ion in GeV per nucleon at entry to lattice].
Col. 2, row 3	<ul style="list-style-type: none"> • 'Non-space charge optics' [non-space-charge or single-particle optics]. • 'Incoh. space charge' [space charge forces that affect the incoherent motion are included]. • 'Coh. space charge' [space-charge forces (images) that affect the coherent motion are included – this applies on central orbit].
Col. 3, row 1	Spare
Col. 3, row 2	<ul style="list-style-type: none"> • 'Off-axis orbit' [an off-axis orbit has been established by tracking and the lattice has been re-expressed for this orbit]. • 'Distorted orbit' [a distorted orbit has been established by tracking and the lattice has been re-expressed for this orbit]. • 'On-axis orbit' [standard case of the central orbit].
Col. 3, row 3	'Coupled optics' [lattice contains skew quadrupoles or solenoids or rotated units that excite coupling and the coupling is switched on].
Col. 4, row 1	<ul style="list-style-type: none"> • 'Eddy currents On' [units for which eddy currents have been calculated have their field parameters modified]. • 'Eddy currents Off'.
Col. 4, row2	<ul style="list-style-type: none"> • 'Scatterers in beam' [scattering/absorbing elements are in the beam and switched on]. • 'Scattering off'.
Col. 4, row3	<ul style="list-style-type: none"> • 'RF cavities on' [includes RFQs] • 'RF cavities off'.

Table 1.2 Status panel messages

Between the top panel and the spreadsheet, there is a panel devoted to information about the current lattice file. This includes:

- Title for the spreadsheet, e.g. 'Lattice elements (on axis)'.
- Name of the lattice, copied from first 18 characters of the file name.
- Classification of lattice as a 'transfer line', 'circular machine' or 'matched section'.
- Optional title that the user can edit directly by clicking on it.
- The date and time of the last calculation affecting the display.

The spreadsheet always has a 'highlighted' cell, line, column or block of cells, although this may be temporarily outside the view of the window. The contents of a selected cell appear in the zoom box and, for lines, columns and blocks, the limits of the selection are given. The mouse actions and keyboard instructions that are common to most spreadsheet windows are listed in Table 1.3. Many routines take the selected cell(s) as the default input, e.g. for printing.

Moving around the "spreadsheet" displays	
Mouse:	
Scroll bars	Use scroll bars to navigate in spreadsheet
Drag small, grey, numbered boxes at the top of columns	Columns can be re-arranged by dragging the small, grey, numbered box in the top line. This change appears in the printed output. If a column is dragged to the edge of the display the display will scroll. To introduce a space, drag an empty column from the far right-hand side.
Keyboard:	
Arrows keys	Moves highlighted box one step and scrolls display if at an edge. If the highlighted box is out of view then the selection jumps to first visible cell on that row or column.
PgUp / PgDn	Scrolls highlighted box one page up / down.
Home / End	Moves highlighted box to beginning of line / end of line.
Ctrl +Home / Ctrl + End	Moves highlighted box to top of file / bottom of file.
Selecting cells, lines, columns and blocks	
Mouse:	
Left click over cell	Selects cell. Contents of cell will appear in zoom box.
Left click and drag diagonally	Selects a block of cells. If cursor moves beyond visible area the spread sheet will scroll. Zoom box shows limits of selection.
Left click in column 1	Selects row of cells
Left click in column 1 and drag vertically	Selects continuous group of lines.
Left click in title line of spreadsheet	Selects column of cells
Left click in title line and drag horizontally	Selects continuous group of columns.
Keyboard:	
Arrow keys	Move highlighted cell to wanted position. If highlighted cell is outside window, it jumps directly to first visible position when an arrow key is pressed. If the edge of the screen is reached the display will scroll.
Shift + → or Shift + ↓ or Shift + ←, or Shift + ↑	Selects a block of cells with its diagonal from the starting position to the final position after moving with arrow keys. If when moving the edge of the screen is reached the display will scroll.
Alt + → or Alt + ←	Selects columns from current position to the left or right respectively.
Alt + ↓ or Alt + ↑	Selects lines from current position upwards or downwards respectively.
Data entry	
Mouse:	
Double left click	Where data entry is allowed, this will open an edit dialogue box in the position of the zoom box.
Keyboard:	
Enter	Providing a single cell is highlighted, an edit box in the position of the zoom box will open as in the case of the double left click above.

Table 1.3 Universal mouse and keyboard commands in spreadsheets

There are also a small number of standard commands that appear in the menu bar of most spreadsheet windows and have keyboard shortcuts (see Table 1.4).

Menu	Menu item	Keyboard shortcut
Options	Quit	Ctrl + Q
	Back to prior window	F9
File	Save	Ctrl + S
	Save as...	Ctrl + A
Tables or Edit	Go to line...	F3
Output	Copy to clipboard	Ctrl+C
	Kopy to notebook	Ctrl+K
	Print current spreadsheet display	Ctrl+P
	Write an export data file	Ctrl+W
Help	Contents	F1

Table 1.4 Standard commands with keyboard short cuts in the menu bar of spreadsheets

1.4.2 Graph / plot windows

Figure 3.4 shows an example graph window. These windows all have a row of 30 control buttons across the bottom. There are also keyboard shortcuts, 'F1' for help and 'F9' for going back to the prior window, 'P' for printing and 'W' for writing a graphics file.

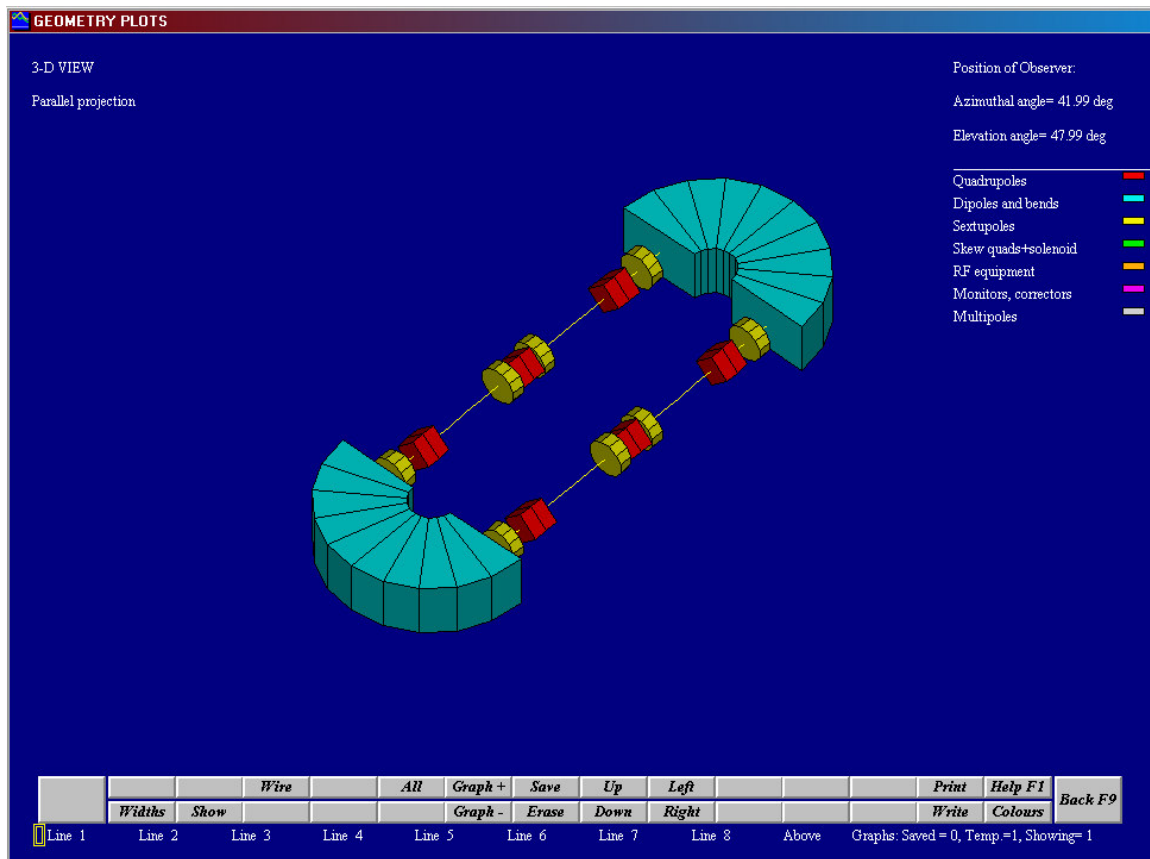


Figure 1.4 An example graph / plot window

1.4.3 Dialogue boxes

WinAGILE can be regarded as a collection of routines that manipulates a data set describing a beam and a lattice. Very often these routines are substantial programs in themselves and are handled through dialogue boxes both for the input of data and the return of results.

(a) Use of the Notebook to store and reload dialogue values

Many dialogue boxes contain the small group of controls shown in Figure 1.5. The two lower buttons either copy the data from the dialogue box to the Windows clipboard or the internal Notebook of the program. Furthermore, copying the time stamp in the internal Notebook and pasting it back into the dialogue box will enable the program to reload the data, so that repetitive calculations are less arduous.

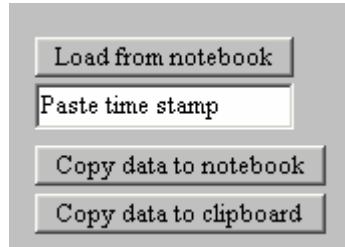


Figure 1.5 A sub-group of controls frequently used in dialogue boxes

(b) Advanced data entry in edit boxes

In many dialogue boxes, the edit boxes are coloured light blue instead of the standard white. The light blue background denotes that the edit box is equipped with an interpreter and can accept mathematically structured input.

- All forms of brackets (.), [..] and {..} are accepted and the recognised operators are: +, -, /, * and ^ for powers. A simple example would be "2^(3/2)".
- The recognised functions are: *sqr*, *sqrt*, *exp*, *log*, *ln*, *fac*, *gamma*, *sin*, *cos*, *tan*, *arcsin*, *arccos* and *arctan*, *sinh*, *cosh*, *tanh*, *arcsinh*, *arccosh* and *arctanh*.
- For π enter "Pi" and for e enter "exp(1)".
- Numerical input can be floating point, integer or in scientific notation 1.234E-3 and the alphabetic input is case insensitive.
- The main database can be accessed, for example "(bx[5]+bz[5])/2" calculates the average of the betatron amplitudes functions β_x and β_z at the entry to the 5th element. Table 1.5 lists the database arrays that can be accessed.

The interpreter is fairly tolerant and will, for example, accept "3Pi" for "3*pi". Brackets are not always needed, but are better typed in, for example "sin5Pi/4" will be understood as "sin(5*Pi/4)", but "sin5Pi/4-1" is ambiguous and may not be interpreted as expected. The interpreter will accept up to 50 characters. Once executed, the formula is not stored.

bx[i]	Twiss, β_x	ax[i]	Twiss, α_x	dx[i]	Dispersion, D_x
bz[i]	Twiss, β_z	az[i]	Twiss, α_z	dz[i]	Dispersion, D_z
l[i]	Length, ℓ	mux[i]	Twiss, μ_x	ddx[i]	Derivative, dD_x/ds
-	-	muz[i]	Twiss, μ_z	ddz[i]	Derivative, dD_z/ds
thex[i]	Hor. bending angle (magnetic lenses).	thez[i]	Vert. bending angle (magnetic lenses).	ang1[i]	Entry edge angle (magnetic lenses).
ethex[i]	Hor. bending angle (electro. lenses).	ethez[i]	Vert. bending angle (electro. lenses).	ang2[i]	Exit edge angle (magnetic lenses).
k[i]	Norm. quad grad (magnetic lenses).	kk[i]	Norm. sext grad. (magnetic lenses)	ks[i]	Norm. skew grad. (magnetic lenses)
ek[i]	Norm. quad grad (electro. lenses).	bs[i]	Norm. solenoid axial field	hg[i]	Half gap/radius of inscribed circle

Table 1.5 Access to arrays in the main database for data entry in edit boxes

1.5 Getting started

WinAGILE requires a 32-bit Windows operating system. The faster the processor, the more comfortable the response, but in practice 1 GHz is sufficient. For a simple installation, create a directory named WINAGILE on a writeable drive and copy the program file (WINAGILE.EXE), its help file (WINAGILE.HLP), the initiation file (WINAGILE.INI optional) and any lattice and data files (*.LAT) into that directory. The lattice file extension (*.LAT) can be associated with the program and a short-cut can be set up for the desktop in the usual way.

The program will not write to the Registry or try to access the internet, but it will create temporary files with the template 'AGxxxxxx.tmp' that may get left in the folder with the program if the program terminates abnormally. If the program is launched from a read-only drive, e.g. a CD-ROM, or network drive, then it searches for a suitable place to write temporary files. This will typically be c:\temp; c:\windows\temp or c:\.

When the program is launched it briefly shows a splash screen and then opens in the Main Window with a "Welcome" message. To quit the program at any time, click the dedicated button 'Quit' (on the left-hand side of the screen), or click 'Quit AGILE' in the 'Options Menu', or type the shortcut Ctrl + Q.

Once the program is running, click the 'File Menu' in the menu bar. 'New' will start a new computation by inviting you to create a new lattice and 'Open' will display a dialogue box for finding existing lattice files. Normally, WinAGILE is distributed with some specimen lattice files. Assuming that a lattice file exists, this can be loaded and its data will appear in the Main Window. The status of the lattice file is given in the 12 status boxes in the Status Bar at the top of the Window. If, by chance, the lattice file is in compressed format, incomplete, or unchecked, the program will suggest that the user clicks 'Edit' in the 'File Menu' to go to the Edit Window. In the Edit Window in the 'Check_Data' menu, click 'Check with decompression (if needed)'. The program will then signal any problems it finds and place the highlighted cell on the variable in doubt.

Once the lattice passes this check, the user will be asked whether he wishes to return to the Main Window.

Once the lattice is loaded and accepted, click on the 'Calculation Menu' and choose what to do with the lattice. If 'Ring' or 'Line' is chosen, the program tries to calculate the matrices, the geometry and the Twiss functions. For transfer lines the program first requests the user to input the Twiss functions at a single point (anywhere) in the line and to provide some emittance information. If the user knows the emittance will not be needed for what he wants to do, he should simply accept the defaults. For rings, there is a brief geometry message and the same request for emittance information. In addition, the dialogue box may also offer switching the RF on/off (if it exists) and a choice on how to treat coupling (again, if it exists). In some cases, there may be a warning that the ring is unstable, but in most cases, the Twiss functions will be successfully displayed. If the program is configured for transverse working, then the transverse Twiss functions will appear and, conversely, if it is configured for longitudinal working. However, this is only a superficial difference. The alternative display can either be found in the 'Table Menu' or the program configuration can be changed to give preference to the other plane by clicking the button 'Trans/Long' on the left-hand side of the screen. This can be done at any time and is a quick and convenient way to jump between the two regimes. For some lattices, it may not be possible to calculate the Twiss functions. In such case, the user will probably find the Teng-Edwards lattice functions, or maybe only the sigma matrices will be available. Whatever the situation, the program will present whatever it can calculate and, only in the most intractable cases, will the user be left with only matrices and tracking.

Once in the Ring or Line Window, the possible calculations and options can be found in the menus. The operation of the program is meant to be intuitive, but in the Help File there are details such as element definitions, sign conventions, etc. If you would like to do something routine, you may find an account of the steps in a "How To..." topic.

The definitions and conventions are described in the next chapter and the routines available in each Window are described in the chapters that follow. Chapter 10 explains how to do some specific tasks.

* * *

Chapter 2 Definitions and conventions

2.1 Lattice database

The lattice database is stored in a spreadsheet with 5000 rows (lattice elements) and 45 columns (parameters). The spreadsheet can be viewed via the menu item ‘Tables | Lattice elements’ that is available in all principal windows. The data can be entered and edited directly in the spreadsheet when in the Edit Window. The philosophy is to allow any data entry in the spreadsheet during the editing phase and then to check the data before leaving the Edit Window. At the time of checking, the program will make simple corrections automatically, while for other problems it will request the intervention of the user. It is also possible to work with a dialogue box that collects all the data relevant to one element (see Chapter 4).

The more exotic elements (scatterers, user-defined matrices, RF cavities, RFQs, helical magnets, electrostatic bends, helical electrostatic lenses, wigglers and the addition of eddy currents to magnetic elements) require additional or *auxiliary data*. This data is stored separately from the main database and is usually limited to 25 data sets per element type plus a set of default data.

2.1.1 Indices, unit names and alias names

Each element entered in the database can be unambiguously identified by its line number or *index* (column 1 in spreadsheet). In addition, each element can have a *unit name* of up to 12 characters (column 2 in spreadsheet) and an *alias name* of up to 30 characters (also column 2 in spreadsheet when toggled). Powered elements, such as quadrupoles, that have the same unit name are assumed to be connected in series. Giving the same unit name to more than one element forces them to be physically identical, except for some special parameters listed in Table 2.1.

The form and dimensions of the aperture (to be used for particle interception) can be assigned individually. In most cases the aperture is simply the vacuum chamber.
The rotation (roll) of the element about the longitudinal axis.
The <i>ramp</i> variable (set by the upstream RF cavity, RFQ or scatterer, see Section 1.2).
The synchronous phase of RF cavities and RFQs.
The cavity mode for RF cavities.

Table 2.1 Parameters that can be assigned individually to elements of the same name

Whenever an element is changed, the program searches for any other elements with the same name and updates them. Elements with no name are treated as unique elements and the program uses their index number for identification. In order to remove an element from a series string and give it new parameters, remove its name and preferably give it a new one. A unit name is case sensitive.

Note that invisible names of an arbitrary number of spaces are deleted automatically at the time of input and replaced by a ‘nullname’ of 12 spaces, which is interpreted as being unnamed. Leading and trailing spaces are automatically suppressed, but internal spaces are left. In addition, the program reserves certain symbols for adding to element names to create special names unique to the program. These symbols are simply ignored at the time of input should the user try to use them. Table 2.2 lists those symbols and their meanings.

< >	Added by the program to the name to indicate entry and exit slices of a subdivided dipole or solenoid.
!!	Added by the program to the name of an element whose optical characteristics changed when it was inverted.
%	Added by the program to the name of a thin-lens quadrupole used for the pseudo-octupole fringe-field correction. This only appears in off-axis orbits.
~	Added by the program to the name of a thin-lens multipole that is added to simulate eddy current effects.
_	Used by the program as a spacer when the index number is added to the element name to create a unique unit name in distorted or off-axis lattices.
{ }	Added by the program to the name of units that are formed by re-combining multiple slices, or repeats, of identical elements. The number of slices, or repeats, that have been recombined is put in the curly brackets. If you wish to remove this addendum in order to return to the base name, use the global renaming of units function.
?	Reserved as a ‘wild’ card in element searches.
@, #, \$, &	Reserved for future usage

Table 2.2 Symbols reserved by the program

The alias name has no influence on the program and can be used to store drawing numbers, position identifiers etc.

2.1.2 Parameters of the lattice elements

Initially the program accepts whatever values the user prescribes, but, before the lattice can be used, the parameters are checked against the requirements and limitations summarised in Table 2.3. In addition, the user often has the choice of specifying fields or normalised parameters. For example a magnetic dipole can be defined with a bending angle in radians or a magnetic field in Tesla. Initially, field values are accepted at face value, but once the central orbit ion or *reference ion* is defined the program re-checks against Table 2.3 to see if the values are physically acceptable. The choice of whether to specify fields or normalised values can be made individually for each element, but must be uniform within an element. Once the reference ion and its energy have been defined the screen displays can be toggled instantly between normalised and field values.

Negative values for lengths, magnet gaps, apertures, RF frequencies etc. are not allowed.
Only drift spaces, thin lenses, markers, pickups, correctors, matrices and rotation planes can have zero length.
A lattice must contain at least one non-zero length element to be viable.
All parameters that are not in the standard definition for an element type are automatically set to zero or a default value.
The general magnetic sector bend with variable edge angles and the cylindrical, spherical and toroidal electrostatic bends are limited to a maximum bending angle of 2.1 rad. To overcome this limit, concatenate two or more bends.
The magnetic rectangular bends are limited to a maximum of 1.4 rad.
Electrostatic deflectors (straight geometry) are limited to a bending angle of 0.2 rad.
The edge angles of a general magnetic bend cannot be greater than 0.7 rad.
All bending elements can bend either horizontally or vertically, but not both. However, the units can be rotated about the longitudinal axis.
Horizontal dipoles, but not vertical dipoles, can have added sextupole components.
Synchronous phase must be in the range $-\pi$ to $+\pi$ and cavity modes must be positive.
The modulation of RFQ vanes (max/min) cannot exceed 2.
Magnetic and electrostatic, normal and skew multipoles cannot have an order higher than decapole.
The line of sight calculated from the geometric data entered with a user-defined matrix must be equal to or less than the path length declared in the main data base.
(Outer –Inner aperture) and (Upper-Lower aperture) cannot be negative or zero.
Magnet pole half gap cannot be negative. (zero means ignore).
The <i>ramp</i> variable for an RFQ must be unity.

Table 2.3 Principal limitations and conventions applied to lattice data

2.1.3 *Element types*

Each lattice element must have a **type** definition (spreadsheet, column 3) of up to 5 letters. The recognised element types are listed in Table 2.4. When the type is input, it can be entered in upper or lower case characters. The routine recognises several 2 or 3-letter abbreviations that can be anywhere in the input string, e.g. ‘dr’ or ‘ss’ are recognised as a ‘drift’ or ‘straight section’ and are converted to ‘DRIFT’ at the moment when the lattice is checked.

DRIFT	Drift space or straight section
MARK	Marker (zero length drift space).
HPU	Horizontal beam position monitor (used by orbit correction routines).
VPU	Vertical beam position monitor (used by orbit correction routines).
HVPU	Combined HPU and VPU (used by orbit correction routines).
HCORR	Horizontal orbit corrector (used by orbit correction routines)
VCORR	Vertical orbit corrector (used by orbit correction routines).
HVCOR	Combined HCORR and VCORR (used by orbit correction routines).
SBEND	Magnetic, combined-function, sector bend with variable edge angles
RBEND	Magnetic, combined-function, rectangular bend
QUADR	Magnetic quadrupole lens
THINQ	Magnetic thin-lens quadrupole
SEXTU	Magnetic sextupole lens
MPOLE	Magnetic multipole thin lens of any order up to decapole (normal and skew)
SKEWQ	Magnetic skew quadrupole
SOLEN	Solenoid with round ends or end plates with slots.
MHELB	Magnetic helical dipole.
MHELQ	Magnetic helical quadrupole
MWIGG	Magnetic dipole wiggler / undulator
EBEND	Electrostatic combined-function (quadrupole only) bend.
EDEFL	Electrostatic parallel-plate deflector.
EQUAD	Electrostatic quadrupole.
EPOLE	Electrostatic multipole thin lens of any order up to decapole (normal and skew)
EHELQ	Electrostatic helical quadrupole
RFCV	Radio-frequency cavity
RFQC1 to 8	Radio-frequency quadrupole cells (8 variants)
ROTPL	Rotation plane
SCATR	Scatterer or absorber.
MATRX	User-defined and program-defined matrix for a section of linear lattice.

Table 2.4 Element types available in WinAGILE

Note that many elements are special cases of a more general element and are only defined for convenience. For example, the SBEND can be configured as a dipole, a quadrupole, a sextupole, a drift space or as any combination of these elements. Equally, the MPOLE can be used for all point lenses including dipole kicks and thin quadrupoles.

TYPE = 'DRIFT'

Column	Attributes to be set in Edit Window
1	Index number set by program.
2	Optional unit name of up to 12 characters [<, >, !, %, ~, _, { }, ?, @, #, \$, & are reserved] and an optional alias name [use 'Alias/Name' button to toggle names].
3	DRIFT It is sufficient to type 'dr' or 'ss' anywhere in the input string in upper or lower case characters.
4	Length along equilibrium orbit [m]. Length can be zero, but not negative.
33	Form of aperture: E = elliptical/round; R = rectangular/square; S = super-elliptical; D = diamond; Complex shapes can be built by combining basic forms. Default = R.
34	Inner aperture wall with respect to axis [m]. Default = -0.1 m.
35	Outer aperture wall with respect to axis [m]. Default = 0.1 m.
36	Lower aperture wall with respect to axis [m]. Default = -0.1 m.
37	Upper aperture wall with respect to axis [m]. Default = 0.1 m.

Notes: Any other attribute that is entered will be automatically set to zero by the routine that checks the data base just before leaving the Edit Window.

A drift space cannot be rotated about its longitudinal axis, but by adding rotation planes at the entry and exit the vacuum chamber can be rotated.

TYPE = 'MARK'

A marker is a zero length drift space. This is a convenience element for making the lattice easier to understand. It is sufficient to type 'ma' or 'rk' anywhere in the input string in upper or lower case characters. Its attributes are the same as for a drift space except that the length is zero. Any length attributed to a marker will be automatically set to zero by the routine that checks data when leaving the Edit Window.

TYPE = 'HPU', 'VPU' and 'HVPU'

Horizontal beam position monitors (HPU), vertical beam position monitors (VPU) and combined horizontal and vertical position monitors (HVPU) are finite or zero length drift spaces with type names that are recognised by the closed-orbit correction routine. These elements have the same attributes as a drift space. As with drift spaces they cannot be rotated about their longitudinal axis, but the addition of rotation planes will rotate the aperture/vacuum chamber. Although rotating alters the aperture, it will not convert a HPU into a VPU, for example. It is sufficient to type combinations such as 'HP', 'PH', 'VP', 'PV', 'HVP', 'VHP' etc. to obtain these elements.

The closed-orbit correction routine uses the beam position at the entrance to the monitor. In order to get the reading at the centre of a beam position monitor, split the monitor in two parts with Name(a) and Name(b) and label the first half as a DRIFT type and the second half as a HPU or VPU or HVPU type, e.g.

NAME = Bpm_102(a)	TYPE = DRIFT	L = 0.15m
NAME = Bpm_102(b)	TYPE = HPU	L = 0.15m.

TYPE = 'HCORR', 'VCORR' and 'HVCOR'

Horizontal orbit correctors (HCORR), vertical orbit correctors (VCORR) and combined horizontal and vertical orbit correctors (HVCOR) are finite or zero length drift spaces with type names that are recognised by the closed-orbit correction routine. These elements have the same attributes as a drift space. As with drift spaces they cannot be rotated about their longitudinal axis, but the addition of rotation planes will rotate the aperture/vacuum chamber. Although rotating alters the aperture, it will not convert a HCORR into a VCORR, for example. It is sufficient to type combinations such as 'HC', 'CH', 'VC', 'CV', 'HVC', 'VHC' etc. to obtain these elements.

The closed-orbit correction routine applies beam kicks at the entrance to the corrector. In order to place the kick at the centre of a corrector, split the corrector in two parts with Name(a) and Name(b) and label the first half as a DRIFT type and the second half as a HCORR or VCORR or HVCOR type, e.g.

NAME Corr_102(a)	TYPE = DRIFT	L = 0.15m
NAME = Corr_102(b)	TYPE = VCORR	L = 0.15m.

TYPE = 'QUADR'

A magnetic quadrupole can have both quadrupole and sextupole field gradients. The sextupole component is included as point kicks distributed with half at the entry to the quadrupole and half at the exit. In the case of a very long quadrupole with an added sextupole gradient, it may be advisable to sub-divide the quadrupole into shorter lengths.

Column	Attributes to be set in Edit Window
1	Index number set by program.
2	Optional unit name of up to 12 characters [<, >, !, %, ~, _ , { }, ?, @, #, \$, & are reserved] and an optional alias name [use 'Alias/Name' button to toggle names].
3	QUADR It is sufficient to type 'qu', 'qf' or 'qd' anywhere in the input string in upper or lower case characters.
4	Length along equilibrium orbit [m]. Length cannot be zero or negative.
9 (16)	Normalised quadrupole gradient, $k = -(1/B\rho)(dB_z/dx)$ [m ⁻²] or field gradient dB_z/dx [T/m] according to status of spreadsheet. Value can be zero.
10 (17)	Normalised sextupole gradient, $kk = -(1/B\rho)(d^2B_z/dx^2)$ [m ⁻³] or field gradient d^2B_z/dx^2 [T/m ²] Value can be zero
30	Radius of inscribed circle defined by magnet poles. (0=ignore)
32	Rotation of unit (BUT not its vacuum chamber) about longitudinal axis. Default = 0. To rotate the unit and its vacuum chamber use rotation planes.
33	Form of aperture: E = elliptical/round; R = rectangular/square; S = super-elliptical; D = diamond; Complex shapes can be built by combining basic forms. Default = R.
34	Inner aperture wall with respect to axis [m]. Default = -0.1 m.
35	Outer aperture wall with respect to axis [m]. Default = 0.1 m.
36	Lower aperture wall with respect to axis [m]. Default = -0.1 m.
37	Upper aperture wall with respect to axis [m]. Default = 0.1 m.

Notes: Any other attribute that is entered will be set to zero just before leaving the Edit Window.
Index numbers in brackets indicate column number when the display is in longitudinal mode.

TYPE = 'SEXTU'

Sextupoles are included in calculations as point kicks that are distributed with half at the entry to the element and half at the exit. In the case of a very long sextupole, it may be advisable to sub-divide the element into shorter lengths.

Column	Attributes to be set in Edit Window
1	Index number set by program.
2	Optional unit name of up to 12 characters [<, >, !, %, ~, _, { }, ?, @, #, \$, & are reserved] and an optional alias name [use 'Alias/Name' button to toggle names].
3	SEXTU It is sufficient to type 'se', 'xt' or 'sx' anywhere in the input string in upper or lower case characters.
4	Length along equilibrium orbit [m]. Length cannot be zero or negative.
10 (17)	Normalised sextupole gradient, $kk = -(1/B\rho)(d^2B_z/dx^2)$ [m^{-3}] or field gradient d^2B_z/dx^2 [T/m^2] Value can be zero.
32	Rotation of unit (BUT not its vacuum chamber) about longitudinal axis. Default = 0. To rotate the unit and its vacuum chamber use rotation planes.
33	Form of aperture: E = elliptical/round; R = rectangular/square; S = super-elliptical; D = diamond; Complex shapes can be built by combining basic forms. Default = R.
34	Inner aperture wall with respect to axis [m]. Default = -0.1 m.
35	Outer aperture wall with respect to axis [m]. Default = 0.1 m.
36	Lower aperture wall with respect to axis [m]. Default = -0.1 m.
37	Upper aperture wall with respect to axis [m]. Default = 0.1 m.

Notes: Any other attribute that is entered will be set to zero just before leaving the Edit Window.
(Outer –Inner aperture) and (Upper-Lower aperture) cannot be negative or zero.
Index numbers in brackets indicate column number when the display is in longitudinal mode.

TYPE = 'SBEND'

A magnetic, sector-bend dipole (SBEND) can bend vertically or horizontally, but **NOT BOTH**. The bending angle must be in the range ± 2.1 radian. For larger bends concatenate shorter sections. If the dipole is defined with a bending field rather than a bending angle, then the program accepts whatever data is given and checks later whether the bending angle is in range once the ion and its energy are known.

A horizontal sector bend dipole can have additional quadrupole and sextupole gradients, **BUT** a vertical sector bend dipole can only have an additional quadrupole gradient. This is meant to avoid confusion by obliging the user to construct what he wants. If the user wishes to turn a horizontal dipole through 90 degrees, this can be done by using the rotation parameter. In this case, the sextupole gradient will appear to the beam as a skew sextupole. If the user wants a normal sextupole plus a vertical bend then individual thin lenses must be added at the entry and exit to the vertically bending unit.

All attributes of a sector dipole can be zero **EXCEPT** the length and the vacuum chamber aperture. The sextupole component is included as point kicks that are distributed with half at the entry to the dipole and half at the exit. In the case of a very

long dipole with a sextupole gradient, it may be advisable to sub-divide the element into shorter lengths.

Column	Attributes to be set in Edit Window
1	Index number set by program.
2	Optional unit name of up to 12 characters [<, >, !, %, ~, _, { }, ?, @, #, \$, & are reserved] and an optional alias name [use 'Alias/Name' button to toggle names].
3	SBEND It is sufficient to type 'sb' or 'di' anywhere in the input string in upper or lower case characters.
4	Length along equilibrium orbit [m]. Length cannot be zero or negative.
5 (12)	Horizontal bending angle in range ± 2.1 [rad] or vertical field, B_z [T].
6 (13)	Vertical bending angle in range ± 2.1 [rad] or radial field, B_x [T].
7 (14)	Entry edge angle [rad] in range ± 0.7 [rad].
8 (15)	Exit edge angle [rad] in range ± 0.7 [rad].
9 (16)	Normalised quadrupole gradient, $k = -(1/B\rho)(dB_z/dx)$ [m^{-2}] or field gradient dB_z/dx [T/m] according to status of spreadsheet. Value can be zero.
10 (17)	Normalised sextupole gradient, $kk = -(1/B\rho)(d^2B_z/dx^2)$ [m^{-3}] or field gradient d^2B_z/dx^2 [T/m ²] Value can be zero. The sextupole gradient is only valid for horizontal bends.
30	Half gap defined by magnet poles. (0=ignore)
31	Fringe field integral, FINT. If in doubt use linear field decay model, FINT = 0.167.
32	Rotation of unit (BUT not its vacuum chamber) about longitudinal axis. Default = 0. To rotate the unit and its vacuum chamber use rotation planes.
33	Form of aperture: E = elliptical/round; R = rectangular/square; S = super-elliptical; D = diamond; Complex shapes can be built by combining basic forms. Default = R.
34	Inner aperture wall with respect to axis [m]. Default = -0.1 m.
35	Outer aperture wall with respect to axis [m]. Default = 0.1 m.
36	Lower aperture wall with respect to axis [m]. Default = -0.1 m.
37	Upper aperture wall with respect to axis [m]. Default = 0.1 m.

Notes: Any other attribute that is entered will be automatically set to zero just before leaving the Edit Window.

Index numbers in brackets indicate column number when the display is in longitudinal mode.

TYPE = 'RBEND'

A magnetic, rectangular-bend dipole (RBEND) has identical attributes to a sector bend dipole. The only differences are:

- the edge angles are half the absolute value of the bending angle by definition,
- the bending angle is limited to the range ± 1.4 [rad].

When the data is checked, the program will automatically set the edge angles to half the absolute bending angle whatever data has, or has not, been entered in the edge-angle cells. The user can exploit this to save time when defining RBENDs. The optical equivalent of a rectangular bend can be obtained by setting the edge angles of the sector bend to half the bending angle.

Column	Attributes to be set in Edit Window
1	Index number set by program.
2	Optional unit name of up to 12 characters [<, >, !, %, ~, _, { }, ?, @, #, \$, & are reserved] and an optional alias name [use 'Alias/Name' button to toggle names].
3	RBEND It is sufficient to type 'rb' anywhere in the input string in upper or lower case characters.
4	Length along equilibrium orbit [m]. Length cannot be zero or negative.
5 (12)	Horizontal bending angle in range ± 1.4 [rad] or vertical field, B_z [T].
6 (13)	Vertical bending angle in range ± 1.4 [rad] or radial field, B_x [T].
7 (14)	Entry edge angle is by definition half of the bending angle. The program will set this.
8 (15)	Entry edge angle is by definition half of the bending angle. The program will set this.
9 (16)	Normalised quadrupole gradient, $k = -(1/B\rho)(dB_z/dx)$ [m^{-2}] or field gradient dB_z/dx [T/m] according to status of spreadsheet. Value can be zero.
10 (17)	Normalised sextupole gradient, $kk = -(1/B\rho)(d^2B_z/dx^2)$ [m^{-3}] or field gradient d^2B_z/dx^2 [T/m ²] Value can be zero. The sextupole gradient is only valid for horizontal bends.
30	Half gap defined by magnet poles. (0=ignore)
31	Fringe field integral, FINT. If in doubt use linear field decay model, FINT = 0.167.
32	Rotation of unit (BUT not its vacuum chamber) about longitudinal axis. Default = 0. To rotate the unit and its vacuum chamber use rotation planes.
33	Form of aperture: E = elliptical/round; R = rectangular/square; S = super-elliptical; D = diamond; Complex shapes can be built by combining basic forms. Default = R.
34	Inner aperture wall with respect to axis [m]. Default = -0.1 m.
35	Outer aperture wall with respect to axis [m]. Default = 0.1 m.
36	Lower aperture wall with respect to axis [m]. Default = -0.1 m.
37	Upper aperture wall with respect to axis [m]. Default = 0.1 m.

Notes: Any other attribute that is entered will be automatically set to zero just before leaving the Edit Window.

Index numbers in brackets indicate column number when the display is in longitudinal mode.

TYPE = 'THINQ'

A thin-lens magnetic quadrupole has zero length by definition and an integrated quadrupole gradient, which can be any value including zero. Internal to the program, a thin-lens quadrupole is identical to a multipole lens (MPOLE) of order 2.

Column	Attributes to be set in Edit Window
1	Index number set by program.
2	Optional unit name of up to 12 characters [<, >, !, %, ~, _, { }, ?, @, #, \$, & are reserved] and an optional alias name [use 'Alias/Name' button to toggle names].
3	THINQ It is sufficient to type 'th' or 'tl' anywhere in the input string in upper or lower case characters.
4	Length along equilibrium orbit [m]. Length is zero by definition.
11 (18)	Integrated normalised gradient, $kt = -(\ell / B\rho)(dB_z/dx)$ [m^{-1}] or integrated field gradient $\ell dB_z/dx$ [T] Value can be zero.
32	Rotation of unit (BUT not its vacuum chamber) about longitudinal axis. Default = 0. To rotate the unit and its vacuum chamber use rotation planes.
33	Form of aperture: E = elliptical/round; R = rectangular/square; S = super-elliptical; D = diamond; Complex shapes can be built by combining basic forms. Default = R.
34	Inner aperture wall with respect to axis [m]. Default = -0.1 m.
35	Outer aperture wall with respect to axis [m]. Default = 0.1 m.
36	Lower aperture wall with respect to axis [m]. Default = -0.1 m.
37	Upper aperture wall with respect to axis [m]. Default = 0.1 m.

Notes: Any other attribute that is entered will be automatically set to zero just before leaving the Edit Window.

Index numbers in brackets indicate column number when the display is in longitudinal mode.

TYPE = 'MPOLE'

A multi-pole magnetic lens has zero length by definition and an integrated field that can be any value including zero. The lens can be set to any order from dipole to decapole with normal or skew orientation. These lenses can also be rotated about the longitudinal axis by an arbitrary angle.

Column	Attributes to be set in Edit Window
1	Index number set by program.
2	Optional unit name of up to 12 characters [<, >, !, %, ~, _, { }, ?, @, #, \$, & are reserved] and an optional alias name [use 'Alias/Name' button to toggle names].
3	MPOLE It is sufficient to type 'mp', 'nl', 'no' or 'mu' anywhere in the input string in upper or lower case characters.
4	Length along equilibrium orbit [m]. Length is zero by definition.
12 (19)	Integrated normalised gradient, $kn(n) = -(\ell / B\rho)(d^{n-1}B_z/dx^{n-1})$ [m^{-1}] or integrated field gradient $\ell d^{n-1}B_z/dx^{n-1}$ [T/m^{n-1}] Value can be zero.
13 (20)	Order of multipole lens, n where $2 n $ = number of poles in lens. Accepted values are: $n = 5$ thin decapole lens. $n = 4$ thin octupole lens. $n = 3$ thin sextupole lens. $n = 2$ thin quadrupole lens (equivalent to the THINQ type). $n = 1$ horizontal dipole point kick. $n = -1$ vertical dipole point kick (this is a skew lens). $n = -2$ thin skew quadrupole lens. $n = -3$ thin skew sextupole lens. $n = -4$ thin skew octupole lens. $n = -5$ thin skew decapole lens.
32	Rotation of unit (BUT not its vacuum chamber) about longitudinal axis. Default = 0. To rotate the unit and its vacuum chamber use rotation planes.
33	Form of aperture: E = elliptical/round; R = rectangular/square; S = super-elliptical; D = diamond; Complex shapes can be built by combining basic forms. Default = R.
34	Inner aperture wall with respect to axis [m]. Default = -0.1 m.
35	Outer aperture wall with respect to axis [m]. Default = 0.1 m.
36	Lower aperture wall with respect to axis [m]. Default = -0.1 m.
37	Upper aperture wall with respect to axis [m]. Default = 0.1 m.

Notes: Any other attribute that is entered will be automatically set to zero just before leaving the Edit Window.

Index numbers in brackets indicate column number when the display is in longitudinal mode.

TYPE = 'SKEWQ'

A magnetic skew quadrupole must have a finite length measured along the equilibrium orbit. Its skew gradient can assume any value including zero. The optical equivalent of a skew quadrupoles can be obtained by rotating a normal quadrupole by $\pi/4$.

Column	Attributes to be set in Edit Window
1	Index number set by program.
2	Optional unit name of up to 12 characters [<, >, !, %, ~, _, { }, ?, @, #, \$, & are reserved] and an optional alias name [use 'Alias/Name' button to toggle names].
3	SKEWQ It is sufficient to type 'sk', 'sq' or 'ew' anywhere in the input string in upper or lower case characters.
4	Length along equilibrium orbit [m]. Length cannot be zero or negative.
14 (21)	Normalised quadrupole gradient, $ks = -(1/B\rho)(dB_x/dx)$ [m^{-2}] or field gradient dB_x/dx [T/m] according to status of spreadsheet. Value can be zero.
32	Rotation of unit (BUT not its vacuum chamber) about longitudinal axis. Default = 0. To rotate the unit and its vacuum chamber use rotation planes.
33	Form of aperture: E = elliptical/round; R = rectangular/square; S = super-elliptical; D = diamond; Complex shapes can be built by combining basic forms. Default = R.
34	Inner aperture wall with respect to axis [m]. Default = -0.1 m.
35	Outer aperture wall with respect to axis [m]. Default = 0.1 m.
36	Lower aperture wall with respect to axis [m]. Default = -0.1 m.
37	Upper aperture wall with respect to axis [m]. Default = 0.1 m.

Notes: Any other attribute that is entered will be automatically set to zero by the routine that checks the data base just before leaving the Edit Window.

Index numbers in brackets indicate column number when the display is in longitudinal mode.

TYPE = 'SOLEN'

A solenoid must have a finite length measured along the equilibrium orbit. Its axial field can assume any value including zero. The ends of the solenoid can be round apertures or horizontal slots or vertical slots. There is also a fourth possibility with no end field. This is used when the body of the solenoid is subdivided into short lengths. If, for example, in a ring the lattice starts at the centre of a solenoid then the section of the solenoid at the entry to the lattice should start with a “no end-field” boundary and the section at the end of the lattice should equally be without an end field. Rotation can affect the transfer when the solenoid has slots.

Column	Attributes to be set in Edit Window
1	Index number set by program.
2	Optional unit name of up to 12 characters [<, >, !, %, ~, _, { }, ?, @, #, \$, & are reserved] and an optional alias name [use 'Alias/Name' button to toggle names].
3	SOLEN It is sufficient to type 'so' or 'oi' anywhere in the input string in upper or lower case characters.
4	Length along equilibrium orbit [m]. Length cannot be zero or negative.
15 (22)	Normalised axial field, $bs = b_s/B\rho$ [m^{-1}] or axial field b_s [T] according to status of spreadsheet. Value can be zero.
16 (23)	Entry field: 0=round, 1=horiz. slot, -1=vert. slot, -999= no end field (for slices in main body)
17 (24)	Exit field: 0=round, 1=horiz. slot, -1=vert. slot, -999= no end field (for slices in main body)
32	Rotation of unit (BUT not its vacuum chamber) about longitudinal axis. Default = 0. To rotate the unit and its vacuum chamber use rotation planes.
33	Form of aperture: E = elliptical/round; R = rectangular/square; S = super-elliptical; D = diamond; Complex shapes can be built by combining basic forms. Default = R.
34	Inner aperture wall with respect to axis [m]. Default = -0.1 m.
35	Outer aperture wall with respect to axis [m]. Default = 0.1 m.
36	Lower aperture wall with respect to axis [m]. Default = -0.1 m.
37	Upper aperture wall with respect to axis [m]. Default = 0.1 m.

Notes: Any other attribute that is entered will be automatically set to zero by the routine that checks the data base just before leaving the Edit Window.

Index numbers in brackets indicate column number when the display is in longitudinal mode.

TYPE = 'MHELB'

A magnetic helical dipole (spin rotator or Siberian snake) must have a finite length measured along its axis. In the first instance, the period of the helix is taken as being equal to the length entered in column 4, the dipole field is taken as 1 T and the sense of rotation takes the default value of clockwise. However, these values can be changed in the Edit Window in the 'Auxiliary Data' menu in the routine 'Edit/Enter additional data for helical dipoles'. The auxiliary database can hold up to 25 individual sets of data, in addition to the default values. Each data set is associated with a unit name and all elements with that name will use that data set. The helical dipole can be split into shorter lengths, e.g. for space charge calculations, but unlike wigglers no indication is given of the special orbit shape within the device.

Column	Attributes to be set in Edit Window
1	Index number set by program.
2	Optional unit name of up to 12 characters [<, >, !, %, ~, _, { }, ?, @, #, \$, & are reserved] and an optional alias name [use 'Alias/Name' button to toggle names].
3	MHELB It is sufficient to type 'helb' or 'held' anywhere in the input string in upper or lower case characters.
4	Length along axis of helix [m]. Length cannot be zero or negative.
5 (12)	Effective horizontal bending angle [rad] or dipole field, B [T].
32	Rotation of unit (BUT not its vacuum chamber) about longitudinal axis. Default = 0. To rotate the unit and its vacuum chamber use rotation planes.
33	Form of aperture: E = elliptical/round; R = rectangular/square; S = super-elliptical; D = diamond; Complex shapes can be built by combining basic forms. Default = R.
34	Inner aperture wall with respect to axis [m]. Default = -0.1 m.
35	Outer aperture wall with respect to axis [m]. Default = 0.1 m.
36	Lower aperture wall with respect to axis [m]. Default = -0.1 m.
37	Upper aperture wall with respect to axis [m]. Default = 0.1 m.

Notes: Any other attribute that is entered will be automatically set to zero by the routine that checks the data base just before leaving the Edit Window.

Index numbers in brackets indicate column number when the display is in longitudinal mode.

TYPE = 'MHELQ'

A magnetic helical quadrupole must have a finite length measured along the straight equilibrium orbit. The gradient is defined in the same way as for a normal quadrupole.

In the first instance, the period of the helix is taken as being equal to the length entered in column 4 and the sense of rotation assumes the default value of clockwise. However, these values can be changed in the Edit Window in the 'Auxiliary Data' menu in the routine 'Edit/Enter additional data for helical quadrupoles'. The auxiliary database can hold up to 25 individual sets of data, in addition to the default values (i.e. default period = length entered in main database and clockwise rotation of helix). Each data set is associated with a unit name and all elements with that name will use that data set.

Column	Attributes to be set in Edit Window
1	Index number set by program.
2	Optional unit name of up to 12 characters [<, >, !, %, ~, _, { }, ?, @, #, \$, & are reserved] and an optional alias name [use 'Alias/Name' button to toggle names].
3	MHELQ It is sufficient to type in upper or lower case characters.
4	Length along equilibrium orbit [m]. Length cannot be zero or negative.
9 (16)	Normalised quadrupole gradient, $k = -(1/B\rho)(dB_z/dx)$ [m ⁻²] or field gradient dB_z/dx [T/m] according to status of spreadsheet. Value can be zero.
32	Rotation of unit (BUT not its vacuum chamber) about longitudinal axis. Default = 0. To rotate the unit and its vacuum chamber use rotation planes.
33	Form of aperture: E = elliptical/round; R = rectangular/square; S = super-elliptical; D = diamond; Complex shapes can be built by combining basic forms. Default = R.
34	Inner aperture wall with respect to axis [m]. Default = -0.1 m.
35	Outer aperture wall with respect to axis [m]. Default = 0.1 m.
36	Lower aperture wall with respect to axis [m]. Default = -0.1 m.
37	Upper aperture wall with respect to axis [m]. Default = 0.1 m.

Notes: Any other attribute that is entered will be automatically set to zero by the routine that checks the data base just before leaving the Edit Window.

Index numbers in brackets indicate column number when the display is in longitudinal mode.

TYPE = 'DWIGG'

WinAGILE pre-supposes that a magnetic wiggler/undulatory has an integer number of cells with a half dipole at the entry and exit to reduce the effect on the closed orbit outside the device to zero. The length of a cell must be finite and is measured along the straight **UNPERTURBED** equilibrium orbit, see Figure 2.1. The aperture is also defined with respect to the **UNPERTURBED** equilibrium orbit. The default data assumes that the ratio of the pole width to the period is 0.25.

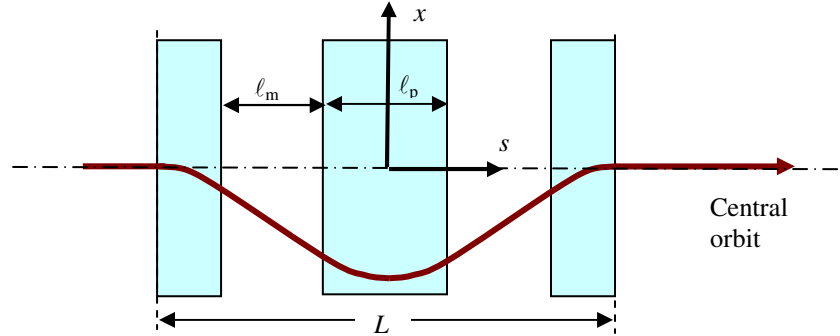


Figure 2.1 Central orbit in a single-cell wiggler/undulator

The magnetic field, the ratio of the pole width to the period, the field harmonics and the number of subdivisions can be changed in the Edit Window in 'Auxiliary I' 'Edit/Enter additional data for magnetic wigglers/undulators'.

Column	Attributes to be set in Edit Window
1	Index number set by program.
2	Optional unit name of up to 12 characters [<, >, !, %, ~, _, { }, ?, @, #, \$, & are reserved] and an optional alias name [use 'Alias/Name' button to toggle names].
3	DWIGG. It is sufficient to type 'wig' anywhere in the input string in upper or lower case characters.
4	Length along UNPERTURBED equilibrium orbit [m]. Length cannot be zero or negative.
30	Radius of inscribed circle defined by magnet poles. (0=ignore)
32	Rotation of unit (BUT not its vacuum chamber) about longitudinal axis. Default = 0. To rotate the unit and its vacuum chamber use rotation planes.
33	Form of aperture: E = elliptical/round; R = rectangular/square; S = super-elliptical; D = diamond; Complex shapes can be built by combining basic forms. Default = R.
34	Inner aperture wall with respect to axis [m]. Default = -0.1 m.
35	Outer aperture wall with respect to axis [m]. Default = 0.1 m.
36	Lower aperture wall with respect to axis [m]. Default = -0.1 m.
37	Upper aperture wall with respect to axis [m]. Default = 0.1 m.

Notes: Any other attribute that is entered will be automatically set to zero just before leaving the Edit Window.

Index numbers in brackets indicate column number when the display is in longitudinal mode.

TYPE = 'EBEND'

An electrostatic, sector-bend (EBEND) can bend vertically or horizontally, but **NOT BOTH** and the bending angle must be in the range ± 2.1 radian. For larger bends concatenate shorter sections. If the bend is defined with an electric field rather than a bending angle, then the program accepts whatever data is given and checks later whether the bending angle is in range once the ion and its energy are known. An electrostatic bend can have additional quadrupole focusing by curving the electrodes in the plane transverse to the bending. This curvature is stored as auxiliary data.

Column	Attributes to be set in Edit Window
1	Index number set by program.
2	Optional unit name of up to 12 characters [<, >, !, %, ~, _, { }, ?, @, #, \$, & are reserved] and an optional alias name [use 'Alias/Name' button to toggle names].
3	EBEND It is sufficient to type 'eb' anywhere in the input string in upper or lower case characters.
4	Length along equilibrium orbit [m]. Length cannot be zero or negative.
18 (25)	Horizontal bending angle in range ± 2.1 [rad] or horizontal electric field, E_x [kV/m].
19 (26)	Vertical bending angle in range ± 2.1 [rad] or vertical electric field, E_z [kV/m].
30	Half gap defined by deflector plates
32	Rotation of unit (BUT not the aperture limits) about longitudinal axis. Default = 0. To rotate the unit and the aperture limits use rotation planes.
33	Form of aperture: E = elliptical/round; R = rectangular/square; S = super-elliptical; D = diamond; Complex shapes can be built by combining basic forms. Default = R. The choice of the form of the aperture is not restricted, but the electrodes will basically impose a quasi-rectangular shape.
34	Inner aperture wall with respect to axis [m]. Default = -0.1 m.
35	Outer aperture wall with respect to axis [m]. Default = 0.1 m.
36	Lower aperture wall with respect to axis [m]. Default = -0.1 m.
37	Upper aperture wall with respect to axis [m]. Default = 0.1 m.

Notes: Any other attribute that is entered will be automatically set to zero just before leaving the Edit Window.

Index numbers in brackets indicate column number when the display is in longitudinal mode.

It is important to note that the bending angles for electrostatic devices and the bending angles for magnetic devices have different columns that are widely spaced in the main spreadsheet. If data is entered in the wrong column, e.g. the bending angle for a dipole is entered in column 18 that is meant for an electrostatic bend, the program will not object in the first instance, but will set the entry to zero during the data check, which may go unnoticed until the program starts to calculate the geometry.

The auxiliary database can hold up to 25 individual sets of data, in addition to the default values for cylindrical electrodes. Each data set is associated with a unit name and all elements with that name will use that data set.

TYPE = 'EDEFL'

A parallel-plate, deflector can bend vertically or horizontally, but **NOT BOTH** and the bending angle must be in the range ± 0.2 radian. For larger bends an EBEND must be used. If the bend is defined with an electric field rather than a bending angle, then the program accepts whatever data is given and checks later whether the bending angle is in range once the ion and its energy are known.

Column	Attributes to be set in Edit Window
1	Index number set by program.
2	Optional unit name of up to 12 characters [<, >, !, %, ~, _ , { }, ?, @, #, \$, & are reserved] and an optional alias name [use 'Alias/Name' button to toggle names].
3	EDEFL It is sufficient to type 'ed' anywhere in the input string in upper or lower case characters.
4	Length along equilibrium orbit [m]. Length cannot be zero or negative.
18 (25)	Horizontal bending angle in range ± 0.2 [rad] or horizontal electric field, E_x [kV/m].
19 (26)	Vertical bending angle in range ± 0.2 [rad] or vertical electric field, E_z [kV/m].
30	Half gap defined by deflector plates.
32	Rotation of unit (BUT not the aperture limits) about longitudinal axis. Default = 0. To rotate the unit and aperture limits use rotation planes.
33	Form of aperture: E = elliptical/round; R = rectangular/square; S = super-elliptical; D = diamond; Complex shapes can be built by combining basic forms. Default = R. The choice of the form of the aperture is not restricted, but the electrodes will basically impose a quasi-rectangular shape.
34	Inner aperture wall with respect to axis [m]. Default = -0.1 m.
35	Outer aperture wall with respect to axis [m]. Default = 0.1 m.
36	Lower aperture wall with respect to axis [m]. Default = -0.1 m.
37	Upper aperture wall with respect to axis [m]. Default = 0.1 m.

Notes: Any other attribute that is entered will be automatically set to zero by the routine that checks the data base just before leaving the Edit Window.

Index numbers in brackets indicate column number when the display is in longitudinal mode.

It is important to note that the bending angles for electrostatic devices and the bending angles for magnetic devices have different columns that are widely spaced in the main spreadsheet. If data is entered in the wrong column, e.g. the bending angle for a dipole is entered in column 18 for an electrostatic bend, the program will not object in the first instance, but will set the entry to zero during the data check, which may go unnoticed until the program starts to calculate the geometry.

TYPE = 'EQUAD'

The program requests the voltage across the electrodes rather than the electric gradient. This is frequently done because the voltage applied across the electrodes of the lens and hence the potential is well defined and calculation can lead to an accurate knowledge of the field. Whereas for magnets, the corresponding pole tip field is difficult to measure and the current in the coil is related non-linearly to the field.

Column	Attributes to be set in Edit Window
1	Index number set by program.
2	Optional unit name of up to 12 characters [<, >, !, %, ~, _, { }, ?, @, #, \$, & are reserved] and an optional alias name [use 'Alias/Name' button to toggle names].
3	EQUAD It is sufficient to type 'eq' anywhere in the input string in upper or lower case characters.
4	Length along equilibrium orbit [m]. Length cannot be zero or negative.
20 (27)	Normalised quadrupole gradient, $k = (1/E\rho)(dEx/dx)$ [m^{-3}] or the electrode voltage [kV] according to status of spreadsheet. Value can be zero.
30	Radius of inscribed circle defined by deflector plates.
32	Rotation of unit (BUT not the aperture limits) about longitudinal axis. Default = 0. To rotate the unit and the aperture limits use rotation planes.
33	Form of aperture: E = elliptical/round; R = rectangular/square; S = super-elliptical; D = diamond; Complex shapes can be built by combining basic forms. Default = R. The choice of the form of the aperture is not restricted, but the best compromise is probably the inscribed circle or square.
34	Inner aperture wall with respect to axis [m]. Default = -0.1 m.
35	Outer aperture wall with respect to axis [m]. Default = 0.1 m.
36	Lower aperture wall with respect to axis [m]. Default = -0.1 m.
37	Upper aperture wall with respect to axis [m]. Default = 0.1 m.

Notes: Any other attribute that is entered will be automatically set to zero.
(Outer –Inner aperture) and (Upper-Lower aperture) cannot be negative or zero.
Index numbers in brackets indicate column number when the display is in longitudinal mode.

TYPE = 'EPOLE'

The nonlinear electrostatic lens has been included, but unfortunately it is not the equivalent of the magnetic version. Magnetic non-linear lenses are points kicks and can be added to lattices very easily to simulate special effects without disturbing the geometry. For electrostatic elements, the concept of a point lens poses problems, because the magnitude of the kick depends on the transit time. For this reason, the EPOLE must have a finite length that corresponds to reality.

The lens can be set to any order from dipole to decapole with normal or skew orientation. These lenses can also be rotated about the longitudinal axis by an arbitrary angle.

It is important to note that the gradient for an EPOLE and the gradient for an MPOLE have different columns, but the order of these two multipoles uses the same column.

Column	Attributes to be set in Edit Window
1	Index number set by program.
2	Optional unit name of up to 12 characters [<, >, !, %, ~, _, { }, ?, @, #, \$, & are reserved] and an optional alias name [use 'Alias/Name' button to toggle names].
3	EPOLE It is sufficient to type 'ep' anywhere in the input string in upper or lower case characters.
4	Length along equilibrium orbit [m]. Length must be realistic and non-zero.
13 (20)	Order of multipole lens, n where $2 n $ = number of poles in lens. Accepted values are: $n = 5$ decapole lens. $n = 4$ octupole lens. $n = 3$ sextupole lens. $n = 2$ quadrupole lens. $n = 1$ horizontal bend. $n = -1$ vertical bend (skew). $n = -2$ skew quadrupole lens. $n = -3$ skew sextupole lens. $n = -4$ skew octupole lens. $n = -5$ skew decapole lens.
21 (28)	Integrated normalised gradient, $kn(n) = -(\ell/E\rho)(d^{n-1}E_x/dx^{n-1})$ [m^{n-1}] or the electrode voltage [kV] according to status of spreadsheet. Value can be zero.
30	Radius of inscribed circle defined by deflector plates.
32	Rotation of unit (BUT not the aperture limits) about longitudinal axis. Default = 0. To rotate the unit and the aperture limits use rotation planes.
33	Form of aperture: E = elliptical/round; R = rectangular/square; S = super-elliptical; D = diamond; Complex shapes can be built by combining basic forms. Default = R. The choice of the form of the aperture is not restricted, but the best compromise is probably the inscribed circle or square
34	Inner aperture wall with respect to axis [m]. Default = -0.1 m.
35	Outer aperture wall with respect to axis [m]. Default = 0.1 m.
36	Lower aperture wall with respect to axis [m]. Default = -0.1 m.
37	Upper aperture wall with respect to axis [m]. Default = 0.1 m.

Notes: Any other attribute that is entered will be automatically set to zero just before leaving the Edit Window.

Index numbers in brackets indicate column number when the display is in longitudinal mode.

TYPE = 'EHELQ'

A helical electrostatic quadrupole must have a finite length measured along the straight equilibrium orbit. The gradient is defined in the same way as for a normal quadrupole.

In the first instance, the period of the helix is taken as being equal to the length entered in column 4 and the sense of rotation assumes the default value of clockwise. However, these values can be changed in the Edit Window in the 'Auxiliary Data' menu in the routine 'Edit/Enter additional data for helical quadrupoles'. The auxiliary database can hold up to 25 individual sets of data, in addition to the default values (i.e. default period = length entered in main database and clockwise rotation of helix). Each data set is associated with a unit name and all elements with that name will use that data set.

Column	Attributes to be set in Edit Window
1	Index number set by program.
2	Optional unit name of up to 12 characters [<, >, !, %, ~, _, { }, ?, @, #, \$, & are reserved] and an optional alias name [use 'Alias/Name' button to toggle names].
3	EHELQ in upper or lower case characters.
4	Length along equilibrium orbit [m]. Length cannot be zero or negative.
20 (27)	Normalised quadrupole gradient, $k = (1/E\rho)(dEx/dx)$ [m^{-3}] or the electrode voltage [kV] according to status of spreadsheet. Value can be zero.
30	Radius of inscribed circle defined by deflector plates.
32	Rotation of unit (BUT the aperture limits) about longitudinal axis. Default = 0. To rotate the unit and the aperture limits use rotation planes.
33	Form of aperture: E = elliptical/round; R = rectangular/square; S = super-elliptical; D = diamond; Complex shapes can be built by combining basic forms. Default = R. The choice of the form of the aperture is not restricted, but the best compromise is probably the inscribed circle or square
34	Inner aperture wall with respect to axis [m]. Default = -0.1 m.
35	Outer aperture wall with respect to axis [m]. Default = 0.1 m.
36	Lower aperture wall with respect to axis [m]. Default = -0.1 m.
37	Upper aperture wall with respect to axis [m]. Default = 0.1 m.

Notes: Any other attribute that is entered will be automatically set to zero just before leaving the Edit Window.

Index numbers in brackets indicate column number when the display is in longitudinal mode.

TYPE = 'RFCV'

An RF cavity must have a finite length that is measured along the straight equilibrium orbit. The cavity is assumed to be rotationally symmetric, so that the rotation has no meaning and is set to zero by definition. There is no simplified RF gap available in WinAGILE. If the user needs to lighten the amount of computation he should reduce the number of integration steps (in the auxiliary data). For fully relativistic cases with constant ion velocity there is no need for more than 1 or 2 integration steps. There is also no need to provide a transit time factor as this is calculated by the program. The harmonics of the cavity field, the number of integration steps and the number of slices the cavity is divided into for viewing are set in the Edit Window in the 'Auxiliary Data' menu in the routine 'Edit/Enter advanced RF cavity data'.

Column	Attributes to be set in Edit Window
1	Index number set by program.
2	Optional unit name of up to 12 characters [<, >, !, %, ~, _, { }, ?, @, #, \$, & are reserved] and an optional alias name [use 'Alias/Name' button to toggle names].
3	RFCAV It is sufficient to type 'cav' or 'gap' anywhere in the input string in upper or lower case characters.
4	Length along equilibrium orbit [m]. Length cannot be zero or negative.
23 (5)	Peak RF voltage [kV]
24 (6)	Harmonic number when in normalised mode and RF frequency [MHz] when in field mode.
25 (7)	Synchronous phase [rad] of synchronous ion at entry to cavity. If the user wishes to adjust the RF according to the synchronous phase at the centre of the cavity use the routine 'Optimise synchronous phase at centre of cavity' in the Calculation Menu of the Main Window.
26 (8)	Cavity mode. Phase shift [rad] between adjacent cavities. If the cavity is uncoupled or it is the first cavity of a string then enter zero.
27 (9)	<i>ramp</i> = 0. The lattice downstream is not adapted for the momentum change introduced by the cavity and the equilibrium orbit of the beam is treated as having a momentum deviation. <i>ramp</i> = 1. The lattice downstream is adjusted for the momentum change introduced by the cavity and the equilibrium orbit of the beam becomes the central orbit of the lattice.
33	Form of aperture: E = elliptical/round; R = rectangular/square; S = super-elliptical; D = diamond; Complex shapes can be built by combining basic forms. Default = R. The choice of the form of the aperture is not restricted, but the best compromise is probably a circle
34	Inner aperture wall with respect to axis [m]. Default = -0.1 m.
35	Outer aperture wall with respect to axis [m]. Default = 0.1 m.
36	Lower aperture wall with respect to axis [m]. Default = -0.1 m.
37	Upper aperture wall with respect to axis [m]. Default = 0.1 m.

Notes: Any other attribute that is entered will be automatically set to zero by the routine that checks the data base just before leaving the Edit Window.

Index numbers in brackets indicate column number when the display is in longitudinal mode.

TYPE = 'RFQC1', 'RFQC2', ...up to ...'RFQC8'

An RFQ cell must have a finite length that is measured along the straight equilibrium orbit. WinAGILE does not allow the RFQ structure to be rotated about its axis because of problems with the synchronous phase. Some flexibility in the structure geometry is provided by the definition of 8 cells that correspond to half and quarter periods at different positions along the structure, see Figure 2.2. Since RFQs operate at low energies the number of integration steps could be important. A little time spent exploring the parameters using the 'RFQ period performance aid' in the Main Window will show the influence of the number of integration steps. In most cases, the default of 24 steps will be more than adequate and the speed of modern processors is such that this will not be a limitation.

Column	Attributes to be set in Edit Window
1	Index number set by program.
2	Optional unit name of up to 12 characters [<, >, !, %, ~, _ , { }, ?, @, #, \$, & are reserved] and an optional alias name [use 'Alias/Name' button to toggle names].
3	RFQC1, RFQC2, RFQC3...RFQC8. The full character string must be entered correctly in upper or lower case characters.
4	Length along equilibrium orbit [m]. Length cannot be zero or negative.
23 (5)	Peak RF voltage [kV]
24 (6)	RF frequency [MHz] when in field mode and harmonic number when in normalised mode. The harmonic number is not useful for RFQs.
25 (7)	Synchronous phase [rad] of synchronous ion at entry to the cell.
28	Modulation of vanes, m = maximum half gap/minimum half gap.
30	Minimum half-gap (minimum radius of inscribed circle radius) [m].
33	Form of aperture: E = elliptical/round; R = rectangular/square; S = super-elliptical; D = diamond; Complex shapes can be built by combining basic forms. Default = R.
34	Left aperture wall w.r.t. axis and looking in beam direction[m]. Default = -0.1 m.
35	Right aperture wall w.r.t. axis and looking in beam direction[m]. Default = 0.1 m.
36	Lower aperture wall w.r.t. axis [m]. Default = -0.1 m.
37	Upper aperture wall w.r.t. axis [m]. Default = 0.1 m.

Notes: All other attributes are automatically set zero or a default value before leaving the Edit Window.
Index numbers in brackets indicate column number when the display is in longitudinal mode.
The most usual configuration for one period would be RFQC1 + RFQC2.

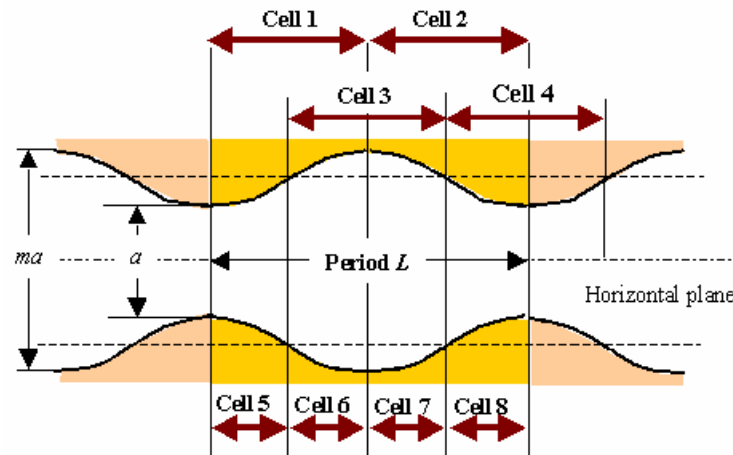


Figure 2.2 Definition of the 8 RFQ cells with respect to the RFQ period defined as shown by the vanes in the horizontal plane

TYPE = 'ROTPL'

Rotation planes were originally introduced for the design of medical gantries, but they can be used for any section of rotated line, the simulation of tilt errors or for creating tilted units. The later is made partially redundant by the fact that most units can be defined with the tilt included. There is a subtle difference however with respect to the vacuum chamber. A unit defined with a tilt does not have a tilted vacuum chamber, whereas the introduction of rotation planes also rotates the vacuum chamber.

Column	Attributes to be set in Edit Window
1	Index number set by program.
2	Optional unit name of up to 12 characters [<, >, !, %, ~, _, { }, ?, @, #, \$, & are reserved] and an optional alias name [use 'Alias/Name' button to toggle names].
3	ROTPL. It is sufficient to type 'ro' or 'pl' anywhere in the input string in upper or lower case characters.
33	Form of aperture: E = elliptical/round; R = rectangular/square; S = super-elliptical; D = diamond; Complex shapes can be built by combining basic forms. Default = R.
34	Left aperture wall w.r.t. axis and looking in beam direction[m]. Default = -0.1 m.
35	Right aperture wall w.r.t. axis and looking in beam direction[m]. Default = 0.1 m.
36	Lower aperture wall w.r.t. axis [m]. Default = -0.1 m.
37	Upper aperture wall w.r.t. axis [m]. Default = 0.1 m.

Notes: All other attributes are automatically set zero or a default value before leaving the Edit Window.
Index numbers in brackets indicate column number when the display is in longitudinal mode.

TYPE = 'SCATR'

An absorber or scatterer must have a finite length. The length is measured along the straight equilibrium orbit. The spreadsheet contains only the length, the beam aperture and the ramp variable (see below). The material constants and other auxiliary data are set in the Edit Window in the 'Auxiliary Data' menu in the routine 'Edit/Enter data for scatterers'. The auxiliary database can hold up to 25 individual sets of data, in addition to the default values for vacuum. Each data set is associated with a unit name and all elements with that name will use that data set.

Column	Attributes to be set in Edit Window
1	Index number set by program.
2	Optional unit name of up to 12 characters [<, >, !, %, ~, _, { }, ?, @, #, \$, & are reserved] and an optional alias name [use 'Alias/Name' button to toggle names].
3	SCATR It is sufficient to type 'sc' or 'ab' anywhere in the input string in upper or lower case characters.
4	Length along equilibrium orbit [m]. Length cannot be zero or negative.
27 (9)	<i>ramp</i> = 0. The lattice downstream is not adapted for the momentum change introduced by the cavity and the equilibrium orbit of the beam is treated as having a momentum deviation. <i>ramp</i> = 1. The lattice downstream is adjusted for the momentum change introduced by the cavity and the equilibrium orbit of the beam becomes the central orbit of the lattice.
33	Form of aperture: E = elliptical/round; R = rectangular/square; S = super-elliptical; D = diamond; Complex shapes can be built by combining basic forms. Default = R.
34	Inner aperture wall with respect to axis [m]. Default = -0.1 m.
35	Outer aperture wall with respect to axis [m]. Default = 0.1 m.
36	Lower aperture wall with respect to axis [m]. Default = -0.1 m.
37	Upper aperture wall with respect to axis [m]. Default = 0.1 m.

Notes: All other attributes that are entered will be automatically set to zero by the routine that checks the data base just before leaving the Edit Window.
Index numbers in brackets indicate column number when the display is in longitudinal mode.

A scatterer cannot be rotated about its longitudinal axis, but by adding rotation planes at the entry and exit its aperture/vacuum chamber can be rotated.

TYPE = 'MATRX'

The user-defined matrix is the most general of all elements. Only the beam path length and the aperture are contained in the data spreadsheet. All other parameters appear as auxiliary data in the Edit Window in the 'Auxiliary Data' menu in the routine 'Edit/Enter user-specified matrix'. The auxiliary database can hold up to 25 individual sets of data, in addition to the default values for a drift space of the length defined in the main database. Each data set is associated with a unit name and all elements with that name will use that data set.

Column	Attributes to be set in Edit Window
1	Index number set by program.
2	Optional unit name of up to 12 characters [<, >, !, %, ~, _, { }, ?, @, #, \$, & are reserved] and an optional alias name [use 'Alias/Name' button to toggle names].
3	MATRX It is sufficient to type 'mat' or 'ix' anywhere in the input string in upper or lower case characters.
4	Length along equilibrium orbit [m]. Length cannot be zero or negative.
27	<i>ramp</i> = 0. The lattice downstream is not adapted for the momentum change introduced by the cavity and the equilibrium orbit of the beam is treated as having a momentum deviation. <i>ramp</i> = 1. The lattice downstream is adjusted for the momentum change introduced by the cavity and the equilibrium orbit of the beam becomes the central orbit of the lattice.
32	Rotation of unit (BUT not the aperture limits) about longitudinal axis. Default = 0. To rotate the unit and the aperture limits use rotation planes
33	Form of aperture: E = elliptical/round; R = rectangular/square; S = super-elliptical; D = diamond; Complex shapes can be built by combining basic forms. Default = R.
34	Inner aperture wall with respect to axis [m]. Default = -0.1 m.
35	Outer aperture wall with respect to axis [m]. Default = 0.1 m.
36	Lower aperture wall with respect to axis [m]. Default = -0.1 m.
37	Upper aperture wall with respect to axis [m]. Default = 0.1 m.

Notes: Any other attribute that is entered will be automatically set to zero by the routine that checks the data base just before leaving the Edit Window.

(Outer –Inner aperture) and (Upper-Lower aperture) cannot be negative or zero.

Index numbers in brackets indicate column number when the display is in longitudinal mode.

2.1.4 Beam aperture

When the database is first initialised, all elements are given a default metallic rectangular beam aperture 20 cm square, which is the boundary on which particles will be lost and electrical images will be calculated. The most likely physical interpretation of this boundary is the vacuum chamber, but for electrostatic lenses and RF equipment the aperture is defined by the electrode structures which are also described by the gap parameter, see Section 2.15. The size and shape of this boundary can be edited and, *unlike other parameters, it can be set individually for all elements whether they are part of a series name or not.* The only restriction on the beam aperture is that it cannot be zero or negative.

In the present version, there is no provision for ceramic chambers apart from switching image forces off. However, the usual image problem in ceramic chambers is that of high-intensity space charge in rapid cycling synchrotrons. In this case, there is usually a metallic cage to shield the ceramic chamber from the beam. RF cages are supported by the program.

As shown in the element descriptions, columns 34 to 37 contain the distances of the walls from the central orbit and column 33 contains the chamber form, of which there are four basic shapes: *E* = elliptical/round; *R* = rectangular/square; *S* = super-elliptical; *D* = diamond. The super-elliptical vacuum chamber, ‘*S*’, is a flattened ellipse. Its cross-section equation is $(x/a)^3 + (z/b)^3 = 1$. In addition to the basic shapes, compound shapes can be specified, e.g. ‘*RREE*’ specifies a rectangular shape in quadrants 1 and 2 and an elliptical/circular shape in quadrants 3 and 4 (taking the quadrants clockwise when looking in the beam direction). Note that ‘*RRRR*’ specifies rectangular in all quadrants, which for simplicity can be entered as ‘*R*’.

2.1.5 *Half gaps or radii and inscribed circles*

The *parameter* is defined as *half of the pole gap in dipoles, half of the plate separation in electrostatic bends* and the *radius of the inscribed circle in lenses and RFQs*. The parameter is used for magnetic images, fringe field calculations, modelling wiggler fields, calculating electrostatic fields and calculating eddy currents.

When the database is first initialised, all elements are given a half-gap or inscribed circle radius of 0.1 m that corresponds to the default aperture. The user can edit this parameter to any positive value. This includes zero for magnetic elements, in which case the program ignores this parameter in calculations. For electrostatic elements however, the half gap must be non-zero.

2.1.6 *Dipole fringe-field integral FINT*

The dipole fringe-field correction requires the dipole half-gap and the ‘*FINT*’ function to be present in the lattice data base. ‘*FINT*’ describes the shape of the fringe field. If in doubt, use 0.15 corresponding to the linear fringe field decay model. The dipole fringe-field correction should be considered once the gap/bending radius > 0.01.

2.2 *Geometry*

2.2.1 *Local beam co-ordinates*

The beam is described with respect to a local, curvilinear co-ordinate system (*x*, *s*, *z*) that follows the equilibrium orbit (central orbit). It is a right-handed system. The relation between this co-ordinate system and the real world is taken account of in the next section by a global survey co-ordinate system (*X*, *Y*, *Z*), see Figure 2.3.

For most calculations in rings, the strict co-ordinate system and sign convention can be ignored and replaced by a convenience system:

Outwards and upwards are positive and the main dipoles are defined as having positive bending angles.

This dissociates the sign of the particle, the direction of motion and the polarity of the fields from the optics and it is left to the user to decide how to connect the power converters to the magnets in a particular case. This is possible because it is widely accepted that normalised parameters can be used.

For calculations in transfer lines, or cases where lines and rings have to be joined, a more strict sign convention has to be applied:

When looking in the beam direction (s):

Rightwards (x, horizontal) and upwards (z, vertical) are positive.

Slopes of trajectories and error kicks etc. are then defined as dx/ds and dz/ds .

However, the bending angles of dipoles are defined as positive when bending to the left or upwards.

Thus, the earlier convenience system of 'outwards and upwards are positive' corresponds to an anticlockwise turning ring and the bending of the dipoles (that determines the direction s of the curvilinear system) is defined as positive. Note that the dipole bending angles are treated differently to dipole kicks and trajectory slopes and that it is the bending angle of the dipoles that determines what happens. If, for example, the main dipoles of a ring are given negative bending angles, then the ring will turn clockwise, the x -co-ordinate will be directed towards the centre of the ring and the dispersion will appear with an unfamiliar sign in the output.

2.2.2 Position and direction in the survey co-ordinate system

The right-handed, local, curvilinear co-ordinate system of the beam (x, s, z) is positioned in the real world in a right-handed, survey co-ordinate system (X, Y, Z), in which (X, Y) defines the horizontal plane. The direction of the beam is defined by a horizontal angle, Θ_H , and an elevation angle, Φ_V , see Figure 2.3. The horizontal angle, Θ_H , is the angle between the projection of the beam trajectory onto the X - Y plane and the X -axis measured in an anticlockwise direction when looking down onto the X - Y plane. The horizontal angle varies from 0 to 2π [radian]. The elevation angle, Φ_V , is the angle between the X - Y plane and the beam axis [radian]. The elevation angle varies between $-\pi/2$ to 0 (below the horizontal plane) to $+\pi/2$ (above the horizontal plane).

The default values for starting a beam are $X = Y = Z = 0$ and $\Theta_H = \Phi_V = 0$, i.e. a beam that starts at the origin and moves along the X -axis, in the horizontal plane.

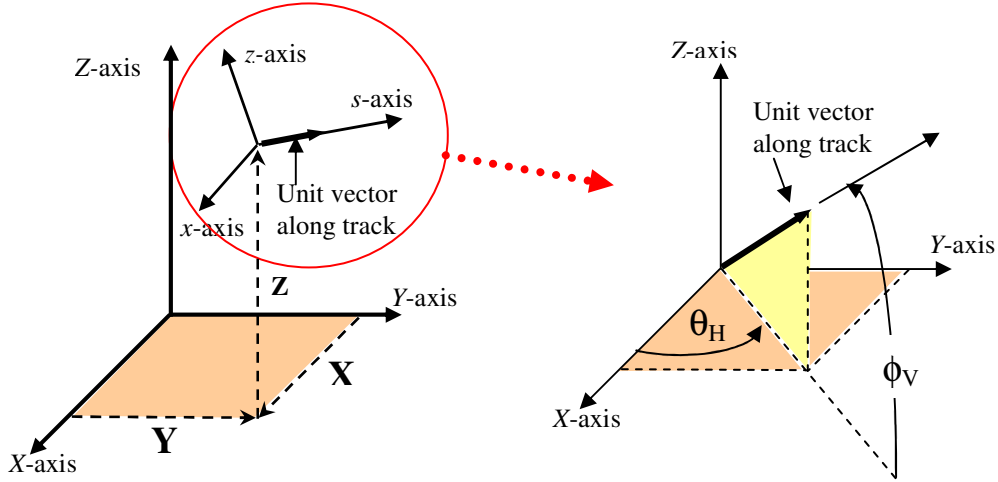


Figure 2.3 Definition of the position and direction of the local co-ordinate system in the global system

2.2.3 Normal-mode roll angle

The rotation angle of the local co-ordinate system about its y-axis corresponds to the angle of the normal modes of oscillation. The normal modes are usually horizontal and vertical, but can easily become tilted. If, for example, the beam is deflected upwards and while it has a positive slope it is deflected to one side the normal modes will become tilted. The curvilinear system follows the normal modes and rolls over accordingly. The roll angle is defined as the rotation about the beam axis measured in the clockwise direction when looking in the beam direction, see Figure 2.4. The roll angle must be in the range $-\pi$ to $+\pi$. The default value is zero, i.e. the normal modes are horizontal and vertical.

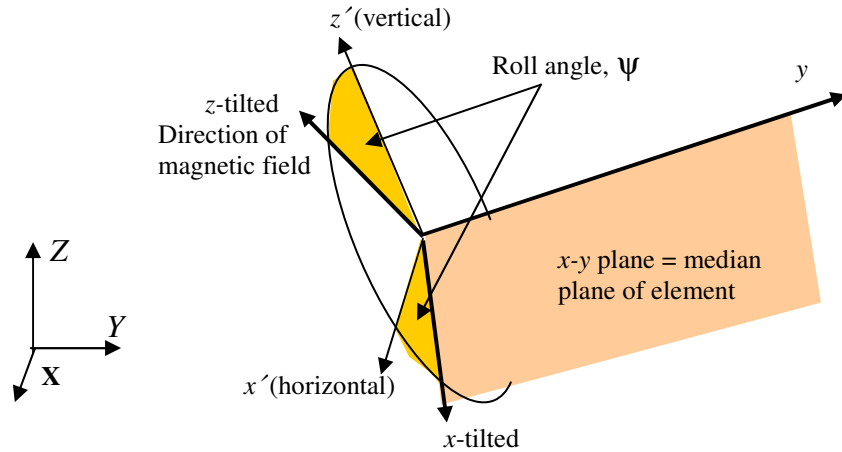


Figure 2.4 Roll angle of local co-ordinate system

2.2.4 Other conventions

Quadrupoles are focusing (F-type) when a parallel beam is made to converge horizontally when looking in the beam direction. The normalised gradient, k , for an F-type quadrupole is negative.

Quadrupoles are defocusing (D-type) when a parallel beam is made to diverge horizontally. The normalised gradient, k , for a D-type quadrupole is positive.

Sextupoles are F-type if they aid the focusing of an F-type quadrupole for positive x . The normalised gradient, kk , for an F-type sextupole is negative.

Sextupoles are D-type if they aid the defocusing of a D-type quadrupole for positive x . The normalised gradient, kk , for a D-type sextupole is positive.

Similarly, higher order normal non-linear lenses are F-type if they aid the focusing of a quadrupole for positive x and the normalised gradients of normal F-type non-linear lenses are negative.

A skew quadrupole with a negative normalised gradient is an F-type quadrupole that has been rotated by 45 degree in the anticlockwise direction.

Similarly, the normalised gradients of all higher order skew lenses have the sign of the normal lens that would have to be rotated by $\pi/2n$ anticlockwise (where $2n$ is the number of poles) to produce the skew lens.

A solenoid has a normalised field $B_z/(Br)$ that is taken as positive when the particles in the axial field turn in an anticlockwise direction.

Another problematic subject is the sign convention for the edge angles of dipoles. The inclination of the end faces (edge angles) of a general dipole are measured with respect to the perpendicular end faces of a sector magnet. In WinAGILE, the sign convention for the edge angles is based on whether the edge is focusing or defocusing in the plane of bending.

An edge angle is defined as negative when it has a focusing action in the plane of bending and is positive when it has a defocusing action.

An edge is focusing (i.e. negative edge angle), if the field integral on the outside of the bend is greater than for the sector magnet and defocusing (i.e. positive edge angle), if it is less than for a sector magnet. Thus, a rectangular magnet has a smaller field integral on the outside of the bend than a sector magnet and it therefore has defocusing edges (i.e. positive edge angles). This criterion has the virtue that it is independent of the sign of the bending angle of the dipole and whether the dipole is bending horizontally or vertically.

Note that a rectangular magnet has defocusing edges, but the bulk central field effect has an exactly equal focusing action, so that overall focusing in a rectangular dipole is zero in the plane of bending and is focusing in the perpendicular plane. This is in contrast to a sector dipole that is focusing in the plane of bending and has zero focusing in the perpendicular plane.

2.2.5 General comments on sign conventions

Unfortunately, the sign conventions described above are not universally used. Although the differences are usually small, they can still cause confusion. Figure 2.5 shows the main characteristics of the WinAGILE, MAD, Trace3D and TRANSPORT sign conventions.

All of the above has deep historical roots. For example, the use of a parameter called the **field index** in the days of **weak-focusing** machines accounts for the negative sign in the definition of k for a focusing quadrupole. Similarly, normalised parameters (i.e. parameters normalised by $B\rho$) have always been used to remove the momentum dependence and to avoid an absolute sign convention, which is why the above sections tacitly ignore the sign of the ion. It is perhaps not a surprise that as lattices become more complicated and varied, the philosophy of **normalised parameters and phenomenological descriptions** occasionally runs into trouble. Some examples are:

- *Consider a synchrotron built in the vertical plane. There is no conceptual problem with the change to horizontal field dipoles, but the required sextupole for this ring would be a skew sextupole in the eyes of the designer of a horizontal ring, because sextupoles behave differently in the two planes. In fact, all lenses of odd order would change from normal to skew when changing from a horizontal ring to a vertical ring. Note that the horizontal field dipole formally belongs in the series of skew lenses, although it is never referred to as such. Of course, this can be solved by considering the vertical plane as being horizontal.*
- *Now consider a horizontal ring with two vertical by-passes. Furthermore, imagine that for space reasons some of the sextupoles controlling the chromaticity must be inside the by-passes where the vertical dispersion is non-zero. Now, the two types of normal and skew sextupoles must appear in the same ring, but with the same phenomenological description.*
- *In colliders and complicated accelerator complexes, there are often elements that are common to two or more beams and frequently the beams are travelling in opposite directions. In these situations, descriptions such as focusing and defocusing can easily become contradictory.*

In the long term, it would be arguably safer and easier to abandon the **normalised parameters and phenomenological descriptions** for an absolute sign system based on the raw fields, but this is not normally done probably because it would require a greater investment of effort when creating the lattice. WinAGILE is typical inasmuch as it works internally with normalised parameters and phenomenological descriptions, but for outputs it can switch between normalised parameters and raw fields. Consequently, it is necessary for the user to define the ion and its charge, so that whenever raw fields are requested the signs can be calculated. Furthermore, one of the above problems is avoided in WinAGILE by only allowing sextupole components in RBEND and SBEND dipoles when the bend is horizontal.

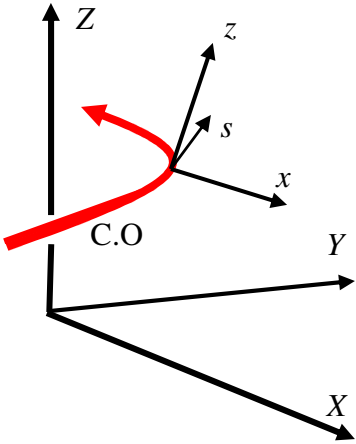
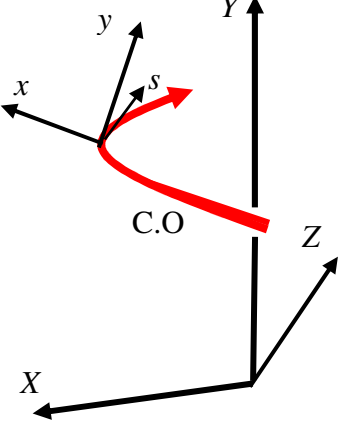
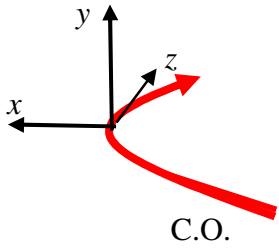
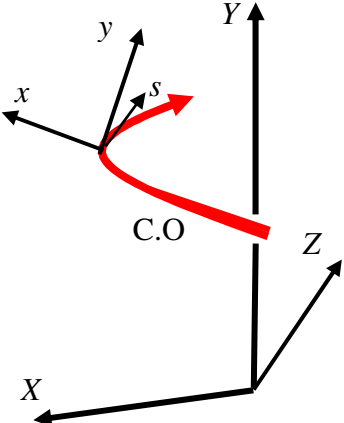
	
Dipoles have positive bending angles to the left when looking in the beam direction.	Dipoles have positive bending angles to the right when looking in the beam direction.
Dipoles have positive bending angles upwards when looking in the beam direction.	Dipoles have positive bending angles upwards when looking in the beam direction.
F-quads have negative k -values.	F-quads have positive k -values.
An anti-clockwise ring has positive bending dipoles and the local beam coordinate 'x' is outwards.	A clockwise ring has positive bending dipoles and the local beam coordinate 'x' is outwards.
(a) WinAGILE	(b) MAD
 <p style="text-align: center;">(No separate survey system)</p>	
Dipoles have positive bending angles to the right when looking in the beam direction	Dipoles have positive bending angles to the right when looking in the beam direction
Dipoles have positive bending angles upwards when looking in the beam direction.	A left bend is obtained by rotating 180° and upwards by rotating -90° and downwards by $+90^\circ$.
F-quads use field index values rather than k -values.	Positive gradients (k) imply F-quads.
A clockwise ring has positive bending dipoles and the local beam coordinate 'x' is outwards.	A clockwise ring has positive bending dipoles and the local beam coordinate 'x' is outwards.
(c) Trace3D	(d) Transport

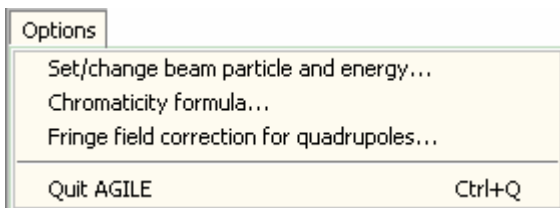
Figure 2.5 Comparison of some sign conventions

* * *

Chapter 3 Main Window

The Main Window hosts basic tasks and routines of general interest and acts as a gateway to the more specialised parts of the program. The menu items are described below. When running the program, it may be that some menu items are greyed indicating that they are not applicable at that particular time.

3.1 Options Menu



The Options Menu contains functions that set the computational environment.

3.1.1 Set/change beam particle and energy...

This menu item initially shows the dialogue box on the left in Figure 3.1. Clicking on the top button “Change beam ion?” activates a second dialogue box from which the user can choose from about 1100 pre-stored ions and exotic particles by scrolling to select the element and then setting the level of stripping and number of neutrons. Limits are set internally on the isotopic range so that only stable or quasi-stable ions can be selected. In addition, there is the possibility of entering data for two user-defined particles. If nothing is done, the default particle is a 1 GeV proton.

Clicking on the button “Change KE/momentum?” activates the dialogue box shown on the right in Figure 3.2. Note that the data to be entered is an average value per nucleon. The user can enter either the kinetic energy or the momentum and the dialogue box will convert between the two values.

For many lattice calculations the beam ion and its energy are not need, but the program may still display the dialogue box on the left in Figures 3.1 and 3.2. This is because the program calculates items such as the longitudinal transfer matrices that need this information. If these features are in fact unnecessary, then accept the defaults to clear the dialogue box. The status bar at the top centre of each principal window shows the currently selected ion and its energy.

The relativistic particle constants (masses etc.) stored in the program form a set of data taken from "Tables of Physical and Chemical Constants" by G.W.C. Kaye and T.H. Labye, 14th edition, (Longman, London and New York) (1973). Other references may have slightly different values, as definitions of basic quantities change. However, beware of changing an isotope mass, for example, to a newer and more 'modern' value using the 'user function', since this would require other basic constants such as the electron charge to be changed inside the program to maintain consistency.

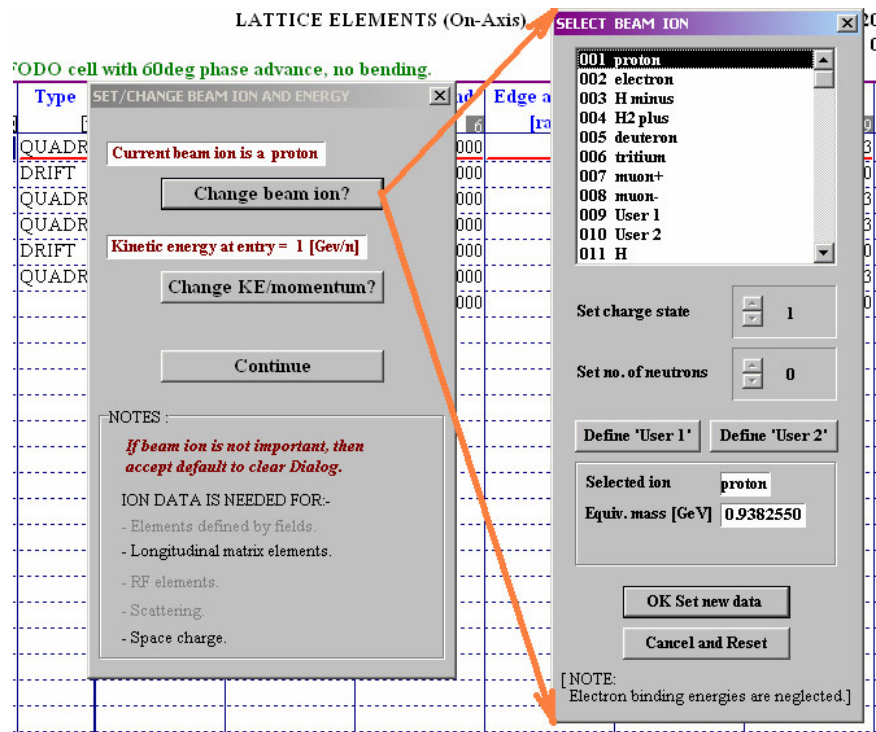


Figure 3.1 Selecting the beam ion

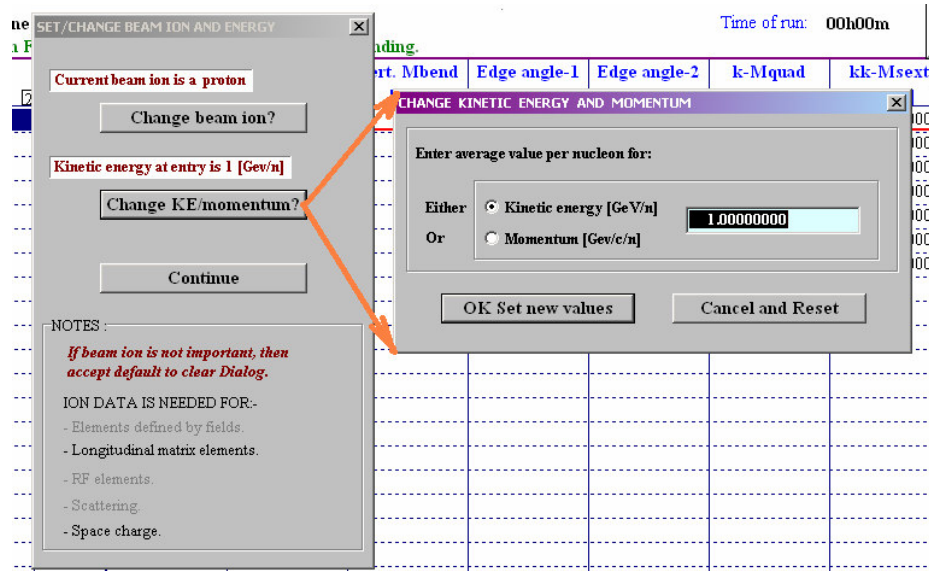


Figure 3.2 Setting the beam's kinetic energy or momentum

3.1.2 Chromaticity formula...

The local chromaticities of a lattice [defined as $dQ/(dp/p)$ in the limit as dp goes to zero] can be calculated using the lattice functions on the equilibrium orbit (this may be the central orbit or an off-axis closed-orbit, or a distorted closed-orbit) and are reported in the Summary Chart after the Twiss Function Display. Chromaticity calculations of this type are notoriously fickle and the results are not always satisfactory. WinAGILE offers

two formulæ. The default is the Hardt, Jäger, Möhl formula and other is a simplified thin-lens formula. The dialogue box (see Figure 3.3) offers the choice between these two formulæ. The status bar at the top centre of each principal window shows which formula is active. If the two formulæ in the program agree, then the lattice is well behaved and the result is reliable. If they differ (which may occur in small rings with edge angles on the dipoles and a large, rapidly varying dispersion function), then this is an indication that more care is needed. In principle, the Hardt, Jäger, Möhl formula is the more accurate, but it is still prudent to investigate the chromatic behaviour by creating off-axis orbits and plotting the tune variation directly.

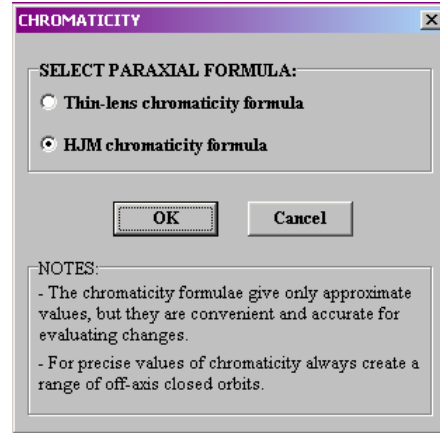


Figure 3.3 Choice of chromaticity formula

3.1.3 *Quadrupole fringe field correction...*

The quadrupole fringe-field dialogue box (see Figure 3.4) offers the option of adding a fringe-field correction to quadrupoles. This correction only affects off-axis orbits. The program introduces the correction by adding thin multipole lenses at the entry and exit to each quadrupole. These lenses are given names of the form #####EN%%%%%%%% and #####EX%%%%%%%% where ##### is the index number of the element being corrected. The correction is relatively weak and need only be considered when the momentum deviation is large (>0.01) and when the dispersion function is large and changing rapidly along the lattice. The status bar at the top of each window shows whether the quadrupole fringe-field correction is active, or not.

The program also applies the more usual dipole fringe-field correction. This correction is active wherever non-zero values have been entered for the half-gap in the dipole and *FINT* that describes the shape of the fringe field. If in doubt over the value of *FINT* use 0.15, which corresponds to a linear fringe-field decay. The dipole fringe-field correction affects the focusing in the plane perpendicular to the bending for orbits of all momenta and should be considered once the gap/bending radius >0.01 .

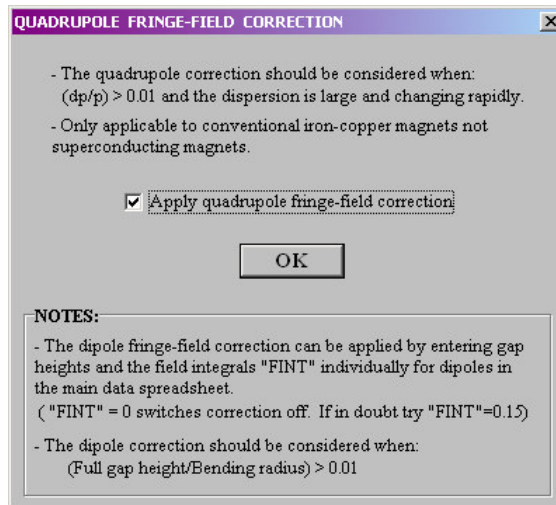
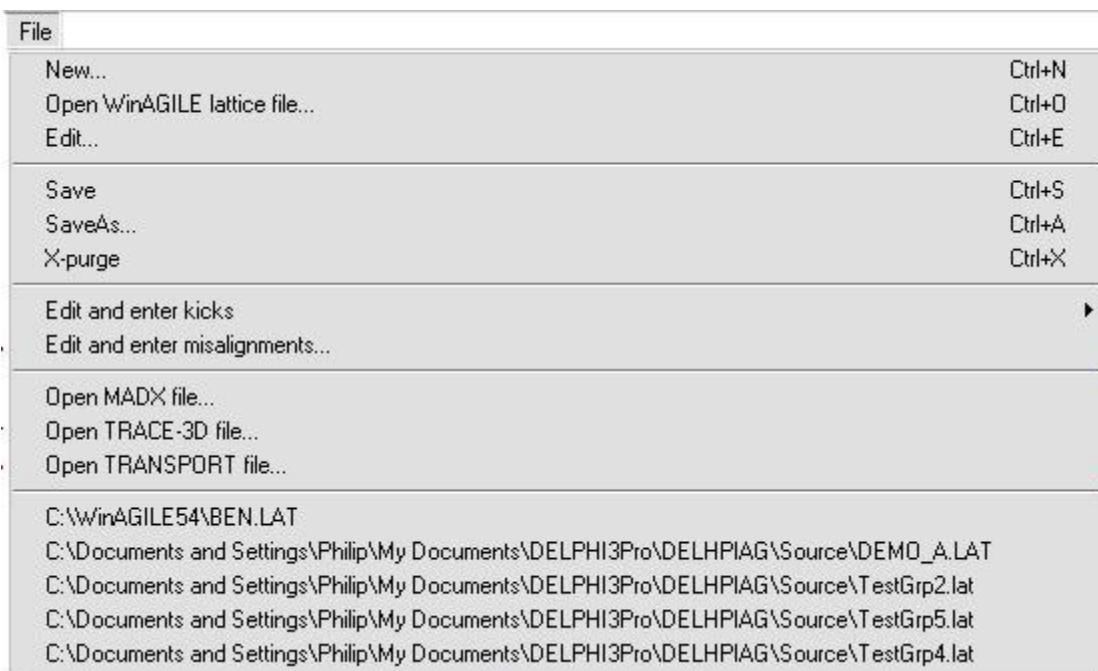


Figure 3.4 Fringe-field correction for quadrupoles

3.1.4 Quit AGILE (Ctrl+Q)

Before the program terminates, temporary files are deleted and the user is offered the possibility of saving the current lattice file and notebook, if this has not been done recently.

3.2 File menu



The File Menu contains the usual file operations as well as providing access to the Kick and Misalignment Windows

3.2.1 New... (Ctrl+N)

This routine launches the creation of a new lattice. An optional title of up to 69 characters is requested before opening the Edit Window for data entry. The optional title, which is referred to as the *user title*, is included in the headers of all data displays and printouts along with the name of the file, the date and the time of the calculation. The user-title can be edited at any time by double clicking within the area of the title (see Figure 3.5).

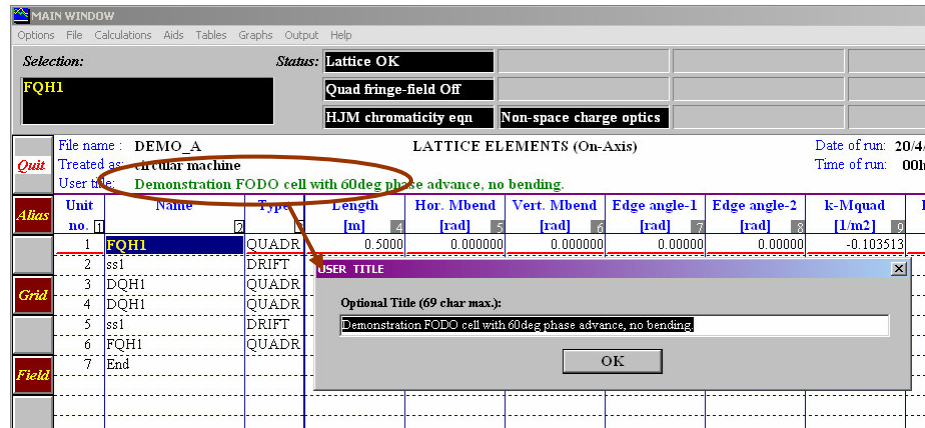


Figure 3.5 Double-click the 'user title' to edit

3.2.2 Open WinAGILE lattice file... (Ctrl+O)

This routine opens a dialogue box requesting the name and path for a lattice file and then tries to open that file. If a problem is encountered a warning is given. If the file has been made by a more recent version of WinAGILE than the current program, a warning is also given and the action can be cancelled or continued. Once the file is opened and the lattice loaded the status bar at the top of the window shows one of the following messages: 'Lattice data OK', 'Lattice in compressed form', 'Lattice data unchecked', or 'No lattice in memory'. If all is well, the listing of the lattice elements will appear on the screen.

If the lattice data is in compressed form, or if it is unchecked, it is necessary to go to the Edit Window and to select 'Check_Data | Check with decompression (if needed)' from the menu bar before the lattice can be used for calculations.

3.2.3 Edit... (Ctrl+E)

This routine calls the Edit Window in order to edit the lattice currently in memory. If there is no lattice in memory, this is equivalent to the action of starting the creation of a new lattice file. The Edit Window is a full screen editor with many editing commands and shortcuts (see Chapter 4). To be viable, a lattice must contain at least one element.

3.2.4 Save (Ctrl+S)

This routine saves the current lattice automatically into the current lattice file. If there is no current lattice file (e.g. when creating a new lattice) the ‘SaveAs’ function is called. The standard extension for lattice files is ‘LAT’.

3.2.5 SaveAs... (Ctrl+A)

This routine opens a dialogue box that requests the file name and path for saving the current lattice. The combo-box labelled ‘Save as type’ offers various possibilities including saving the lattice in the format of the earlier program version 3.5, see Figure 3.6. Element types that are not recognised by the earlier version are replaced by drift spaces. It may also be that certain auxiliary information such as beam characteristics will not be stored in the earlier version. In addition, there are the possibilities of saving the lattice according to the current version either as created, or with only raw fields or in normalised form. The standard extension for a lattice file is ‘LAT’.

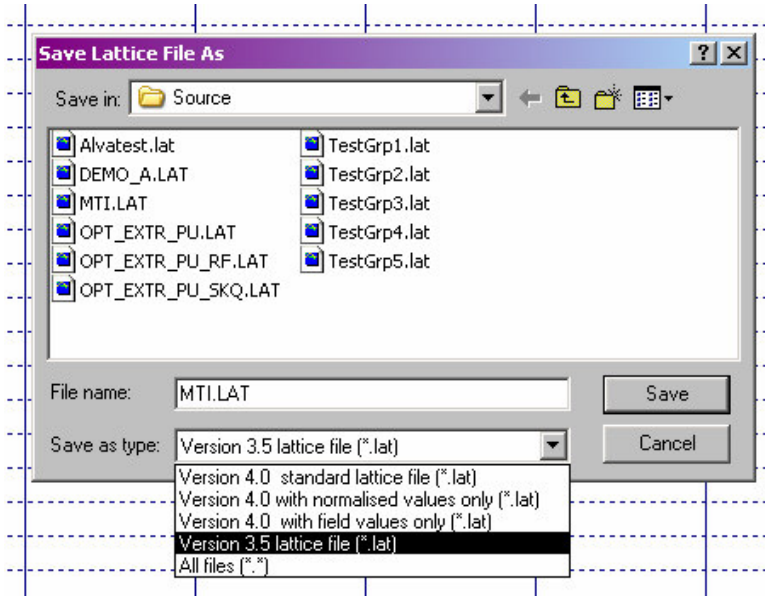


Figure 3.6 ‘SaveAs’ dialogue box showing options for file saving

3.2.6 X-purge (Ctrl+X)

This function erases the data base currently in memory and sets default values. First, the user is prompted to save the lattice and its notebook, if this was not done recently.

3.2.7 Edit and enter kicks

This command activates a sub-menu with four items that all call the Kick Window for the creation or editing of kick files, see Chapter 9. The first item concerns point dipole kicks [rad], the second thin-lens quadrupole kicks [T], the third thin-lens skew quadrupole kicks [T] and the fourth point momentum kicks [dp/p]. Before a kick file can be created a lattice needs to be in memory. Many of the commands in the Kick Window use the lattice structure, for example for the addition of errors to dipoles with a specific name. A kick file can be stored on disk and recalled later. When a kick file is

reloaded a check is made that the number of lattice elements corresponds to the current lattice in memory and that the type of kick is that expected. The kick file does not contain lattice element names, so apart from these simple checks, it is up to the user to be sure that the kick file is the correct one. A kick file normally has the extension 'KCK'.

3.2.8 Edit and enter misalignments...

This command calls the Misalignment Window for the creation or editing of element misalignment files, see Section Chapter 9. Misalignment files are similar in their handling to kick files. They normally have the extension 'MIS'.

3.2.9 Open MAD-X file...

The first dialogue box, see Figure 3.7, explains the differences between the sign conventions of MAD-X and WinAGILE.

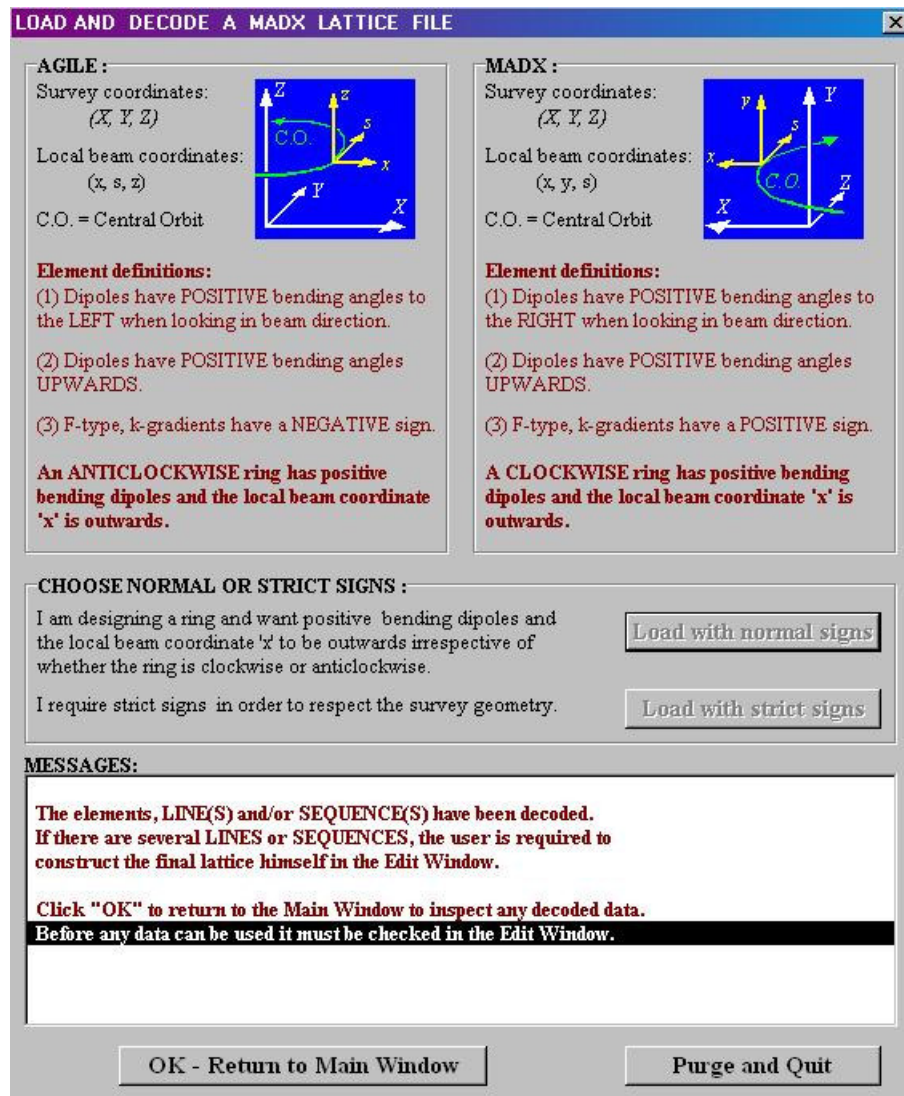


Figure 3.7 Dialogue for loading a MAD-X file into WinAGILE

Before calling a second dialogue for finding and opening the MAD-X file, the user is requested to choose between two options for the file translation.

- **Strict signs** which will give the same survey geometry, but will give opposite signs for the local beam co-ordinates.
- **Normal signs** i.e. the local beam co-ordinate system is positive outwards for a ring and upwards. In strict terms, this means an anticlockwise turning ring in WinAGILE and a clockwise turning ring in MADX. However, for many applications the absolute sense of rotation is not important.

Since MAD-X files can be highly structured, no guarantee is given that a satisfactory translation will be made. If problems are encountered warnings are given in a list box that the user can scroll through. There are also fundamental incompatibilities that will be flagged, such as finite length OCTUPOLES and EDGE elements. WinAGILE is coded to read mathematical expressions for variables, elements that rely on previously defined variables, lines and sequences, but the insertion of sequences or lines in other sequences or lines is not included. If the translation is incomplete, the user will be presented with the individual lines and sequences and will be required to construct the final lattice from those elements. If the lines or sequences are absent, then only the lattice elements are presented. MAD-X users sometimes divide their data into several files. For the purposes of WinAGILE, it is necessary to put the variables, elements and lines/sequences all in one continuous file.

Once the file is loaded, the status bar at the top of the window will show the message: '*Lattice data unchecked*' (or '*No lattice in memory*') and if all is well, the listing of the lattice elements will appear on the screen. However, the data cannot be used until it has been checked in the Edit Window under the menu item 'Check_Data | Check with decompression (if needed)'.

It may be that the extension 'mad' has been reserved for Microsoft Access and machine protection programs will consider '*.mad' files as malicious software. For this reason, the dialogue boxes initially offer no extension or the extension 'txt', but the 'mad' extension is included in the options.

3.2.10 Open TRACE-3D file...

The opening and loading of a Trace-3D file is handled through a dialogue box similar to that used for MAD-X file, see Figure 3.7. Equally, no guarantee is given that a satisfactory translation will be possible. The most important fundamental difference between the two programs concerns RFQ cells. Trace-3D works with the high-level variables AV and V/r^2 , which hides the vane modulation, whereas WinAGILE requires the explicit values for the vane modulation, the minimum aperture radius (half gap) and RF voltage. Unfortunately, only the user can resolve this difference for his particular application. Once the data is loaded, it cannot be used until it has been checked in the Edit Window under the menu item 'Check_Data | Check with decompression (if needed)'.

3.2.11 Open TRANSPORT file...

Not available in the version.

3.2.12 Recently-used files

Directly under the File menu, the program displays the five last-used lattice files. A single click on the file name will reload the lattice.

3.3 Calculations menu

Calculations	
Ring or matched section	Ctrl+R
Transfer Line...	Ctrl+L
Matrices	
Geometry..	Ctrl+G
Fit geometry to an end point...	
Fit geometry to close a ring...	
Optimise synchronous phase at centre of cavity...	

The Calculations Menu provides access to the more specialised Ring and Line Windows. It also hosts some calculations that do not depend specifically on the classification of the lattice as a ring or transfer line.

Almost all lattice calculations will pass via this menu to either the Ring or Line Windows. **Rings** and **Matched Sections** are characterised by having their input and output lattice functions equal, which unambiguously determines the lattice functions. **Transfer lines** require the user to define the beam shape at some point along the line, since any one of an infinite number of beams can transit through the line and the program cannot choose the appropriate set of lattice functions for itself. A matched section differs from a ring by not being geometrically closed. An example of a matched section would be a FODO cell in a ring or periodic transfer line. In practice, the program will do whatever is asked. It will calculate one turn of a ring as if it were a line and will equally calculate a line as a matched section by forcing the periodic condition of input equals output. This choice is up to the user.

3.3.1 Ring or matched section (Ctrl+R)

This command initiates the calculation of a lattice as a ring or matched section before calling the Ring Window for more specialised calculations. If the data is not suitable (e.g. unchecked or in compressed form), a warning is given with brief directions on how to check or decompress the lattice. If the lattice was last used as a transfer line, the program asks for confirmation of the change. If the routine finds the ion has not been chosen, the dialogue box on the left in Figure 3.1 is activated with the title 'Confirm beam ion'. If the lattice has no elements defined by fields, then the dialogue can be cleared by accepting the defaults.

The routine then enters a complex sequence of actions that are described in Section 1.2 in Steps 2 to 4. Providing no unexpected problems are found the routine will display either a dialogue box of the form shown in Figure 3.8 for uncoupled lattices or Figure 3.9 for coupled lattices.

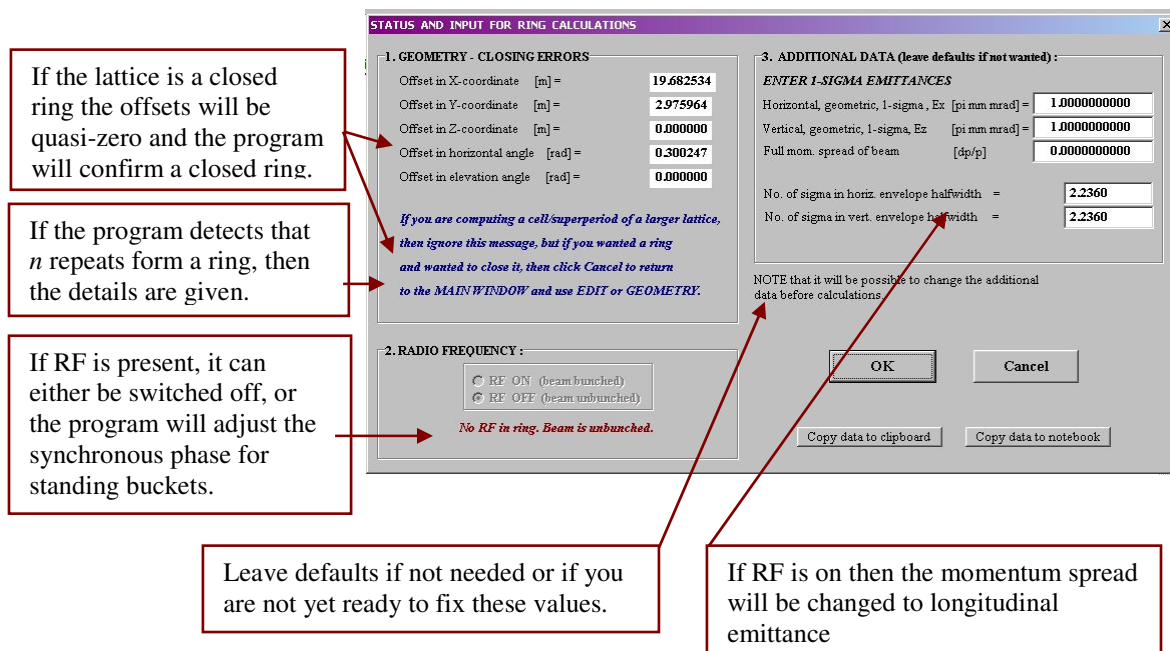


Figure 3.8 First example of the dialogue showing the status for ring/matched section calculations

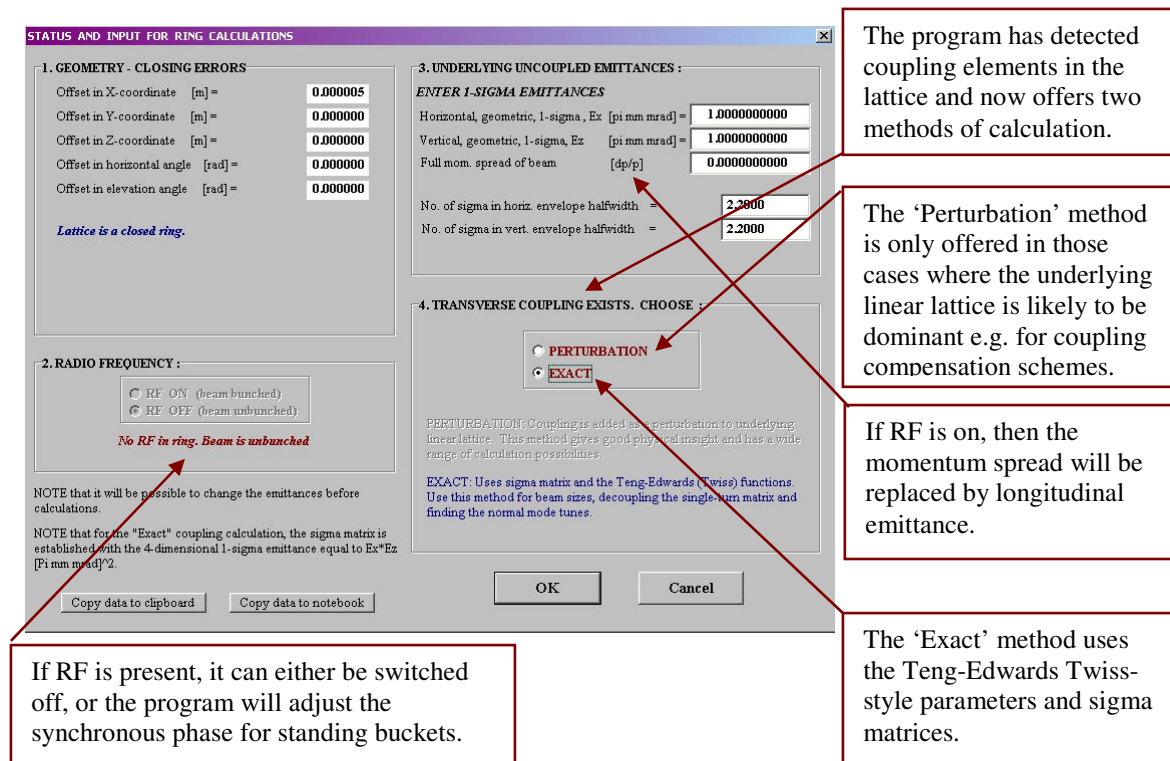


Figure 3.9 Second example of the dialogue showing the status for ring/matched section calculations

For uncoupled lattices (Figure 3.8), the program will continue by calculating the Twiss functions and the sigma matrices. This calculation may fail if the ring is unstable,

in which case the traces of the single-turn matrices will be shown, see Figure 3.10. If the user can do nothing to correct the situation, he can still continue with matrices only.

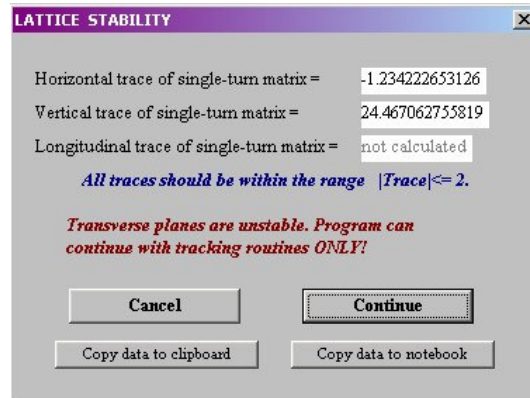


Figure 3.10 Example of an unstable lattice

For coupled lattices (Figure 3.9), the situation is more complicated. If the coupling is due to skew quadrupoles and/or solenoids, the program offers two ways of continuing:

Method 1. Perturbation. When calculating the lattice, the program replaces the skew quadrupoles and solenoids by drift spaces and calculates the Twiss functions in the usual way. The coupling can then be evaluated as a perturbation to the uncoupled machine. The driving terms of the coupling resonances and the excitation of vertical dispersion and median plane tilt can all be calculated. There are also routines for the control or compensation of coupling. Method 1 gives the best physical insight into what is happening and offers the widest range of possibilities.

If the lattice contains other sources of coupling, such as rotated dipoles or rotation planes that cause whole sections of the lattice to rotate, then the program deems that the replacement of coupling elements by drift spaces will not lead to a useful uncoupled lattice and Method 2 is applied directly.

Method 2. Exact In this mode, the program attempts to treat the lattice with two exact formalisms: the Teng-Edwards (Twiss-styled) parameters and SIGMA matrices. There is a risk that the lattice will be sensitive or unstable and one or both formalisms will not work. In these cases, warning messages will be given and, in the worst case, the user will only be able to continue using the raw transfer matrices.

The Teng-Edwards formulation produces an output similar to the normal Twiss output. However, this is a mathematical equivalence and not a physical one. The Twiss-type parameters are expressed in a new co-ordinate system based on the normal modes that appear as 4-D ellipsoids that change shape and orientation as they turn round the ring. However, the tunes $Q1$ and $Q2$ of the normal modes are the values most likely to be recorded by monitors if the ellipsoids are not tilted too much. It is also possible in Method 2 to decouple the single-turn matrix seen at a given point and possibly for a limited region around that point with the help of 4 coupling elements. Note that this does not decouple the machine. In fact, the coupling may be stronger than before. An analogy

is that of the closed orbit. The orbit can be perfect over a limited region after correction and extremely distorted elsewhere.

The SIGMA formalism includes the coupled dispersion functions. The advantage of this formalism is that it gives exact beam sizes. One reason for difficulties may be the poor closure of the SIGMA matrices. This may occur when slots are used in the end-plates of solenoids. Solenoids can have ends of the types 0, +1 or -1 (ends of type -999 should only be used between the slices of an internally divided solenoid). The slot types +1 and -1 are less stable and probably, therefore, not strictly symplectic.

Note that independent of whether Method 1 or Method 2 has been chosen, the matrix display will still show the correct 6×6 matrices for the coupling elements and that the tracking of single particles and distributions and the plotting of maps always use the correct 6×6 matrices. If the lattice is so exotic as to be intractable by the standard methods, the user is still left with the choice to continue with matrices only.

3.3.2 *Transfer line...(Ctrl+L)*

This command initiates the calculation of the lattice as a *transfer line* before calling the Line Window for more specialised calculations. If the data is not suitable (e.g. unchecked or in compressed form), a warning is given with brief directions on how to check or decompress the lattice. If the lattice was last used as a ring or matched section, the program asks for confirmation of the change and then presents, as defaults, the Twiss functions at the entry to line from the last ring calculation. If the routine cannot find confirmation that the ion has been chosen, the dialogue box that appears on the left in Figure 3.1 is activated with the title ‘Confirm beam ion’. If the lattice has no elements defined by fields, then the dialogue can be cleared by accepting the defaults. The routine then enters a complex sequence of actions that are described in Section 1.2 in Steps 2 to 4. Providing no unexpected problems are found the routine will display a dialogue box of the form shown in Figure 3.11. This dialogue summarises the status and the input requirements for making further calculations in the Line Window. In order to have arrived at the status shown in Figure 3.11, the geometry and matrices will have already been calculated.

For line calculations, it is necessary to supply the Twiss functions at some point along the line, since the Twiss functions are not uniquely determined by periodicity as for a ring or a matched section. The specified point is usually the entry to the line, but it can be anywhere. In anticipation of the end of the line being the next most-used reference point, the index number of the last element is displayed.

If the routine detects powered skew quadrupoles or solenoids, it will issue a message that the line will be analysed using the sigma matrix formalism, since the Twiss formalism is no longer valid in this case. However, the beam is assumed to be uncoupled at the input to the line and the normal Twiss functions are requested at this point. When coupling is present, the emittances become necessary input rather than optional input as in the uncoupled case, because the ratio of the emittances affects the coupling. The first output presented by the program is the listing of the sigma matrix elements and the statistical emittances along the whole lattice. The $\langle x^2 \rangle$ and $\langle z^2 \rangle$ terms give the sigmas (i.e. effectively the beam widths) of the beam distribution. This display replaces the usual Twiss display.

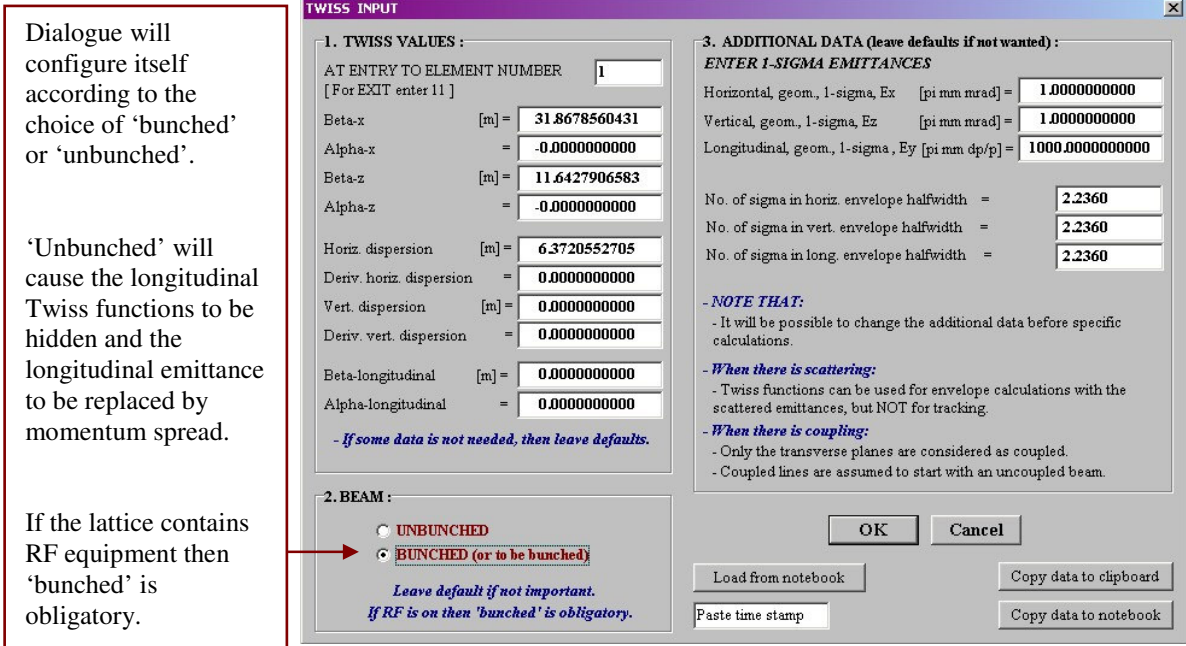


Figure 3.11 Example of the dialogue showing the status for line calculations

3.3.3 Matrices (Ctrl+M)

This routine calculates the first-order, linear, 6×6 transfer matrices for each element in the lattice. If the data is not suitable for some reason (e.g. unchecked or in compressed form), a warning is given with brief directions on how to check or decompress the lattice. If the routine cannot find confirmation that the beam ion has been chosen, the dialogue box that appears on the left in Figure 3.1 is activated with the title 'Confirm beam ion'. If the lattice has no elements defined by fields, there are no scatterers and the user does not wish to calculate space charge, then the dialogue can probably be safely cleared by accepting the defaults.

When calculating the matrices, finite-length sextupoles are replaced by drift spaces and point non-linear lenses by identity matrices. When tracking, non-linear lenses are represented by point kicks. In the Main Window, the matrices are always calculated in their basic form without space charge or any other special feature. At the end of the lattice, there is one summary line that contains the 6×6 matrix for a single turn in a ring or a single transit in a line.

The matrix elements are displayed in the spreadsheet, but they can also be viewed in 'block' form by double-clicking anywhere along the line belonging to an element, see Figure 3.12. The 'block' form can be copied to the clipboard or to the note book. This feature is also available in the spreadsheets for Twiss functions, tracking and beam envelopes.

6 x 6 TRANSFER MATRIX

hh1 0.93760441	hh2 2.56331952	hv11 0.00000000	hv12 0.00000000	hs11 0.00000000	hs12 0.67821810	x
hh21 -0.04716461	hh22 0.93760441	hv21 0.00000000	hv22 0.00000000	hs21 0.00000000	hs22 0.51266273	
vh11 0.00000000	vh12 0.00000000	vv11 0.92688634	vv12 2.55387962	vs11 0.00000000	vs12 0.00000000	z
vh21 0.00000000	vh22 0.00000000	vv21 -0.05516380	vv22 0.92688634	vs21 0.00000000	vs22 0.00000000	
sh11 -0.51266273	sh12 -0.67821810	sv11 -0.00000000	sv12 -0.00000000	ss11 1.00000000	ss12 0.49459348	ds
sh21 0.00000000	sh22 0.00000000	sv21 0.00000000	sv22 0.00000000	ss21 0.00000000	ss22 1.00000000	

Matrix for element no. **6**

Element name **BD**

Alias name **Dipole Drwg 010-234A**

OK

Copy data to clipboard

Copy data to notebook

Figure 3.12 Dialogue box showing the ‘block’ form of a 6×6 transfer matrix for a dipole

3.3.4 Geometry... (Ctrl+G)

This routine calculates the geometry of the lattice. Once the standard checks described at the start of Section 3.3.3 have been made, the user is required to situate the lattice in the global co-ordinate system by specifying the position and direction coordinates at the entry to the lattice using the geometry dialogue shown in Figure 3.13. There are no restrictions on the position or direction of the entry to a lattice.

When the lattice file is saved, its geometric entry coordinates are also stored and will appear as defaults in the dialogue box the next time the routine is used. It is possible to copy the geometric coordinates to the Windows clipboard and the lattice notebook. The same values can also be reloaded into the dialogue box on a later occasion by finding the data in the lattice notebook, copying the time stamp, pasting it into the dialogue box and clicking ‘Load from notebook’.

CALCULATION OF SURVEY GEOMETRY

ENTER/EDIT:
 Define entry to lattice in global survey coordinate system [X,Y,Z].
 This defines start of local beam coordinate system [x,s,z].

X co-ordinate [m] = 0.00000000
 Y co-ordinate [m] = 0.00000000
 Z co-ordinate [m] = 0.00000000

Horizontal angle of central orbit [rad] = 0.00000000
 Roll angle about central orbit [rad] = 0.00000000
 Elevation angle of central orbit [rad] = 0.00000000

Compute **Cancel**

NOTES:
 - [XYZ] is the global survey co-ordinate system.
 - [xsZ] is the local beam co-ordinate system aligned with the median plane of the lattice elements and the central orbit.
 - The HORIZONTAL ANGLE is angle between the X-axis and the projection of the central orbit on the X-Y plane.
 - The ROLL ANGLE is the angle between the x-axis and the intersection between the x-z and X-Y planes.
 - The ELEVATION ANGLE is the angle between the central orbit and its projection on the X-Y plane.

Load from notebook Copy to clipboard
 Paste time stamp Copy to notebook

Figure 3.13 Dialogue box for calculation of lattice geometry

3.3.5 *Fit geometry to an end point...*

Adjusting the geometry to a particular end point is a typical problem for a transfer line, where the general layout is known from the surrounding equipment and buildings and the parameters must be adjusted to conform to these boundary conditions. The fitting procedure can be carried out with a full lattice, or a "skeleton" lattice of drift spaces and dipoles.

Once the standard checks described at the start of Section 3.3.3 have been made, the program opens the dialogue box shown in Figure 3.14 that requests the entry co-ordinates and angles of the transfer line in Section 1 and the desired exit values in Section 2. This routine cannot control the roll angle of the median plane and, for this reason, it is not included in the desired exit values. The routine is basically designed to work with dipoles and drift spaces that are aligned vertically or horizontally. Small roll angles and slopes can be tolerated in the elements used for the optimisation, but the fitting efficiency is reduced. When this occurs, warning messages are given and the user can decide whether to continue or abandon.

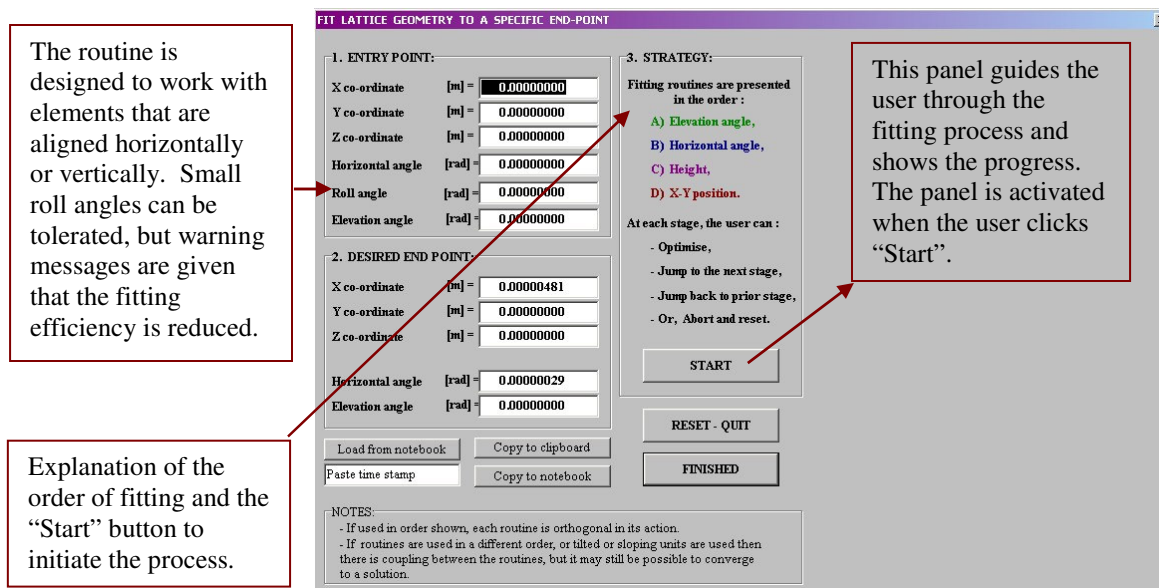


Figure 3.14 Dialogue box for fitting lattice to an end point (View 1)

Section 3 of the dialogue box shows the order in which the fitting routines are presented once the “Start” button is clicked:

- (a) elevation angle, ϕ_v ,
- (b) horizontal angle, Θ_h ,
- (c) height Z , and finally,
- (d) X-Y position.

The fitting routines appear in the panel on the far right-hand side in Figure 3.14. This panel configures itself according to the routine that is currently requested. The general layout is always the same, but the details of the number and type of elements to be used changes. Figure 3.15 shows the situation once the user has reached the stage of optimising the X-Y position at the end of the line. The buttons at the top of the panel, “NEXT routine” and “PRIOR routine”, allow the user to move backwards and forwards to iterate a solution or jump steps. Below these buttons, there is always a list box in which all the named elements that can be used for fitting are shown. The act of selecting an element in this list box automatically enters its index number in the edit box directly below with a default step size. If the user prefers he can enter the index and step size directly without passing via the list box. In fact, this is essential when using elements that have no name, since unnamed elements do not appear in the list box. Once the required number of variables has been chosen, click the button ‘(Re-)compute’ and a fit will be made. The fitting routines will signal problems as they arise. If for example, the vertical dipole that has been chosen to adjust the elevation angle is tilted, it will also affect the horizontal angle and the user will be warned of this. He can either accept the error, or choose another dipole. Similar problems arise with drift spaces that slope. It may also be that a negative length is required for the X-Y fitting or that the drift spaces are parallel and therefore degenerate. It is also possible with the X-Y fitting routine to make a single parameter “best” fit by entering the name of one drift space twice. The result of the fit is shown at the bottom of the panel.

The fitting can be aborted and the original lattice reset at any time. Note that if an element used for fitting is repeated many times in the lattice be sure that the step size is sufficiently small since many elements changing by the same step can cause a lattice to wind up like a spring and the fitting routine will not be able to optimise correctly.

FIT LATTICE GEOMETRY TO A SPECIFIC END-POINT

1. ENTRY POINT:

X co-ordinate [m] = -9.19621800
Y co-ordinate [m] = 3.55380700
Z co-ordinate [m] = 0.00000000
Horizontal angle [rad] = 4.66866250
Roll angle [rad] = 0.00000000
Elevation angle [rad] = 0.00000000

2. DESIRED END POINT:

X co-ordinate [m] = 36
Y co-ordinate [m] = -15.86857663
Z co-ordinate [m] = -0.00010513
Horizontal angle [rad] = 6.28300790
Elevation angle [rad] = 1.57079600

3. STRATEGY:

Fitting routines are presented in the order :
A) Elevation angle,
B) Horizontal angle,
C) Height,
D) X-Y position.

At each stage, the user can :
- Optimise,
- Jump to the next stage,
- Jump back to prior stage,
- Or, Abort and reset.

4. CHOOSE 2 DRIFT SPACES TO ADJUST X-Y POSITION

NEXT routine **PRIOR routine**

EITHER, click on unit in list box
OR, enter element index directly below

To force a single-parameter fit, enter same unit twice.

Enter index of element to be changed = 108
Enter index of 2nd element = 108
Enter step size for fitting = 0.010000
Enter step size for 2nd element = 0.010000

(Re)-COMPUTE

	X-position	Y-position
Desired value =	36.00000000	-15.86857663
Actual value =	36.00000000	-15.86857663
Residual error =	-0.00000000	0.00000000

NOTES:
- If used in order shown, each routine is orthogonal in its action.
- If routines are used in a different order, or tilted or sloping units are used then there is coupling between the routines, but it may still be possible to converge to a solution.

Load from notebook **Copy to clipboard** **Paste time stamp** **Copy to notebook** **RESET - QUIT** **FINISHED**

‘NEXT’ and ‘PRIOR’ buttons allow user to treat routines in a different order and to iterate to a solution.

Select elements. Only named elements appear in List Box. To use an unnamed element enter its index below.

Selected variables with their indices.

Figure 3.15 Dialogue box for fitting lattice to an end point (View 2)

3.3.6 Fit geometry by closing a ring...

The ring closing procedure can be carried out with a full lattice, or a "skeleton" lattice of drift spaces and dipoles. Once the standard checks described at the start of Section 3.3.3 have been made, the program opens the dialogue box shown in Figure 3.16, which is reminiscent of Figure 3.15 for fitting to an end point, except that input and output coordinates are no longer needed.

Although most rings will require only the horizontal angle and possibly the X-Y position to be corrected, the fitting routine also offers elevation angle and height for those rings with a vertical by-pass. The fitting routines are presented in a certain order once the "Start" button is clicked:

- (a) elevation angle, ϕ_v ,
- (b) horizontal angle, Θ_h ,
- (c) height Z, and finally,
- (d) X-Y position.

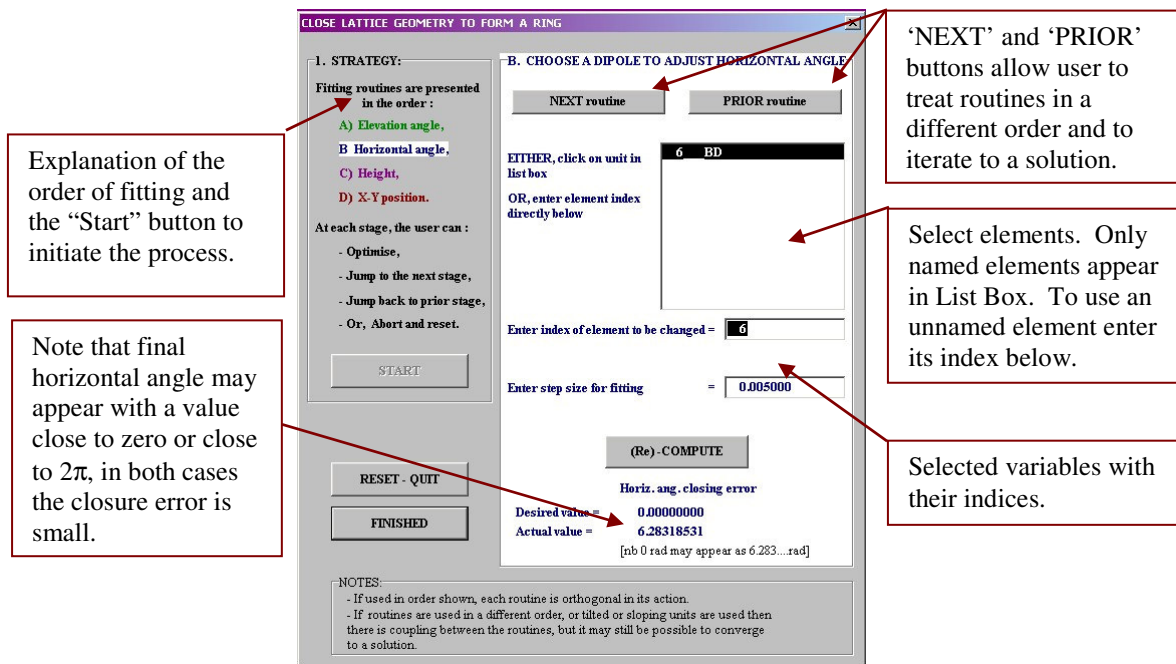


Figure 3.16 Dialogue box for closing a ring

The panel on the right in Figure 3.16 configures itself according to the routine that is currently requested. The general layout is always the same, but the details of the number and type of elements to be used changes. Figure 3.16 shows the situation once the user has reached the stage of optimising the horizontal angle. The buttons at the top of the panel, "NEXT routine" and "PRIOR routine", allow the user to move backwards and forwards to iterate a solution or to jump a step. Below these buttons, there is a list box in which all the named elements that can be used for fitting are shown. The act of selecting an element in this list box automatically enters its index number in the edit box directly below with a default step size. If the user prefers he can enter the index and step size directly without passing via the list box. In fact, this is essential when using elements that have no name, since unnamed elements do not appear in the list box. Once the required number of variables has been chosen, click the button '(Re)-compute' and a fit will be made. The fitting routines will signal problems as they arise.

The fitting can be aborted and the original lattice reset at any time. When closing a ring the variables used are likely to be parameters that are repeated many times in the lattice, for example the main dipole bending angle. In this case, be sure that the step size is sufficiently small since many elements changing by the same step can cause a lattice to wind up like a spring and the fitting routine will not be able to optimise correctly. Also, in the case of the horizontal angle, the lattice must start with a closing error of less than π radians or the routine is likely to fail. A warning message is given.

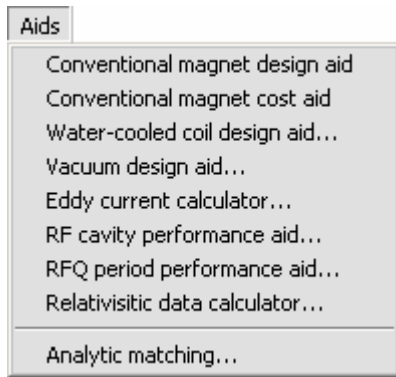
3.3.7 Optimise synchronous phase at centre of cavity...

In some cases, the synchronous phase is known at the centre of the cavity and not at the entrance where the WinAGILE input data requires it. To overcome this problem, set some approximate values for the synchronous phase at the cavity entrance while creating the lattice and then use this command and dialogue box, see Figure 3.17,

to calculate and load the exact phase needed at the cavity entrance to obtain the required value at the centre.

Figure 3.17 Dialogue box for setting synchronous phase at cavity centre

3.4 Aids menu



The Aids Menu contains a number of design aids and calculators useful for accelerator design

3.4.1 Conventional magnet design aid

Providing the memory contains a valid lattice file with values for the magnet half gaps and the radii of the inscribed circles for the lenses, the menu item ‘Conventional magnet design aid’ will be available. The display may be preceded by a request for (or confirmation of) the beam particle type and energy. As shown in Figure 3.18, the routine displays for each dipole, quadrupole, skew quadrupole and sextupole the strength of the main field component, the pole-tip field, the steel length of the yoke and the number of ampere-turns per pole, assuming a conventional copper-iron magnet. The steel length and number of ampere turns are estimated taking into account saturation over the range of 0-2 T for a high-carbon steel of a type typically used for magnet manufacture. However, the results should be treated very much as estimates, since the details of the design and

the quality of the steel in any particular case are unknown to the program. When the pole-tip field is greater than 2 T, as is the case for the super-ferric dipole in Figure 3.18, there is a comment drawing attention to this fact.

MAIN WINDOW

OptionsFileCalculationsAidsTablesGraphsOutputHelp

Selection:

Status:

Lattice OK with ion

Beam ion: proton

Attn. Out of range, B(pole)>2T

Hard-edge model

5.000000 [GeV/u](entry)

HJM chromaticity eqn

Non-space charge optics

Coupled optics

File name : Test

Treated as:

User title: Example

ESTIMATED CONVENTIONAL MAGNET PARAMETERS

Alias	Unit no.	Name	Type	B-pole tip [T]	Yoke length [m]	NI/pole [A]	Comment	End
Grid	1	D1	DRIFT	0.0000	0.00000	0.0000		"
	2	QF	QUADR	-1.5582	0.43200	62000.4118		"
	3	D2	DRIFT	0.0000	0.00000	0.0000		"
	4	QD	QUADR	1.5282	0.48482	61451.9855		"
NB	5	D4	DRIFT	0.0000	0.00000	0.0000		"
	6	SX1	SEXTU	0.4890	0.25000	12970.5152		"
	7	D3	DRIFT	0.0000	0.00000	0.0000		"
	8	BD	SBEND	3.9118	2.86653	4036020.9006	Validity?, B>2	"
	9	D4	DRIFT	0.0000	0.00000	0.0000		"

Figure 3.18 Spreadsheet display for the estimated conventional magnet parameters

3.4.2 Conventional magnet cost aid

This menu item activates the graphical aid shown in Figure 3.19 for estimating conventional magnet costs. It is necessary to know the total weight and type of the magnets to be ordered. Simple dipoles with flat coils are on the lower edge of the price band, complicated dipoles with bedstead coils and quadrupoles are in the middle of the price band and finally sextupoles are at the top of the band. This guide is based on medium and large orders and will not perform well for very small series. An alternative formula for dipoles is also given, again for large orders (i.e. >500 ton). In this case, it is necessary to know the iron and copper weights individually.

Both formulæ were derived in the 1990s and may require some indexation.

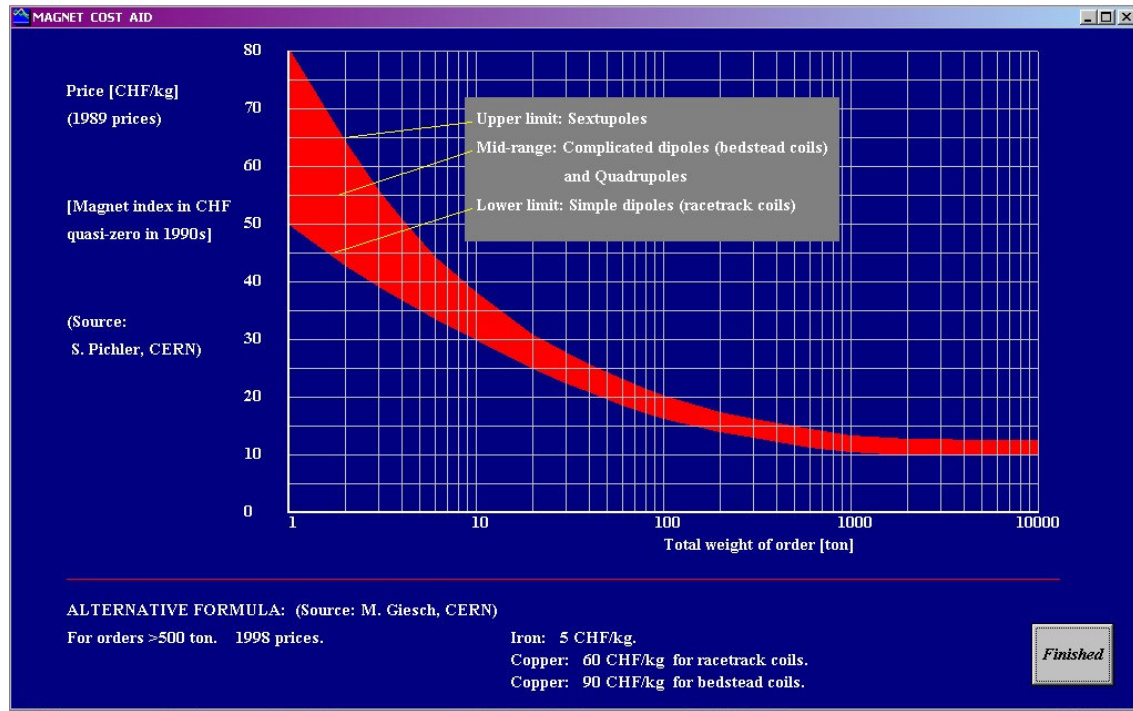


Figure 3.19 Conventional magnet cost aid

3.4.3 Water-cooled coil design aid...

This standalone routine for the calculation of water-cooled coils operates as a specialised calculator, see Figure 3.20. By entering the requested data for the available water pressure, water inlet temperature, conductor material, conductor dimensions and the electrical current in the left hand box and clicking 'Compute', the routine returns with the water volume flow, type of flow (turbulent or laminar), water speed, temperature rise, power dissipation, coil resistance when hot, coil weight and the rate of temperature rise that would exist if the cooling water were to be cut off. The routine considers a single water circuit. It is up to the user to arrange individual circuits in series or parallel groups to get the total power dissipation and water flow.

Among the data calculated is the rate of temperature rise that would occur if the cooling water were to be cut off. This is critical for coils that operate at high current densities ($>20 \text{ A/mm}^2$), since this indicates the time the interlock system would have in order to avoid permanent damage to the coil.

For lattice magnets, typical values are turbulent flow at 3-5 m/s, 4-8 bar pressure drop, 3-5 A/mm^2 and 10-30 degree temperature rise. Higher water velocities can lead to a lot of noise and erosion of the water circuit. Note that round conductors can be treated by setting the width equal to the height and the radius of the edge rounding to half the width. It is preferable to avoid brazes within the coil length. The maximum length of conductor is limited by the billet size for drawing which is about 120 kg for copper. Current-wall septa have more extreme values with $>50 \text{ A/mm}^2$. In these cases, the greatest danger is a cut in the cooling-water supply.

WATER-COOLED COIL DESIGN AID...

INTRODUCTION:

- Parameters are for a SINGLE water circuit.
- It is left to the user to combine water circuits in series/parallel to get the overall temperature rise, power consumption, etc.
- For circular conductors, set conductor height = width and the rounding radius = 0.5*width.

INPUT:

Conductor height [mm] = 8.9

Conductor width [mm] = 8.9

Diameter of water hole [mm] = 4.1

Radius of edge rounding [mm] = 0.5

Conductor length in a single water circuit [m] = 45

Available water pressure for a single circuit [bar] = 5

Input water temp. [deg C] = 19.2

Current [A] = 250

Conductor material: ☐ Copper ☒ Aluminium

COMPUTE

OUTPUT FOR SINGLE WATER CIRCUIT:

Water speed [m/s] = 1.59935

Water flow [l/min] = 1.26693

Water flow (turbulent/laminar) TURBULENT

Water temp. rise [deg C] = 13.42589

Power dissipation [watt] = 1185.006

Resistance (hot) [ohm] = 0.018960

Current density [A/mm²] = 3.79980

Conductor mass [kg] = 7.98791

RESPONSE TIME:

If the cooling water is cut off, then the temperature will rise at the rate of;

0.164 [deg C/s]

Load from notebook Paste time stamp QUIT Copy data to clipboard Copy data to notebook

NOTES:

- Typical values for lattice magnets: turbulent water flow at 3-5 m/s; current density 3-5 A/mm²; pressure drop 4-8 bar; temperature rise 10-30 deg C.
- For septa, parameters are more extreme: try not to exceed a current density of 50 A/mm²; try not to exceed a water velocity of 10 m/s; try to use a square conductor to facilitate bends in both planes.
- For high current densities, check the rate of temperature rise if the water stops and be sure the temperature interlock is fast enough to cut the current.
- The maximum length of conductor without a braise is limited by the maximum billet size for drawing (120 kg approx. for Cu).

Figure 3.20 Specialised calculator for water-cooled coils

3.4.4 Vacuum design aid...

This standalone routine for the calculation of vacuum systems of transfer lines and accelerators operates as a specialised calculator, see Figure 3.21.

By entering in the left hand box, the requested data for the vacuum pump spacing, the pumping speed per pump, the major and minor radii of the vacuum pipe (equal when circular), the specific surface area (i.e. the area exposed to vacuum in cm² over a 1 m length of the vacuum system), the specific out-gassing rate of the vacuum pipe material (typical values are given in the dialogue box), the absolute temperature and the molecular weight of the residual gas and clicking 'Compute', the routine returns with the maximum pressure mid-way between the pumps, the average pressure along the system and the minimum pressure at the pump ports, the conductance of a pipe equal in length to the distance between pumps and finally the specific conduction of the pipe. The specific surface area should include the surface area of the vacuum pipe plus additional contributions for bellows, screens, pickup plates and shields. The routine is based on an infinite array of uniformly-spaced pumps on a uniform vacuum pipe. This is rather simplistic, but it gives a first indication of what is required.

VACUUM SYSTEM DESIGN AID

INTRODUCTION:
Equilibrium pressure calculation for an infinite array of uniformly spaced vacuum pumps connected to a vacuum pipe of circular or elliptical cross-section

INPUT:

Pump separation	[m] =	5
Pump speed	[l/s] =	100
Major half-width of pipe	[m] =	0.07
Minor half-width of pipe	[m] =	0.037
Specific surface area 'seen' by vacuum	[cm ² /m] =	3654
Specific out-gassing rate (enter #.#E##)	[Torr l/s/cm ²] =	5.0E-12
Absolute temperature	[K] =	293
Molecular weight of residual gas	=	2

COMPUTE

QUIT

OUTPUT:

Maximum pressure	[Torr] =	1.04E-0009
Average pressure	[Torr] =	1.00E-0009
Minimum pressure	[Torr] =	9.13E-0010
Conductance of pipe (over separation lgth)	[l/s] =	88.39
Specific conductance of pipe	[l m/s] =	441.97

Copy data to clipboard

Copy data to notebook

Paste time stamp

Load from notebook

NOTES:

- The specific surface area 'seen' by vacuum should be increased to account for shields, screens, bellows etc.
- Some typical outgassing rates are:

Stainless steel baked for 24 hour at 300 deg. C	= 1 E-12
Stainless steel, unbaked after 100 hour pump-down	= 2 E-9
Copper after 10 hour pump-down	= 5 E-9
Brass after 10 hour pump-down	= 2.5 E-8
Teflon and aluminium after 10 hour pump-down	= 1 E-7
Araldites B, D and F after 10 hour pump-down	= 8 E-7, 5 E-7, and 4 E-7
Nylon after 10 hour pump-down	= 4.5 E-6
- Molecular weights: 2(H₂), 28(N₂), 32(O₂).

Figure 3.21 Specialised calculator for vacuum systems with regularly distributed pumping

3.4.5 Eddy current calculator...

This routine allows the user to investigate the effects of eddy currents in the yokes and vacuum chambers of conventional dipoles and quadrupoles and also to store the auxiliary data needed to include eddy current field effects in lattice calculations (up to 25 data sets). The calculator can only be applied to magnets that are in that lattice. The routine commences with the dialogue shown in Figure 3.22. The two list boxes on the far left show all the dipoles and quadrupoles that are potentially available for the calculation. Double clicking an element moves the selected item into the corresponding right hand list box where it is attributed default data for eddy current calculations. If units have been attributed data on an earlier occasion, they will already be in one of the right hand list boxes. Selecting an element in one of the right hand list boxes activates the buttons down the right hand side. Double clicking an element selected in a right hand list box is equivalent to clicking the top button.

- **'Calculate eddy currents'.** Opens a new dialogue box to refine data and calculate eddy currents (see Figures 3.22a and 3.22b).
- **'Show field errors'.** Opens a graph Window that shows the field errors while stepping through a user-specified ramp.

- ‘Delete selection’. Removes the selected item from the list box and deletes its eddy current data.

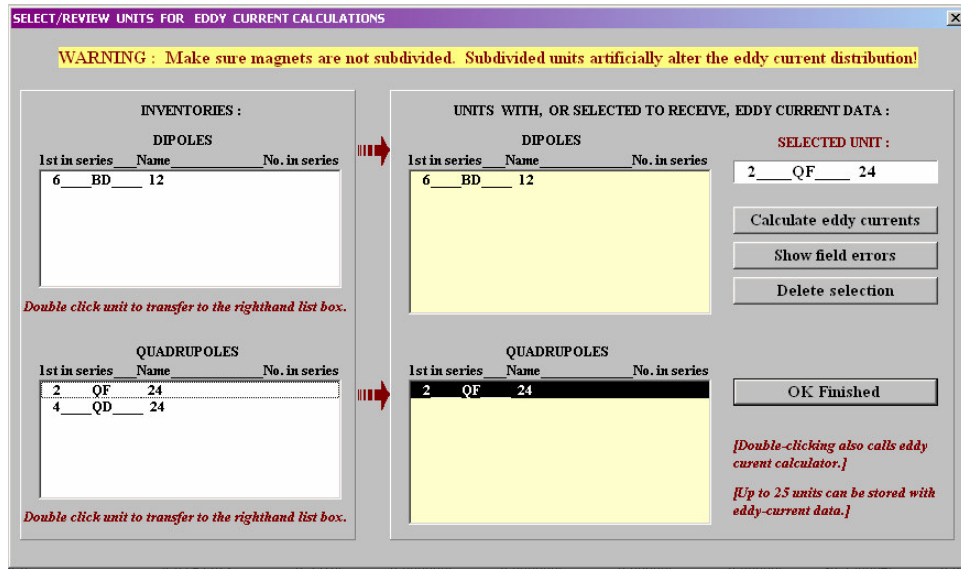
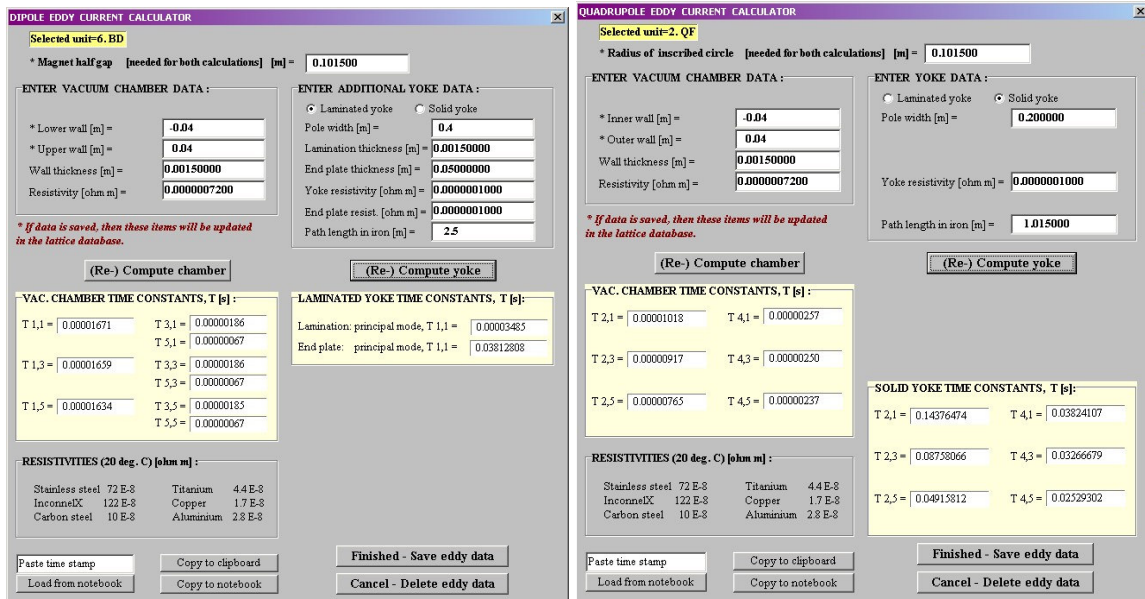


Figure 3.22 Inventory dialogue for choosing units for eddy current calculations

Clicking ‘Calculate eddy currents’ opens the dialogue box shown in Figure 3.23a for a dipole or the dialogue box shown in Figure 3.23b for a quadrupole. Both dialogues offer the possibility of having a laminated (see example in Figure 3.23a) or solid yoke (see example in Figure 3.23b). The calculation can be cycled with different values as many times as wanted. Some resistivity values for typical materials are included in the dialogues.



(a) Example of a laminated dipole

(b) Example of a non-laminated quadrupole

Figure 3.23 Dialogue boxes for data collection and calculation of eddy currents

Clicking the button ‘Show field errors’ launches the graph window shown in Figure 3.24. In this example, the right hand graph shows the field ramp and the cluster of points (yellow) at the start of the ramp where calculations have been made. The left hand graph shows the corresponding field errors across the magnet pole due to the eddy currents. A similar picture would be obtained by focusing on the end of the ramp. The routine starts with default data for a linear current ramp to be sampled at 24 uniformly spaced steps. The default data is usually too coarse to show more than a constant maximum field error, but the boundary conditions can be modified, see below:

- **Scale.** Changes the scales of both graphs and the thicknesses of graph lines, axes and grid lines.
- **Options.** Sets the field levels and duration of the ramp, the sampling period, the starting time of the first calculation. It is possible to switch off either the yoke or the vacuum chamber time constants.
- **Step+1.** Triggers the routine to make one step in time and to calculate the integrated field error on the median plane.
- **Step+24.** Triggers the routine to execute and display 24 steps (25 curves).
- The window is also equipped with the usual buttons for printing, writing graphic files, modifying screen colours, opening the on-line help and returning to the prior window.

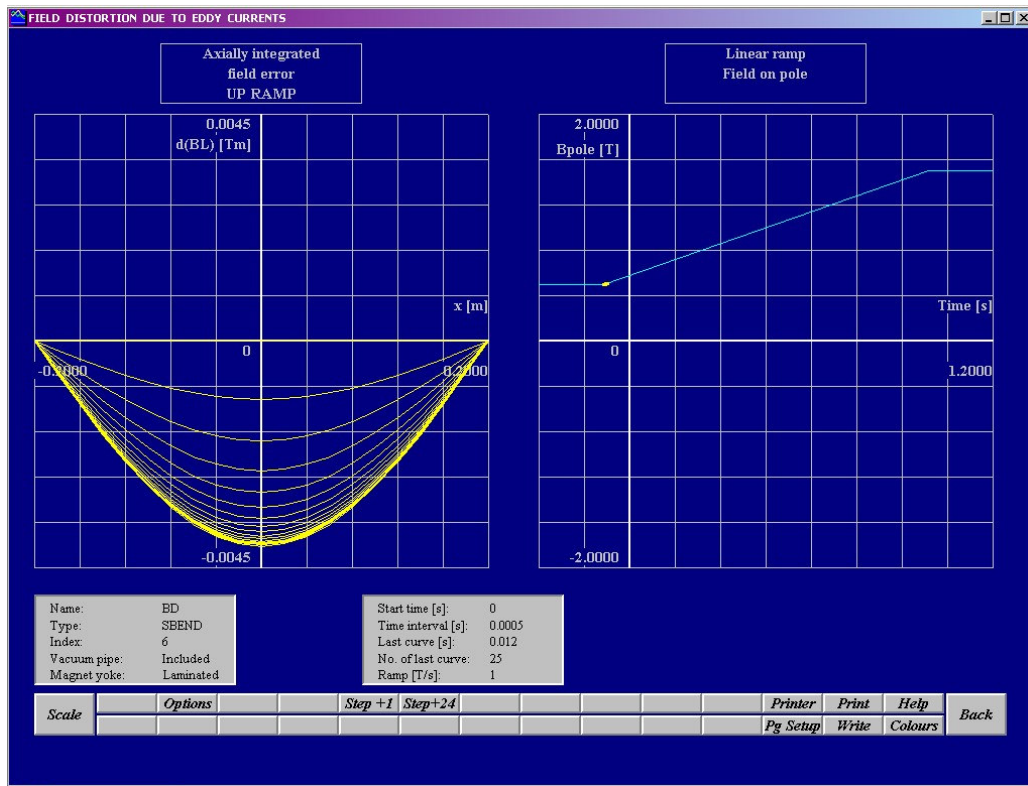


Figure 3.24 Field errors due to at the start of a linear ramp

3.4.6 RF cavity performance aid...

This routine enables the user to investigate the performance of an RF cavity. The routine is only available once a lattice has been loaded and can only be applied to cavities that are in that lattice. Changing the cavity length, the type of ion and the

harmonic distribution of the axial field can only be done in the main program, but the energy of the incoming particle, the cavity voltage, frequency and synchronous phase at entry and the number of integration steps can all be varied without affecting the main lattice database. . The routine displays the dialogue box shown in Figure 3.25.

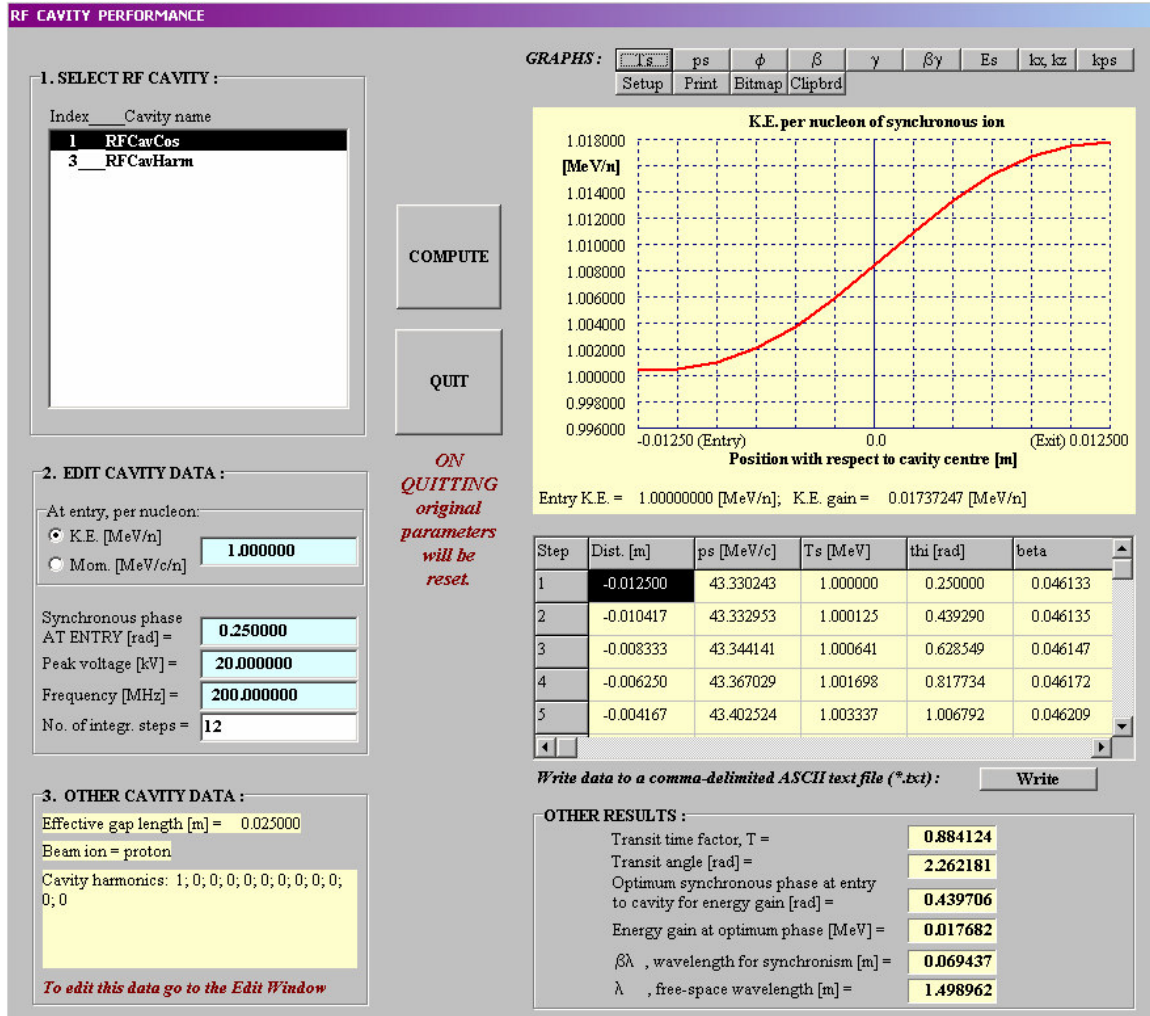


Figure 3.25 Calculator for investigating RF cavity performance

On the top left hand side of Figure 3.25, a list box shows all the cavities in the lattice from which the user should select the cavity to be investigated. The parameters that can be freely set are directly below the list box. The length of the cavity, the type of ion and the harmonic distribution can only be edited in the main program. By clicking 'Compute' the routine will read the data and calculate the beam characteristics when traversing the cavity. The data base table on the right hand side at the bottom will then fill up with data, which can be shown graphically using the buttons above the graph space. The example in Figure 3.25 shows the kinetic energy gain of the particle as it moves through the cavity. Below the data table there are some additional parameters such as the *transit time factor*, the maximum energy that can be absorbed from the cavity, the synchronous phase needed at the input to achieve this maximum energy gain,

etc. The 'Write' button under data base table can be used to create a comma-delimited ASCII text file of the table's contents. The graphs can be printed or stored as bitmaps. Clicking 'Quit' clears the dialogue box and re-sets the original data in the data base.

3.4.7 RFQ period performance aid...

This routine enables the user to investigate the performance of an RFQ cell. The routine is only available once a lattice has been loaded and can only be applied to RFQs that are in that lattice. Changing the type of ion can only be done in the main program, but the energy of the incoming particle, the period length, the vane voltage, frequency and synchronous phase at entry, vane modulation and the number of integration steps can all be varied without affecting the main lattice database. The routine displays the dialogue shown in Figure 3.26.

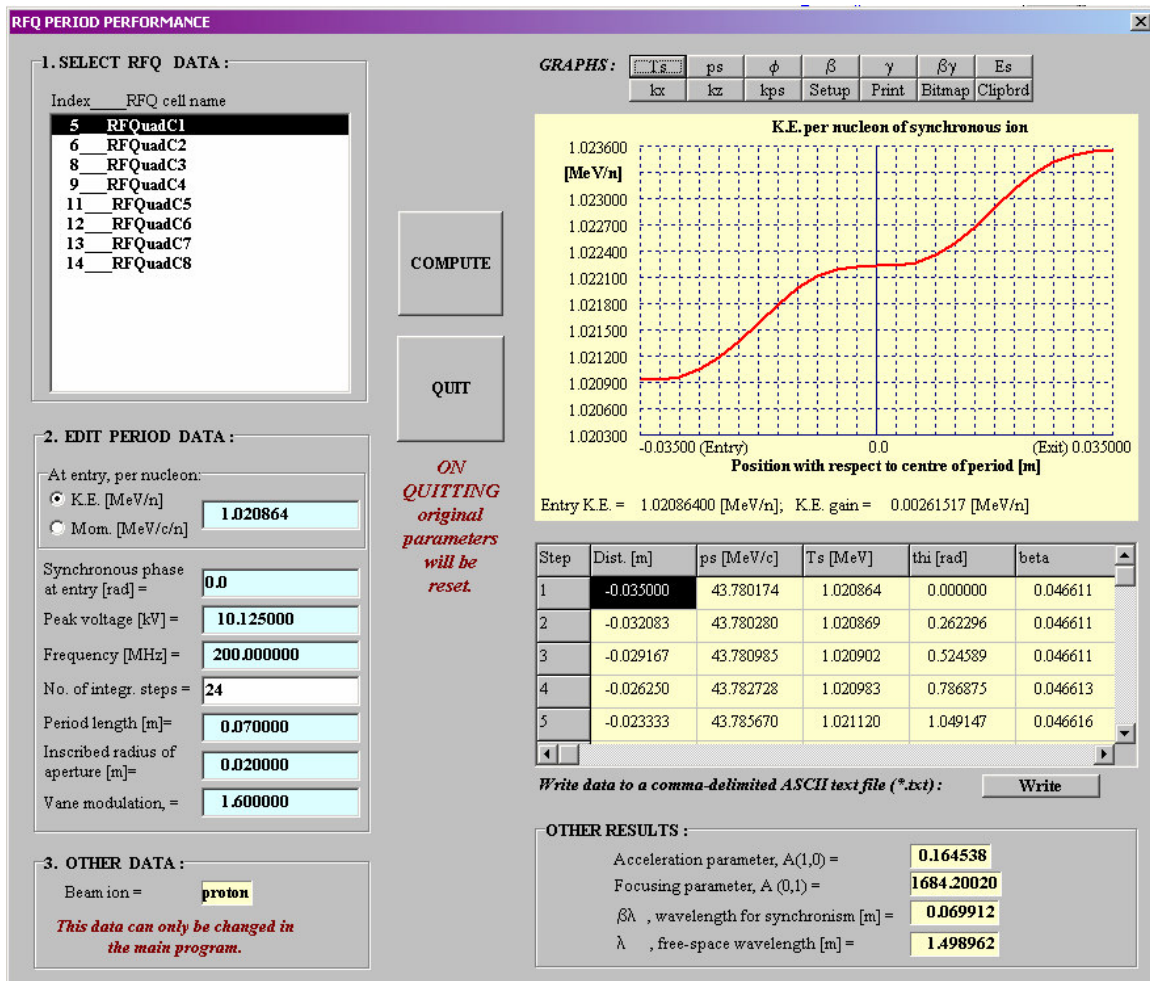


Figure 3.26 Calculator for investigating the performance of an RFQ period

On the top left hand side of Figure 3.26, a list box shows all the RFQ cells in the lattice from which the user should select the cell to be investigated. The parameters that can be freely set are directly below the list box. By clicking 'Compute', the routine will read the data and calculate the beam characteristics when traversing one RFQ period

comprising the chosen cell. The data base table on the right hand side at the bottom will then fill up with data, which can be shown graphically using the buttons above the graph space. The example in Figure 3.26 shows the kinetic energy gain of the particle as it moves through one RFQ period. Below the data table there are some additional parameters such as the *acceleration and focusing parameters*. The ‘Write’ button under data base table can be used to create a comma-delimited ASCII text file of the table’s contents. The graphs can be printed or stored as bitmaps. Clicking ‘Quit’ clears the dialogue box and re-sets the original data in the data base.

3.4.8 Relativistic calculator...

Figure 3.27 Calculator for relativistic beam data

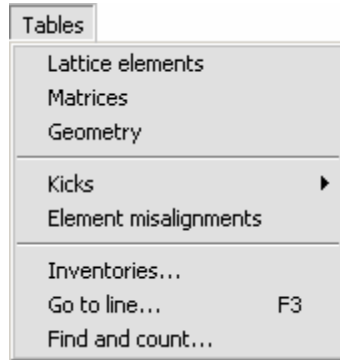
The dialogue box shown in Figure 3.27 requests the particle type and its kinetic energy or momentum (the corresponding value is then calculated by the routine and can be accessed by clicking on the appropriate radio button). Once this basic data is entered, the ‘(Re)-Compute’ button calculates a set of frequently-used relativistic parameters and displays them. By selecting ‘Set new data’, the new values can be stored for the current lattice in the main program. Alternatively, the ‘Re-set original values’ returns the

memory to its original condition, so allowing the dialogue box to be used as a relativistic calculator^{*}. The data from the dialogue can be copied to the clipboard or the notebook.

3.4.9 *Analytic matching...*

The analytic matching functions are not installed in this version. They are available in versions 3.5 and earlier.

3.5 *Tables Menu*



The Tables Menu provides access to the data currently in memory as well as some utility functions

3.5.1 *Lattice elements*

This command displays the lattice elements with their parameters (i.e. name, type, length, bending angles, edge angles, quadrupole gradient, etc.). Once the type of beam ion has been confirmed the on-axis kinetic energy, the on-axis momentum, the on-axis relativistic gamma, the on-axis relativistic beta and the average momentum deviation are also listed in the extreme right-handed columns of the spreadsheet. The lattice elements can be edited by entering the Edit Window (see Chapter 4) and the display can be printed by selecting 'Output | Print Display'.

3.5.2 *Matrices*

This command displays the matrix elements for each of the lattice elements. At the end of the listing, there is a summary line containing the single 6×6 matrix for a full turn in a ring or a single transit in a line. The matrix can also be viewed in 'block' form by double-clicking anywhere along the line belonging to an element. The elements of the 6×6 matrices are labelled as:

^{*} Note: The relativistic and particle constants included in the program form a consistent set of data taken from "Tables of Physical and Chemical Constants" by G.W.C. Kaye and T.H. Labye, 14th edition, (Longman, London and New York) (1973). Other references may have slightly different values as definitions of basic quantities change. However, beware of changing an isotope mass, for example, to a more 'modern' value using the 'user function', since this would also require other constants (electron charge etc.) to be changed inside the program to maintain consistency.

$$\begin{pmatrix} hh11 & hh12 & hv11 & hv12 & hs11 & hs12 \\ hh21 & hh22 & hv21 & hv22 & hs21 & hs22 \\ \hline vh11 & vh12 & vv11 & vv12 & vs11 & vs12 \\ vh21 & vh22 & vv21 & vv22 & vs21 & vs22 \\ \hline sh11 & sh12 & sv11 & sv12 & ss11 & ss12 \\ sh21 & sh22 & sv21 & sv22 & ss21 & ss22 \end{pmatrix}.$$

This matrix corresponds to the column vector $(x, dx/ds, z, dz/ds, ds, dp/p)$. The higher-order lenses have identity matrices and are included in calculations as point kicks. The display can be printed by selecting ‘Output | Print Display’.

3.5.3 *Geometry*

This command displays the geometry and other survey data at the entry to each of the lattice elements with an exit line for the last element. The display can be printed by selecting ‘Output | Print Display’.

3.5.4 *Kicks*

This command activates a sub-menu with four items. The first item concerns point dipole kicks [rad], the second thin-lens quadrupole kicks [T], the third thin-lens skew quadrupole kicks [T] and the fourth point momentum kicks $[dp/p]$. In all cases, the kicks are set at the entrances to the lattice elements. Any display can be printed by selecting ‘Output | Print Display’.

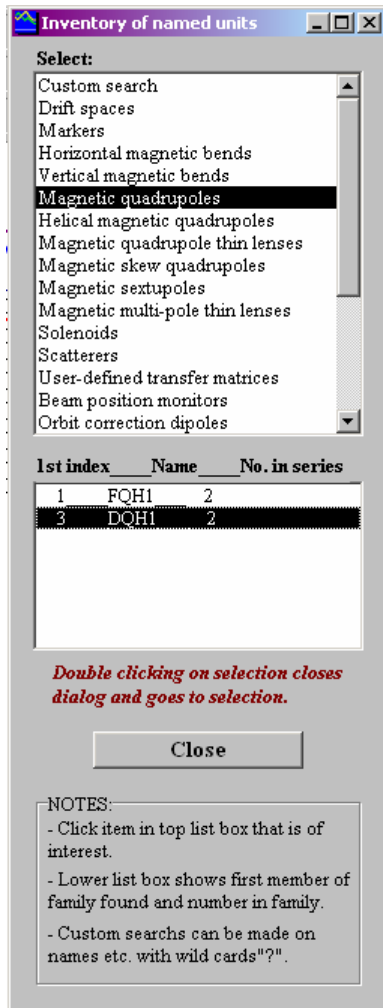
3.5.5 *Element misalignments*

This command displays the values of misalignments of various types at the entrances to the lattice elements. The display can be printed by selecting ‘Output | Print Display’.

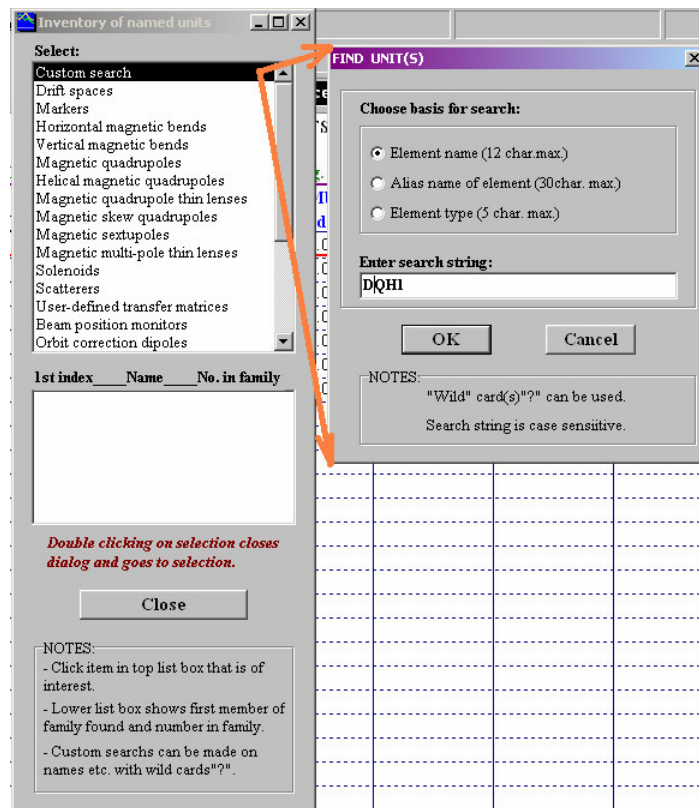
3.5.6 *Inventories...*

The ‘Inventories’ dialogue box is shown Figure 3.28. In the example (a), the category of ‘Magnetic quadrupoles’ has been selected in the upper list box. The lower list box shows that there are two series named ‘FQH1’ and ‘DQH1’ starting at elements 1 and 3 respectively and each series contains 2 units. If an entry in the lower list box is double clicked, then focus is transferred to the first element in that series.

Note that the first category in the upper list box is ‘Custom search’, for which a second dialogue box is opened for the search details, as shown in example (b). The symbol ‘?’ can be used as a multi-use wild card e.g. ‘?QD?50?’.



(a) Inventories



(b) Custom search in inventories

Figure 3.28 Lattice element inventory

3.5.7 Go to line... (F3)

This command sets the focus on a particular element/line number, see Figure 3.29.

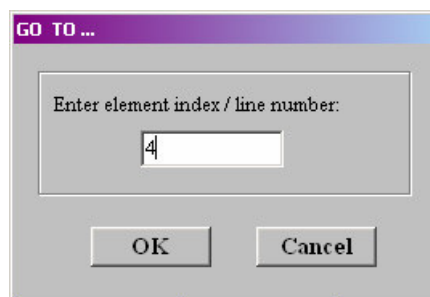


Figure 3.29 Setting the focus on a particular element/line number

3.5.8 Find and count...

The 'Find and count' dialogue box is shown Figure 3.30. This routine is similar to the 'Inventories' routine, but it has the useful feature of being able to step through the lattice from one member of a series family to the next while counting. The symbol '?' can be used as a multi-use wild card e.g. '?QD?50?'.

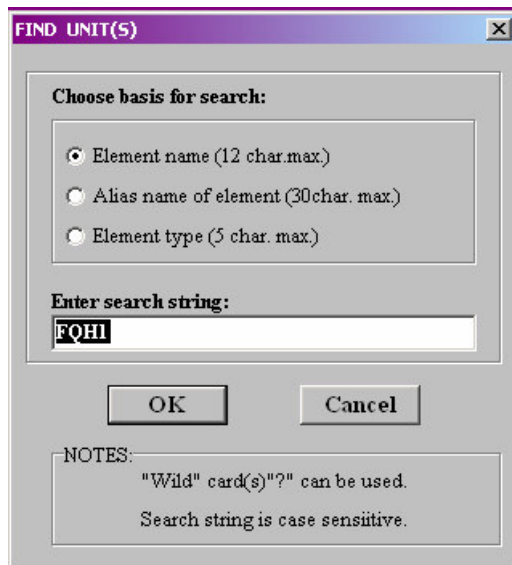
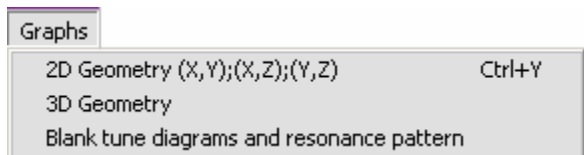


Figure 3.30 Finding particular units and stepping through series

3.6 Graphs Menu



The Graphs Menu provides access to the data currently in memory

3.6.1 2D Geometry (X,Y);(Z,X);(Y,Z) (Ctrl+Y)

This routine displays three views of the lattice (plan, elevation and side elevation) in the survey co-ordinate system, see Figure 3.31. For details of the graphical display and its controls see Section 8.4.1.

3.6.2 3D Geometry

This routine displays a 3D survey view of the lattice. The view can be changed by changing the observer's position to the left/right and up/down. For details of the graphical display and its controls see Section 8.4.2.

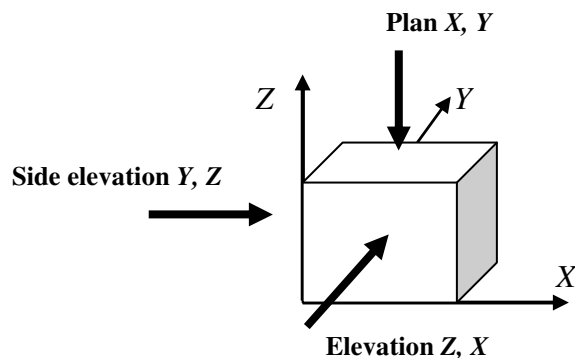
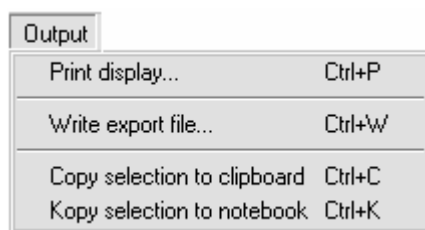


Figure 3.31 2D survey geometry views

3.6.3 Blank tune diagrams and resonance pattern

It is often useful to have blank tune diagrams of either Q_x or Q_z versus $\Delta p/p$ or Q_x versus Q_z . for sketching ideas for working line manipulations. This menu item displays blank diagrams of all types that are scaleable and can have the non-linear resonance pattern up to 9th order included. Blank diagrams can be printed using the button 'Print' at the bottom of the graph window on the right-hand side. For details of the graphical display and its controls see Section 8.14.

3.7 Output Menu



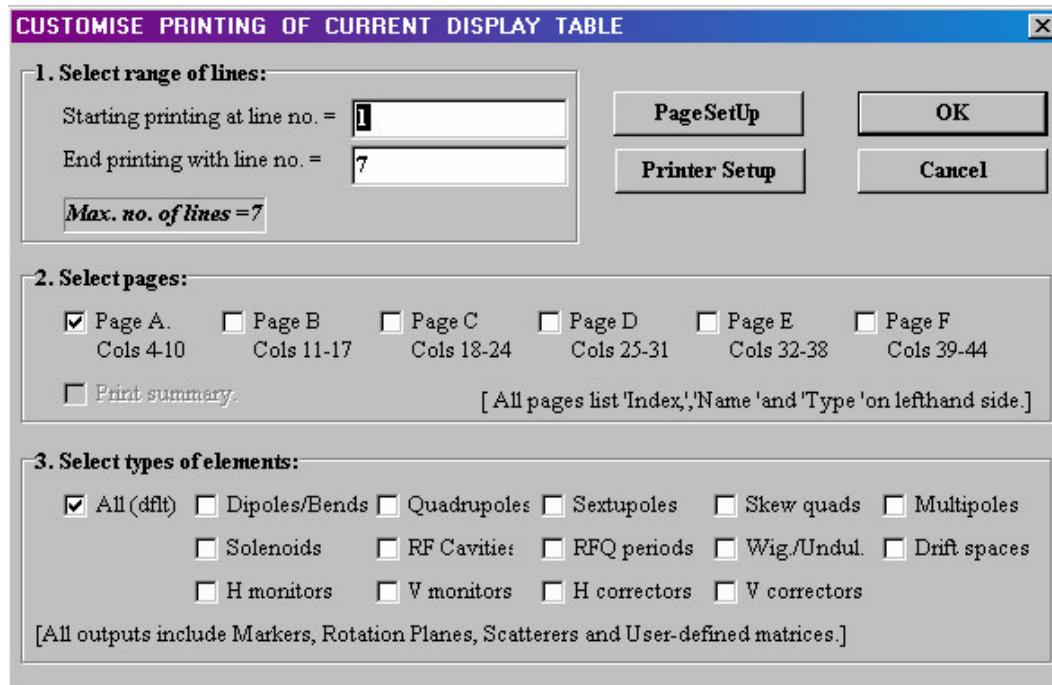
The Output Menu provides methods for exporting numerical data

3.7.1 Print display... (Ctrl+P)

This command prints the current parameter listing. Since the spreadsheet displays can be many thousands of lines long and many tens of columns wide, there are several options for choosing sub-sets of the data for printing. The print dialogue box is shown in Figure 3.32. When printing, the columns are spread across pages A, B, C... so that an output will contain pages 1A, 1B,...2A, 2B, 3A, 3B etc. When pages are not needed by a display they are grayed in the dialogue box. Each page has the index number, the name and the type of the element down the left-hand side. Any sub-set of lines and any selection of pages with or without the summary chart (if it exists) can be chosen in the dialogue box. It is also possible to choose to print only certain categories of elements.

The need to fill out the line limits and pages in the dialogue box can often be avoided by first selecting on the screen what is to be printed and then clicking 'Print display'. It is also possible to be more radical and to re-order the columns before

printing. This is done by clicking on the small numbered grey boxes in the column titles and dragging the columns to the desired positions.



CUSTOMISE PRINTING OF CURRENT DISPLAY TABLE

1. Select range of lines:

Starting printing at line no. =

End printing with line no. =

Max. no. of lines = 7

2. Select pages:

☒ Page A. Cols 4-10 ☐ Page B. Cols 11-17 ☐ Page C. Cols 18-24 ☐ Page D. Cols 25-31 ☐ Page E. Cols 32-38 ☐ Page F. Cols 39-44

☐ *Print summary.* [All pages list 'Index','Name' and 'Type' on lefthand side.]

3. Select types of elements:

☒ All (dft) ☐ Dipoles/Bends ☐ Quadrupoles ☐ Sextupoles ☐ Skew quads ☐ Multipoles

☐ Solenoids ☐ RF Cavities ☐ RFQ periods ☐ Wig./Undul. ☐ Drift spaces

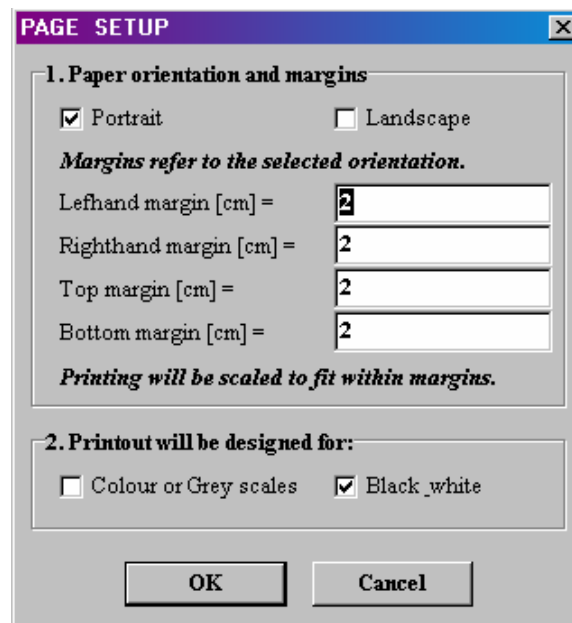
☐ H monitors ☐ V monitors ☐ H correctors ☐ V correctors

[All outputs include Markers, Rotation Planes, Scatterers and User-defined matrices.]

Buttons: PageSetUp, OK, Printer Setup, Cancel

Figure 3.32 Print dialogue box for spreadsheet displays

The 'Page setup' button in the dialogue box opens a new dialogue box (see Figure 3.33) that allows the user to set the margins on the printout, set the paper orientation and instruct the program to prepare the output for including colour or only using black and white. Finally, there is a 'Printer Setup' button.



PAGE SETUP

1. Paper orientation and margins

☒ Portrait ☐ Landscape

Margins refer to the selected orientation.

Lefthand margin [cm] =

Righthand margin [cm] =

Top margin [cm] =

Bottom margin [cm] =

Printing will be scaled to fit within margins.

2. Printout will be designed for:

☐ Colour or Grey scales ☒ Black white

Buttons: OK, Cancel

Figure 3.33 Page setup dialogue box for spreadsheet displays

3.7.2 Write export file... (Ctrl+W)

This command allows the user to create data files for export from the data currently in memory using the dialogue box shown in Figure 3.34. In the example shown, the input data and the matrices are also available, but have not been selected for inclusion in the file. The separator for the data can be chosen to suit the application that will receive the data, e.g. a tab will be recognised by MS-EXCEL, a comma is recognised by AUTOCAD. The dialogue box automatically grays data choices that are not available.

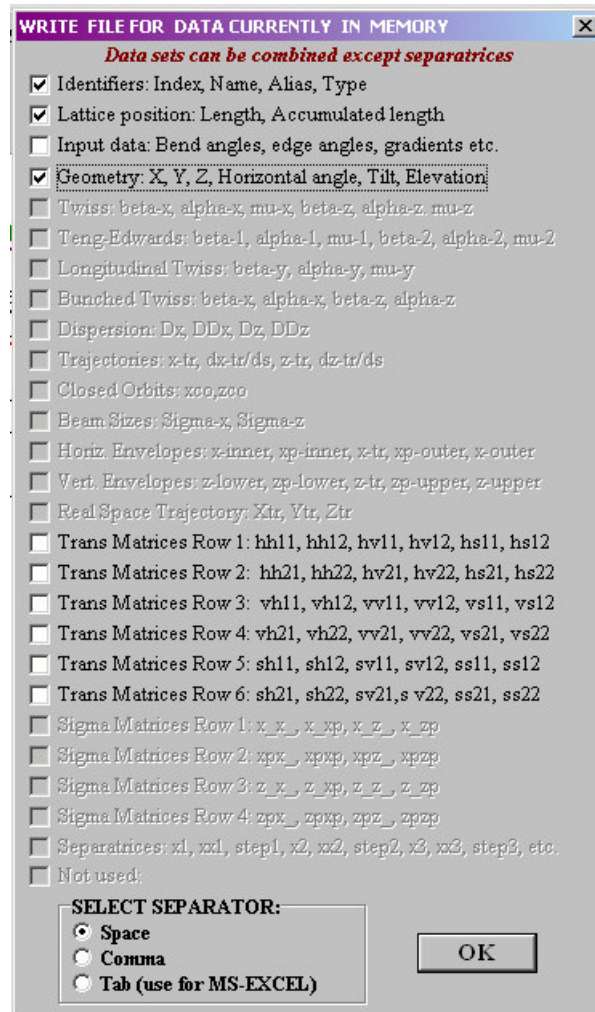


Figure 3.34 Creating an export file

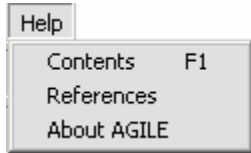
3.7.3 Copy selection to clipboard (Ctrl+C)

This command copies whatever is selected in the spreadsheet display to the Windows clipboard. Selections can be made beyond the visible display by dragging. Note that by clicking on a column title the whole column will be selected. Ctrl+C will then copy the whole column to the clipboard and Ctrl+V can paste that column directly into EXCEL. It is quicker to export a limited number of columns to EXCEL in this way than by creating an export file

3.7.4 Kopy selection to clipboard (Ctrl+K)

This command copies whatever is selected in the spreadsheet display to the lattice notebook. The lattice notebook is automatically stored with the lattice file.

3.8 Help (F1)



The Help Menu provides a standard help file, key references and copyright and disclaimer notices

3.9 Other functions

Double-clicking a cell in the spread sheet selects the element on that row and displays a synopsis of its attributes in a single dialogue box, see Figure 3.35. In the example shown, the normalised gradient is defined, but the field gradient [T/m] is not yet available because the ion and the ion energy have not yet been specified.

SUMMARY OF ELEMENT DATA

Element name: **QF (1)** Element/line no.=7

Alias name:

Type: **QUADR (magnetic quadrupole)**

Length [m] = **0.175**

Rotation round axis [rad] = **0**

Not used	-	Not used	-
Not used	-	Not used	-
Norm. k-quad. [m-2] =	-0.31357054698	Quad. gr. dBz/dx [T m-1] =	* See below
Norm. kk-sext. [m-3] =	0	Sext. gr. d2Bz/dx2 [T m-2] =	0
Not used	-	Not used	-
Not used	-	Not used	-

Form of aperture : **RRRR**

Left wall [m] = **-0.1**

Right wall [m] = **0.1**

Lower wall [m] = **-0.1**

Upper wall [m] = **0.1**

Half gap/inscr. rad. [m] = **Not available**

R=rectangular (square),
E=elliptical (circular),
S=super-elliptical,
D=diamond.

Upper

Left **Right**

Lower

OK

* To be calculated once reference ion and energy are known.

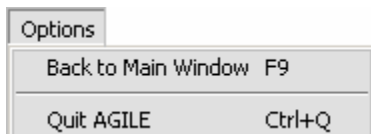
Figure 3.35 Synopsis of an element's attributes obtained by double-clicking on the corresponding row in the spreadsheet

* * *

Chapter 4 Edit Window

The Edit Window is accessed via the Main Window and is dedicated to the creation and editing of the lattice data. The menu items described below are all high level editing functions that can simplify the manipulation of complicated lattices. The alternative of direct data entry cell-by-cell by hand is described at the end of this chapter.

4.1 Options Menu



The Options Menu contains functions that control the computational environment.

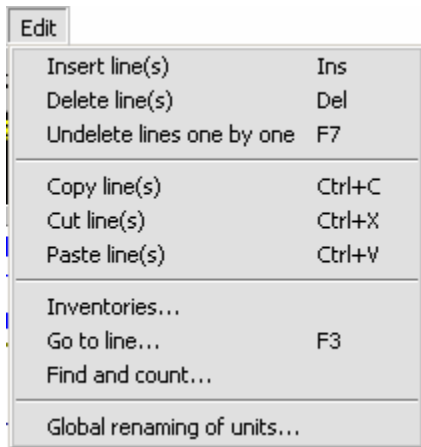
4.1.1 Back to the Main Window (F9)

This command forces an immediate return to the Main Window, even if the lattice data is incomplete or unchecked.

4.1.2 Quit AGILE (Ctrl+Q)

Before the program terminates, temporary files are deleted and the user is offered the possibility of saving the current lattice file and notebook, if this has not been done recently.

4.2 Edit Menu



The Edit Menu contains basic editing functions for the spreadsheet.

4.2.1 Insert line(s) (Ins)

This command inserts an empty line at the position of the highlighted cell by stepping the display down one line starting with the highlighted cell. If several lines are highlighted (either by dragging the mouse down column 1 or by dragging the mouse diagonally across a rectangular area), then the command inserts that many empty lines immediately before the highlighted area. After the insertion of the empty lines, the highlight is left in the second column of the top line of the inserted lines.

4.2.2 *Delete line(s) (Del)*

This command deletes the line in which highlighted cell is placed. If several lines are highlighted (either by dragging the mouse down column 1 or by dragging the mouse diagonally across a rectangular area), then the command deletes all the lines covered by the highlighted area. After deletion of the selected lines, the highlight is placed: in second column at the line position of the top line deleted.

4.2.3 *Undelete lines one by one (F7)*

This command undeletes the last line deleted (one at a time) and inserts it at the position of the highlighted cell making it possible to insert undeleted lines at any position within the lattice (not just the position of the original deletion). If several lines are highlighted (either by dragging the mouse down column 1 or by dragging the mouse diagonally across a rectangular area), then the insertion is made at the position of the top line of the selected area. After the undeletion of lines, the highlight is positioned in the second column of the top line of the undeleted lines. Up to 50 lines can be undeleted one after the other (this limit is set by the deleted lines' memory). Once the memory is empty the routine continues by inserting empty lines.

4.2.4 *Copy line(s) (Ctrl+C)*

This command copies the line with the highlighted cell, or the batch of lines if several lines have been highlighted (either by dragging the mouse down column 1 or by dragging the mouse diagonally across a rectangular area). Copied lines can be pasted multiple times at any position selected by the highlighted cell. The maximum number of lines that can be copied at one time is 50 (this limit is set by the cut & paste memory).

4.2.5 *Cut line(s) (Ctrl+X)*

This command cuts (and stores) the line with the highlighted cell or the batch of lines if several lines have been highlighted (either by dragging the mouse down column 1 or by dragging the mouse diagonally across a rectangular area). Cut line(s) can be pasted multiple times at any position selected by the highlighted cell. The maximum number of lines that can be cut at one time is 50 (this limit is set by the cut & paste memory).

4.2.6 *Paste line(s) (Ctrl+V)*

This command pastes line(s) at the position of the highlighted cell or at position of the top line of a batch of highlighted lines, by pushing the line or lines with the highlight downwards. Alternatively, by dragging the mouse down column 1 to select whole lines, the selected lines can be deleted and the stored lines inserted.

The same line(s) can be pasted multiple times at any position(s) selected by the highlighted cell. The maximum number of lines that can be pasted at one time is 50 (this limit is set by the cut & paste memory).

4.2.7 Inventories... (see Section 3.4.6)

4.2.8 Go to line... (see Section 3.4.7)

4.2.9 Find and count... (see Section 3.4.8)

4.2.10 Global renaming of units...

It is sometimes necessary to revise the naming scheme of a lattice and this may involve hundreds of changes. This routine changes all instances of a particular name, either to a new name or to the empty state of no name. Finally, it returns the number of units affected, see Figure 4.1. When first opened, the dialogue box offers a default name taken from the position of the highlighted cell, or the top left-hand corner of a highlighted area. If the new name already exists, then a warning is given.

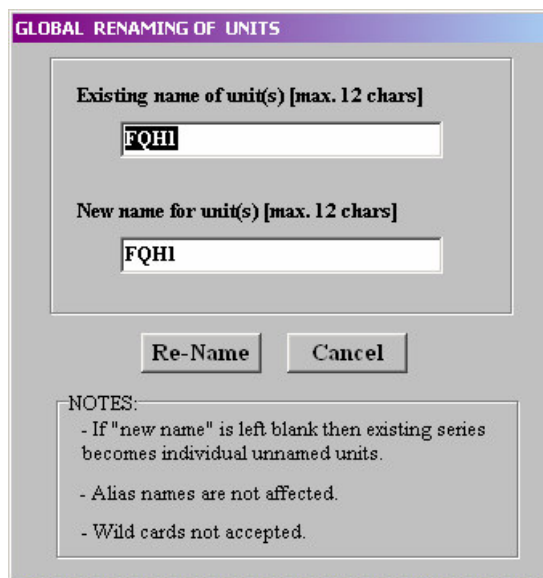
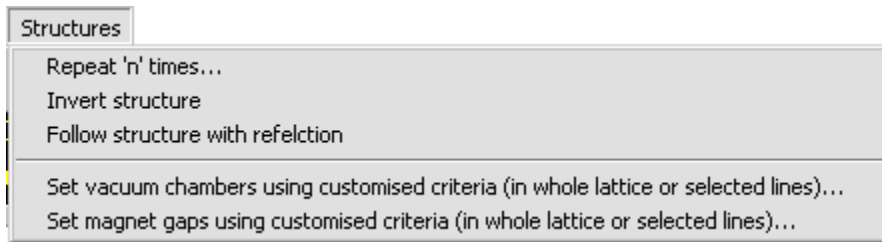


Figure 4.1 Global renaming of units

4.3 Structures Menu



The Structures Menu contains utility routines for lattice creation.

4.3.1 Repeat 'n' times...

Drag the left-hand mouse button in the first column to select the block of lines in the lattice that is to be repeated. Select the menu item 'Repeat 'n' times...' and place in the dialogue box the number of repeats (n) to be made (see Figure 4.2). The routine then marks the selected structure with double parenthesisises ((...)) followed by the number of repeats to be made. The lattice can be stored in this compact form, or it can be expanded for calculations (see Section 4.5).

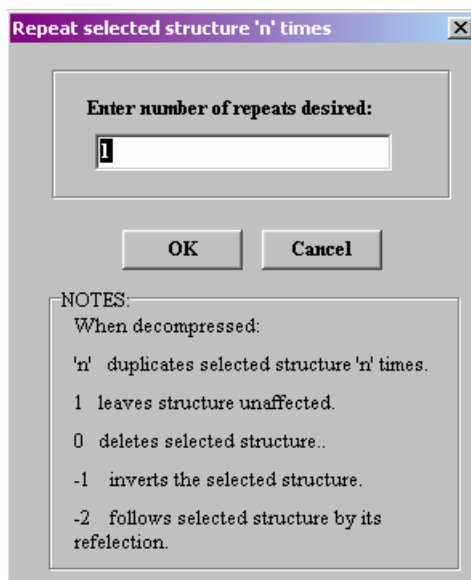


Figure 4.2 Operating on sections of lattice

If preferred an empty line can be inserted on either side of the block that is to be repeated and the double parentheses can be entered by hand in column 3 with the number of repeats after the closing parentheses.

The routine can also be used to delete, invert and reflect structures according to the value of n used.

$n = -2$, follow structure with its reflection.

$n = -1$, invert structure.

$n = 0$, purge structure.

$n = 1$, leave structure unchanged.

$n = 2$ to 99, repeat structure n times.

4.3.2 *Invert structure*

Drag the left-hand mouse button in the first column to select the block of lines in the lattice that is to be inverted and select the menu item 'Invert structure'. The routine then marks the selected structure with double parentheses ((...)) followed by -1. The lattice can be stored in this compact form, or it can be expanded for calculations (see Section 4.5).

4.3.3 *Follow structure with reflection*

Drag the left-hand mouse button in the first column to select the block of lines in the lattice that is to be followed by its reflection and select the menu item 'Follow structure with reflection'. The routine then marks the selected structure with double parentheses ((...)) followed by -2. The lattice can be stored in this compact form, or it can be expanded for calculations (see Section 4.5).

4.3.4 Set vacuum chambers using customised criteria (in whole lattice or selected lines)...

This routine calls the dialogue box shown in Figure 4.3. The routine can be made to treat subsets of the lattice using the search tools in Sections 1 and 2 of the dialogue box. The wild card '?' is accepted in searches. If a restricted section of lattice is to be treated, this can be pre-selected by selecting those lines in the lattice before calling the routine. The selection lines will then appear in Section 3 of the dialogue box. Alternatively, the limits can be typed directly into the edit boxes. Sections 4 and 5 of the dialogue box deal with the vacuum chamber definition.

SET VACUUM CHAMBER DIMENSIONS AND FORM IN SELECTED UNITS

1. CHOOSE BASIS FOR SELECTION :

☒ Element name (12 char.max.)
☐ Alias name of element (30char. max.)
☐ Element type (5 char. max.)

2. ENTER SEARCH STRING :

FOH1

3. DEFINE RANGE IN LATTICE :

Enter index of first element
Enter index of last element
[For full lattice enter indices 1 to 12]

4. ENTER WALL POSITIONS ON AXES :

Left wall position [m]
Right wall position [m]
Lowermost wall position [m]
Uppermost wall position [m]
[Positions are with respect to the central orbit]

5. SELECT FORM OF VACUUM CHAMBER :

☐ Rectangular and square.
☐ Elliptical and circular.
☐ 'Super' elliptical i.e. $(x/a)^3 + (z/b)^3 = 1$.
☐ 'Diamond' shaped.
☐ Compound shape.

Chamber: RRRR

Confirm one by one Apply to all units found Quit

NOTES:

- "Wild" card(s) "?" can be used in the search string.
- To select all elements enter a single "?" in the search string. Note the search string is case sensitive.
- The vacuum chamber wall is better interpreted as the BOUNDARY FOR BEAM LOSS. Thus it is used to define pickup plates, collimators, dumps blocks, septa wires etc. For RF cavities, RFQs and electrostatic lenses use a circular "chamber" with the inscribed radius defined by the electrodes/poles.

Figure 4.3 Setting vacuum chamber sizes and forms

The simple vacuum chamber forms (rectangular, elliptical, 'super' elliptical and diamond) can be selected directly in Section 5, but the option of a compound shape opens a further dialogue box, see Figure 4.4. Although, the dialogue boxes refer to the vacuum chamber, it would be more exact to refer to the boundary upon which particles will be lost. The example shown in Figure 4.4 could, for example, represent the aperture set by inclined collimators. Note that it is also possible to type all the data into the spreadsheet directly, but for large groups of elements the dialogue box is more efficient.

SET VACUUM CHAMBER DIMENSIONS AND FORM IN SELECTED UNITS

1. CHOOSE BASIS FOR SELECTION :

AID FOR DEFINING COMPOUND VACUUM CHAMBERS

4th QUADRANT

☐ Rectangular or square (R)

☒ Diamond shaped (D)

☐ Elliptical or circular (E)

☐ Super elliptical (S)

1st QUADRANT

☐ Rectangular or square (R)

☐ Diamond shaped (D)

☐ Elliptical or circular (E)

☒ Superelliptical (S)

Upper

Left Right

Lower

3rd QUADRANT

☐ Rectangular or square (R)

☐ Diamond shaped (D)

☐ Elliptical or circular (E)

☒ Super elliptical (S)

2nd QUADRANT

☐ Rectangular or square (R)

☐ Diamond shaped (D)

☐ Elliptical or circular (E)

☐ Super elliptical (S)

Apply Cancel

NOTES:

- Chambers can be entered directly into spreadsheet by typing the code letters in the order of the quadrants (clockwise) e.g. EERR.
- Chamber is placed symmetrically between, left, right, upper and lower limits.
- Although the program refers to a vacuum chamber, it is more strictly the BOUNDARY on which particles are LOST.
- More sophisticated boundary shapes are possible via 'collimators' and 'apertures'. [NOT INSTALLED].

4. ENTER WALL POSITIONS ON AXES :

Left wall position [m] -0.1

Right wall position [m] 0.1

Lowermost wall position [m] -0.1

Uppermost wall position [m] 0.1

[Positions are with respect to the central orbit]

5. SELECT FORM OF VACUUM CHAMBER :

☐ Rectangular and square.

☐ Elliptical and circular.

☐ 'Super' elliptical i.e. $(x/a)^3 + (z/b)^3 = 1$.

☐ 'Diamond' shaped.

☒ Compound shape.

Chamber: RRRR

Apply to all units found Quit

ch string. Note the search string is case sensitive.

BOUNDARY FOR BEAM LOSS.

umps blocks, septa wires etc.

a circular "chamber" with the inscribed radius defined by the

Figure 4.4 Defining compound vacuum chamber shapes

Note that vacuum chambers are an exception to the rule that elements with the same name have the same parameters. This routine will set the vacuum chamber of units of the same name within a selected section of lattice, while leaving it unchanged in elements of the same name outside the selected section of lattice.

4.3.5 Set half gaps using customised criteria (in whole lattice or selected lines)...

This routine calls the dialogue box shown in Figure 4.5. The routine can be made to treat subsets of the lattice using the search tools in Sections 1 and 2 of the dialogue box. The wild card '?' is accepted in searches. If a restricted section of lattice is to be treated, this can be pre-selected by selecting those lines in the lattice before calling the routine. The selection lines will then appear in Section 3 of the dialogue box. Alternatively, the limits can be typed directly into the edit boxes. Section 4 of the dialogue box deals with the magnet gaps. Note that the value required for dipoles is the half pole gap and for lenses and RF equipment it is the radius of the inscribed circle. It is possible to type all the data into the spreadsheet directly, but for large groups of elements the dialogue box is more efficient.

SET MAGNET GAPS IN CUSTOM-SELECTED UNITS

1. CHOOSE BASIS FOR SELECTION :

☐ Element name (12 char.max.)
☐ Alias name of element (30char. max.)
☒ Element type (5 char. max.)

2. ENTER SEARCH STRING :

QUADR

3. DEFINE RANGE IN LATTICE :

Enter index of first element: 3

Enter index of last element: 6

[For full lattice enter indices 1 to 12]

4. ENTER GAP DATA :

For dipoles enter half-height of magnet gap [m]
 OR
 For lenses enter radius of inscribed circle [m]

0.1

Apply to all units found

Confirm one by one

Quit

NOTES:

- "Wild" card(s) "?" can be used in search. The search string is case sensitive.
- Gap data is used for the calculation of pole-tip fields, magnetic images, electric gradients and Einzel lenses.
- Gap data is not needed for drift spaces or thin lenses.

Figure 4.5 Setting half gaps

4.4 Auxiliary Data Menu

Auxiliary_Data

- Edit/enter data for scatterers...
- Edit/enter user-specified matrices...
- Edit/enter advanced RF cavity data...
- Edit/enter additional data for RFQs...
- Edit/enter additional data for helical magnetic dipoles...
- Edit/enter additional data for electrostatic bends...
- Edit/enter additional data for helical quadrupoles...
- Edit/enter additional data for magnetic wigglers/undulators...

The Auxiliary Data Menu is used to create and edit the additional data needed for special units.

4.4.1 Edit/enter data for scatterers...

This command activates a dialogue box (see Figure 4.6) for creating and editing up to 25 sets of data for scatterers and absorbers. The program uses the data in two ways:

- *Tracking single particles and distributions through lattices with scatterers. The calculation includes multiple coulomb scattering, energy loss by ionisation, losses from inelastic collisions. Elastic nuclear scattering can be included for protons only.*
- *Modification of the Twiss parameters to plot envelope functions and emittance plots. This includes multiple coulomb scattering, energy loss by ionisation, losses from inelastic collisions, BUT NOT elastic nuclear scattering.*

To compare results from the two methods the elastic nuclear scattering should be switched off. The nuclear elastic scattering produces wide, low-density tails that affect the statistical fitting quite strongly, but in practice are probably lost on the walls. The main reason for including this form of scattering is for simulating collimators, where the additional loss could be important especially for superconducting magnets.

EDIT/ENTER DATA FOR SCATTERERS...

1. SELECT UNIT
Lattice index no.
This is a fresh assignment.
Please enter new data.
NOTE: All units with the same name will be equally affected.

2. MATERIAL
☐ Pb
☐ Au
☐ W
☐ Cu
☐ Fe
☐ Al
☐ C
☐ Be
☐ Air
☒ Lucite
☐ Water
☐ Vacuum

3. COULOMB SCATTERING (can be edited)
 Density [g/cm³] =
 Radiation length [g/cm²] =
 Shape [ONLY 'Flat'] =

4. NUCLEAR SCATTERING (can be edited)
 Nucl. collision lgth [g/cm²] =
 Nucl. inel. col. lgth [g/cm²] =
 - To switch off all nuclear scattering enter zero in both boxes.
 - To switch off inelastic collisions put zero in the second box only.

5. INTEGRATION STEPS:
 No. of integration steps =
 Used for particle tracking ONLY. See HELP.
 Rec. step [m] = 0.0687

6. SUB-DIVISION
 No. of slices for viewing =
 If not a submultiple of the number of integration steps, program will adjust values.

Buttons: DELETE AND SET A DRIFT, Load from notebook, Copy data to clipboard, Set MeV defaults, Set GeV defaults, Paste time stamp, Copy data to notebook, Save this data set, Finished.

NOTES:
 - Nuclear inelastic collisions are used to calculate transmission efficiency [default below 1 GeV 100%].
 - Nuclear elastic is ONLY available for protons and when tracking.
 - The scatterer can be subdivided into slices to view the beam at intermediate points. This has no effect on the accuracy of the calculation when in Twiss mode or when tracking particles. The number of integration steps, however, does affect tracking.

Figure 4.6 Entering auxiliary data for a scatter or absorber

Section 1 of the dialogue box guides the selection of the unit in the lattice. If the unit is pre-selected before activating the dialogue box, then it appears directly in this section. Section 2 offers the choice of 12 materials. The internally stored parameters for these materials appear in Section 3. These values can be edited, if better values are known. In fact, it is possible to ‘trick’ the program into calculating for a material not explicitly in the list by entering the values required. Note that human tissue is close to water with energy loss values within 2%. Section 5 sets the integration step for use when tracking single ions and Section 6 is for the number of points at which the calculation is to be viewed (i.e. the number of slices that will appear in the lattice listing for the scatterer). After setting the data for a scatterer, click 'Save this data set' to update the memory. Moving to a new scatterer, or clicking 'Finished' quits the dialogue box without updating the memory.

When creating a lattice, enter just one element for each scatterer in the main database and then use the auxiliary data dialogue box (see Figure 4.6) to set the number of slices the scatterer is to be split into for viewing. This is the only safe way to subdivide scatterers. Note that the number of integration steps is a separate parameter and that it has to be a multiple of the number of slices into which the scatterer is divided. It is possible to make copies of a scatterer and paste it elsewhere in the lattice, but be sure to copy all slices. The program makes consistency checks and colour codes the slices so that adjacent scatterers are easily distinguished. If at a later time, the subdivision is edited the program will revise the whole lattice.

Some notes:

- *In line with the general philosophy of the program, all elements of the same name are attributed with the same scattering data. However, scatterers can only be ‘active’ in the Line Window and are highlighted with the colour cyan. Lattices that formally*

include scatterers can be treated in the Ring Window, but only once the scatterer has been deactivated by being set to vacuum. Scattering and energy losses can alternatively be ‘switched off’ by entering zero for the radiation and collision lengths.

- *The multiple coulomb scattering is based on the Highland formula. All particles can be treated except, H minus, H2 plus and electrons. The type of ion can only be changed in the main program.*
- *The nuclear elastic scattering is calculated from the total nuclear collision length and the inelastic nuclear collision length. This is only applied by the program to protons when in the tracking mode. In almost all cases, it is acceptable (if not better) to neglect this effect as being too small to see in a practical beam (the thin tails are lost). However, statistically the effect of a few large angle particles can strongly affect the emittance values. Putting the total nuclear collision length equal to the inelastic length, puts the elastic scattering to zero, while maintaining the inelastic scattering. The nuclear inelastic scattering is a cause of particle loss.*
- *The default data for nuclear scattering is for protons. The inelastic losses are set to zero below 1 GeV and an average value for multi-GeV energies. The data can be edited. Putting zero for both the total nuclear collision length and the inelastic length switches off all nuclear scattering. When starting a problem, it is recommended that the nuclear scattering is switched off.*
- *The step size is only used when particle tracking. When using the Twiss representation, the calculations are best done without sub-division of the scatterer. This is because the Highland formula, upon which the scattering envelopes are based, is a fit to the behaviour in a thick scatterer including energy loss and it is not a description of the physics. The fit is generally quoted over the range 10^{-3} radiation lengths to 100 radiation lengths and at the edges of this range the error mounts to 11%. Clearly, there is no point in subdividing a thick scatterer so that the thickness of the slices approaches the lower edge of the range where the accuracy is poor. The policy followed in AGILE is to always apply the formula in a single application.*
- *WARNING: If the lattice is substantially altered during editing, then the scattering data may become inconsistent. The program tries to maintain the auxiliary scattering data in a consistent manner, but it may not succeed. The user should therefore always check after editing that the scattering data is still consistent with the modified lattice.*
- *WARNING: The energy loss can be so great that the particle stops. If this happens outside magnetic elements, then the optics is always correct. Thus, the program can be used to track the beam size inside a scatterer right down until the beam has stopped. This is meant for beams that are used for cancer therapy. If the beam is passing through magnetic elements as it slows down, then the results may not be correct, according to the situation.*

4.4.2 Edit/enter user-specified matrices...

This command activates a dialogue box (see Figure 4.7) for creating and editing up to 25 sets of data for user-specified matrices. Apart from the matrix elements, it is necessary to enter additional geometric and energy data because the matrix itself is limited to describing changes within the local curvilinear coordinate system. After setting the data for a matrix, click 'Save this data set' to update the memory. Moving to a new matrix, or clicking 'Finished' quits the dialogue box without updating the memory. These elements cannot be subdivided.

1. SELECT UNIT:

Lattice index

This is a new assignment.
Please enter data.

NOTE: All units with the same name will be equally affected.

2. MATRIX ELEMENTS:

hh1 = <input type="text" value="1.0"/>	hh12 = <input type="text" value="0.0"/>	hv11 = <input type="text" value="0.0"/>	hv12 = <input type="text" value="0.0"/>	hs11 = <input type="text" value="0.0"/>	hs12 = <input type="text" value="0.0"/>
hh21 = <input type="text" value="0.0"/>	hh22 = <input type="text" value="1.0"/>	hv21 = <input type="text" value="0.0"/>	hv22 = <input type="text" value="0.0"/>	hs21 = <input type="text" value="0.0"/>	hs22 = <input type="text" value="0.0"/>
vh11 = <input type="text" value="0.0"/>	vh12 = <input type="text" value="0.0"/>	vv11 = <input type="text" value="1.0"/>	vv12 = <input type="text" value="0.0"/>	vs11 = <input type="text" value="0.0"/>	vs12 = <input type="text" value="0.0"/>
vh21 = <input type="text" value="0.0"/>	vh22 = <input type="text" value="0.0"/>	vv21 = <input type="text" value="0.0"/>	vv22 = <input type="text" value="1.0"/>	vs21 = <input type="text" value="0.0"/>	vs22 = <input type="text" value="0.0"/>
sh11 = <input type="text" value="0.0"/>	sh12 = <input type="text" value="0.0"/>	sv11 = <input type="text" value="0.0"/>	sv12 = <input type="text" value="0.0"/>	ss11 = <input type="text" value="1.0"/>	ss12 = <input type="text" value="0.0"/>
sh21 = <input type="text" value="0.0"/>	sh22 = <input type="text" value="0.0"/>	sv21 = <input type="text" value="0.0"/>	sv22 = <input type="text" value="0.0"/>	ss21 = <input type="text" value="0.0"/>	ss22 = <input type="text" value="1.0"/>

3. SHIFTS IN LOCAL COORDINATES BETWEEN THE INPUT AND OUTPUT OF THE MATRIX:

C.O. shift in x [m] = Shift in horiz. angle of C.O. [rad] =

C.O. shift in s [m] = Shift in roll angle of median plane [rad] =

C.O. shift in z [m] = Shift in elevation of C.O. [rad] =

Fractional gain/loss in momentum of synchronous particle =

Figure 4.7 Entering auxiliary data for a user-specified matrix

4.4.3 Edit/enter advanced RF cavity data...

This command activates a dialogue (see Figure 4.8) for creating and editing up to 25 sets of advanced data for RF cavities. Section 1 of the dialogue box guides the selection of the cavity in the lattice. If the unit is pre-selected before activating the dialogue box, then it appears directly in this section. Section 2 requests the harmonics of the axial field on the cavity axis. The harmonics are scaled internally to give an integral of unity through the cavity. Section 3 sets the number of integration steps and Section 4 the number of points at which the calculation is to be viewed (i.e. the number of slices that will appear in the lattice listing for the cavity). After setting the data for a cavity, click 'Save this data set' to update the memory. Moving to a new cavity, or clicking 'Finished' quits the dialogue box without updating the memory. The default data corresponds to a cavity with a cosine field distribution, 12 integration steps and a single element in the database.

When creating a lattice, enter just one element for each cavity in the main database and then use the auxiliary data dialogue box (see Figure 4.8) to set the number of slices the cavity is to be split into for viewing. This is the only way to sub-divide RF

cavities. Note that the number of integration steps is a separate parameter and that it has to be a multiple of the slices into which the cavity is cut. It is possible to make copies of a cavity and paste it elsewhere in the lattice, but be sure to copy all slices. The program makes consistency checks and colour codes the slices so that adjacent cavities are easily distinguished. If at a later time, the subdivision is edited the program will revise the whole lattice.

The program calculates the ion's progress through the cavity by changing the relativistic parameters in steps, but integrating analytically the faster changing RF fields. This calculation works from first principles and does not need the transit time to be provided. A little time spent exploring the parameters using the 'RF cavity performance aid' in the Main Window will show the influence of the number of integration steps. Except at the very lowest energies, the default of 12 steps is in fact more than adequate and the speed of modern processors is such that this is not a problem.

EDIT/ENTER ADVANCED DATA FOR RF CAVITIES

1. SELECT UNIT :
 Lattice index no.
 StaticText1
 StaticText2
 NOTE: All units with the same name will be equally affected.
 SET defaults for harmonics
 Load from notebook
 Paste time stamp
 Copy data to notebook
 Copy data to clipboard

2. FOURIER HARMONICS :
 A1 =
 A3 =
 A5 =
 A7 =
 A9 =
 A11 =
 A13 =
 A15 =
 A17 =
 A19 =
 A21 =
 A23 =
 [Note: Program scales harmonics to give integral of unity.]

3. INTEGRATION STEPS :
 No. of integration steps =
 Minimum is 1 (use when fully relativistic).
 Recommended value = 12.

4. SUB-DIVISION :
 No. of slices for viewing =
 If not a submultiple of the number of integration steps, program will adjust values.
 No. of slices has no effect on the accuracy.

Save this data set
 Finished

Figure 4.8 Entering auxiliary data for RF cavities

4.4.4 Edit/enter additional data for RFQs...

Because most of the data needed for RFQs is included in the main lattice database, the additional data needed (see Figure 4.9) concerns only the number of integration steps and the number of positions for viewing the calculation. Furthermore, the same choice applies to all RFQ cells. The default data specifies 12 integration steps and a single element for each cell.

The program calculates the ion's progress through the RFQ cell by changing the relativistic parameters in steps, but analytically integrating the faster changing RF fields. A little time spent exploring the parameters using the 'RFQ period performance aid' in

the Main Window will show the influence of the number of integration steps. In most cases, the default of 24 steps is in fact more than adequate and the speed of modern processors is such that this is not a problem.

When creating a lattice, enter just one element for each RFQ cell in the main database and then use the auxiliary data dialogue box (see Figure 4.9) to set the number of slices the RFQ cell is to be split into for viewing. This is the only way to sub-divide RFQ cells. Note that the number of integration steps is a separate parameter and that it has to be a multiple of the slices into which the RFQ cell is cut. It is possible to make copies of an RFQ cell and paste it elsewhere in the lattice, but be sure to copy all slices. The program makes consistency checks and colour codes the slices so that adjacent RFQ cells are easily distinguished. If at a later time, the subdivision is edited the program will revise the whole lattice.

EDIT/ENTER DATA FOR RFQ CELLS

1. INTEGRATION STEPS PER CELL :

No. of integration steps =

Minimum is 1 (use only when fully relativistic).
Recommended value = 12.

2. SUB-DIVISION PER CELL :

No. of slices for viewing =

If not a submultiple of the number of integration steps, program will adjust values automatically.
The number of slices has no effect on the accuracy.

OK Save data **Cancel**

Note that to enter an additional RFQ cell it is necessary to enter one element in the lattice for each slice.

Figure 4.9 Entering auxiliary data for RFQ cells

4.4.5 Edit/enter additional data for helical magnetic dipoles...

The auxiliary data for helical magnetic dipoles comprises the field on the axis of the helix, the sense of rotation and the number of slices the element is to be divided into for viewing, see Figure 4.10. The default for the period is the length entered in the main database for the device and the default sense of rotation is clockwise. There is no need for integration steps to be specified as the matrices are calculated analytically. Note the length of the device when first entered is taken as the length of one period. To add a second period, paste in a second copy.

When creating a lattice, enter just one element for each helical dipole period in the main database and then use the auxiliary data dialogue box (see Figure 4.10) to set the

number of slices the helical dipole period is to be split into for viewing. This is the only way to sub-divide helical dipoles. It is possible to make copies of a helical dipole and paste it elsewhere in the lattice, but be sure to copy all slices as partial periods will not function correctly. The program makes consistency checks and signals apparent errors. If at a later time, the subdivision is edited the program will revise the whole lattice.

AUXILIARY DATA FOR HELICAL DIPOLES...

1. SELECT UNIT :

Edit/enter lattice index

Auxiliary helix data saved.

Edit or choose new unit?

2. ENTER DATA :

Field on axis [T]

No. of slices for viewing =

Sense of rotation of helix :

☒ Clockwise, looking in beam direction

☐ Anticlockwise, looking in beam direction

Save this data

Finished

NOTES :

- All units with the same name will be equally affected.

Figure 4.10 Entering auxiliary data for helical magnetic dipoles

4.4.6 *Edit/enter additional data for electrostatic bends...*

Most of the data required for electrostatic bends is included in the main database*. The auxiliary data dialogue box (see Figure 4.11) is limited to the transverse curvature (1/radius of curvature) of the electrodes. Up to 25 units can be assigned different curvatures. The default is zero curvature, which corresponds to cylindrical electrodes. Spherical electrodes have a transverse curvature equal to the main curvature and the general case of toroidal electrodes can have any transverse curvature. When the main and transverse curvatures have equal signs then the action of the transverse curvature is defocusing in the plane of bending.

Electrostatic bends can be sub-divided manually in the main data base.

* Note that the bending angles for magnetic dipoles and electrostatic bends are in different columns at widely-spaced positions.

AUXILIARY DATA FOR ELECTROSTATIC BENDS

1. SELECT UNIT :

Edit/enter lattice index

Showing default data.

New data can be entered.

NOTE: All units with the same name will be equally affected.

2. ENTER TRANSVERSE CURVATURE :

Transverse curvature as defined by the equi-potential surface passing through the central orbit.

Curvature (1/radius of curvature) [1/m]=

NOTES :

CYLINDRICAL ELECTRODES: Have zero curvature.

SPHERICAL ELECTRODES: Curvature equals curvature of central orbit.

TORROIDAL ELECTRODES: Can have any curvature positive or negative. Positive is of the spherical class and negative is of the "saddle" class.

Figure 4.11 Entering auxiliary data for electrostatic bends

4.4.7 *Edit/enter additional data for helical quadrupoles...*

The auxiliary data for magnetic and electrostatic helical quadrupoles comprises the period of the helix, the sense of rotation and the number of slices the element is to be divided into for viewing, see Figure 4.12. The default for the period is the length entered in the main database for the device and the default sense of rotation is clockwise. There is no need for integration steps to be specified as the matrices are calculated analytically. Note the length of the device need not be an integer number of periods.

When creating a lattice, enter just one element for each helical quadrupole in the main database and then use the auxiliary data dialogue box (see Figure 4.12) to set the other parameters. Using the auxiliary data is the only way to sub-divide helical quadrupoles. It is possible to make copies of a helical quadrupole and paste it elsewhere in the lattice, but be sure to copy all slices. The program makes consistency checks and will signal suspected errors.

AUXILIARY DATA FOR HELICAL QUADRUPOLES...

1. SELECT UNIT :

Edit/enter lattice index

Showing default data.

New data can be entered.

2. ENTER DATA :

Period of helix [m] =

No. of slices for viewing =

Sense of rotation of helix :

☒ Clockwise, looking in beam direction

☐ Anticlockwise, looking in beam direction

NOTES :

- All units with the same name will be equally affected.
- Length need NOT be an exact number of periods.

Figure 4.12 Entering auxiliary data for helical quadrupoles

4.4.8 *Edit/enter additional data for magnetic wigglers/undulators...*

Figure 4.13 shows the dialogue box for creating and editing up to 25 sets of additional data for wigglers/undulators. Section 1 of the dialogue box guides the selection of the wiggler in the lattice. If the unit is pre-selected before activating the dialogue box, then it appears directly in this section. Section 2 requests the ratio of the pole width to the period, the number of slices the wiggler unit is to be subdivided into, the maximum field on the median plane and finally the sense of the bending (left or right) in the first half pole when looking in the beam direction. Section 3 requests the odd harmonics of orders 1 to 11 of the vertical field on the median plane. When first investigating structures these harmonics will not be known and the program can be asked to supply calculated values according to a simple model. After setting the data, click 'Save this data' to update the memory. Moving to a new wiggler, or clicking 'Finished' quits the dialogue box without updating the memory. The default data corresponds to a wiggler with the ratio of the pole width to the period equal to 0.25, a single element for one period and a simple cosine field distribution.

When creating a lattice, enter just one element for each wiggler period in the main database and then use the auxiliary data dialogue box (see Figure 4.13) to set the number of slices the wiggler period is to be split into for viewing. This is the only way to sub-divide wigglers/undulators. It is possible to make copies of a wiggler and paste it elsewhere in the lattice, but be sure to copy all slices. The program makes consistency checks and colour codes the slices so that adjacent wiggler periods are easily

distinguished. If at a later time, the subdivision is edited the program will revise the whole lattice.

AUXILIARY DATA FOR DIPOLE WIGGLERS/UNDULATORS...

1. SELECT UNIT :
 Edit/enter lattice index:
 Auxiliary wiggler data saved.
 Edit or choose new unit?

2. ENTER DATA :
 Pole length/Period:
 No. of slices for viewing =
 Max. field on median plane [T] =
 Direction of bending for entry half-pole:
☒ Left, looking in beam direction
☐ Right, looking in beam direction

3. FOURIER HARMONICS :
 Vertical field on median plane;
 $B_z = \text{Sum} (B_n \cdot \cos(nL_s))$
 $L = 2\pi/\text{period}$, n is odd
 B1 =
 B3 =
 B5 =
 B7 =
 B9 =
 B11 =
 Calculate default harmonics
 Reset previous data
 Save this data
 Finished

NOTES :
 - All units with the same name will be equally affected.
 - Original length in lattice defines the period.
 - One period contains, in the following order, : one half pole+one full pole+one half pole.
 - Wigglers are assumed to be in horizontal plane.

Figure 4.13 Entering auxiliary data for wigglers/undulators

4.5 Check Date Menu

Check_Data
 Check with decompression (if needed) F2
 Check without decompression F8

The Check Data Menu must be used before a lattice will be accepted for computation.

4.5.1 Check with decompression (if needed) (F2)

This command first decompresses any embedded structures that are to be repeated, inverted or reflected (see Section 4.3) and then performs a series of checks and tests on the data. Trivial problems are corrected automatically, but the user is requested to deal with more complex problems. In this event, the program places the highlight on the cell with the incorrect data. Once all problems have been cleared, the dialogue box shown in Figure 4.14 is displayed. In the example, the routine has also signalled that some elements contain raw field data rather than normalised data and hence some further checking will be needed once the ion and its energy are known. The routine also points out some ambiguous data that could be correct, but may be an error.

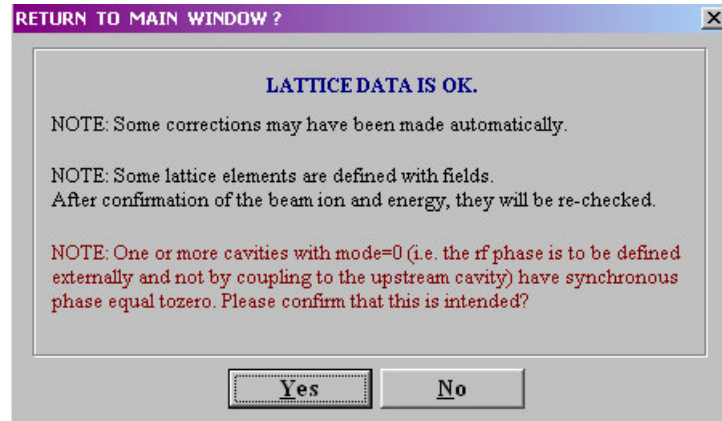
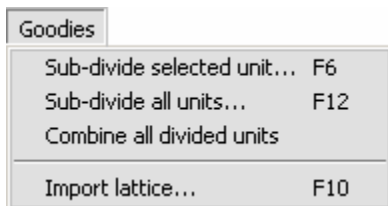


Figure 4.14 Data correct and return to Main Window

4.5.2 *Check without decompression (F8)*

This command is the same as the previous command in Section 4.5.1, except that the lattice is left in its compressed form after checking.

4.6 *Goodies Menu*



The Goodies Menu provides some useful utility functions.

4.6.1 *Sub-divide selected unit... (F6)*

Subdividing elements is useful for interpreting results and producing smooth graphs, as well as being essential for certain space-charge and coupling calculations. For example, it would be prudent to sub-divide a solenoid physics detector that was sitting in a low-beta region, but probably unnecessary to sub-divide the short skew quadrupoles. Some elements can be sub-divided by hand, but it is far safer to use the routines provided. This routine will subdivide the currently selected unit and all units with the same name into equal length slices, see Figure 4.15. If a block of elements is selected the routine asks the user to deselect the block and select a single unit. When a dipole with non-zero edge angle is subdivided, the central slices become sector bends with zero edge angles. The two end slices carry the end fields of the original dipole and are renamed by the addition of "<" and ">" signs to indicate the entry and exit slices, which are optically different to the central slices. When a solenoid is subdivided, the central slices are not given end fields and in the data base they carry the code -999 in columns 16 and 17. The two end slices carry the end fields of the original solenoid and are also renamed by the addition of "<" and ">" signs. Zero-length elements, electric deflectors (not electric bends) and user-defined matrices cannot be subdivided. RF cavities, RFQs, scatterers, helical elements and wigglers cannot be subdivided by this routine. These units have to be referred to the 'Auxiliary Data' menu where they can be treated.

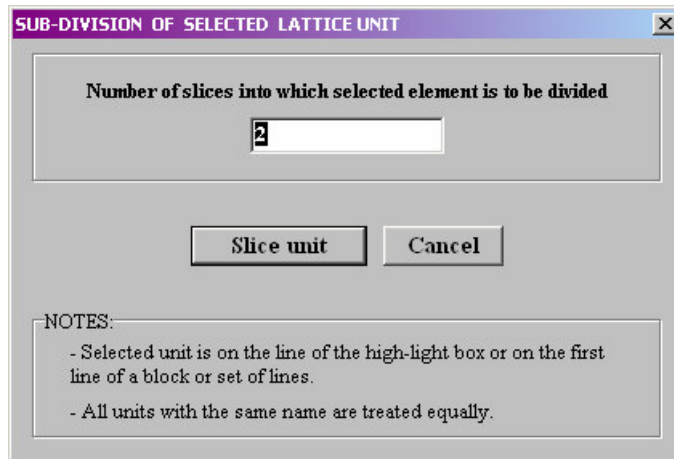


Figure 4.15 Dialogue box for the subdivision of a single unit

4.6.2 Sub-divide all units... (F12)

The sub-division of ALL elements is recommended when calculating space charge. Subdivision is also useful for interpreting results and producing smooth graphs. Some elements can be sub-divided by hand, but it is far safer to use the routines provided. This routine will subdivide all units into slices not exceeding the user-given value, see Figure 4.16. When a dipole with non-zero edge angle is subdivided, the central slices become sector bends with zero edge angles. The two end slices carry the end fields of the original dipole and are renamed by the addition of "<" and ">" signs to indicate the entry and exit slices, which are optically different to the central slices. When a solenoid is subdivided, the central slices are not given end fields and in the data base they carry the code -999 in columns 16 and 17. The two end slices carry the end fields of the original solenoid and are also renamed by the addition of "<" and ">" signs. Zero-length elements, electric deflectors (not electric bends) and user-defined matrices cannot be subdivided. RF cavities, RFQs, scatterers, helical elements and wigglers cannot be subdivided by this routine. These units have to be referred to the 'Auxiliary Data' menu where they can be treated.

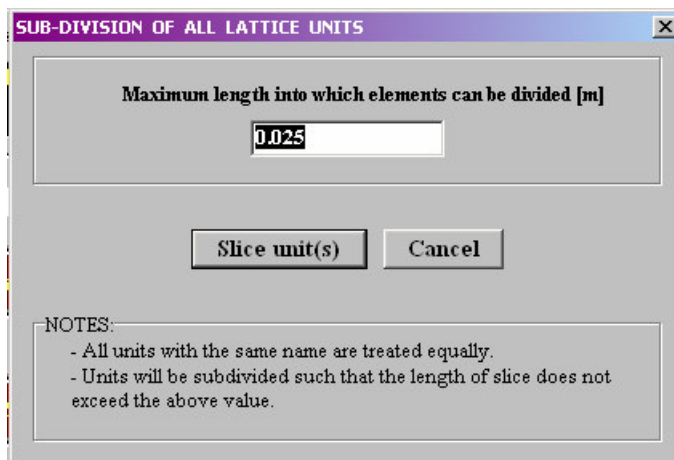


Figure 4.16 Dialogue box for the subdivision of all units

4.6.3 *Combine all divided units*

This command is included to help the user retrieve a compact lattice from one that has been subdivided. The routine does not completely reverse the action of the subdivision routines in Sections 4.6.1 and 4.6.2 because there are some ambiguities. The main differences are that dipoles with edge angles and solenoids will still have separate end sub-units while their central regions are recombined. When a series of sub-units are combined the new unit is renamed by adding '(*n*)' to the name, where *n* is the number of slices that were combined. The recombination of RF cavities, RFQs, scatterers, helical elements and wigglers cannot be done by this routine. These units have to be referred back to the 'Auxiliary Data' menu.

Note that the vacuum chamber of the first sub-unit determines the vacuum chamber for the new combined unit. If, for example, the original unit was subdivided in order to create a special longitudinal profile for the vacuum chamber this information would be lost.

4.6.4 *Import lattice... (F10)*

This command will import a lattice file into the current lattice. After specifying the name and path of the file to be imported, the imported lattice is inserted at the position of the highlighted cell, or at the position of the first highlighted line of a selected block. The highlighted line and all the following lattice is moved down to make space. If a block of lattice was selected, it is NOT deleted at the time of importation. Once the importation is complete, the program leaves a highlighted cell or line at the top of the imported lattice. If the combination of the current and new lattices exceeds the available memory a warning is given and the action can either be aborted or continued. If the action is continued, then that part of the lattice that is displaced downwards and overflows the memory is lost. Auxiliary data belonging to the imported lattice is also lost.

This procedure is very useful, but also potentially dangerous. For example, there is no check on what is imported and it is possible, for example, to import elements with names that already exist, but with different parameters. The present data checking routines will not find this and it is up to the user to control the situation. If same-name elements with different parameters do exist, then the program results will be erratic both during editing and execution. This does have the advantage that it is also possible to "trick" the graph routines into combining graphs from different lattices, but again it is up to the user to control what is happening.

4.7 *Help (F1)*



The Help Menu provides a standard help file

4.8 Other functions

4.8.1 Direct data entry (double-click cell)

Data, whatever it relates to, can be entered or edited directly by double clicking the relevant cell in the lattice-data spreadsheet when in the Edit Window. After double-clicking the cell, an edit box opens above the spreadsheet on the left hand side, see Figure 4.17. As an alternative to double-clicking, select the cell and press the ‘Enter’ key. After data entry, the lattice database is updated and the dialogue box cleared by clicking ‘OK’ or pressing the ‘Enter’ key a second time.

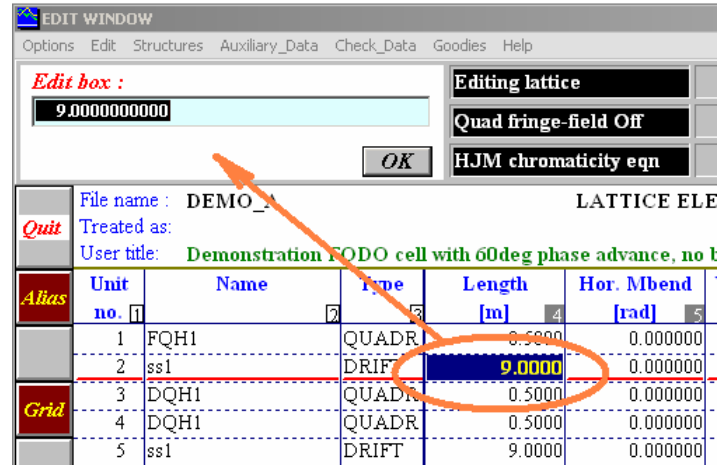


Figure 4.17 Direct entry/editing of data

In some special cases, the “Edit box” has the additional buttons shown in Figure 4.18. This occurs for attributes, such as the vacuum chamber size, that can differ between elements of the same name.

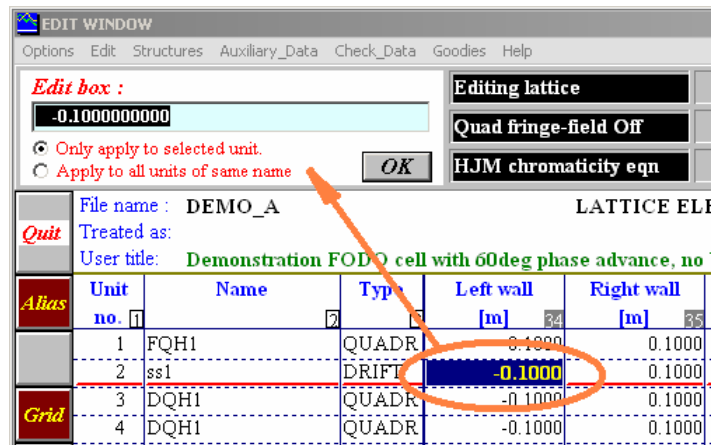


Figure 4.18 Direct entry/editing of data for attributes that can differ between units of the same name

4.8.2 Collected data entry by element

A variant of the direct data entry shown above is to view and edit all the data for one element in a single dialogue box that is called up by double clicking the appropriate row in column 1.

Initially the dialogue box appears as shown in Figure 4.19 (a). At this stage, the dialogue box needs to know the element type in order to configure itself to ask for the correct parameters. If the user enters a type, with or without a name, and clicks the “Configure dialog for element type” button the dialogue will respond by asking for all the relevant parameters. Figure 4.19 (b) shows the example of a sector bend (SBEND). If the user has activated the dialogue box from a line in the spreadsheet with existing data, then the name and type (if they exist) will appear as defaults. It is still necessary to click “Configure” to add or change data. Another possibility is to enter the name of an existing element and to click “Configure”. The routine will then find that element and, if its type is defined, the dialogue will configure accordingly and fill with the existing data. Some very basic checks will be made on the data, but it is still necessary to activate the checking routine in the ‘Check Data’ menu (see Section 4.5) before a lattice can be used for calculations.

Note that an element can be defined with normalised parameters or with fields, but not both. The upper set of radio buttons on the right sets this choice. The lower set of radio buttons applies to those attributes that can differ between elements of the same name. The dialogue box will reconfigure and display messages according to the situation.

(a) Presentation on opening

(b) After clicking ‘Configure’

The element type is needed in order to configure the dialogue box. The type can be entered directly or via the name of an existing element. Click “Configure” to continue.

If the routine recognises the element type, the dialogue box will configure itself to ask for the relevant data. Only very basic checks are made on the input at this stage.

Figure 4.19 Dialogue box for editing all attributes of an element

* * *

Chapter 5 Ring Window

The Ring Window is accessed via the Main Window and is dedicated to the calculation of rings and matched sections, i.e. lattices for which the input and output Twiss parameters are constrained to be equal. Scatterers and RFQs are not allowed in this Window and RF cavities have to be reset with standing buckets. The menu items are described below.

5.1 Options Menu

Options	
Incoherent space charge (coasting 2D) ON...	
Coherent space charge (coasting 2D) ON...	
Space charge OFF...	
Eddy currents (linear ramp) ON...	
Eddy currents OFF...	
Back to Main Window	F9
Quit AGILE	Ctrl+Q

The Options Menu contains functions that set the computational environment.

5.1.1 Incoherent space charge (coasting 2D) ON...

(a) Application to central orbit

It is advisable to subdivide the lattice elements into short lengths. When the routine is first called, it checks the maximum length of the elements and displays this value with a recommended value for the calculation, see Figure 5.1.

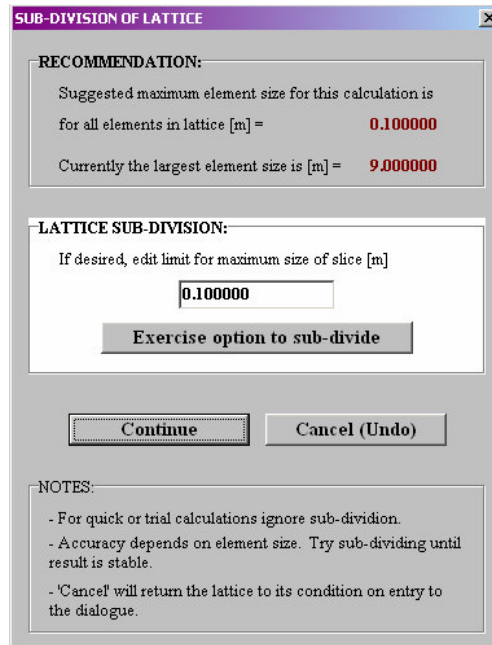


Figure 5.1 First dialogue box for incoherent space charge

For important work, it is prudent to repeat the calculations with different subdivisions of the elements to check if the results are stable. The user may also prefer to perform the sub-division in advance in the Edit Window, see Section 4.6. In this way, a more detailed preparation can be made by subdividing different elements in the lattice to different degrees. If the user is only testing the procedure, then he can omit the sub-division to save time and pass to the next stage. Note that if a sub-divided lattice is saved, the program will prompt the user to supply a new name in order to avoid the original master file being over-written.

INCOHERENT SPACE-CHARGE (2D) FOR A COASTING BEAM...

1. STATUS :
 Current ion is a **proton**
 K.E./nucleon [GeV/u] = **1.0000000**
 Momentum/nucleon [GeV/c/u] = **1.6960277**
To change these parameters please return to the Main Window

2. SELECT DISTRIBUTION:
☒ Uniform 2-D distrib, boundary ellipse at 2 sigma
☐ Elliptic 2-D distrib, boundary ellipse at 2.24 sigma
☐ Parabolic 2-D distrib, boundary ellipse at 2.45 sigma
☐ Gaussian 2-D distrib, boundary ellipse >2.5 sigma

3. SET BEAM CURRENT (or # of particles) :
 EITHER
☒ Beam current [A] = **0.01**
 OR
☐ Number of particles =
For rings put total number of particles e.g. 2.5E10
For lines put number of particles per metre.

4. ENTRY EMITTANCES :
 Horiz. 1 sigma emitt. [pi mm mrad] = **1.0000000**
 Vert. 1 sigma emitt. [pi mm mrad] = **1.0000000**
 Full mom. spread of beam [dp/p] = **0.000000**

5. IMAGE FORCES (optional) :
☒ Include images (electric+magnetic)
☐ Do NOT include images

6. RF CAGE
 Is there an rf cage? ☐ Yes ☒ No
The vacuum chamber and magnet gap sizes in the lattice file will be used to determine the electric and magnetic image forces.

Copy data to notebook Paste time stamp
 Copy data to clipboard Load from notebook
 Continue Cancel

Figure 5.2 Second dialogue box for incoherent space charge parameters

After the sub-division dialogue box has been cleared, the routine displays the dialogue box shown in Figure 5.2. To change the parameters in Section 1 it is necessary to return to the main program. The transverse beam distribution can be set in Section 2 and the beam current in Section 3. As an alternative to the beam current, the total number of ions can be specified for a ring or the number of ions per metre for a line. Bunched beams can be simulated by introducing the peak current. The calculation can include images if desired. The dialogue box will adapt Sections 5 and 6 accordingly. The electric images can either be calculated from the data supplied with the lattice for the vacuum pipe, or alternatively, an RF cage can be specified that is set at a constant distance from the beam, or at a maximum distance that is scaled by the local square roots of the betatron amplitude functions. The dimensions of the RF cage are based on the beam size that is calculated by linearly adding the betatron and momentum contributions. The magnetic images are calculated from the data supplied with the lattice for the magnet

gaps, but if no data exists, warning messages are given. In the case of the vacuum pipe, the default values are usually too large to give reasonable estimates of the images. It is up to the user to check that the program has reasonable values with which to work. The status bar at the top centre-right of the window shows the space-charge regime that is being used, if any. Note that the entries in the dialogue box can be copied to the clipboard and the internal notebook and that they can be pasted back into the dialogue for future calculations.

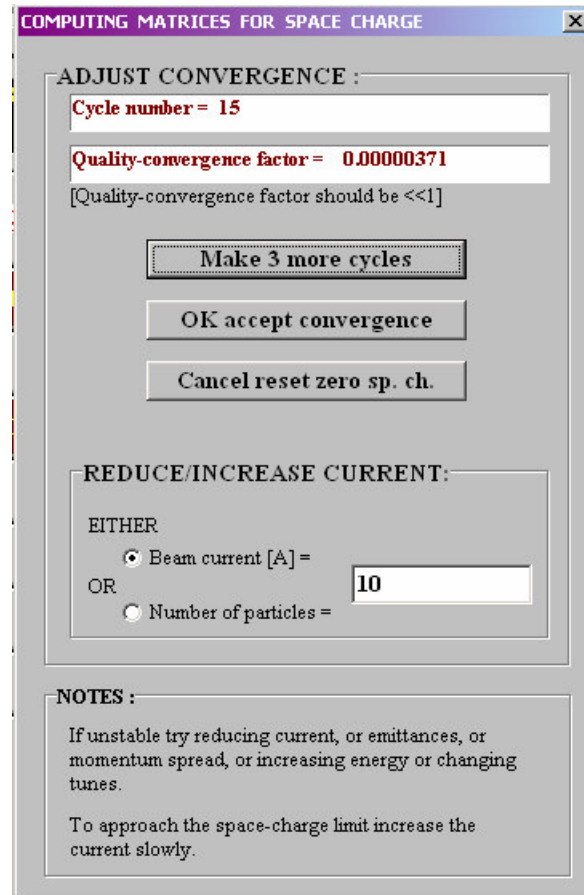


Figure 5.3 Third dialogue box for the convergence of the incoherent space charge calculation

Finally, the program displays an interactive dialogue box, see Figure 5.3, that follows the convergence of the iterations when applying the space charge to the beam. The quality convergence factor should be much less than unity, as shown in Figure 5.3. If the convergence fails, try reducing the current (in the same dialogue box) and then make three more cycles. It is often necessary to approach strong space charge slowly.

Note that the routine is not equipped to handle the effect of synchrotron radiation.

(b) Application to an off-axis orbit

Space charge can also be applied to off-axis orbits, but there are some restrictions:

- *Any subdivision of the lattice must be done before the off-axis orbit is created.*
- *The off-axis orbit must first be created without space charge.*
- *When the space charge is applied to an off-axis orbit images are excluded.*

5.1.2 Incoherent space charge (bunched 3D) ON...

Not available in this version.

5.1.3 Coherent space charge (coasting 2D) ON...

It is advisable to subdivide the lattice elements into short lengths. When the routine is first called, it checks the maximum length of the elements and displays this value with a recommended value for the calculation using the dialogue box in Figure 5.1. This is the same dialogue box as used for the incoherent space charge, except that the recommended subdivision length will be different.

It is prudent to repeat the calculations with different sub-divisions of the elements to check if the results are stable. This is important when an RF cage is present because the cage follows the shape of the beam. However, when the images are formed in the vacuum chamber, a more coarse subdivision that follows the changes in the vacuum chamber is sufficient. This is because, in the latter case, the beam shape is not needed as the image forces are calculated from the line charge of the beam and the routine includes the image forces analytically in finite length elements, NOT as thin-lens kicks. The user may prefer to perform the sub-division in advance in the Edit Window, see Section 4.6. In this way, a more detailed preparation can be made by subdividing different elements in the lattice to different degrees. If the user is only testing the procedure, then he can omit the sub-division to save time and pass to the next stage. Note that if a sub-divided lattice is saved, the program will prompt the user to supply a new name in order to avoid the original master file being over-written.

After the sub-division dialogue box has been cleared, the routine displays the dialogue box in Figure 5.4. This is similar to the dialogue box shown for incoherent space charge except that the beam distributions are now missing, since the images are calculated from the line charge and the transverse distributions have no influence. To change the parameters in Section 1, it is necessary to return to the main program. The beam current can be set in Section 2. As an alternative to the beam current, the total number of ions can be specified for a ring, or the number of ions per metre for a line. Coherent space charge effects are derived from electric and magnetic images. The electric images can either be calculated from the data supplied with the lattice for the vacuum pipe, or alternatively, an RF cage can be specified that is set at a constant distance from the beam, or at a maximum distance that is scaled by the square roots of the betatron amplitude functions. The presence and design of the RF cage are set in Section 3 and the dialogue box will adapt according to the situation. When an RF cage is present, Section 4 will appear and request the beam emittances. Note that the emittances are for

calculating the dimensions of the RF cage and not the beam. The cage size is defined by linearly adding the betatron and momentum contributions. The magnetic images are calculated from the data supplied with the lattice for the magnet gaps, but if no data exists, warning messages are given. The status bar at the top centre-right of the window shows the space-charge regime that is being used, if any. Note that the entries in the dialogue box can be copied to the clipboard and the internal notebook and that they can be pasted back into the dialogue for future calculations.

COHERENT SPACE CHARGE (2D) FOR A COASTING BEAM...

1. STATUS :
 Current ion is a **12C 6+**
 K.E./nucleon [GeV/u] = **1.0000000**
 Momentum/nucleon [GeV/c/u] = **1.6918762**
To change these parameters please return to the Main Window

2. SET BEAM CURRENT (or # of particles) :
 EITHER
☒ Beam current [A] = **0.0000000**
 OR
☐ Number of particles = **0.0000000**
For rings put total number of particles e.g. 2.5E10
For lines put number of particles per metre.

3. RF CAGE
 Is there an RF cage? ☒ Yes ☐ No
 Is cage to beam clearance constant or scaled by root of betatron amplitude.
☒ Constant ☐ Scaled
 Enter gap from beam to RF cage [m] = **0.020000**

4. ENTRY EMITTANCES :
 Horiz. 1 sigma emitt. [pi mm mrad] = **14.00000000**
 Vert. 1 sigma emitt. [pi mm mrad] = **14.00000000**
 Full mom. spread of beam [dp/p] = **0.000000**

Copy data to notebook Paste time stamp
 Copy data to clipboard Load from notebook
 Continue Cancel

Figure 5.4 Second dialogue box for coherent space charge parameters

Finally, the program displays the same interactive dialogue box as used for the incoherent space charge, see Figure 5.3. If the images are formed in the vacuum chambers and the magnet gaps, then the calculation does not need to cycle and arrives directly at the final solution. The beam may be unstable, but the result is final. If the images are formed in an RF cage, then the program needs to cycle until the cage dimensions and the focusing forces are consistent with the beam sizes. This generally converges quicker than in the incoherent case. The quality convergence factor should be much less than unity, as shown in Figure 5.3. If the convergence fails, try reducing the current (in the same dialogue box) and then make three more cycles.

Coherent space charge cannot be applied to off-axis orbits. Note that, unlike the incoherent case, the routine is not affected by synchrotron radiation altering the beam cross-section unless there is an RF cage.

5.1.4 Space charge OFF...

This command switches off all space charge (incoherent and coherent) and sets the display for the non-space-charge transverse Twiss parameters. However, the procedure differs slightly if the current lattice is an off-axis orbit, or if it is the central orbit.

On central orbit, the program also offers the option of restoring the undivided lattice, providing the subdivision was made at the time the space charge was switched on, see Figure 5.5. On an off-axis orbit, the file structure is such that this reversal of the subdivision cannot be carried out at that moment. However, if the machine is returned to the central orbit and providing there have not been too many intermediate actions, the menu item ‘File | Recover original lattice’ will in most cases restore the original undivided lattice

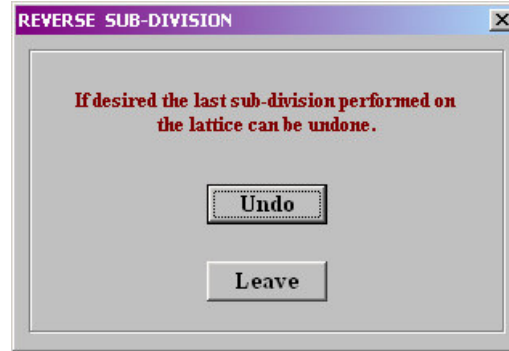


Figure 5.5 Dialogue box for the reversal of the last subdivision of the lattice

5.1.5 Eddy currents (linear ramp) ON...

In machines that ramp, it is often important to evaluate the effects of eddy currents to help decide the materials to be used for vacuum chambers, the thickness of laminations and end plates, the shaping of the ramping functions and the dynamic correction of high-order multipoles. The basic investigation of the time constants of the vacuum chambers and yokes of dipoles and quadrupoles has already been covered in the Main Window in the menu ‘Aids | Eddy current calculator...’, which is described in Section 3.4.5. The present routine uses the eddy current data generated in this way to predict the optical performance of the machine at defined points on the ramp.

The routine first checks that:

- *Eddy current data exists.*
- *The lattice is defined by normalised variables. Elements that are defined by fields would present the ambiguity of whether they should be kept constant or ramped.*
- *No space charge has been applied. Space charge can be applied after the eddy current data has been included in the lattice. This ensures that the quadrupole error is correctly taken into account.*

A dialogue box requesting the details of the ramp is then displayed, see Figure 5.6. Note that the ramp can be defined in terms of beam momentum or beam energy and that the order of the values determines whether the ramp is ‘up’ or ‘down’. Internally, the routine looks at each magnet that has eddy current data and determines the field change on the magnet pole for the defined ramp. The dialogue also requests the point in time during the ramp that the calculation should be made. In most cases, it is sufficient to look at the maximum effects and to choose a point that is a few time constants into the ramp. In this respect, note that field errors depend only on geometry and ramp rate (the steel

permeability is assumed to be constant) and that optically the errors have their greatest effect at low energies.

Figure 5.6 Dialogue box for defining the machine ramp with eddy currents

Once the dialogue box is cleared, it will be seen that the underlying display of the lattice has been updated with thin multipole lenses introduced at the entries and exits of the dipoles and quadrupoles. The additional multipole lenses are powered so as to simulate the optical effects of the eddy currents. The dipoles have sextupole and decapole lenses, while the quadrupoles have quadrupole and octupole lenses. The multipole corrections added by the program can be seen by requesting “Tables | Lattice elements”. There is also a dipole field error that is simulated by the creation of a kick file that introduces dipole kicks at the entry and exit of each dipole. The dipole errors can be viewed by requesting “Tables | Kicks... | Dipole kicks”. The effect of the dipole errors can be seen by running a closed-orbit calculation, see Section 5.3.7. The closed orbit will look similar to a dispersion trajectory, since the dipoles and quadrupoles effectively correspond to different energies due to the different time constants (time lags). In a practical machine, it is relatively easy to synchronise the dipole and quadrupole ramps to remove the fundamental dipole and quadrupole errors. This is done numerically by checking the box in the previous dialogue. Most machines have individually powered sextupoles that can be used to correct the sextupole error. Higher order errors are usually left uncorrected.

Eddy currents can also be applied to off-axis orbits, or eddy currents can be applied and then the off-axis orbit can be created. Combining eddy currents and an off-axis orbit will greatly increase the lattice file size. Any function that cannot be used in a consistent way with eddy currents will be 'greyed'. For example, tracking a distribution of particles is not allowed with eddy currents because the time constants may be of the order of a few revolutions and it is not clear that the program will be doing something

meaningful. However, the way to circumnavigate this restriction is to save the lattice once the eddy current condition is applied. The "Save" function will be "greyed", but the "SaveAs" is available to prompt the use of a new file name, otherwise the original data is lost. The new file will be a simple lattice with the multipole lenses that simulate the eddy current condition. To get the full eddy current effect, the dipole kick file must also be saved. The use of this file will not be restricted, but it is up to the user to check if what he does is meaningful.

Note that while eddy currents are active the kick file containing the dipole errors is protected and all editing of kick files is prohibited.

Warning Do not subdivide units when calculating eddy currents. Sub-division enforces non-physical constraints on the eddy current distribution.

5.1.6 Eddy currents OFF...

Switches the effects of eddy currents off and returns to the original, on-axis, non-space-charge lattice condition.

5.1.7 Back to the Main Window (F9)

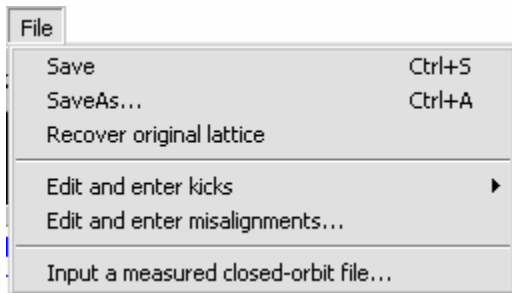
This command returns to the Main Window.

5.1.8 Quit AGILE (Ctrl+Q)

Before the program terminates, temporary files are deleted and the user is offered the possibility of saving the current lattice file and notebook, if this has not been done recently.

5.2 File Menu

The File Menu contains some basic file utilities



5.2.1 Save (Ctrl+S)

This routine saves the current lattice automatically into the current lattice file. If there is no current lattice file (e.g. when creating a new lattice) the 'SaveAs' function is called. The standard extension for lattice files is 'LAT'.

5.2.2 SaveAs... (Ctrl+A)

This routine opens a dialogue box requesting the file name and path for saving the current lattice. The option is also given of saving the lattice in the format of an earlier version via the combo-box labelled 'Save as type'. Saving with an earlier format may cause some information to be lost. Element types that are not recognised by the earlier

version are replaced by drift spaces. It may also be that certain auxiliary information such as beam characteristics will not be stored by the earlier versions. The normal extension for a lattice file is 'LAT'.

5.2.3 *Recover original lattice*

This command recovers and loads a lattice held in a backup file. This lattice always corresponds to central orbit without space charge or eddy currents, as saved from before the last major change.

5.2.4 *Edit and enter kicks*

This command activates a sub-menu with four items: the first item concerns point dipole kicks [rad], the second thin-lens quadrupole kicks [T], the third thin-lens skew quadrupole kicks [T] and the fourth point momentum kicks [dp/p]. Many of the commands in the Kick Window, see Chapter 9, use the lattice structure, for example for the addition of errors to dipoles with a specified name. A kick file can be stored on disk and recalled later. When a kick file is reloaded a check is made that the number of lattice elements corresponds to the current lattice in memory and that the type of kick is that expected. The kick file does not contain lattice element names, so apart from these simple checks, it is up to the user to be sure that the kick file is the correct one. A kick file normally has the extension 'KCK'.

5.2.5 *Edit and enter misalignments...*

This command calls the Misalignment Window for the creation or editing of element misalignment files (see Chapter 9). Misalignment files are similar in their handling to kick files. They normally have the extension 'MIS'.

5.2.6 *Import a measured closed-orbit file...*

This routine imports closed-orbit files that list the beam positions at the beam monitors only (the lattice must contain the corresponding number of elements of the types HPU, HVPU and VPU). These orbits can then be corrected in the closed-orbit correction routine in the closed-orbit plot window, see Section 8.5. The input file should be formatted so that:

- *It contains only the orbit readings for one plane. Whether the plane is the horizontal or the vertical is requested by the dialog box for reading the file.*
- *The readings must be in beam order, expressed in [m] and must be separated by space(s) and/or non-numeric characters. For example 0.001 0.002 0.003 0.000 -0.0001 etc. No special termination is required, but all monitors in the relevant plane must have a reading to maintain the sequence. If a reading is missing then add 0.0. The readings can be edited or removed later.*
- *If the file does not fit the number of monitors in the lattice, then a warning message is given.*

The input of the file is initiated by the dialogue box shown in Figure 5.7. Once the file is loaded and 'Continue' is clicked the program displays the readings in the closed-orbit plot window as a bar chart. The orbit can then be corrected, see Section 8.5.

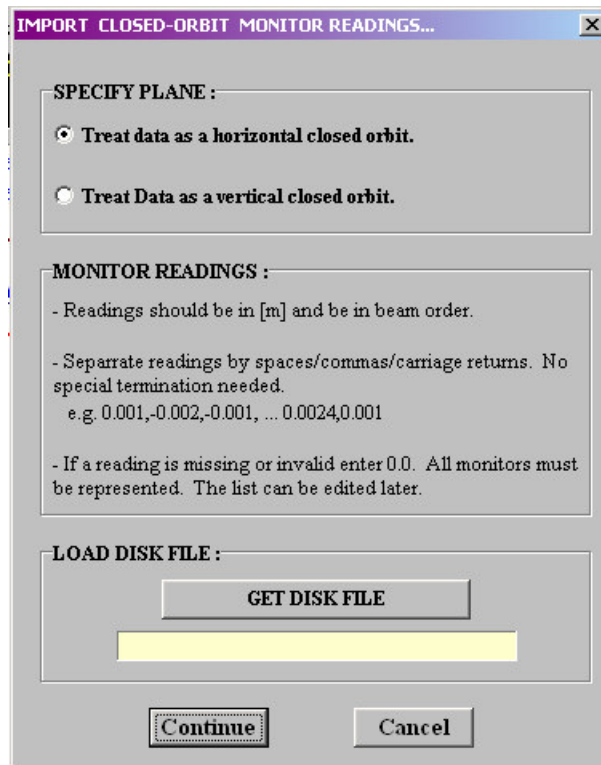
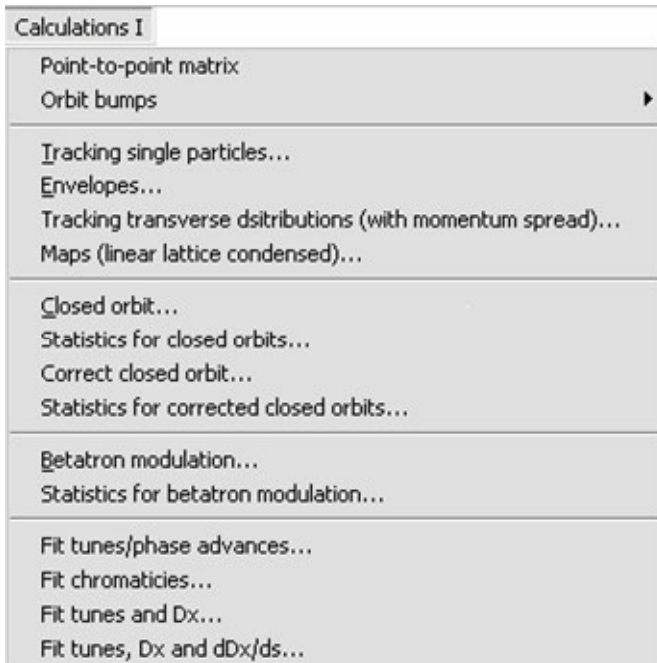


Figure 5.7 Dialogue box for the input of measured closed orbits

5.3 Calculations I



The Calculations I Menu contains the most frequently used routines for a ring.

5.3.1 Point-to-point matrix

The initial dialogue box (not shown) asks for the index of the element at whose entry the matrix is to start and the index of the element at whose exit the matrix is to end. If a block of cells is highlighted on the screen, then the entry and exits lines of the block will be offered by the routine as defaults. A typical result is shown in Figure 5.8.

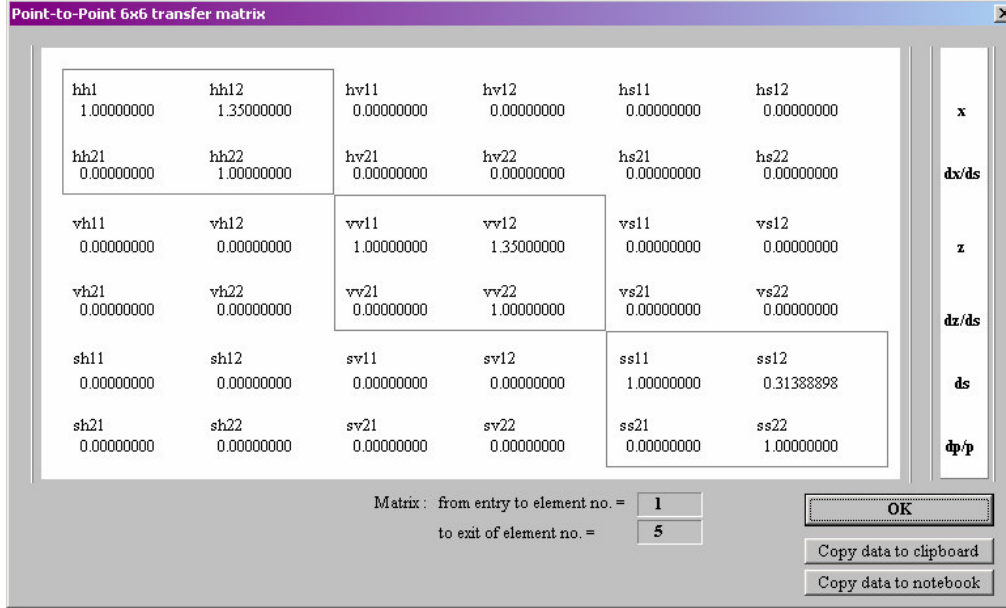


Figure 5.8 Dialogue box for showing point-to-point 6×6 matrices

If the matrix includes space charge forces, this is indicated on the display and if the matrix is a scatterer a warning is given that only a drift space matrix is shown. Similarly, sextupoles appear as drift-space matrices and other high-order nonlinear lenses are given 1:1 matrices. The effect on the beam of these non-linear elements is represented in the lattice by point kicks.

The routine can calculate across the ‘first-last’ element boundary for rings and matched sections. Matrices can be copied to the clipboard and/or notebook.

5.3.2 Orbit bumps

This menu item leads to a sub-menu containing routines for calculating ‘3-magnet’, ‘4-magnet’ and ‘5-magnet’ bumps. Figure 5.9 shows the dialogue box for the 3-magnet bump, which is typical of the group.

3-MAGNET BUMP

1. CHOOSE ENVIRONMENT :

☒ Delete existing kicks ☐ Add bump to existing kicks

2. CHOOSE PLANE :

☒ Horizontal ☐ Vertical

3. ENTER DATA :

Positions of kicks :

1st kick at entry to element number =

2nd kick at entry to element number =

3rd kick at entry to element number =

Reference Point :

At entry to element number =

Amplitude of bump [m] =

[If when calculated bump has zero amplitude at reference point, then 1st kick will be set to 0.01 radian.]

NOTES :

- Beam envelopes can be added later.
- Bumps are calculated with linear optics.
- Any momentum kicks and/or misalignments are switched off.

Figure 5.9 Dialogue box for designing closed-orbit bumps

The routine calculates the dipole kicks needed at user-specified positions to create a closed-orbit bump with the specified characteristics. The user has the choice of creating the bump in the horizontal or vertical plane and whether the bump kicks should be added to the existing dipole kicks, or whether the existing kicks should be deleted first. Warnings are given if the supplied data is, in some way, unusable. The routine works with linear optics (e.g. the presence of sextupole components is ignored).

The bump routines can be applied to on-axis, off-axis and distorted orbits in rings and transfer lines. These routines are however strictly transverse and the longitudinal plane is ignored. For rings and matched sections, the entry index can be greater than the exit index. In this case, the routine operates across the "first-last" element boundary. The numerical results are shown in the main spreadsheet display. The bump trajectories can also be viewed in the closed-orbit plots window that can be opened via the Graphs menu, or by clicking the 'Graph' button on the right hand side of the window.

5.3.3 Tracking single particles...

The particle tracking routine can take into account all effects known to the program and calculate with the highest precision possible within the capabilities of the program. The user must specify a starting point in the lattice and the particle position in six dimensional phase space, see Figure 5.10. He is also asked to define whether dipole kicks, misalignments and momentum kicks should be included. The number of turns that can be stored is limited by the memory, but the range can be effectively increased by specifying a number of turns to be performed before recording the values. If there are no RF elements in the lattice, then ignore the lead/lag parameter. The fractional momentum deviation is with respect to the momentum of the equilibrium orbit about the particle is oscillating, which may be the original central closed orbit or an off-axis closed orbit, if this has been calculated. Note that the input data for the dialogue box can be copied to the clip board and the notebook and that this data can be reloaded into the dialogue box on later occasions.

Item 6 in the dialogue box offers the option of tracking the longitudinal motion with all non-linearities included, or using transfer matrices that are only valid within the linear region of the RF bucket. The inclusion of the linear matrix routine allows the user to check the degree of non-linearity in the longitudinal motion.

DATA INPUT FOR SINGLE-PARTICLE TRACKING

1. ENTER TRACKING DATA:

Start tracking at entry to element no. =

Horizontal position, x [m] =

Horizontal slope, dx/ds =

Vertical position, z [m] =

Vertical slope, dz/ds =

Lead/lag wrt synchronous particle [m] =

Fractional momentum deviation from equilibrium orbit, dp/p =

2. DIPOLE KICKS: [To edit dipole kicks, exit dialog, open File menu and Edit kicks...]

☒ Track without kicks

☐ Track with current kicks

3. MOMENTUM KICKS: [To edit momentum kicks, exit dialog, open File menu and Edit kicks...]

☒ Track without kicks

☐ Track with current kicks

4. MISALIGNMENTS: [To edit misalignments, exit dialog, open File menu and Edit misalignments...]

☒ Track without misalignments

☐ Track with current misalignments

5. MULTI-TURN:

Lattice is a closed ring. Memory can store 31 turns

Enter no. of first turn to be stored Enter no. of turns to be stored

6. OPTIONS FOR LONGITUDINAL TRACKING

☒ Exact tracking with non-linear module.

☐ Fast tracking, valid only in linear region of rf bucket.

NOTES:

- Beam envelopes can be calculated after tracking.
- Dipole kicks, momentum kicks and misalignments can all co-exist.

Buttons: Continue, Cancel, Copy data to notebook, Paste time stamp, Copy data to clipboard, Load from notebook

Figure 5.10 Dialogue box for single-particle tracking

The numerical results are shown in the main spreadsheet display and the trajectories can be viewed in the transverse and longitudinal trajectory graphs, which can be opened via the Graphs menu, or by clicking the 'Graph' button on the right hand side of the window.

5.3.4 Envelopes...

The beam envelope routine uses linear optics to calculate the transverse and longitudinal beam boundaries. This is meant mainly for deciding vacuum chamber apertures and bunch lengths. If a track exists in the memory from an earlier calculation, the user can opt to base the beam envelopes on this track, see Figure 5.11. If he decides not to do this, or if there is no track in the memory, the envelopes are based directly on the current equilibrium orbit as defined by the dispersion function. Envelopes can also be calculated on off-axis and distorted orbits as described in Sections 5.4.1 and 5.5.4. Note that if stationary RF buckets are set, then momentum offsets are excluded.

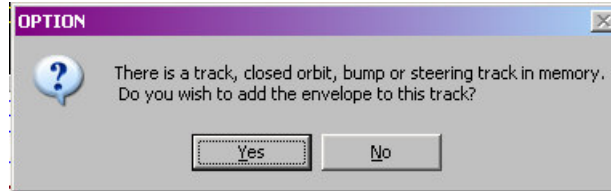


Figure 5.11 Dialogue box for choosing the base orbit

Once the reference orbit for the envelope has been chosen, the routine presents a second dialogue box, see Figure 5.12, requesting data for the construction of the envelopes. This entails specifying the incoming 1-sigma emittances, the number of sigma that define the edge of the beam (this can be different in the two planes) and the full momentum spread of the beam. If the envelopes are to be added to a reference trajectory, then the program assumes that this track contains any effects due to a momentum deviation. If a reference trajectory is not to be used, then the user is also asked to give the mean deviation in momentum with respect to the central orbit. In the presence of coupling, the incoming emittances are assumed to be uncoupled.

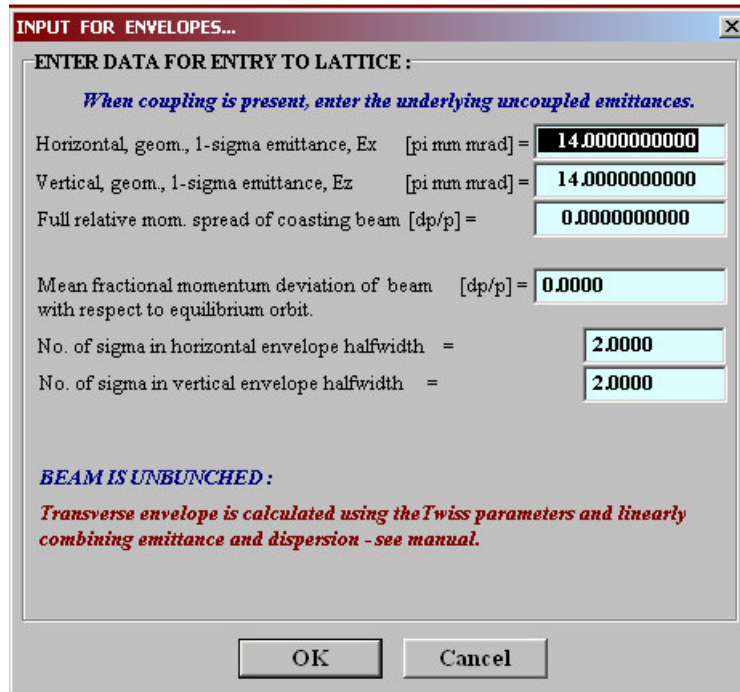


Figure 5.12 Dialogue box for setting the envelope parameters

The envelope routine works in different ways according to the situation and signals the choices taken by setting the appropriate messages in the dialogue box:

- ***Uncoupled lattices with unbunched beams.*** The normal Twiss parameters are used to calculate the betatron and momentum envelopes. These two contributions are added linearly to the reference track to provide the maximum, or worst-case, values. In transfer lines, scattering is incorporated into the Twiss parameters.
- ***Uncoupled lattices with bunched beams.*** Special bunched-beam, pseudo Twiss parameters are used to express the transverse beam size; as is done in programs such as TRACE 3D. The pseudo Twiss functions combine the betatron and momentum contributions. The method used for merging the two contributions is described in the physics manual.
- ***Transversely coupled lattices with unbunched beams.*** Sigma matrices are used with the coupled dispersion functions to calculate the betatron and momentum envelopes. These two contributions are added linearly to the reference track to provide the maximum, or worst-case, values. In transfer lines, scattering is incorporated into the sigma matrix.
- ***Transversely coupled lattices with bunched.*** Special bunched-beam, pseudo Twiss parameters are used to express the transverse beam size. The pseudo Twiss functions combine contributions from sigma matrices and the coupled dispersion functions. The method used for merging the two contributions is described in the physics manual.
- ***Longitudinal envelopes for bunched beams*** are obtained from the longitudinal Twiss parameters. The envelopes correspond to the length of the bunch for the given emittance, the momentum spread in the bunch and the time lead/lag of the head and tail of the bunch.

The spreadsheet display tabulates the transverse beam envelopes in metres. In each plane, there are columns for the overall envelope, the momentum envelope and the trajectory upon which the envelopes are based. For bunched beams the columns for the momentum envelope are marked 'Included' since the betatron envelope and the momentum envelope are considered jointly.

The transverse beam envelopes can also be viewed in the transverse beam envelopes window with or without the vacuum chamber wall in the two planes versus distance, or as cross-sections with the vacuum chamber in real space or in phase space. The latter can be made into an animated display that travels along the lattice. The longitudinal bunch parameters are given in metres, nanoseconds and keV and graphical displays can be found in the longitudinal beam envelopes window. Both graphics windows can be opened via the Graphs menu, or by the 'Graph' button on the right hand side of the window, but in this case the status of the 'Trans/Long' button on the left hand side of the window determines which graphics window will be opened.

5.3.5 Tracking transverse distributions (with momentum spread)...

This routine tracks any distribution of up to 10'000 particles around the machine with an animated display showing the beam and the vacuum chamber in real space or in transverse phase space in real or normalised co-ordinates. It is also possible to add self-space-charge forces (but not images) while tracking. The location and magnitude of particle losses are recorded along the lattice, see Section 8.10. Longitudinal effects are not included in these displays. The detailed simulation of what happens to the beam in a defined vacuum aperture makes this routine useful for practical problems such as multi-turn injection and internal beam dumping.

The user must first complete the dialogue box shown in Figure 5.13 in order to set the environment to be used with respect to the addition of field errors and misalignments. Once this dialogue box has been cleared, the program automatically opens the transverse distribution tracking window, see Section 8.10.

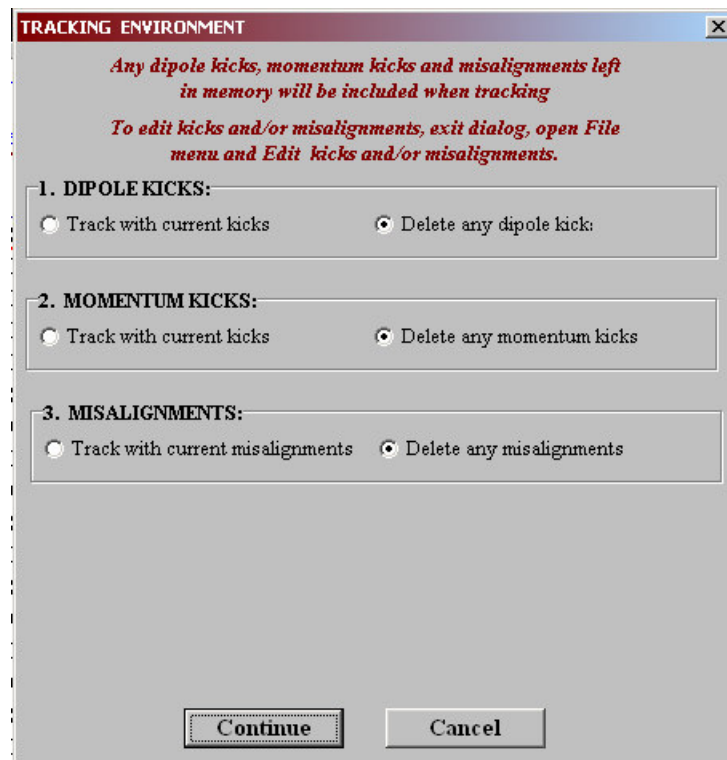


Figure 5.13 Dialogue box for the environment for tracking particle distributions

5.3.6 Maps (linear lattice condensed)...

This routine is only applicable to closed rings. The user specifies his observation point and the routine builds a phase-space map of single-particle or multi-particle beams at that point. The map routine differs in a number of ways from the tracking of particle distributions. The detailed interaction with the vacuum chamber around the machine is removed (apart from at the point of observation) and the linear regions in the lattice are condensed for speed of computation. The simplification of the single-turn matrix also removes the possibility of adding field errors and misalignments

and applying space charge. Consequently, this routine is better suited to studies such as resonances, resonant extraction and dynamic aperture.

The user must first complete the dialogue box shown in Figure 5.14 in order to set the observation point and the level of precision to be applied. Normal precision is sufficient in most cases and high precision is best kept for final calculations of sensitive parameters, such as the area of the last stable phase-space triangle in an extraction scheme. High precision is approximately equivalent in terms of accuracy to the use of second-order matrices. However, a different technique is used that first performs a dipole tracking to establish an off-momentum orbit and then constructs a new magnetic lattice that fits the geometrical data of this orbit, see Section 5.4.1. In this way, the focusing about the off-axis orbit is represented to a better precision. Consequently, the high-precision option costs heavily in computation time if there is a wide range of momenta in the particle distribution. Note that high precision cannot be applied to lattices that excite coupling. Once the dialogue box has been cleared, the program automatically opens the transverse map window, see Section 8.11.

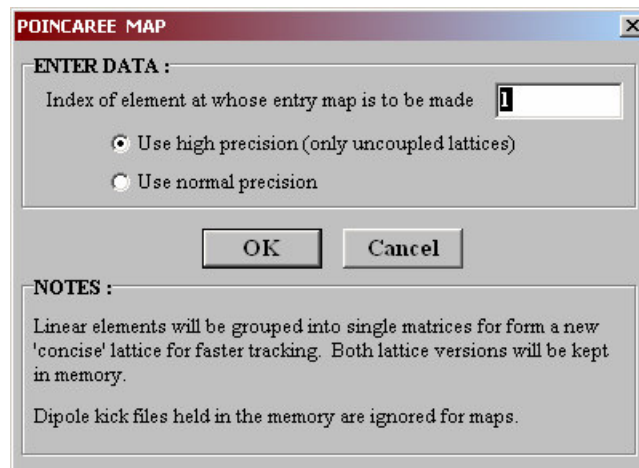


Figure 5.14 Dialogue box for the environment for plotting maps

Tip: For speed, make sure that all particles of the same momentum are either loaded or created together. If the input sequence jumps back and forth between two or more different momenta, the calculation time is increased.

5.3.7 Closed orbit...

This routine calculates the expected closed-orbit distortion in a machine due to a one or more point dipole kick errors. The orbit distortion is calculated as a perturbation to the ideal machine. The linear paraxial equation for doing this relies on the betatron amplitudes and phases not being changed significantly by the addition of the dipole errors. The routine is approximate, but has the advantage of being fast and ideal for creating statistics for the closed-orbit prognosis in a new machine, see Section 5.3.8.

The user must first complete the dialogue box shown in Figure 5.15 in order to establish the data file containing the field errors. The option of editing or preparing a new data set takes the user directly to the Kick File Editing Window, see Chapter 9. Upon leaving the kick window or upon clearing the dialogue box, the numerical results

are shown in the main spreadsheet display and the closed orbit can be viewed in the closed-orbit plots window, see Section 8.5. The closed-orbit plots window can be opened via the Graphs menu, or by clicking the 'Graph' button on the right hand side of the window. If a high-precision closed orbit is required, then the 'Distorted Orbit' routine should be used in preference, see Section 5.4.4.

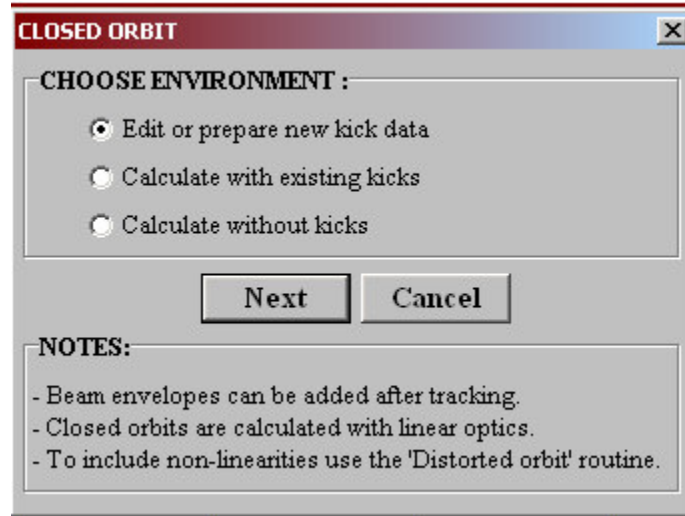


Figure 5.15 Dialogue box for launching a closed-orbit calculation

5.3.8 Statistics for closed orbits...

An important task for machine designers is to determine the tolerances required for magnet alignment, field quality and power converter precision. The sensitivity of the closed orbit to such errors provides one way of looking at these problems. The present routine makes a statistical analysis of up to 1000 closed orbits generated with random dipole errors that are in turn generated from random misalignments and random powering errors.

Figure 5.16 shows the dialogue box that requests the basic information needed for the statistics calculation. The user can opt to use kick and misalignment errors already held in memory or to create new errors. The next stage is to specify the number of orbits to be generated and whether the generated data (peak-to-peak amplitude distribution, mean amplitude and root mean square amplitude) is to be for the whole machine, or for the beam excursions at one particular point.

When the dialogue box is cleared, the routine will either start calculating (if the user has chosen to use existing errors), or the routine will open the Kick Window (see Chapter 9) for the creation of new errors. When creating a data set, use the functions offered in the Kick Window to set up generators for systematic and random errors e.g. shift all dipoles randomly in the axial direction with a distribution with an rms of 2 mm. The program 'remembers' the choices and the rms values given and then generates different sets of data for the statistical analysis. Both random and systematic errors can be included, but there must be at least one random error. Do not enter individual errors by hand. Any errors entered in this way will be ignored after the first calculation. Also, only enter an error of a given type with a given destination once, e.g. a random dipole

kick with an rms of 0.001 Tm to be applied to all sextupoles of the family named SF1. If the same family is given a second error of the same type, the two errors are summed on the screen and are taken into account for single calculations, but for statistics the routine applies only the error that was entered most recently.

The maximum number of calculations for a statistical analysis is 1000, but it is recommended that a test run is made with say 20 calculations.

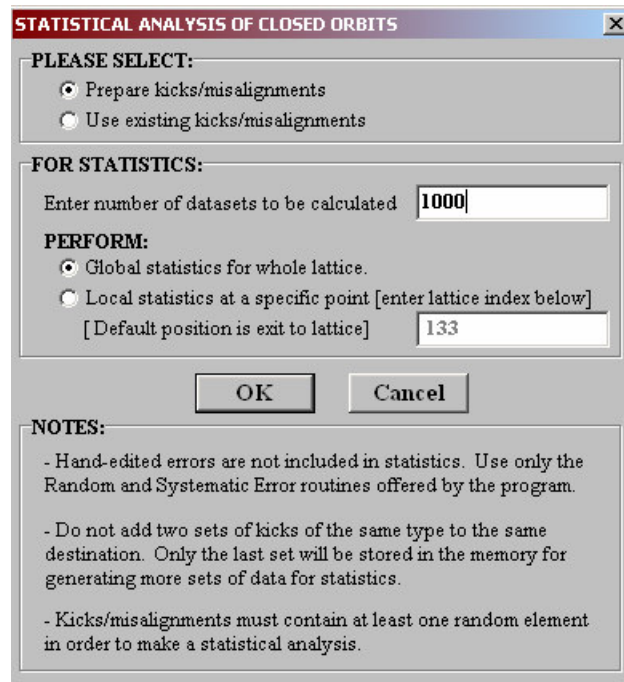


Figure 5.16 Dialogue box for preparing a closed-orbit statistics calculation

Once the errors have been created in the Kick Window, click 'Back to prior window' in the 'Options Menu'. The calculation of the closed orbits then commences. This may take an appreciable time according to the size of the lattice, the extent of the errors and the power of the computer. A bar graph will record the progress and, if the user wishes, he can terminate the calculation at any time by pressing the right-hand mouse button. In this event, the routine will stop and present the results for whatever number of machines has been calculated up to that point.

Figure 5.17 shows an example from an analysis of the horizontal closed-orbit errors created by random axial shifts of the main lattice dipoles. Note that since no vertical kicks are generated by axial shifts, there is no distortion in the vertical plane. Consequently, all the vertical orbits are represented by a single bar in the first bin of the lower histogram. Along the bottom of the window there are several buttons for adjusting the display:

- **Absol./Pk-Pk** Toggles between the values for the maximum peak-peak, average peak-peak and the rms peak-peak and the values for the absolute maximum excursion, average of the absolute excursion and rms of the absolute excursions.
- **Off (V)/On (V)/On (HV)** Toggles between horizontal display, vertical display and both displays.

- **+v div**. Increases the number of vertical divisions in the graphs.
- **-v div** Decreases the number of vertical divisions in the graphs.
- **+ bin** Increases the number of bins into which the closed orbits are put.
- **-bin** Decreases the number of bins into which the closed orbits are put.
- The window is also equipped with the usual buttons for printing, writing graphic files, modifying screen colours, opening the on-line help and returning to the prior window.

When the graph window is closed, the statistics routine ends and the screen display shows the Twiss functions. However, the last closed orbit that was calculated will still be available in numerical form in “Tables | Closed orbits, in graphical form in “Graphs | Closed orbits” and the kicks will be in “Tables | Kicks... | Dipole kicks”.

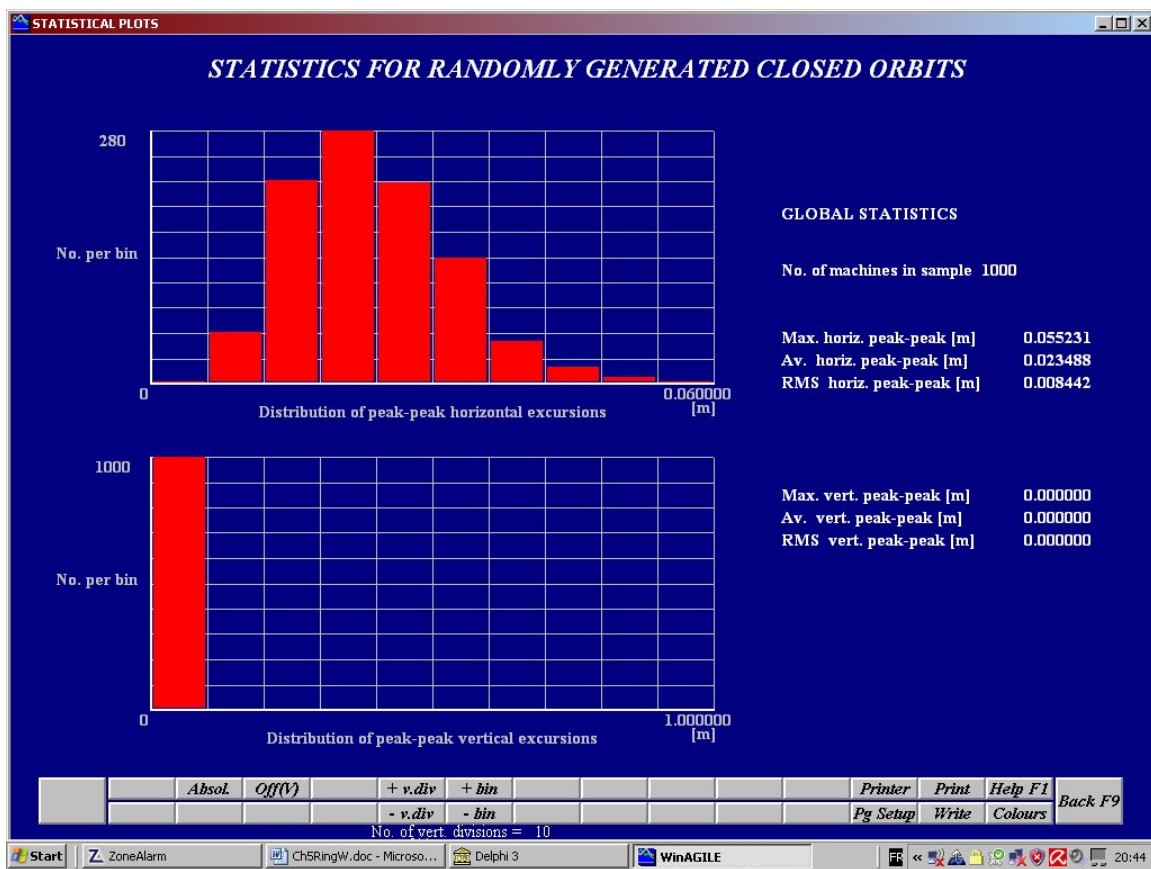


Figure 5.17 Graph window showing closed-orbit, statistics results

5.3.9 Correct closed orbit...

The user will be advised if there is no closed orbit stored in memory and briefly how to create a distorted closed orbit or import one. If a closed orbit already exists in memory the user is requested to display this orbit in the graph mode and then to use the correction button in the graph window, see Section 8.5.

5.3.10 *Statistics for corrected closed orbits...*

Another important task for machine designers is to determine how well the proposed layouts of correctors and beam position monitors will perform when correcting typical closed orbits. The present routine makes statistical analyses before and after the correction of up to 1000 closed orbits generated by random dipole errors that are in turn generated from random misalignments and random powering errors.

Figure 5.18 shows the dialogue box that requests the basic information needed for the statistical analyses. The left hand side of the dialogue box deals with the parameters for the correction routine and the right hand side deals with the parameters for creating the distorted orbits:

- In the top left group box, define the plane that is to be treated by checking the appropriate button.
- The next stage is to create lists of the correctors and beam monitors that are to be used. At an advanced design stage, the correctors and monitors would already be in the lattice labelled as element types HCORR, VCORR, HVCOR, HPU, VPU and HVPU, but if this not the case, the program has a backup mode that uses all the entry points to quadrupoles as possible positions for correctors and monitors. In the next two group boxes, check the buttons for either the backup mode, or the use of elements defined in the lattice. In the example in Figure 5.18, there is the choice between monitors in the lattice and the backup mode, but for correctors only the backup mode is offered because the correctors have not yet been placed in the lattice.
- In the backup mode, there are almost certainly more elements available than are really needed and when using installed elements, it may be of interest to switch some elements off to see how well the system can cope. For these reasons, it is possible to edit the lists of correctors and monitors by clicking the button 'Edit monitor and/or corrector lists' midway down the left hand side of the dialogue box. In this case, a new dialogue box will appear that is described at the end of this Section. In addition to simply removing correctors and monitors, there are also other options for use in special situations.
- Once the monitor and corrector lists are created, the next group box offers the user the possibility of setting a limit on the number of correctors to be used out of the list of available elements. It is good practice to limit the number of correctors. This helps to avoid extreme situations where a very large number of power converters are trying to stabilise currents close to zero, or several correctors oppose each other with large kicks to get a small residual effect. It also means that the operational reliability is improved.
- At the bottom of the left hand side of the dialogue box, there is a group box that offers three numerical methods for calculating the orbit correction.
- At the top of the right hand side of the dialogue box, there is a group box for either setting up new random errors for creating the distorted orbits, or for accepting the errors set up in an earlier run.
- The next group box on the right hand side sets the number of orbits to be treated (max. 1000) and whether the generated data (peak-to-peak amplitude distribution, mean amplitude and root mean square amplitude) is to be for the whole machine, or for the beam excursions at one particular point.
- Finally, click 'Compute' to continue or 'Cancel' to quit.

STATISTICAL ANALYSES OF CLOSED-ORBITS BEFORE AND AFTER CORRECTION

CHOOSE PLANE :

☒ Correct HORIZONTAL plane
☐ Correct VERTICAL plane

PLEASE SELECT:

☒ Prepare dipole error kicks
☐ Use existing dipole error kicks

CHOOSE MONITOR OPTION :

☒ Use elements of types HPU, VPU, HVPU
☐ Use entries to quadrupoles for positions (backup mode)

FOR STATISTICS:

Enter number of datasets to be calculated

PERFORM:

☒ Global statistics for whole lattice.
☐ Local statistics at a specific point [enter lattice index below]
[Default position is exit to lattice]

CHOOSE CORRECTOR OPTION :

☐ Use elements of types HCORR, VCORR, HVCOR
☒ Use entries to quadrupoles for positions (backup mode)

NO. OF CORRECTORS TO BE USED :

Enter number of correctors to be used

Total no. of monitors available =
Total no. of correctors available =

COMPUTATIONAL METHOD :

☒ Least Squares Fit + Gauss-Jordan (fast)
☐ Singular Value Decomposition Fit (safe)
☐ Least Squares Fit + Householder Trans. + Tri-diag.

NOTES:

- Hand-edited errors are not included in statistics. Use only the Random and Systematic Error routines offered by the program.
- Do not add two sets of kicks of the same type to the same destination. Only the last set will be stored in the memory for generating more sets of data for statistics.
- Kicks/misalignments must contain at least one random element in order to make a statistical analysis.
- A maximum of 1000 randomly generated orbits can be treated.

Figure 5.18 Dialogue box for preparing closed-orbit statistics before and after correction

On clicking 'Compute', the program will either start calculating and correcting closed orbits (if the user has chosen to use existing errors), or the program will open the Kick Window, see Chapter 9, for the creation of new errors. When creating new errors, use the functions offered in the Kick Window to set up generators for systematic and random errors e.g. shift all dipoles randomly in the axial direction with a distribution with an rms of 2 mm. The program remembers the choices and the rms values given and then generates different sets of data for the statistical analysis. Both random and systematic errors can be included, but there must be at least one random error. Do not enter individual errors by hand. Any errors entered in this way will be ignored after the first calculation. Also, only enter an error of a given type with a given destination once, e.g. a random dipole kick with an rms of 0.001 Tm to be applied to all sextupoles of the family named SF1. If the same family is given a second error of the same type, the two errors are summed on the screen and are taken into account for single calculations, but for statistics the routine applies only the error that was entered most recently. Once the parameters have been set to generate errors, press F9 or click 'Return to Prior Window' in the 'Option Menu'. The calculation then commences. This may take an appreciable time according to the size of the lattice, the extent of the errors and the number of correctors to be used. A bar graph will record the progress and, if the user wishes, he can terminate the calculation at any time by pressing the right-hand mouse button. In this event, the routine will stop and present the results for whatever number of machines has been calculated up to that point.

Figure 5.19 shows an example from an analysis of the horizontal closed-orbit errors created by random transverse shifts of the main quadrupoles. The upper histogram shows parameters for the uncorrected orbits and the lower histogram shows the same

parameters for the corrected orbits. Note that although 1000 machines were requested only 837 appear in the analysis. This is because 163 machines were either unstable (close to a resonance) or the correction computation broke down numerically. Along the bottom of the window there are several buttons for adjusting the display. When the graph window is closed, the screen display reverts to showing the Twiss functions. However, the last closed orbit that was calculated will still be available in numerical form in “Tables | Closed orbits, in graphical form in “Graphs | Closed orbits” and the kicks will be in “Tables | Kicks... | Dipole kicks”.

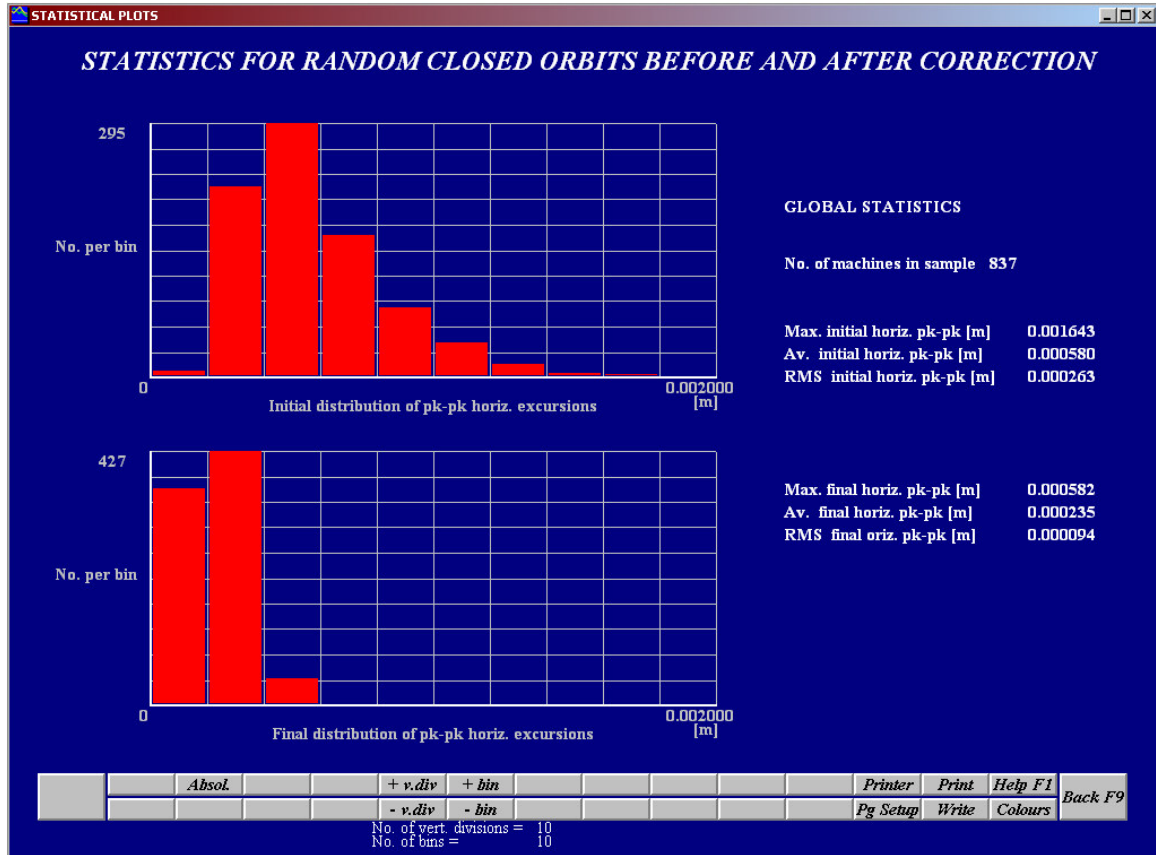


Figure 5.19 Graph window showing statistical results for the uncorrected and corrected closed orbits

Figure 5.20 shows the additional dialogue box, mentioned earlier, for editing the lists of available beam monitors and correctors. The example chosen corresponds to Figure 5.18 with installed beam monitors (on the left), but the backup mode for correctors (on the right). The editing functions are:

- To remove either a monitor or a corrector, highlight the unit and click the relevant ‘Remove’ button. The unit is then labelled “Removed” in the list.
- Monitor readings can be forcibly set to zero by highlighting and clicking the ‘Zero-Set’ button. Two monitors have been treated in this way in the example. This means that their reading (i.e. the distortion) is considered as zero and the correction routine then corrects only that part of the orbit that has non-zeroed monitors.

- Correctors can be discounted and then removed using the ‘Discount-Rem’ button. This means that the effect of any kick presently in the corrector is numerically removed from the measured orbit and, in addition, the corrector is not used for the correction. In this way, a local bump can be maintained with its kicks unchanged while the orbit is corrected.
- Correctors can also be discounted as above, but then still used for the correction. This means that the effect of any kick presently in the corrector is numerically removed from the measured orbit, as above, except that in this case the corrector is still considered for the correction process.
- To reverse any of the above actions, repeat the same procedure.

EDIT VERTICAL PICKUPS AND CORRECTORS

USING QUADRUPOLE POSITIONS FOR CORRECTORS

EDIT BEAM MONITORS :

8	HV pickup	
14	HV pickup	
20	HV pickup	Removed
27	HV pickup	
33	HV pickup	
39	HV pickup	
44	HV pickup	
50	HV pickup	
58	HV pickup	Removed
62	HV pickup	
68	HV pickup	
74	HV pickup	
82	HV pickup	
88	HV pickup	
94	HV pickup	Removed
100	HV pickup	
106	HV pickup	
112	HV pickup	
117	HV pickup	
123	HV pickup	Removed
131	HV pickup	
140	HV pickup	SetZero
146	HV pickup	SetZero
152	HV pickup	

REMOVE/RESTORE monitor
REMOVE

ZERO-SET/RESTORE monitor, so that region is not corrected
ZERO-SET

Note that buttons toggle status

EDIT ORBIT CORRECTORS :

7	QF (1)	Removed
9	QF (1)	
13	QD	Removed
15	QD	
19	QF (1)	Removed
21	QF (1)	
26	QF (2)	Removed
28	QF (2)	
32	QD	Removed
34	QD	
38	QF (2)	
40	QF (2)	Removed
43	QF (2)	
45	QF (2)	Removed
49	QD	
51	QD	Removed
57	QF (2)	
59	QF (2)	Dis-Rem
61	QF (1)	Dis-Rem
63	QF (1)	
67	QD	
69	QD	Dis-Use
73	QF (1)	Removed
75	QF (1)	

REMOVE/RESTORE corrector
REMOVE

DISCOUNT effect of current kick and then REMOVE or RESTORE
DISCOUNT-REM

DISCOUNT effect of current kick, but then USE for correction or RESTORE
DISCOUNT-USE

Note that buttons toggle status

NOTES :

- To correct a local region, ZERO-SET the monitors that are outside that region.
- Use DISCOUNT-REM or DISCOUNT-USE to correct an orbit while leaving a local bump in place.
- DISCOUNT-REM discounts the effect of the current kick and then REMOVES the corrector.
- DISCOUNT-USE discounts the effect of the current kick, but allows the corrector to be used.

OK
Cancel

Figure 5.20 Dialogue box for editing the lists of beam monitors and correctors available for use

5.3.11 Betatron modulation...

This routine calculates the changes in the betatron amplitude functions due to a given distribution of gradient errors. The routine opens with the dialogue box shown in Figure 5.21. The user can opt for using an existing gradient error distribution or for creating a new distribution. The results are given in tabulated form and show the variables β , $\Delta\beta$, $\Delta\beta/\beta$ and ΔQ or $\Delta\mu$ (for transfer lines). The calculation is exact using the matrices of the kicks (not a perturbation equation). However, since the calculation is made for the point kicks given in the kick file, it will differ slightly from distributed errors calculated by editing the gradients of long units directly.

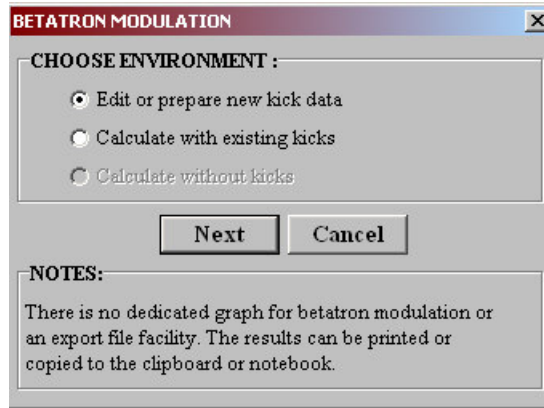


Figure 5.21 Dialogue box launching a betatron-modulation calculation

5.3.12 Statistics for betatron modulation...

This routine makes a statistical analysis of a lattice with up to 1000 randomly generated quadrupole error distributions that are in turn generated from random misalignments and random powering errors.

Figure 5.22 shows the dialogue box that requests the basic information needed for the statistics calculation. The user can opt to use kick and misalignment errors already held in memory, or to create new errors. The next stage is to specify the number of error sets to be generated and whether the analysis is to be for the whole machine, or the changes in the betatron amplitude at one particular point.

When the dialogue box is cleared, the routine will either start calculating (if the user has chosen to use existing errors), or the routine will open the Kick Window, see Chapter 9, for the creation of new errors. When creating a new data set, use the functions offered in the Kick Window to set up generators for systematic and random errors. The program remembers the choices and the rms values given and then generates different sets of data for the statistical analysis. Both random and systematic errors can be included, but there must be at least one random error. Do not enter individual errors by hand. Any errors entered in this way will be ignored after the first calculation. Also, only enter an error of a given type with a given destination once. If the same family of magnets is given a second error of the same type, the two errors are summed on the screen and are taken into account for single calculations, but for statistics the routine applies only the error that was entered most recently. Once the errors have been created in the Kick Window, click 'Back to prior window' in the 'Option Menu'. The calculation then commences. This may take an appreciable time according to the size of the lattice, the extent of the errors and the power of the computer. A bar graph will record the progress and, if the user wishes, he can terminate the calculation at any time by pressing the right-hand mouse button. In this event, the routine will stop and present the results for whatever number of machines has been calculated up to that point.

The display of the statistical data is very similar to that for closed-orbit distortion illustrated in Figure 5.17. When the graph window is closed, the statistics routine ends and the screen display shows the normal Twiss functions.

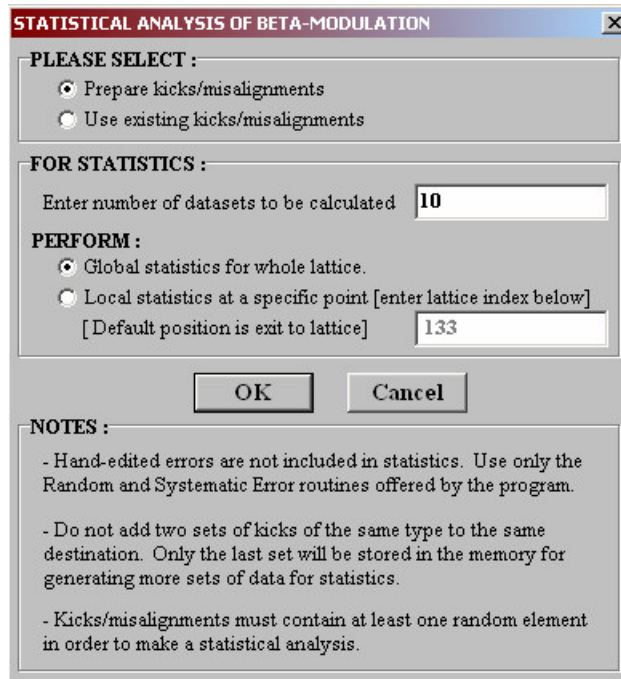


Figure 5.22 Dialogue box for launching a statistical analysis of betatron-modulation

5.3.13 *Fit tunes/phase advances...*

With respect to many parameters, lattices can be remarkably linear. For example, closed-orbit distortion versus dipole kick strength remains quasi-linear over the aperture in a typical accelerator. Changes in phase advance, or tune, versus quadrupole strength are also quasi-linear over the working range of most accelerators. Similarly, chromaticity is quasi-linear with sextupole strength. In fact, the solution of two simple linear equations iterated two or three times is all that is needed to set tune values and chromaticity to a high accuracy. In WinAGILE, this is called '*fitting*' to distinguish this simple technique from numerical matching.

This fitting routine launches the dialogue box shown in Figure 5.23 for adjusting the tunes of a ring, or the phase advances in a matched cell or transfer line, to specified values. The user needs to enter the tunes or phase advances desired (the dialogue box adapts automatically to the situation) and then to select two units with gradient variables for the routine to adjust. These units can be chosen from the list box, or specified explicitly in the edit boxes below the list box. If the two units are degenerate (i.e. have the same effect, or are just the same unit entered twice), then a best fit is made with the first variable.

The routine will anticipate unstable situations in transfer lines, when the betatron amplitude functions start to increase exponentially, and give a warning message. If a ring becomes unstable during fitting, then try reducing the step sizes or changing the sign of the steps. Fitting is an iterative process and can be re-launched if the result is not accurate enough. When fitting with space charge, try reducing the step size below the program default values and seeing whether the result is unchanged. Space charge fitting takes longer because the routine adds additional cycles to stabilise the result.

Tip: To force a single-parameter best fit, enter the same unit twice in the dialog box.

2-PARAMETER FIT FOR UNCOUPLED TUNES

1. ENTER DESIRED TUNES:
Horizontal, Qx =
Vertical, Qz =

2. CHOOSE FROM INVENTORY:
EITHER click on unit in inventory
OR enter directly below the indices and increments to be used

7	QF (1)
11	MB
13	QD
26	QF (2)

3. UNITS/SERIES AND STEP-SIZES TO BE USED:
Index of 1st unit or unit in 1st series =
Index of 2nd unit or unit 2nd series =
Step-size for 1st unit/series =
Step-size for 2nd unit/series =

(Re-) Compute

STATUS:

	Qx	Qz
Desired values =	1.680000	1.720000
Obtained values =	1.680001	1.720000
Residual errors =	0.000001	-0.000000

OK - accept new values Reset original values

NOTES:
- To force a "best fit" with one variable only, enter the same index for both units.
- The inventory box shows only named units, but any unit with a gradient can be used by entering its index directly.

Figure 5.23 Dialogue box for fitting tunes values or phase advances in a lattice

5.3.14 Fit chromaticities...

Figure 5.24 shows the dialogue box for fitting the chromaticities using two sextupole gradients. If the two units are degenerate (i.e. have the same effect, or are just the same unit), then a best fit is made with the first variable. In closed rings, the dialogue box refers to tune rather than phase advance. To force a single-parameter best fit, enter the same unit twice.

2-PARAMETER FIT FOR CHROMATICITIES

1. ENTER DESIRED CHROMATICITIES:

Horizontal, $dQ_x/dp/p$ =
-3.1

Vertical, $dQ_z/dp/p$ =
-1.0

2. CHOOSE FROM INVENTORY:

EITHER click on unit in inventory
OR enter directly below the indices and increments to be used

7 QF (1)
11 MB
13 QD
24 XC (D)
26 QF (2)
53 XC (F)
79 XR (Reson)

3. UNITS/SERIES AND STEP-SIZES TO BE USED:

Index of 1st unit or unit in 1st series =
24

Index of 2nd unit or unit in 2nd series =
53

Step-size for 1st unit/series =
0.017854

Step-size for 2nd unit/series =
0.010000

(Re-) Compute

STATUS:

	$dQ_x/dp/p$	$dQ_z/dp/p$
Desired values =	-3.100000	-1.000000
Obtained values =	-3.100000	-1.000000
Residual errors =	-0.000000	-0.000000

OK - accept new values
Reset original values

NOTES:

- To force a "best fit" with one variable only, enter the same index for both units.
- The inventory box shows only named units, but any unit with a sextupole gradient can be used by entering its index directly.

Figure 5.24 Dialogue box for chromaticities in a lattice

5.3.15 Fit tunes and D_x ...

The two-parameter fitting used in the previous two sections can be extended to a three-parameter fitting to control the tunes and the dispersion at a single point in a ring. Figure 5.25 shows the three-parameter fitting dialogue box that enables the user to specify the desired values and to choose the three units with quadrupole gradients to be used by the. The three units must not be degenerate (i.e. have the same effect).

This routine can be used to explore the tuning range of a machine while maintaining some control over the dispersion, for example by keeping the dispersion zero in a dispersion-free region. It can also be used to create the so-called Terwilliger scheme that introduces zero dispersion regions periodically around the machine. The fitting is an

iterative process and can be re-launched if the result is not accurate enough. Note that this routine is more sensitive than the previous two routines.

3-PARAMETER FIT FOR HORIZONTAL DISPERSION AND TUNES

ENTER DESIRED TUNES AND DISPERSION:

Horizontal, Q_x = Index of position for D_x =
 Vertical, Q_z = Horiz. dispersion, D_x =

CHOOSE FROM INVENTORY:

EITHER click on unit in inventory
 OR enter directly below the indices and increments to be used

7
 11
 13
 26

UNITS/SERIES AND STEP-SIZES TO BE USED:

Index of 1st unit or unit in 1st series = Step-size for 1st unit/series =
 Index of 2nd unit or unit in 2nd series = Step-size for 2nd unit/series =
 Index of 3rd unit or unit in 3rd series = Step-size for 3rd unit/series =

(Re-) Compute

STATUS:

	Q_x	Q_z	D_x
Desired values =	1.620000	1.720000	0.100000
Obtained values =	1.620000	1.720000	0.100000
Residual errors =	0.000000	-0.000000	-0.000000

OK - accept new values Reset original values

Figure 5.25 Dialogue box for fitting tunes and the dispersion function at a single point

5.3.16 Fit tunes, D_x and $dD_x/ds...$

This routine extends the previous section to four-parameter fitting. The dialogue box is similar to Figure 5.25, except that the user is also asked to supply the desired derivative of the dispersion and an additional gradient variable.

This routine can be used to explore the tuning range of a machine while maintaining a more complete control over the optics than allowed by the routine in the previous section. It can also be used to create the so-called Terwilliger scheme that introduces zero dispersion regions periodically around the machine. The fitting is an iterative process and can be re-launched if the result is not accurate enough. This routine is the most sensitive of the fitting routines presented so far.

5.4 Calculations II

Calculations II
Create Twiss on off-axis, closed orbit...
Reset central orbit
Reset Twiss on a stored, off-axis, closed orbit...
Create Twiss on a distorted, closed orbit...
Reset undistorted central orbit
Linear coupling, driving terms, single or summed...
Statistics for linear coupling...
Resonance driving terms (2nd-5th order), single or summed...
Fit coupling, difference and/or sum resonance, single or summed, with or without 'Twiss' tune control...
Decouple single-turn matrix...
Decouple single-turn matrix with 'Teng' tune control...

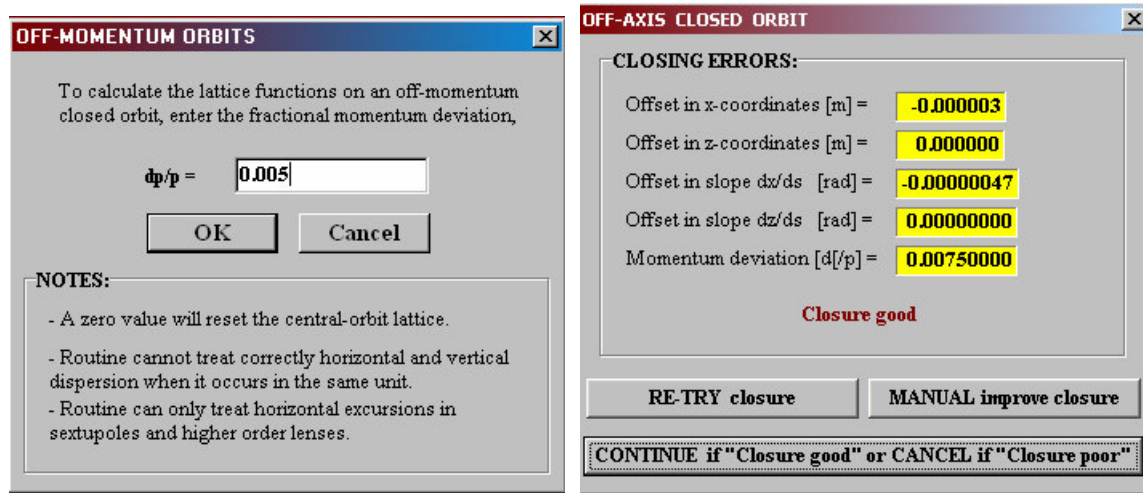
The Calculations II Menu contains some less-frequently used routines for a ring.

5.4.1 Create Twiss on off-axis, closed orbit...

The philosophy of WinAGILE is to calculate special conditions such as off-axis orbits on line. In the case of an off-axis orbit, the program first performs a dipole tracking to find the geometry of the orbit. Based on this orbit, a new sequence of elements is created. For example, all quadrupoles are changed to sector bends (type=SBEND) with a dipole field matching the bending angle, edge angles matching the geometry of the trajectory and gradients modified for the momentum change. Additional thin lenses are also introduced to better describe the action of thin lenses with the off-axis beam. This makes it possible to calculate the focusing on the off-axis orbit more precisely than is possible by developing the fields in power series about the original central orbit. The program simply regards the sequence of modified elements as a new lattice and calculates the Twiss functions on the 'new' central orbit. The user will find certain routines are "greyed" because they may not respond quite as expected. For example, coherent space charge is not allowed off-axis, because the program does not have the off-axis image coefficients. These restrictions can be circumnavigated by saving the lattice. A lattice saved in this way has all the elements renamed by concatenating the original name with the element number to create new individual names, since in most cases the elements will have different parameters. It is up to the user to ensure that the new lattice is used in a meaningful manner.

In principle, off-axis orbits can be calculated in this way with the same precision as the original central orbit. This is not quite the case, because the new lattice will, in general, be a much more complicated lattice than the original one and there is also the question of the accuracy of the dipole tracking. Dipole tracking is, however, intrinsically more accurate than off-axis focusing calculations, so there is a net gain in applying this method. To be prudent, WinAGILE limits off-axis orbit calculations to $\pm 4\%$. Eight off-axis orbits can be calculated and stored in addition to the original central orbit.

The user is requested to supply the momentum deviation of the off-axis orbit to be created, see Figure 5.26 (a). The routine then searches for the off-axis closed orbit and presents the search results in the dialogue box shown in Figure 5.25 (b).



(a) Set dp/p

(b) Handle closure

Figure 5.26 Dialogue boxes for calculating an off-axis orbit

The routine will judge the quality of the off-axis orbit on the basis of the closing errors that are listed in the dialogue box. If the closure is rated as ‘poor’, the user can click on the ‘RE-TRY closure’ button and, if this is ineffectual or diverges, the ‘MANUAL improve’ routine can be tried. The manual routine allows the user to input seed values for the search. In difficult cases, it may be necessary to proceed in steps, gradually approaching the final orbit. If the closure is still rated as ‘poor’, quit the routine by clicking the ‘CONTINUE’ button. A message will say the orbit could not be found and the machine will be reset to the original central orbit. If, however, the closure is rated as ‘good’, click the ‘CONTINUE’ button and the routine will display the trajectory of the off-axis orbit with respect to the original central orbit. This is a temporary display. If a record of the trajectory is required, it must be printed before selecting a new action such as calculating the Twiss parameters. Subsequent calculations and displays are based on the newly-created lattice, which can be saved and considered, in most respects, as an entirely new machine. Resetting the original central orbit, or a new off-axis orbit, is treated in the following sections.

5.4.2 *Reset central orbit*

Resets the original central orbit upon which the off-axis orbit was based.

5.4.3 *Reset Twiss on a stored, off-axis, closed orbit...*

The program can store and recall up to 8 off-axis equilibrium orbits in addition to the original central orbit. The data for these orbits is held in memory and can also be used for plotting the *working line* in the *tune diagram*. Figure 5.27 shows the dialogue box for selecting and setting one of the stored orbits and Figure 5.28 shows an example of a working line. The plot is made automatically by clicking on ‘Tune diagram’ in the

Graphs menu. If no off-axis orbits exist then the central orbit values, known as the *working point* is plotted.

SELECT EQUILIBRIUM ORBIT TO BE RE-RUN

SELECT EQUILIBRIUM ORBIT TO BE SET :

	dp/p	Entry x	Entry dx/ds	Entry z	Entry dz/ds	x-closure	dx/ds-closure	z-closure	dz/ds-closure	Qx	Qz
<input type="radio"/> Central orbit (1)	0.000000	0.000000	0.00000000	0.000000	0.00000000	0.000000	0.00000000	0.000000	0.00000000	3.21000	3.17999
<input type="radio"/> Off-axis (2)	-0.005000	-0.009595	0.00000000	0.000000	0.00000000	-0.000002	-0.00000028	0.000000	0.00000000	3.22674	3.20080
<input checked="" type="radio"/> Off-axis (3)	-0.010000	-0.019043	0.00000000	0.000000	0.00000000	-0.000004	-0.00000046	0.000000	0.00000000	3.24376	3.22280
<input type="radio"/> Off-axis (4)	-0.015000	-0.028344	0.00000000	0.000000	0.00000000	-0.000004	-0.00000040	0.000000	0.00000000	3.26108	3.24596
<input type="radio"/> Off-axis (5)	0.005000	0.009754	-0.00000000	0.000000	0.00000000	-0.000005	-0.00000078	0.000000	0.00000000	3.19355	3.16040
<input type="radio"/> Off-axis (6)	0.010000	0.019648	-0.00000000	0.000000	0.00000000	-0.000003	-0.00000059	0.000000	0.00000000	3.17736	3.14206
<input type="radio"/> Off-axis (7)	0.015000	0.029674	-0.00000000	0.000000	0.00000000	0.000005	0.00000104	0.000000	0.00000000	3.16145	3.12498
<input type="radio"/> Off-axis (8)	No data	No data	No data	No data	No data	No data	No data	No data	No data	No data	No data
<input type="radio"/> Off-axis (9)	No data	No data	No data	No data	No data	No data	No data	No data	No data	No data	No data

Index of currently set active orbit is : 7

Data at entry to element number 1

Copy to clipboard Copy to notebook OK Cancel

Figure 5.27 Dialogue box for resetting the original central orbit or a stored off-axis orbit

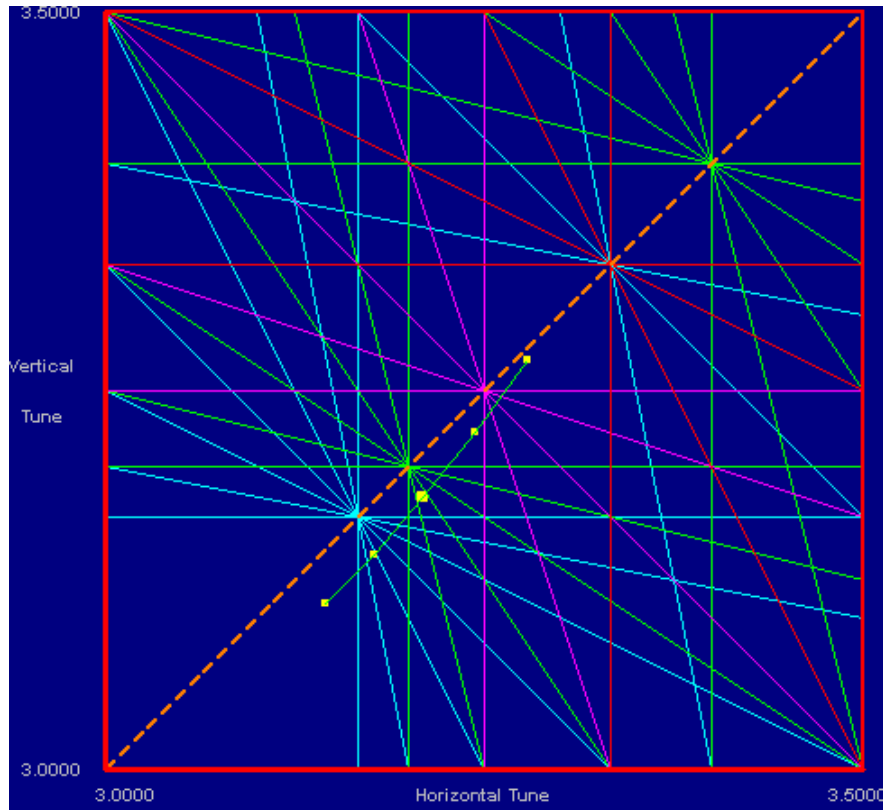


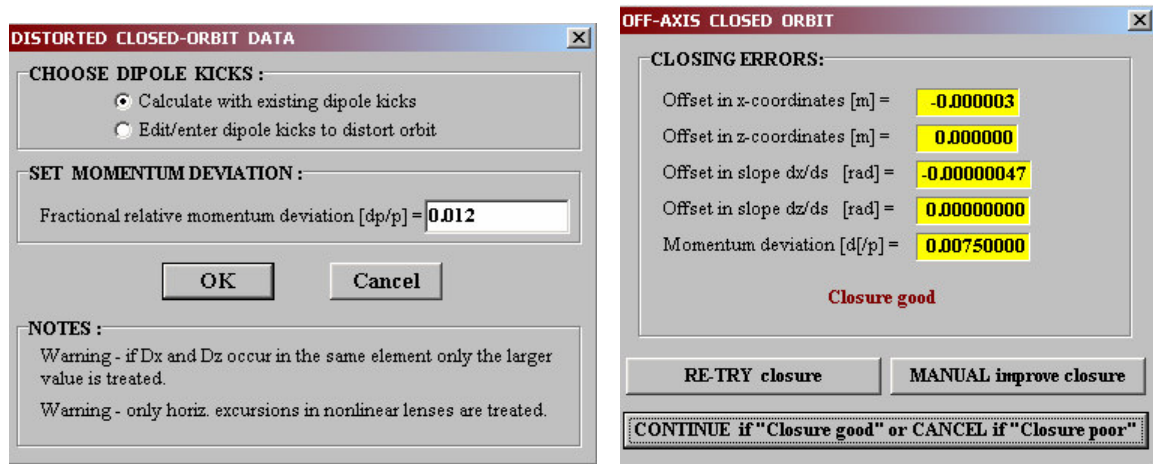
Figure 5.28 An example of a working line plot in the tune diagram

5.4.4 Create Twiss on a distorted closed orbit...

This routine can be used to obtain accurate Twiss values on a strongly-distorted closed orbit that may also be an off-momentum orbit. Typically, this situation might arise for an injection orbit that is off-momentum and has a large amplitude bump to take the beam passed a septum magnet. Another application would be to produce a high-

precision closed-orbit bump by taking into account the distortion of the betatron amplitude functions caused by the bump itself.

Figure 5.29 (a) shows the opening dialogue that requests the user to either create the dipole kicks causing the distortion, or to use the dipole kicks currently in memory, and to enter the momentum deviation, which can of course be zero. The routine then searches for the distorted closed orbit and presents the search results in the dialogue box shown in Figure 5.29 (b). How well the orbit is closed and how the subsequent results are presented are the same as described in Section 5.4.1.



(a) Set closed-orbit distortion and dp/p

(b) Handle closure

Figure 5.29 Dialogue boxes for calculating an off-axis orbit

5.4.5 *Reset undistorted central orbit*

Resets the original central orbit upon which the distorted orbit was based.

5.4.6 *Linear coupling, driving terms, single or summed...*

For accurate coupling calculations, it is advisable to sub-divide the skew quadrupoles and solenoids into slices of 5 cm or less, according to the situation. When the linear coupling routine is first called, it checks the maximum length of these elements and displays this value with a recommended value for the calculation, see Figure 5.30 (a). The user then has the possibility of sub-dividing the lattice before entering the main routine. If the user is only testing, he can ignore the sub-division to save time. For important work, it is prudent to repeat the coupling calculations with different sub-divisions of the elements to check if the results are stable. If preferred, the sub-division can be performed much earlier in the 'Goodies' Menu in the Edit Window. This has the advantage that elements with small betatron amplitude functions can be divided more finely than those with large betatron amplitude functions.

After closing the sub-division dialogue, a second dialogue box is opened that requires the user to choose between the options of: Delete current skew quadrupoles kicks, Calculate with current skew quadrupole kicks or Edit skew quadrupole kicks before calculating, see Figure 5.30 (b).

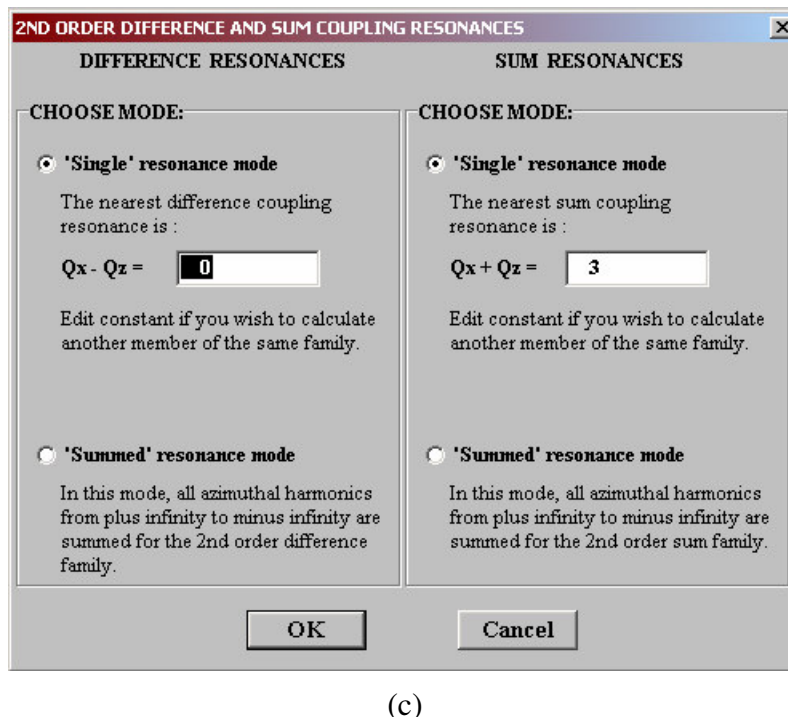
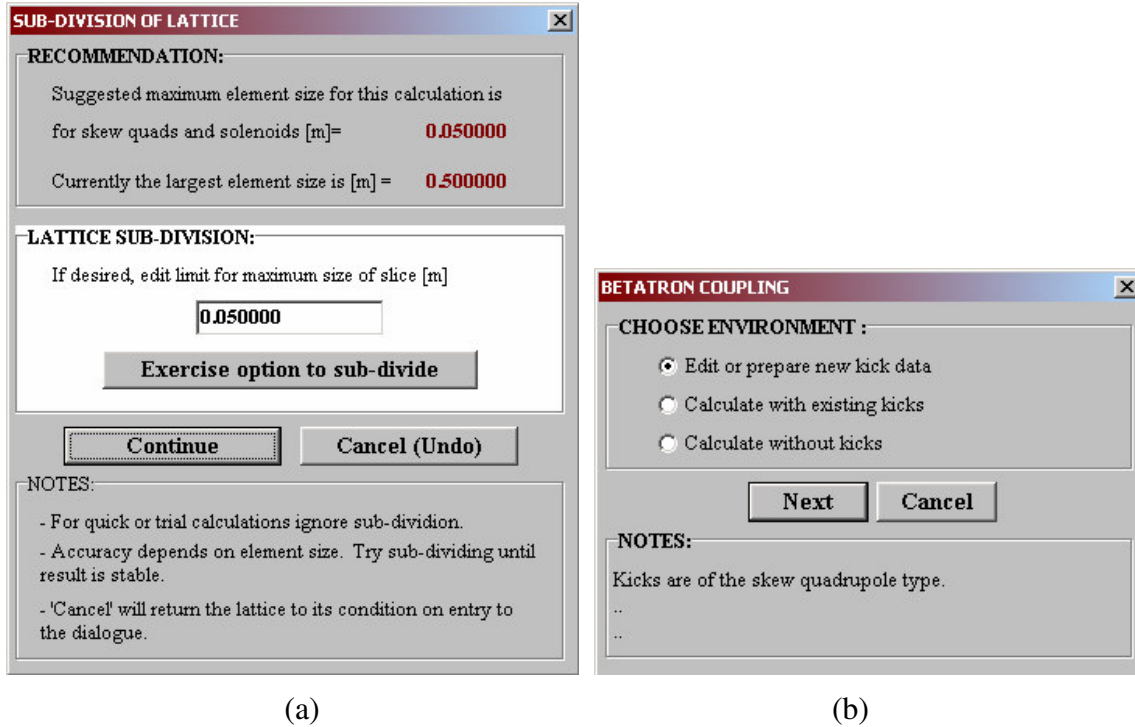


Figure 5.30 Dialogue boxes for preparing a coupling calculation

After closing the kick dialogue, a third dialogue box is opened that offers the user several options concerning the driving terms to be calculated, see Figure 5.30(c). A brief description of the options is given below, but a more detailed analysis can be found in the physics manual.

- **Single resonances** In this case, the routine calculates the real and imaginary parts of the coupling coefficients and associated parameters for a specific difference resonance ($Q_x - Q_z = p$) and a specific sum resonance ($Q_x + Q_z = q$), from the skew quadrupoles, the skew quadrupole kicks and the solenoids in the machine: The program proposes the values of p and q for the two resonances that are closest to the current tune values, but the user can override these choices.
- **Summed resonances** In this case, the routine calculates the real and imaginary parts of the coupling coefficients and associated parameters for the combined action of all resonances in the family, i.e. the routine sums over the azimuthal harmonics p and q from minus infinity to plus infinity.

Directly after the calculation, the display shows the coupling coefficients for the difference resonance(s), followed by the coupling coefficients for the sum resonance(s) and some other parameters such as the betatron and dispersion tilts.

The results can be viewed graphically as a coupling vector in the complex plane, see Section 8.12. To compensate a resonance, it is sufficient to add coupling vectors until the vector diagram gives a closed figure. This is done automatically by the fitting routines described in Section 5.4.9. Efficient corrections are recognised by being compact diagrams with minimum-sized vectors.

The distribution of skew quadrupoles and solenoids will, in general, excite an infinite number of sum and difference coupling resonances. The driving terms of all these resonances can be found by editing the integers p and q , the Fourier harmonics of the magnetic error distribution. If all the resonances are compensated, then the single-turn matrix should have zero off-axis (i.e. coupling) coefficients, but, in practice, the values are only quasi-zero. If only one or two resonances are compensated, then there will be, in general, substantial residual off-axis coefficients. These terms represent the effects of the driving terms of the more distant resonances. Other numerical methods for decoupling the single-turn matrix are described in Sections 5.4.10 and 5.4.11.

Note that once the lattice has been sub-divided for a coupling calculation, it remains sub-divided after leaving the routine. If the user tries to save the new lattice, the program forces a 'SaveAs' to prompt the user to supply a new name in order that the original master file is not over-written.

5.4.7 *Statistics for linear coupling...*

This routine makes a statistical analysis of up to 1000 cases of a machine with random coupling errors. Sub-division of the lattice is not offered by this routine because it may lead to excessive calculation times. However, the user can circumvent this restriction by splitting the lattice in advance in the Edit Window or in a linear coupling calculation, see Section 5.4.6.

The first dialogue box, shown in Figure 5.31, sets the main parameters for the statistical analysis. First the user is offered the choice between using a set of random errors that have already been defined, or of entering the Kick Window to create or edit skew quadrupole errors, see Chapter 9. Next the user must define the number of sets of data to be generated. The maximum is 1000. Finally, the user is asked whether he wants to calculate the moduli of the coupling coefficients, which are global parameters for the

whole machine, or whether he wishes to view the distribution of coupling vectors in complex phase space at one particular point in the lattice, which he must specify.

STATISTICAL ANALYSIS OF COUPLING

PLEASE SELECT :

☒ Prepare kicks/misalignments

☐ Use existing kicks/misalignments

FOR STATISTICS :

Enter number of datasets to be calculated

PERFORM :

☐ Statistics for moduli of coupling coefficients.

☒ Statistics for coupling vectors [enter position index below].

[Default position is exit to lattice]

OK **Cancel**

NOTES :

- Hand-edited errors are not included in statistics. Use only the Random and Systematic Error routines offered by the program.
- Do not add two sets of kicks of the same type to the same destination. Only the last set will be stored in the memory for generating more sets of data for statistics.
- Kicks/misalignments must contain at least one random element in order to make a statistical analysis.

Figure 5.31 Dialogue box for preparing a coupling statistics calculation

When creating the skew quadrupole errors, use the built-in routines in the Kick Window e.g. add a random skew quadrupole kick to all quadrupoles with an rms of 0.001 T. The program remembers the choices and the rms values given and then generates different sets of data for the statistical analysis. Both random and systematic errors can be included, but there must be at least one random error. Do not enter individual errors by hand. Any errors entered in this way will be ignored by the routine, since it can only remember those errors generated by the built-in routines. Also, only enter an error of a given type with a given destination once, e.g. a random skew quadrupole kick with an rms of 0.001T to be applied to all sextupoles of the family named SF1. If the same family is given a second error by the same routine, the two errors are summed on the screen and are taken into account for single calculations, but for statistics the routine applies only the error that was entered most recently.

Once the errors have been created in the Kick Window, click 'Back to prior window' in the 'Options Menu'. The routine will then display the dialogue box shown in Figure 5.30(c). The options offered in this dialogue box for specifying the resonances to be calculated have already been described in Section 5.4.6. When this last dialogue box is cleared the calculation of the sum and difference coupling coefficients commences. This may take an appreciable time according to the size of the lattice, the number of errors and the power of the computer. A bar graph will record the progress and, if the user wishes, he can terminate the calculation at any time by pressing the right-hand mouse button. In this event, the routine will stop and present the results for whatever number of machines have been calculated up to that point.

Finally, the results are presented in numerical and graphical form that can be printed, or written to a post script file. Figure 5.32 shows an example of 500 coupling vectors generated by random quadrupole tilts. The green line shows the average vector. When the statistical display is closed, the screen display reverts to the Twiss functions.

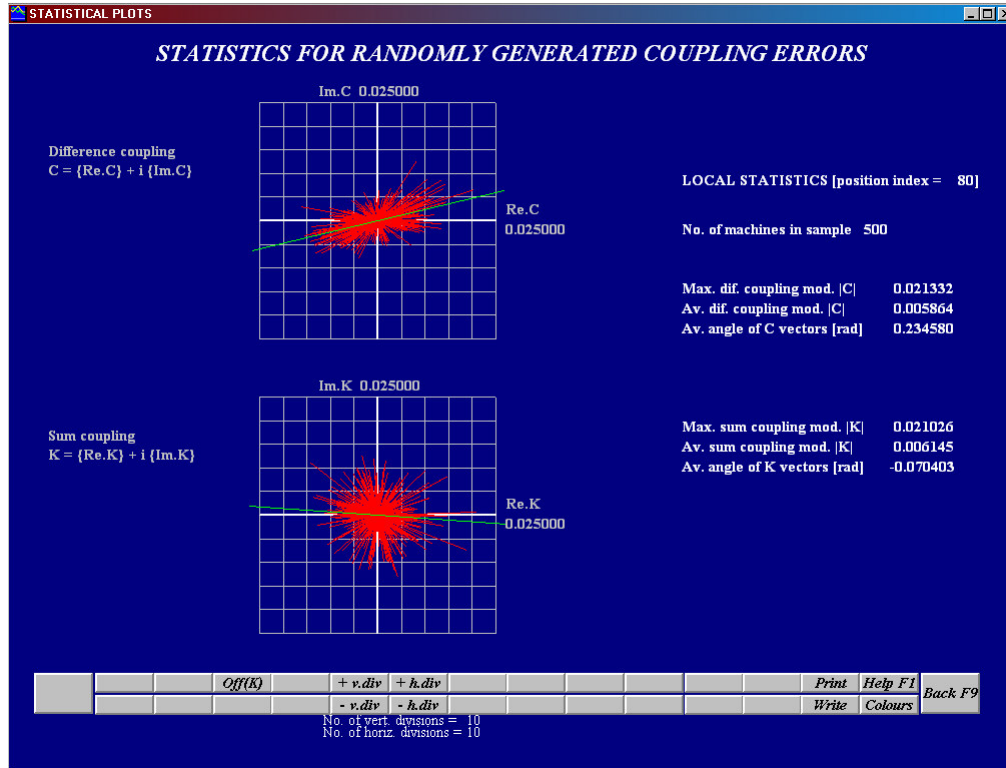


Figure 5.32 Statistical analysis of the coupling vectors generated by random quadrupole tilts

5.4.8 Resonance driving terms (2^{nd} - 5^{th} order), single or summed...

This routine calculates the driving terms of any sum or difference resonance in the range 2nd to 5th order, $n_x Q_x \pm n_z Q_z = p$. It includes the special case of second-order linear coupling with solenoid fields that can also be calculated separately as described in Section 5.4.6.

The user is given the choice of calculating a single resonance in the family that he has specified, or the driving term for the combined action of all the resonances in that family (i.e. summed over the azimuthal harmonic p). Until a new or different calculation is performed, the listing of the driving terms can be recalled for display.

Resonance driving terms are used in the context of nonlinear resonance compensation schemes and in the context of linear coupling. In the former case, the driving term is usually denoted by K and in the latter by C . Unfortunately, there is often a difference between the two definitions of a factor of 2 i.e. $C = 2K$. In WinAGILE, the C definition is used throughout, so the K -type driving terms may appear to be too small by a factor of 2 compared to some other programs.

SPECIFY NON-LINEAR RESONANCE

$n_x.Q_x + n_z.Q_z = p$

[where, $n_x = 0, 1, 2, 3, 4, 5$
 $n_z = 0, +/-1, +/-2, +/-3, +/-4, +/-5$]

SPECIFY RESONANCE :

$n_x =$ $n_z =$

[n_z can be negative]

SPECIFY 'SINGLE' OR 'SUMMED' :

☒ 'Single' resonance harmonic $p =$

☐ 'Summed' resonance family (i.e. all values of p summed)

NOTES:

$|n_x| + |n_z| = 2$ is driven by quadrupoles (skew/normal).
 $|n_x| + |n_z| = 3$ is driven by sextupoles (skew/normal).
 $|n_x| + |n_z| = 4$ is driven by octupoles (skew/normal).
 $|n_x| + |n_z| = 5$ is driven by decapoles (skew/normal).
 If n_z is ODD driving lenses are SKEW.
 If n_z is EVEN driving lenses are NORMAL.
 See HELP for the definition of driving terms.

Figure 5.33 Calculation of non-linear resonance driving terms

5.4.9 *Fit coupling difference and/or sum resonance, single or summed, with or without 'Twiss' tune control...*

This routine is similar to the fitting routines in Sections 5.3.13 to 5.3.16. The user is offered several variants for setting the coupling in a ring to any desired value including zero. The routine treats the coupling as a perturbation to a normal uncoupled lattice described by the Twiss parameters. Difference-coupling resonances and sum-coupling resonances can be treated individually or together and as single or summed resonances. In addition, the Twiss uncoupled tunes can also be controlled. According to the choices made by the user, 2 or 4 or 6 variables will be required (i.e. 2 or 4 coupling variables of the skew quadrupole or solenoid types and 2 normal quadrupole gradients).

For coupling calculations, it is advisable to cut the skew quadrupoles and solenoids into slices of 5 cm or less according to the situation. When the routine is first called, it checks the maximum length of these elements and displays this value with a recommended value for the calculation, see Figure 5.30(a). The user then has the possibility of sub-dividing the lattice before entering the main coupling routine. If the user is only testing the procedure then he can ignore the sub-division to save time. For important work, it is prudent to repeat the coupling calculations with different sub-divisions of the elements to check if the results are stable. If the user wishes he can perform the sub-division in advance in the Edit Window, in the Goodies Menu. Note that once the lattice has been sub-divided, it remains sub-divided after the calculation. If the user tries to save the new sub-divided lattice, the program forces a 'SaveAs' in order that the original master file is not over-written.

On closure, the sub-division dialogue is replaced by Figure 5.30(b), which allows the user to add/edit skew quadrupole kicks if desired. In the event that the user chooses to create a distribution of skew quadrupole kicks, control passes directly to the Kick Window. On leaving the Kick Window, a new dialogue box is opened, see Figure 5.34, for setting the resonances to be included and their mode and whether the uncoupled tunes should also be taken into account. The choices relevant to the resonances have already been described in Section 5.4.6.

2 TO 6 PARAMETER FITTING FOR COUPLING

1. SELECT ANY COMBINATION FOR FITTING :

- ☒ Difference coupling resonance (Allow 2 variables)
- ☒ Sum coupling resonance (Allow 2 variables)
- ☒ Tunes, Qx, Qz (uncoupled) (Allow 2 variables)

2. GIVE DETAILS OF RESONANCES :

DIFFERENCE RESONANCE:

☒ **'Single' resonance mode**

The nearest difference coupling resonance is :

Qx - Qz =

Edit constant if you wish to calculate another member of the same family.

☐ **'Summed' resonance mode**

In this mode, all azimuthal harmonics from plus infinity to minus infinity are summed for the 2nd order difference family.

SUM RESONANCE:

☒ **'Single' resonance mode**

The nearest sum coupling resonance is :

Qx + Qz =

Edit constant if you wish to calculate another member of the same family.

☐ **'Summed' resonance mode**

In this mode, all azimuthal harmonics from plus infinity to minus infinity are summed for the 2nd order sum family.

NOTES :

- Two-variable fitting is fast and reliable. Four-variable fitting also works well. Six-variable fitting is more sensitive.
- For six-parameter fitting, fit coupling first, then add the tunes for the fine adjustment.
- Fitting uses the underlying uncoupled tunes. These are easier to compute than the Teng-Edwards tunes. However, the routine is mainly conceived for compensating coupling and the uncoupled tunes are easier to understand.

Figure 5.34 Setting the resonances to be controlled and whether the uncoupled tunes are to be included

Note that it is possible to mix the single and summed modes by having say single mode for the difference resonance and summed mode for the sum resonances. There is some evidence that when the working point is near to a resonance, then this resonance should be compensated as a single resonance, but when the working point is far from all members of the family, then the dynamic aperture is best served by making a compensation of the summed effect.. It is frequently the case that a machine works close to the difference coupling resonance, but midway between sum coupling resonances.

Finally, the routine displays a 6-parameter fitting dialogue box similar to those already described in Sections 5.3.13 to 5.3.16. The user then selects the position in the ring where he wants to set the coupling vectors, selects his variables and sets the desired end values into the edit boxes before clicking the calculation button, see Figure 5.35. In

the chosen example, the coupling is being set to zero and the tunes are being maintained constant. If the real and imaginary parts of both the sum and difference resonances to are set to zero when in the summed resonance mode, then the off-diagonal coupling terms in the single-turn transfer matrix will be made quasi-zero at the position in the lattice where the calculation is made.

6-PARAMETER FIT FOR DIFFERENCE AND SUM COUPLING WITH TUNE CONTROL

1. ENTER OBSERVER'S POSITION AND DESIRED COUPLING COEFFICIENTS AND TUNES :

Index of position for fit =

C real = K real = Qx =

C imaginary = K imaginary = Qz =

2. CHOOSE 2 SKEW QUADS/SOLENOIDS :

EITHER click on unit in inventory

OR enter directly below the indices and increments to be used

9
20
41
126
131

3. CHOOSE 2 UNITS WITH QUAD. GRADIENT :

EITHER click on unit in inventory

OR enter directly below the indices and increments to be used

7
10
12
23

4. UNITS/SERIES AND STEP-SIZES TO BE USED:

Index of 1st unit or unit in 1st series = Step-size for 1st unit/series =

Index of 2nd unit or unit in 2nd series = Step-size for 2nd unit/series =

Index of 3rd unit or unit in 3rd series = Step-size for 3rd unit/series =

Index of 4th unit or unit in 4th series = Step-size for 4th unit/series =

Index of 5th unit or unit in 5th series = Step-size for 5th unit/series =

Index of 6th unit or unit in 6th series = Step-size for 6th unit/series =

(Re-) Compute

STATUS :

	C real	C imag	K real	K imag	Qx	Qz
Desired =	0.000000	0.000000	0.000000	0.000000	1.666600	1.720000
Obtained =	0.003848	-0.000274	0.003406	-0.001812	1.666600	1.720000
Residuals =	0.003848	-0.000274	0.003406	-0.001812	0.000000	-0.000000

OK - accept new values Reset original values

Figure 5.35 6-parameter fitting dialogue for difference and sum coupling

By clicking 'OK – accept new values', the dialogue is closed and the state of the coupling in the machine is displayed. The result can also be viewed graphically by clicking the button 'Graph' on the main display. An example is shown in Figure 5.36. In the example the initial error vector is shown in pink. The correction vectors are shown in grey and the final result would be shown in cyan if it had a finite amplitude. The labels can be switched on/off.

As with all the fitting routines, the more parameters that are used the more sensitive the fit becomes. The computation button can be clicked again to improve the fit obtained and if any variables are found to be degenerate then routine will notify the user.

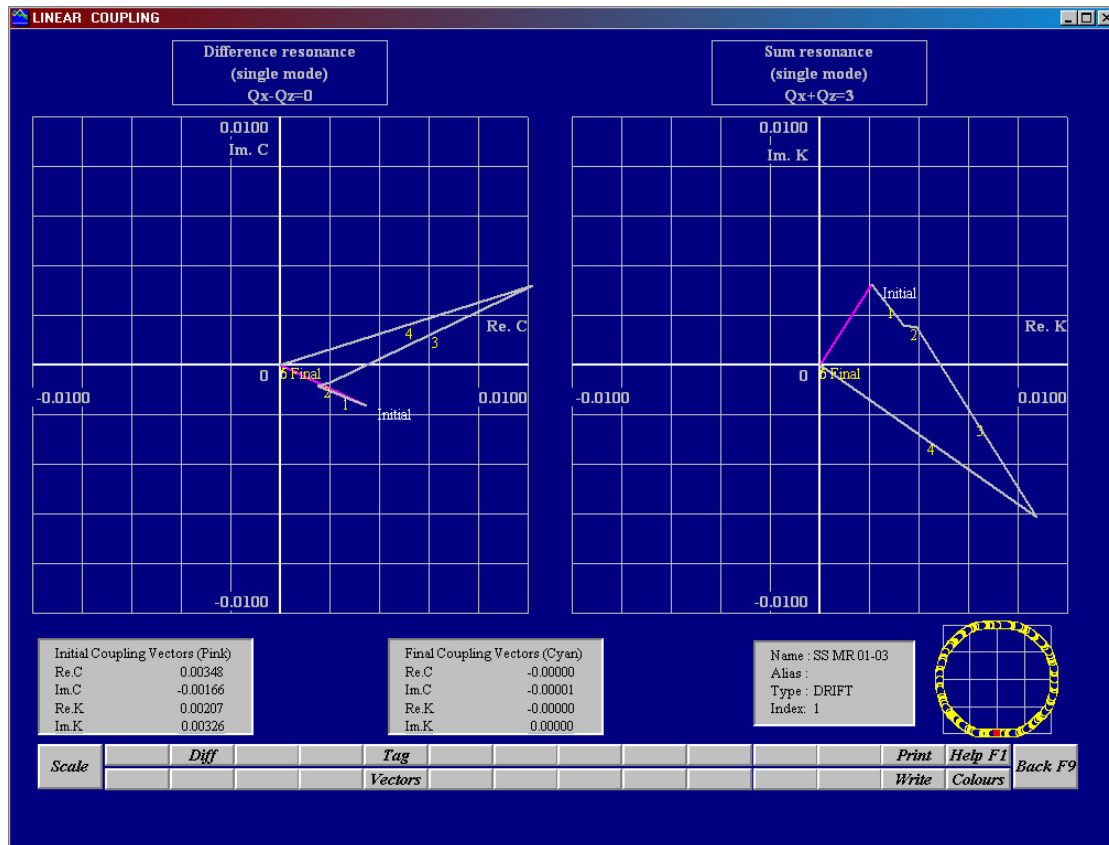


Figure 5.36 Graphical representation of coupling compensation with tune control

5.4.10 Decouple single-turn matrix...

This routine is only applicable to closed rings in the presence of coupling. It is very similar to the fitting routines already described, except that instead of fitting high-level parameters such as the tune values, it operates directly on the 4 off-axis matrix coefficients in the single-turn matrix to set them to zero. In general, the coupled matrix for a single-turn in a machine can only be decoupled at one point at a time by suitably powering four coupling elements.

The routine displays the 4-parameter fitting dialogue box shown in Figure 5.37. The usual comments apply that the calculation can be cycled and the variables (skew quadrupoles and/or solenoids) cannot be degenerate. Note that although the single-turn matrix can be decoupled in this way, this does not mean that the whole machine is decoupled. In fact, the resonance excitation may be worse than before at some points. A close analogy is that of the closed orbit. The orbit can be perfect over a limited region while being extremely distorted elsewhere. If the aim is to limit the excitation of particular coupling resonances, then it is more prudent to try the resonance fitting routines described in the previous section.

If the user accepts the fit proposed by the routine, he can close the dialogue box by clicking 'Accept new values' and the single-turn matrix at the specified point will be displayed to confirm that the off-axis 2×2 blocks are zero.

4-PARAMETER FIT TO DECOUPLE SINGLE-TURN MATRIX

1. ENTER POSITION IN LATTICE :
Index of position where single-turn matrix is to be de-coupled =

2. CHOOSE SKEW QUADS OR SOLENOIDS :

EITHER click on unit in inventory

OR enter directly below the indices and increments to be used

9
20
41
126
131

skq0
skq1
skq2
skq3
skq4

3. UNITS/SERIES AND STEP-SIZES TO BE USED:

Index of 1st unit or unit in 1st series =	20	Step-size for 1st unit/series =	0.001000
Index of 2nd unit or unit in 2nd series =	41	Step-size for 2nd unit/series =	0.001000
Index of 3rd unit or unit in 3rd series =	126	Step-size for 3rd unit/series =	0.001000
Index of 4th unit or unit in 4th series =	131	Step-size for 4th unit/series =	0.001000

STATUS OF OFF-AXIS MATRIX ELEMENTS:

	hvl1	hvl2	hv21	hv22
Desired values =	0.0	0.0	0.0	0.0
Obtained values =	-0.000000	0.000000	-0.000000	0.000000
Residual errors =	0.000000	0.000000	0.000000	0.000000

NOTES:

- In general, the matrices will only be decoupled at the specified point and for a limited surrounding region.
- At other places in the ring, the effects of coupling may still exist and may be stronger than before.
- In particular, the closest coupling resonance to the working point may be stronger.

Figure 5.37 4-parameter fitting dialogue for decoupling the single-turn matrix

5.4.11 Decouple single-turn matrix with Teng tune control...

This is a fitting routine for decoupling the single-turn matrix of a closed ring. It is similar in conception to the routine described in Section 5.4.10, but it is only applicable to coupled rings calculated in the 'exact mode' using the Teng-Edwards parameters in place of the Twiss parameters. If control of the tunes is not needed, then the simpler and faster routine described in Section 5.4.10 should be used instead.

As before, decoupling the single-turn matrix is, in general, only possible at one point in the ring. Since the Teng-Edwards tunes are included, 6 coupling variables are needed. The usual comments apply that the calculation can be cycled, the variables (skew quadrupoles and/or solenoids) cannot be degenerate and the routine can be rather sensitive. The routine presents the dialogue box shown in Figure 5.38. When the 'Accept - new values' is clicked, the routine displays the single-turn matrix at the

specified point to confirm that the off-axis 2×2 blocks are zero. This routine can be rather sensitive.

6-PARAMETER FIT TO DECOUPLE SINGLE-TURN MATRIX WITH TUNE CONTROL

1. ENTER POSITION IN LATTICE AND NORMAL MODE TUNES :

Index of position where single-turn matrix is to be de-coupled =

Normal mode tune 1 (horizontal tune once decoupled), Q1 =

Normal mode tune 2 (vertical tune once decoupled), Q2 =

2. CHOOSE 4 SKEW QUADS/SOLENOIDS :

EITHER click on unit in inventory
OR enter directly below the indices and increments to be used

9	skq0
20	skq1
41	skq2
126	skq3
131	skq4

3. CHOOSE 2 UNITS WITH QUAD. GRADIENT :

EITHER click on unit in inventory
OR enter directly below the indices and increments to be used

7	QF (1)
10	MB
12	QD
23	QF (2)

4. UNITS/SERIES AND STEP-SIZES TO BE USED:

Index of 1st unit or unit in 1st series =	<input type="text" value="9"/>	Step-size for 1st unit/series =	<input type="text" value="0.001000"/>
Index of 2nd unit or unit in 2nd series =	<input type="text" value="126"/>	Step-size for 2nd unit/series =	<input type="text" value="0.001000"/>
Index of 3rd unit or unit in 3rd series =	<input type="text" value="131"/>	Step-size for 3rd unit/series =	<input type="text" value="0.001000"/>
Index of 4th unit or unit in 4th series =	<input type="text" value="20"/>	Step-size for 4th unit/series =	<input type="text" value="0.001000"/>
Index of 5th unit or unit in 5th series =	<input type="text" value="12"/>	Step-size for 5th unit/series =	<input type="text" value="0.001000"/>
Index of 6th unit or unit in 6th series =	<input type="text" value="23"/>	Step-size for 6th unit/series =	<input type="text" value="0.001000"/>

(Re-) Compute

STATUS OF OFF-AXIS MATRIX ELEMENTS:

	h _v 11	h _v 12	h _v 21	h _v 22	Q1 mode	Q2 mode
Desired =	0.0	0.0	0.0	0.0	1.660000	1.720000
Obtained =	0.000000	0.000000	0.000000	0.000000	1.666592	1.720022
Residuals =	-0.000000	-0.000000	-0.000000	-0.000000	0.006592	0.000022

OK - accept new values Reset original values

NOTES:

- In general, the matrices will only be decoupled at the specified point and for a limited surrounding region.
- At other places in the ring, the effects of coupling may still exist and may be stronger than before.
- In particular, the closest coupling resonance to the working point may be stronger.

Figure 5.38 6-parameter fitting dialogue box for decoupling the single-turn matrix with Teng tune control

5.5 Tables

The Tables menu provides access to the results currently held in memory. The menu items are greyed if no results are available, or if the results are not consistent with the current lattice. In Chapter 3 for the Main Window, many of the functions have already been described, so only the new functions will be described here.

5.5.1 Sigma matrices

Sigma matrices are a beam description based on statistical functions. Providing the transfer matrices are known and there are no numerical instabilities the sigma matrices can usually be evaluated. They have the advantage of being able to give the

beam sizes when the beam is coupled. In the case of uncoupled beams (transverse 4×4), the sigma matrices reduce to combinations of the Twiss functions and the emittances. The screen display shows: the sigma matrix for each element, the dispersion functions, the statistical emittances in the horizontal plane, the vertical-plane and for the 4-dimensional ellipsoid and some other general beam parameters. All parameters refer to the entries of the elements and an exit line is added at the end of the listing.

The matrix elements can be more conveniently viewed by double-clicking anywhere along the line belonging to an element to show the matrix in 'block' form, see Figure 5.39.

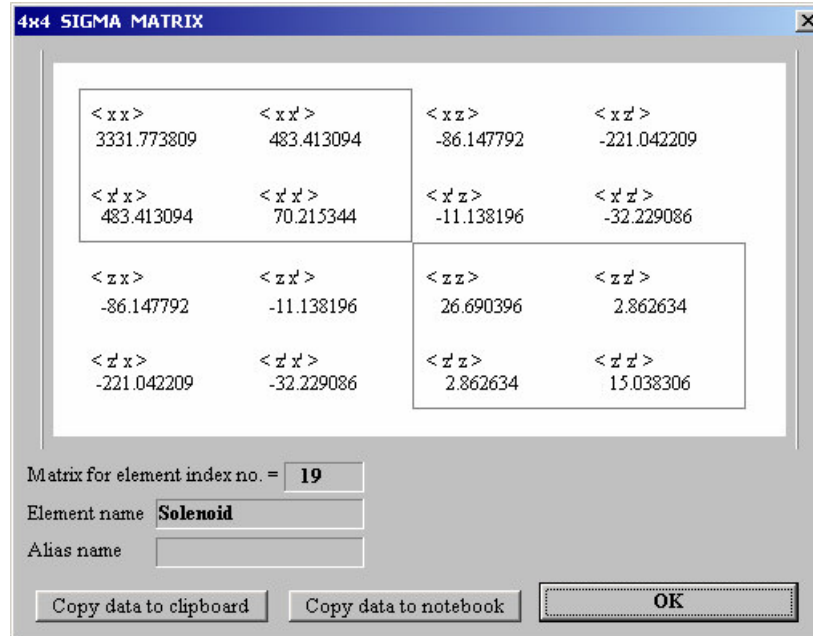


Figure 5.39 A sigma matrix in a line with solenoids and skew quadrupoles

5.5.2 *Twiss transverse parameters*

This routine lists the transverse Twiss parameters on the screen and some derived parameters for the current equilibrium orbit. All parameters refer to the entries of the elements and an exit line is added at the end of the listing. There is also a summary chart at the end of the listing. When scatterers are present in the beam, the display changes slightly and the betatron phase advances, which have no meaning in a scatterer, are removed. There is a graphical representation of the Twiss data available in the Twiss plot window accessed via the Graph menu and via the "Graph" button (near top right of the screen) when the data display for Twiss functions is active.

5.5.3 *Twiss longitudinal parameters*

This routine lists the longitudinal Twiss parameters on the screen and some derived parameters for the current equilibrium orbit. The longitudinal Twiss parameters are only applicable to the linear motion inside the RF bucket. They can be found in such programs as Trace-3D, but they are used far less often than their transverse equivalents. All parameters refer to the entries of the elements and an exit line is added at the end of

the listing. There is also a summary chart at the end of the listing. A graphical representation of the longitudinal Twiss data is available in the Twiss plot window accessed via the Graph menu and via the "Graph" button (near top right of the screen) when the data display for longitudinal Twiss functions is active.

5.5.4 *Normalised dispersion*

This routine lists the normalised dispersion through the lattice. The graphical representation of this data, which is accessed via the Graph menu and via the "Graph" button (near top right of the screen) when the data display for normalised dispersion is active, is useful for lattice design purposes.

5.5.5 *Twiss-style transverse bunched-beam parameters*

This is a special use of the Twiss formalism that can lead to confusion and apparent conflict with the normal transverse Twiss parameters. It is included here for comparison with programs such as Trace-3D and is only available when RF is applied to the beam.

In a bunched beam, the RF is controlling the momentum spread, which contributes to the transverse beam size in finite-dispersion regions. Since beam size is an important and observable parameter, it is useful to have a way of representing this with the action of the RF included. The bunch parameters use the Twiss formalism to represent the total beam size due to betatron and dispersion effects, but care should be taken to indicate that this is a special use of Twiss and emittance parameters. One consequence is that single particles cannot be tracked with these functions and the phase function that can be constructed by integrating the inverse of the betatron amplitude has no physical meaning.

The data listing shows the effective Twiss parameters and the corresponding emittances that change in the presence of dispersion. All parameters refer to the entries of the elements and an exit line is added at the end of the listing. A graphical representation of the bunched Twiss data is available and is accessed via the Graph menu and via the "Graph" button (near top right of the screen) when the data display for bunched Twiss functions is active.

5.5.6 *Teng-Edwards transverse and Twiss longitudinal parameters*

Twiss parameters are not calculable in the presence of coupling. One alternative is the use of the Teng-Edwards parameters. These parameters converge on the uncoupled Twiss parameters as the coupling reduces to zero.

This routine lists the Teng-Edwards parameters for the transverse plane and the longitudinal Twiss parameters for the longitudinal plane. All parameters refer to the entries of the elements and an exit line is added at the end of the listing. There is also a summary chart that contains the unperturbed tunes values and the Teng-Edwards, normal-mode tunes Q1 and Q2.

The Teng-Edwards parameters describe coupled lattices in a new co-ordinate system that isolates the normal modes. The equivalence between these parameters and

the normal Twiss parameters is a mathematical one rather than a practical one. For example, these betatron amplitude functions cannot be used to predict beam envelopes in real space, although in many cases the differences may not be large. It is convenient to call 'Normal mode 1' the 'near-horizontal mode' and 'Normal mode 2' the 'near-vertical mode'. A graphical representation can be accessed via the Graph Menu and via the "Graph" button (near top right of the screen) while the Teng-Edwards data display is active.

5.5.7 Trajectories transverse and longitudinal

This routine lists the results of tracking calculations in 6D space. Graphical representations of the transverse and longitudinal trajectories can be accessed via the Graph menu. Clicking the "Graph" button (near top right of the screen) will also show the trajectory plot for either the transverse trajectory or the longitudinal trajectory according to the status of the mode button on the left hand side of the screen.

5.5.8 Losses

When tracking particle distributions, the vacuum chamber can be set and the particle losses recorded. The data describing the losses, it can be accessed via this menu item. The listing shows the losses along the lattice and the repartition between the inner, outer, lower and upper walls.

5.5.9 Closed orbits

This routine lists a closed orbit with the kick distribution used to generate the closed orbit. All parameters refer to the entries of the elements and an exit line is added at the end of the listing. There is also a summary chart at the end of the listing that shows the peak-to-peak, the average and the rms values of the distortion in both planes. A graphical representation can be accessed via the Graph Menu and via the "Graph" button (near top right of the screen) while the closed-orbit data display is active.

5.5.10 Envelopes

The screen listing for envelope calculations adapts according to the situation. When RF equipment is present and powered, the overall beam envelope due to betatron and momentum effects is calculated using the "bunched-Twiss" parameters. If coupling is present the envelope is calculated using the sigma matrices. In normal cases and lines with scattering, the Twiss functions are used. The envelope may be added to a trajectory or directly to the central orbit. All parameters refer to the entries of the elements and an exit line is added at the end of the listing. There is also a summary chart at the end of the listing. A graphical representation of the envelopes can be accessed via the Graph Menu and via the "Graph" button (near top right of the screen) while the envelope data display is active.

5.5.11 Betatron modulation

This routine lists the modulation of the betatron amplitude functions for a given quadrupole error distribution. All parameters refer to the entries of the elements and an exit line is added at the end of the listing.

5.5.12 Chromatic variables

When an off-axis orbit is active, the chromatic variables and the W -vector can be listed for this orbit with respect to the original central orbit. The W -vector quantifies the chromatic effects and provides a way of designing sextupole correction schemes. All parameters refer to the entries of the elements and an exit line is added at the end of the listing. The W -vector can be viewed via the Graph Menu and via the "Graph" button (near top right of the screen) while the chromatic data display is active.

5.5.13 Linear coupling, diff. and sum resonances, single and summed modes

This routine lists the coupling coefficients and associated data for the second-order difference and sum resonances at all points round a ring. All parameters refer to the entries of the elements and an exit line is added at the end of the listing. Note that in a coupled machine two tilts can be at work. The normal modes of the betatron oscillations will be modified and tilted and, if the skew quadrupoles are in finite dispersion regions, dispersion will also be excited in the orthogonal plane giving rise to a tilt depending on momentum deviation. The difference between the unperturbed phase advances is listed, since this is useful when choosing positions for skew quadrupoles/solenoids for a coupling compensation scheme.

5.5.14 Resonance driving terms (2nd-5th order), single or summed

The routine displays the data associated with single or summed resonances at all points round a ring. All parameters refer to the entries of the elements and an exit line is added at the end of the listing. Resonance driving terms are used in the contexts of nonlinear resonance compensation schemes and linear coupling. In the former the driving term is usually denoted by K and in the latter by C . Unfortunately, there is often a difference between the two definitions of a factor of 2 i.e. $C = 2K$. In WinAGILE, the C definition is used throughout, so the non-linear K -type resonance driving terms may appear to be too small by a factor of 2 compared to some other programs.

5.5.15 Separatrices

Directly after a phase-space map has been calculated and the 'Sptrx I' routine has been used, it is possible to view the raw data for the separatrix plots and to write this data to a file in standard spreadsheet format, see 'Write export file'.

The listing of the separatrix data is shown on the screen under the column headings : x_1 , dx_1/ds , *step*, x_2 , dx_2/ds , *step*, x_3 , dx_3/ds , *step*, x_{p1} , dx_{p1}/ds , *step*, x_{p2} , dx_{p2}/ds , *step*, x_{p3} , dx_{p3}/ds , *step*, *area*.

x_1 , x_2 , x_3 are the positions of the on-resonance particle on the three separatrices that spiral outwards at the plane of the map.

dx_1/ds , dx_2/ds , dx_3/ds are the longitudinal slopes of the particle trajectory at the plane of the map.

x_{p1} , x_{p2} , x_{p3} are the points and dx_{p1}/ds , dx_{p2}/ds , dx_{p3}/ds are the slopes of the separatrices for the off-resonance particle.

The off-resonance points are in two groups. Lines 1-332 contain the first unstable separatrix and points 333-666 contain the last stable triangle. Points beyond the aperture are replaced by 9999.

The steps correspond to the change in position from the current line to the next.

‘area’ is the area enclosed by the three points x_{p1} , x_{p2} , x_{p3} .

If the routine ‘Sptrx II’ is used, this data is modified for internal use.

5.5.16 Summary of eddy current data

This dialogue box, see Figure 5.40, summarises the current status of the eddy current data. It lists individual units and the first unit of any series of units that have stored eddy current parameters. In each case, it shows whether the unit has data for the vacuum chamber and/or the magnet yoke.

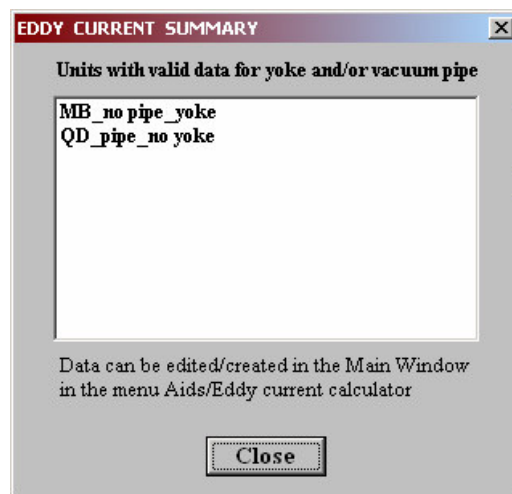


Figure 5.40 Dialogue box showing the current status of the eddy current data

5.5.17 Other derived data

This dialogue box, see Figure 5.41, collects together several parameters derived from the lattice data. Note that the user can choose between numerically calculated and analytically calculated data. As can be seen from the example, the majority of the parameters can be calculated analytically, but where this is not available the numerical option can still be taken.

OTHER BASIC DATA

LIMITS:
 From entry to element index no.
 To exit of element index no.
Total no. of elements is 134

INTEGRATION:
☐ Numerical
☒ Analytic

NOTES:
 Maximum and minimum values are found from the listings. Differences may occur after subdividing elements.

Average beta-x	[m] =	8.743742
Average of square root of beta-x	[m ^{1/2}] =	not available
Average beta-z	[m] =	8.491778
Average of square root of beta-z	[m ^{1/2}] =	not available
Average Dx	[m] =	3.155533
Average Dz	[m] =	0.000000
Maximum beta-x	[m] =	16.077030
Minimum beta-x	[m] =	3.772570
Maximum beta-z	[m] =	14.807138
Minimum beta-z	[m] =	2.847754
Maximum Dx	[m] =	8.262129
Minimum Dx	[m] =	-0.000001
Maximum Dz	[m] =	0.000000
Minimum Dz	[m] =	0.000000
Length of beam path	[m] =	75.240000
Revolution/transit time	[1E-6 s] =	0.286623

Figure 5.41 Dialogue box showing a selection of

5.5.18 Synchrotron radiation data

This function is a synchrotron radiation calculator, see Figure 5.42. The user can set the beam ion and its energy and then calculate the synchrotron radiation integrals and several other parameters for the current lattice. The routine is only applicable to closed rings. When the user exits, the original beam data is reset.

SYNCHROTRON RADIATION DATA

SET BEAM ION :
 Current beam ion is a

SET ENERGY/MOMENTUM OF ION :
 Average value per nucleon :
☒ Kinetic energy [GeV/n]
☐ Momentum [Gev/c/n]

SYNCHROTRON RADIATION DATA :

S.R. Integrals :
 I1 [m] = I2 [m-1] =
 I3 [m-2] = I4 [m-1] =
 I5 [m-1] =

Partition numbers :
 Jx = Jz =
 Js =

Damping constants :
 Tx = Tz =
 Ts =

Other parameters :
 Energy loss/particle/turn [keV] =
 Lattice damping constant D~ =
 Sigma E/E [o/oo] =
 Equilibrium emittance [pi mm mrad] =

Figure 5.42 Calculator for synchrotron radiation parameters

5.5.19 Condensed lattice data

This command activates a sub-menu with two items that display the matrices and Twiss functions of the *condensed lattice* that is generated when a phase-space map is

created. In a condensed lattice, the sections that contain only linear elements are reduced to single matrix elements. This is done to make tracking faster with large numbers of particles.

5.6 *Graphs*

Most of the data displays accessed through the Tables menu have graphical equivalents accessed through the Graphs menu. The graphical displays often have a range of controls and new calculation features. These are described in more details in the plot windows described in Chapter 8.

5.7 *Output*

The Output menu provides the same functions as described for the Main Window.

5.8 *Help (F1)*

The Help Menu provides a standard help file, key references and copyright and disclaimer notices.

* * *

Chapter 6 Line Window

The Line Window is accessed via the Main Window and is dedicated to the calculation of transfer lines. Transfer lines are not required to have their input and output Twiss parameters equal. They are intrinsically more general structures and, unlike rings and matched sections, scatterers and RFQs are allowed and RF cavities are not restricted to non-accelerating buckets.

Although lines and rings have many fundamental differences, the Ring Window contains nearly all the menu items appearing in the Line Window. Furthermore, the descriptions already given need no further elaboration. The one major addition is “Matching (Ctrl+A)” in the “Calculations” menu. This menu item is the gateway to the Numerical Matching Window, which is described separately in Chapter 7.

* * *

Chapter 7 Numerical Matching Window

The Numerical Matching Window is accessed from the Line Window through the menu item “Matching” in the “Calculations” menu. The Numerical Matching Window is wholly dedicated to the matching of transfer lines and modules to sets of prescribed conditions. When the window first opens, a summary is presented of the principal steps to be taken to perform matching, see Figure 7.1.

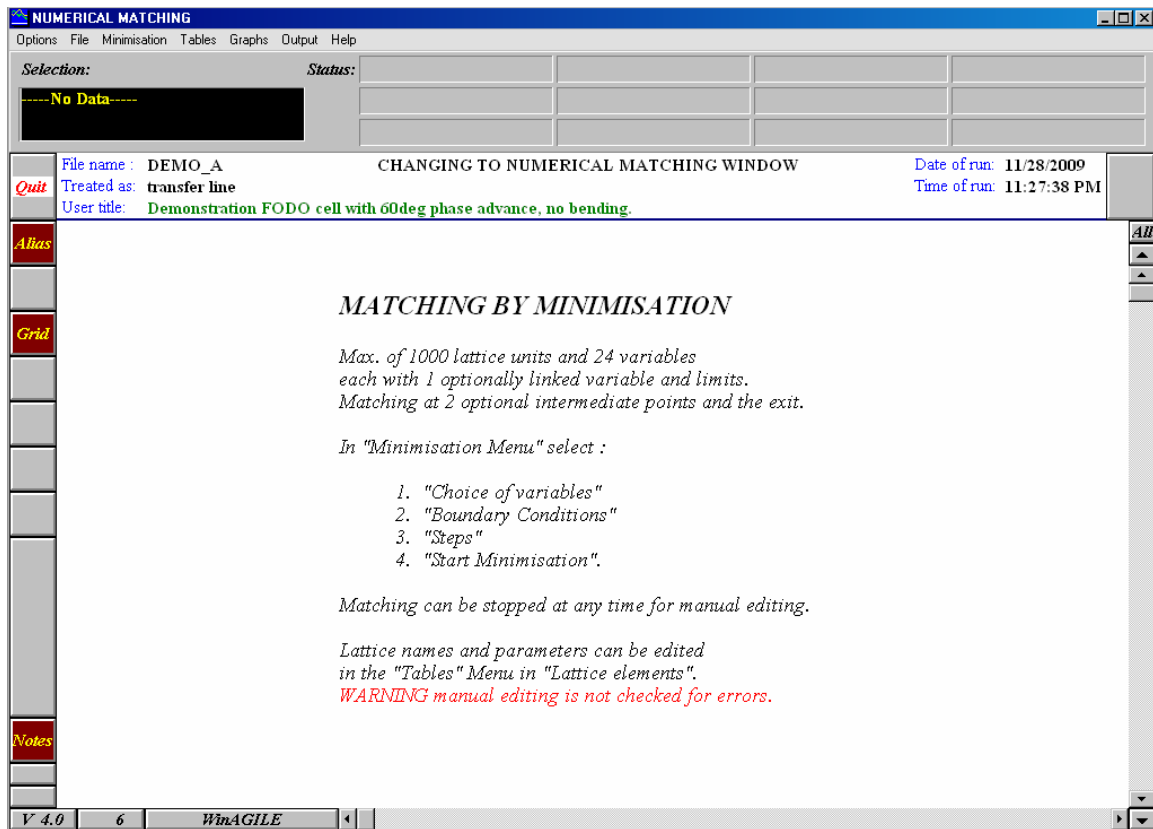


Figure 7.1 Opening the Numerical Matching Window

The lattice on which the matching is to be performed is limited in size to 1000 units. In general, it is more efficient to split up a matching problem into modules with small numbers of elements in each module. This makes the 1000 limit look generous, but for lattices that have been sub-divided for matching with space charge for example, the limit can be quickly reached. In addition, the number of variables that can be used simultaneously is limited to 24.

The first three steps (“Choice of variables”, “Boundary conditions” and “Steps”) can be made in any order and can be revisited at any time and, in most cases, the default values for the Steps used by the numerical search routines can be left unchanged. The minimum input needed for matching is one variable and one boundary condition. The user has complete freedom to stop matching, edit parameters and restart as he sees fit.

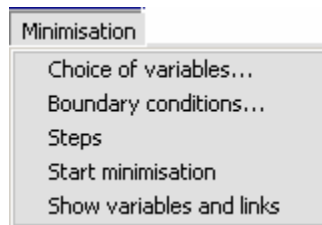
Sometimes when matching, it is necessary to intervene manually and make changes to correct or influence the matching process. **By entering the Tables menu and**

selecting “Lattice elements”, it is possible to edit the lattice in the same way as in the Edit Window. Editing is enabled on this table because it is extremely convenient, but the user should be aware that all the checks that exist in the Edit Window are not applied here. This means that it is very easy to corrupt data invalidating results and ultimately causing the program to crash.

7.1 Options and File Menus

These two menus contain only the basic functions for returning to the prior window, quitting WinAGILE and saving lattices, all of which are self-explanatory and appear in descriptions made earlier for the Main Window.

7.2 Minimisation



The Minimisation Menu contains the principle functions for setting the environment for numerical matching.

7.2.1 Choice of variables...

This routine presents a full-screen editor that displays all the lattice elements in beam order with their parameters. If a parameter is both suitable for matching and not yet used for matching, it is marked "FREE". By double clicking on the cell in the chart, the variable is chosen and its designation changes to "VAR.1" or VAR.2" or whatever is the next free matching number. If a mistake is made, the variable can be de-selected by double clicking a second time. When a variable is de-selected, the remaining variables are automatically renumbered. If the variable chosen belongs to a series chain of elements (i.e. elements with the same name), then all those elements will display the same designation "VAR.#". By scrolling downwards longer structures can be scanned and variables selected up to a maximum of 24. An example is shown in Figure 7.2.

In the example in Figure 7.2, six independent variables have been selected. ‘Variables 2’ and ‘Variable 3’ are the gradients of individual quadrupoles. This class of matching variable is the most convenient and the most used. ‘Variable 1’ is repeated three times down the list because it is the bending angle of three dipoles connected in series. Of course, matching with bending angles is only possible when the geometry is not fixed. “Variables 5” and ‘Variable 6’ are drift lengths. Drift lengths make very powerful matching variables, but they are only usable while the geometry is still flexible. ‘Variable 4’ is an edge angle. This provides a very useful variable that can still be applied once the geometry is fixed, but of course it can no longer be used once the magnet is designed. Note that the edge angles of the three series dipoles are not offered as being ‘Free’ for matching. This is because these dipoles are rectangular bends and their edge angles are set to half the bending angle by definition.

File name : TestLine		SELECTION OF MATCHING PARAMETERS						Date of run: 12/8/2009	
Treated as: transfer line								Time of run: 1:15:02 PM	
User title: Proton injection line									
Unit no.	Name	Type	Length [m]	H-bend [rad]	V-bend [rad]	Edge angle-1 [rad]	Edge angle-2 [rad]	Grad k.kL [m-2.m-1]	Skewq. ks [m-2]
1	Ref. Point S	MARK	--	--	--	--	--	--	--
2	SSS IN 11	DRIFT	VAR. 5	--	--	--	--	--	--
3	HBD IN	RBEND	Free	VAR. 1	--	--	--	Free	--
4	SSS IN 12	DRIFT	Free	--	--	--	--	--	--
5	HBD IN	RBEND	Free	VAR. 1	--	--	--	Free	--
6	SSS IN 13	DRIFT	Free	--	--	--	--	--	--
7	QUF IN 13	QUADR	Free	--	--	--	--	VAR. 3	--
8	SSS IN 14	DRIFT	Free	--	--	--	--	--	--
9	HBD IN	RBEND	Free	VAR. 1	--	--	--	Free	--
10	SSS IN 15	DRIFT	Free	--	--	--	--	--	--
11	QUF IN 15	QUADR	Free	--	--	--	--	VAR. 2	--
12	SSS IN 16	DRIFT	Free	--	--	--	--	--	--
13	MSI IN 16	RBEND	Free	Free	--	--	--	Free	--
14	SSS IN 17	DRIFT	Free	--	--	--	--	--	--
15	MSI IN 17	RBEND	Free	Free	--	--	--	Free	--
16	SSS IN 18	DRIFT	VAR. 6	--	--	--	--	--	--
17	MSI-IN-18	SBEND	Free	Free	--	Free	VAR. 4	Free	--
18	Ref-Pt-B	MARK	--	--	--	--	--	--	--

Figure 7.2 Screen editor for the selection and definition of matching variables

By scrolling horizontally, three additional types of matching variable become visible. The skew quadrupole gradient and solenoid field are foreseen for matching coupled lattices with the Teng-Edwards functions rather than Twiss functions. However, this is still in an experimental stage and should not be used in the current version of the program. The third variable, labelled as “Move in lattice”, provides two new features:

- When an element has individually named drift spaces on either side, then "Move in Lattice" makes the position of the element a matching variable while keeping the sum of the two drift spaces constant. When used with quadrupoles, this provides an extra matching variable without affecting the overall geometry.
- When the element is a scatterer and the following element is a drift space, then "Move in Lattice" makes the thickness of the scatterer a matching variable, such that the combined distance of the scatterer thickness plus the following drift space remain constant. For this function, do not subdivide the scatterer.
- Note that if other drift spaces have the same name as one of the drift spaces used by “Move in lattice”, then these drift spaces will also be changed in unison. Thus, it is possible to move a series of quadrupoles in a regular lattice in a symmetric way, but in most cases individual names for the drift spaces will be needed in order to preserve the overall geometry.

Scrolling rightwards further reveals columns for setting maximum and minimum limits on gradients, angles and lengths. Quadrupoles and dipoles already have a default minimum of 0.01 m because these elements are not allowed to have zero length. If when setting the limits for a variable, the variable is outside the range specified, then the routine offers to edit the value to bring it within range.

Each matching variable can be linked to one other matching variable. To do this, first select a master variable in the way described above. Directly after, select the LINK variable by double clicking with the right-hand mouse button. If this is an allowable combination, then the variable will be labelled "LINK V.2" or whatever number the master variable has. A dialogue box then appears which ask for the link ratio. For example, if the series F-quadrupole is to be linked to the series D-quadrupole chain

and all the quadrupoles are mechanically equal and the idea is to have one power supply, then the link ratio would be -1. Link variables can be de-selected by double clicking, this time with the left-hand mouse button.

The allowable combinations for links are:

- a pair of lengths (note that the link ratio must be positive in this case),
- a pair of gradients,
- a pair of edge angles from SBEND dipoles (note that RBEND edge angles are not variables and cannot be selected),
- a pair of bending angles (note that angles from an RBEND and an SBEND can be linked),
- a pair of thin lenses.

The selection of variables can be left at any time by clicking a new item in the menu bar. The variables selected up to that point will be retained even if you return temporarily to the Line Window.

7.2.2 Boundary conditions...

Initially the program surveys the lattice checking for coupling and scatterers before opening the dialogue box shown in Figure 7.3

Figure 7.3 Definition of basic requirements for a matching problem

In the case of an uncoupled line (as in Figure 7.3), the dialogue will indicate that matching can be performed with Twiss functions, dispersion functions and betatron phase advances. In the case of a coupled lattice, the Twiss functions will be replaced by the Teng-Edwards functions, but as this is still at an experimental routine, this option should not be used in the current version. In the event, that scatterers are found, item 5 (which is greyed in Figure 7.3) will appear and the user will be able to select emittances or beam sizes as possible boundary conditions, but the betatron phases advances will be removed since they have no strict meaning. This is the only situation in which emittances can be

adjusted as part of the minimisation process. Of course, if the routine is to adjust the emittances, then the thicknesses of the one or more scatterers must be included in the matching variables.

In all cases, item 6 in the dialogue box offers the user the possibility of defining boundary conditions at up to two intermediate points. Finally, there are some special conditions such as “ $\beta_x = \beta_z$ at the exit to the line” that can be applied. The most frequently used of these special conditions is that the β_x and β_z values at intermediate points should not be used as matching values, but as limiting values. The comments in the dialogue box are reproduced below in view of their importance:

- All weights are initially set to zero.
- To add/remove a boundary condition add/remove the corresponding weight.
- Typical weights are 1-10 for dimensions/sizes and 10-100 for angles/derivatives.

By clicking “Next” the user passes to a full screen editor for the boundary conditions, as shown in Figure 7.4.

NUMERICAL MATCHING

Options File Minimisation Tables Graphs Output Help

Edit box :
10.0000000000

OK

Lattice OK with ion Beam ion: 12C 6+ Eddy currents Off
 Quad fringe-field Off 0.007000 [GeV/n](entry) On-axis orbit
 HJM chromaticity eqn Non-space charge optics

File name : TestLine Date of run: 12/8/2009
 Treated as: transfer line Time of run: 5:55:17 PM
 User title: Proton injection line

DEFINING BOUNDARY CONDITIONS

Alias	Parameter		Entry point 1	Mid-point 1 12	Mid-point 2 15	Exit point 19			
Grid	Beta-x [m]	Existing	16.41697297	40.63598330	45.45394805	49.69008857			
		Wanted	16.41697297	40.63598330	45.45394805	30.00000000			
		Weight	N.A.	0.00000000	0.00000000	0.00000000	1.00000000		
Grid	Alpha-x	Existing	0.00000000	-0.89251123	-1.00478136	-0.82627094			
		Wanted	0.00000000	-0.89251123	-1.00478136	0.00000000			
		Weight	N.A.	0.00000000	0.00000000	0.00000000	10.00000000		
Grid	Beta-z [m]	Existing	2.86501303	9.68530734	1.06001106	4.10698758			
		Wanted	2.86501303	9.68530734	1.06001106	5.00000000			
		Weight	N.A.	0.00000000	0.00000000	0.00000000	1.00000000		
Grid	Alpha-z	Existing	0.00000000	2.85360347	0.56546351	-2.12324596			
		Wanted	0.00000000	2.85360347	0.56546351	0.00000000			
		Weight	N.A.	0.00000000	0.00000000	0.00000000	10.00000000		
Grid	D-x [m]	Existing	0.00000000	-1.45933392	-0.16131722	-0.00000002			
		Wanted	0.00000000	-1.45933392	-0.16131722	-0.00000002			
		Weight	N.A.	0.00000000	0.00000000	0.00000000	0.00000000		
Grid	DD-x	Existing	0.00000000	0.56258457	0.31127430	0.00000002			
		Wanted	0.00000000	0.56258457	0.31127430	0.00000002			
		Weight	N.A.	0.00000000	0.00000000	0.00000000	0.00000000		
Grid	D-z [m]	Existing	0.00000000	0.00000000	0.00000000	0.00000000			
		Wanted	0.00000000	0.00000000	0.00000000	0.00000000			
		Weight	N.A.	0.00000000	0.00000000	0.00000000	0.00000000		
Notes	DD-z	Existing	0.00000000	0.00000000	0.00000000	0.00000000			
		Wanted	0.00000000	0.00000000	0.00000000	0.00000000			
		Weight	N.A.	0.00000000	0.00000000	0.00000000	0.00000000		

V 4.0 18 WinAGILE

Figure 7.4 Screen editor for boundary conditions

The potential boundary conditions β_x , α_x etc. are listed down the left hand side and each one has three lines labelled “Existing”, “Wanted” and “Weight”. The lines with a light yellow background can be edited by double clicking and using the Edit box on the top left of the screen, as is done in the Edit Window. Thus the input value to the line for a given parameter can be edited and the values at intermediate points and at the exit can

also be edited. Equally the weights for each parameter at each point can be edited. It is useful to remember that boundary conditions can be released quickly by adding zero weights and replaced by restoring a finite weight. Scrolling down the page, or dragging the window to a larger size, will bring into view the betatron phases, the emittances or the beam sizes according to the case.

The definition of the boundary conditions can be left at any time by clicking a new item in the menu bar. The choices made up to that point will be retained even if you return temporarily to the Line Window.

7.2.3 Steps...

This dialogue box shows the initial values that will be used by the minimisation routines. This can normally be ignored, but occasionally during the matching process, one loses track of the sizes, in which case this routine provides direct access and the possibility to edit the values.

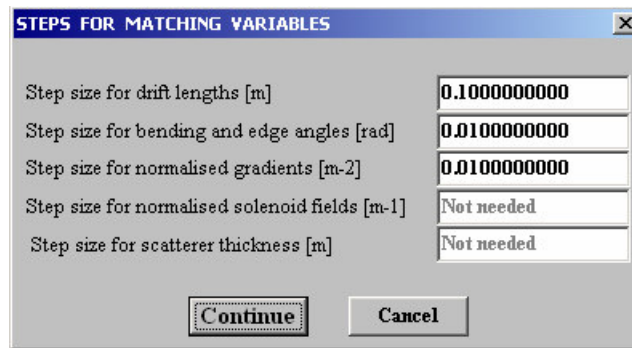


Figure 7.5 Dialogue box for matching steps

7.2.4 Show variables and links

This is a simple dialogue showing the variables that are linked and the link ratios.

7.2.5 Minimisation

This menu item opens the Minimisation Plot Window in which the numerical matching is performed. It is possible to move back and forth between the two windows at any time and to re-enter the routines described above and then to resume matching. The plotting window is described separately in the next section.

7.3 Minimisation Plot Window

The Minimisation Plot Window is divided into four sections. The status of the match is shown in the top box, the centre box shows a graph giving a visual feedback of the Twiss functions or beam sizes as the matching progresses, the right hand box has the global mismatch factor and some instructions at the start of matching, but these are replaced during a minimisation with a graphic representation of how the matching variables are reacting. Finally across the bottom, there are a number of control buttons. Figure 7.6 shows the plot window at the start of a matching session.

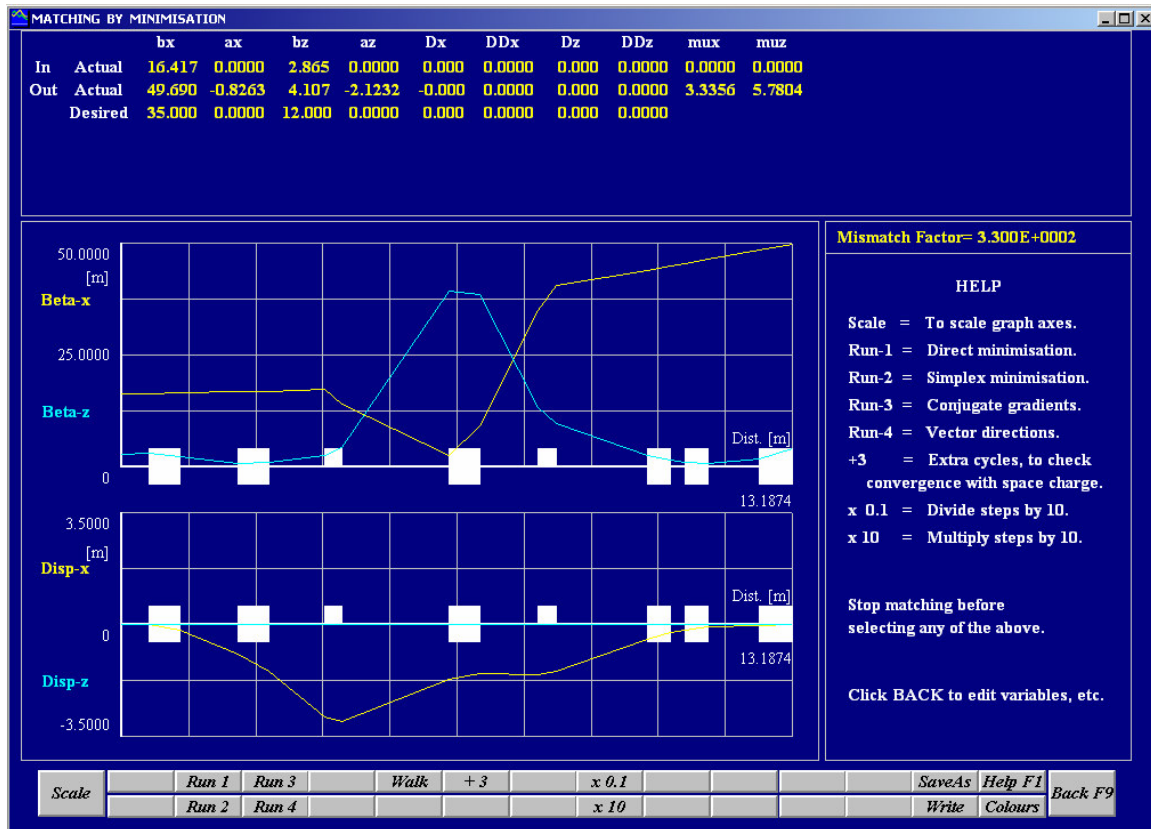


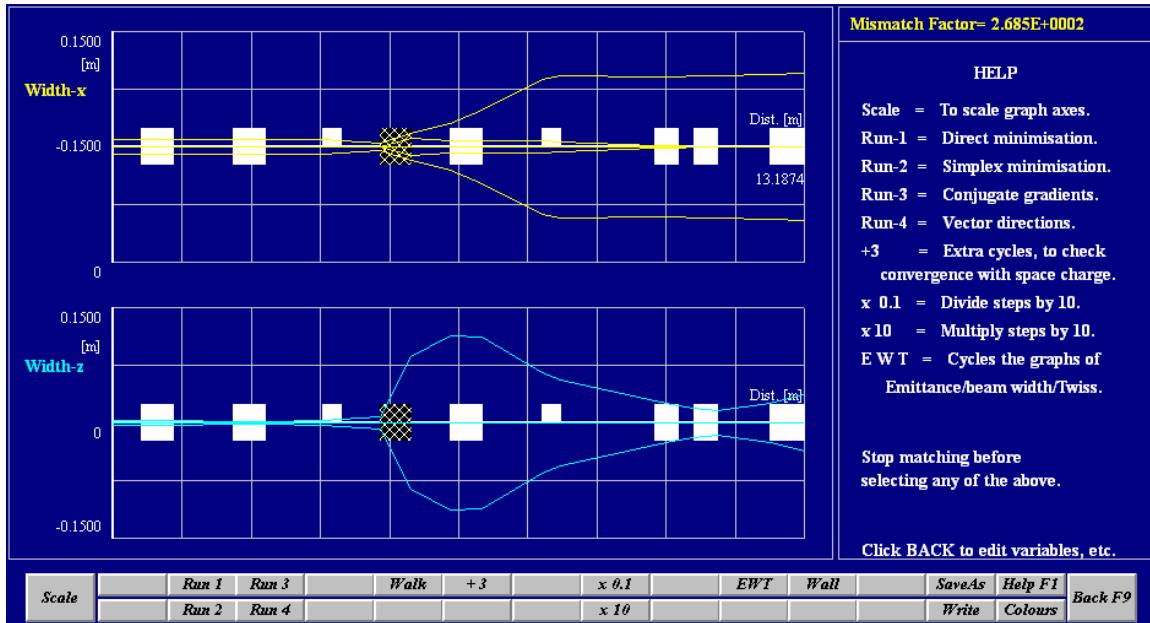
Figure 7.6 The Minimisation Plot Window ready to start matching

- **Scale** allows the user to scale the graph. In most cases, the automatic rescaling, which occurs each time the minimisation is stopped, should be sufficient.
- **Run-1** starts a simple minimisation routine that directly operates on the available matching variables. Although the routine automatically reduces the step size as the minimisation progresses, it is sometimes advantageous to stop and re-start the routine to reset all the variables (while the minimisation is running the button will be labelled “Stop”). Sometimes this routine can be used to de-block the other routines when they become trapped in a local minimum.
- **Run-2** launches the well-known Simplex routine and is perhaps the most efficient out of the routines available. As with Run-1, it is sometimes advantageous to stop and to restart the routine.
- **Run-3** constructs conjugate gradients for the minimisation rather than working with amplitudes. When other routines exhibit instability at small mismatch values, this routine can sometimes improve the match further.
- **Run-4** constructs vectors for the minimisation. This routine is the most recent and is, in most cases, outperformed by the other routines.
- **Walk**. Not available in this version.
- **3+** causes the routine to cycle the lattice three times to ensure that when matching with incoherent space charge the solution is stable.
- **x0.1** reduces the minimisation steps or the size of the initial simplex by a factor 10. Since the routines internally handle their step sizes, this is generally not needed.
- **x10** increases the minimisation steps or the size of the initial simplex by a factor 10. This complements the previous control and also helps to escape from local minima.

- **EWT** only appears when there is a scatterer in the lattice and the option to optimise, either the emittances, or the beam widths, has been chosen. This button allows the user to toggle between the graphs of the Twiss functions, the beam emittances and the beam sizes. An example of the Twiss plot has already been given in Figure 7.6 and examples of the emittance and beam width plots are given in Figure 7.7.



(a) Emittance plot for a single scatterer represented by the hatched rectangle



(b) Beam size plots in a line with a scatterer represented by the hatched rectangle

Figure 7.7 Examples of the emittance and beam width plots used in the minimisation window

- **SaveAs** prompts the user for a new file name and then saves the lattice. Since it is easy to diverge from the wanted situation, it is important to save frequently.

- ***Write, Help and Colours*** provide standard functions.
- ***Back F9*** returns control to the Numerical Matching Window where matching variables, boundary conditions, steps and lattice parameters can be edited. The direct editing of the lattice parameters is done by selecting "Lattice parameters" in the Tables Menu of the Numerical Matching Window and double clicking on the parameter that needs to be changed. Please note that although the program limits the editing to matching parameters and element names, it does not check the validity of the changes, so the user must be careful.

7.4 ***Tables, Graphs and Help***

As mentioned earlier, the lattice elements table can be edited directly in those columns containing matching parameters and element names. Note that the program does not check the validity of changes, so it is easy to cause inconsistencies.

All the other functions offered in the Tables, Graphs and Help menus have been described earlier in Chapter 3.

* * *

Chapter 8 Plot Windows

Much of the calculation power of WinAGILE and the interaction with the user is concentrated in the plot windows. The plot windows provide graphical presentations of the data and in many cases the plots are animated. The various plot windows are accessed either through the Calculation Menus or through the Graphs Menus.

The Minimisation Plot Window has already been described in Chapter 7 (Section 7.3), the Eddy-current Plot Window was described in Chapter 3 (Section 3.4.5) and the Statistics Plot Window was also described in Chapter 3. The remaining plot windows are described in this chapter in an approximate order of familiarity.

8.1 Transverse Twiss Plot Window

8.1.1 Uncoupled, unbunched lattices with no scatterers

The transverse Twiss and dispersion functions for “normal” lattices are perhaps the most used graphs in accelerator design, see Figure 8.1. This window can be accessed from the Line Window, the Ring Window and the Numerical Matching Window.

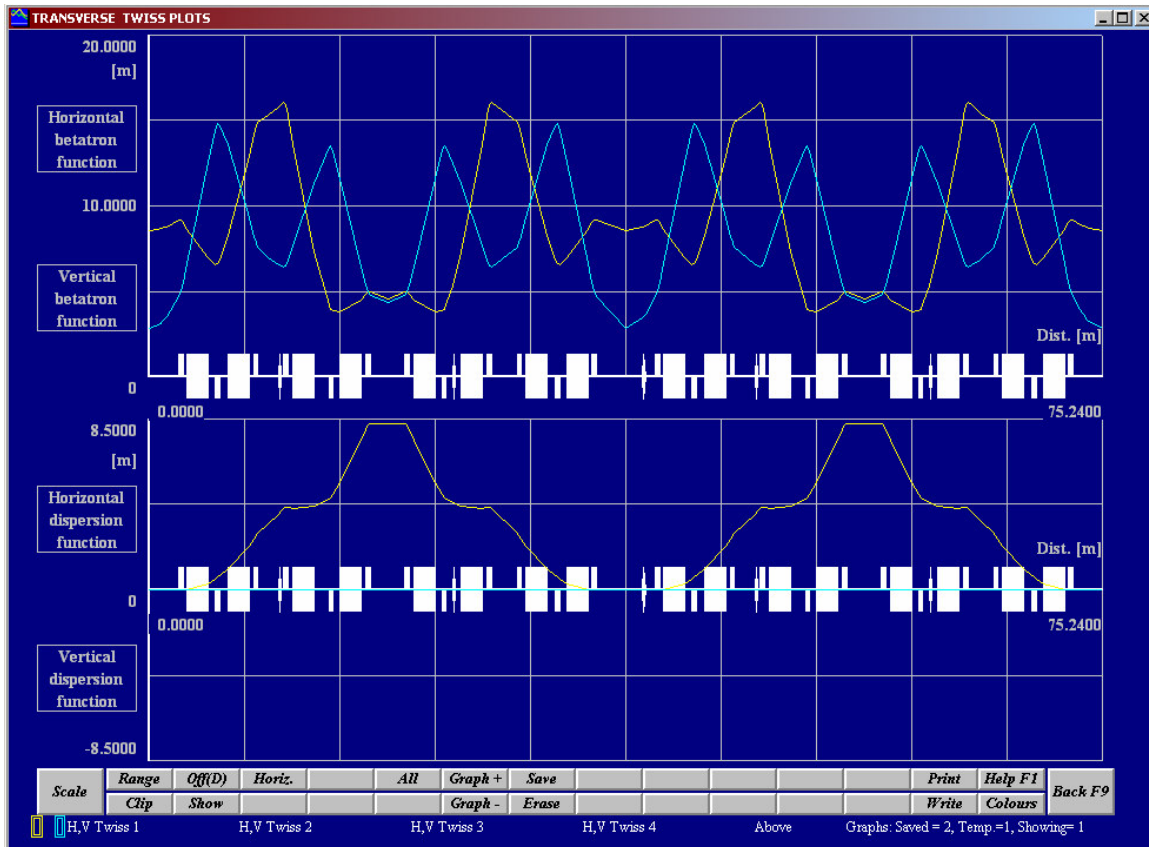


Figure 8.1 Transverse Twiss and dispersion functions

The upper graph shows the horizontal and vertical betatron amplitude functions and the lower graph shows the horizontal and vertical dispersion functions. The abscissa

of both graphs is distance along the lattice and there are symbols that indicate the type of lattice unit at each position, see Section 8.15 for the key. This bar of symbols is common to many of the plot windows.

Across the bottom of the screen there are thirty control buttons. Some buttons offer a single function like “scaling”, others toggle between two or more states like “phase space” and “real space”. All toggle buttons show a label that indicates what will be obtained on clicking and not what is currently active. Many of the control buttons are common to nearly all the plot windows and the same buttons, or similar buttons, are usually grouped in the same positions.

On the right hand side below the control buttons there is a brief indication of the number of graphs stored, whether the most recent graph is a temporary graph or is already stored and which graph is currently being displayed.

The buttons in the Transverse Twiss Plot Window are listed below:

- **Scale** allows the user to change the vertical scale of the graphs. This may be needed if several graphs are to be compared and the automatic scaling routine has to be over-ridden.
- **Range** allows the user to select the part of the lattice to be displayed on the horizontal axis.
- **Clip** allows the user to limit the drawing of the graph lines to a limited section of the lattice.
- **Off (D)/On (D)** toggles the dispersion graphs on/off.
- **Horiz./Vert./Horiz.+vert.** allows the user to toggle between the horizontal graphs by themselves, the vertical graphs by themselves and the horizontal and vertical graphs together, as in Figure 8.1.
- **Show** gives the user some control over the types of elements that are symbolised along the axis. The elements that can be optionally ignored are horizontal and vertical beam position monitors, horizontal and vertical orbit correctors, sextupoles, multipoles, solenoids and skew quadrupoles. All other elements are always indicated by an icon.
- **All/Single** toggles between showing a single graph and all the graphs of this type currently held in memory. When showing all graphs, the first 8 are colour coded according to the small rectangular icons below the control buttons. Graphs 9 and above all have light grey traces. Note that if the clipping limits are set before saving, then each graph can be individually tailored. Alternatively, the graphs can be clipped to a common value when they are collectively displayed.
- **Graph +** steps up through the graphs currently held in memory. This function requires the single graph mode to be active and will automatically switch to that mode. The graphs can be identified by their colours and by the index number displayed below the control buttons on the right hand side.
- **Graph –** steps down through the graphs currently held in memory. This function requires the single graph mode to be active and will automatically switch to that mode. The graphs can be identified by their colours and by the index number displayed below the control buttons on the right hand side.
- **Save** saves the currently displayed graph, unless it is already saved in which case the command is ignored. Up to 25 graphs can be stored, after which the oldest files are erased to make way for the newer ones. This function with the **Button All** makes it possible to superimpose multiple graphs with say varying tune or different space charge conditions. When graphs are saved in this way, the files are only kept while they are consistent with the current lattice. Loading a new lattice, for example, causes all the temporary files to be purged (there is one exception for geometry files, see Section 8.4).

- **Erase** erases the current graph if the single graph mode is active and all the graphs if the combined graph mode is active.
- **Print** opens a dialogue box from which the user can control how the program generates the image, for example, the black & white option will use dashed lines and solid lines to distinguish graph lines whereas the colour option will use colours. The routine will then instruct the default printer unless the user clicks the Printer button to change printers, or give more advanced instructions. This routine is common to all plot windows.
- **Write** opens a dialogue box giving three options for writing image files. The first option is a black & white drawing of the screen graph in “eps” format, the second is a colour drawing of the screen graph also in “eps” format and the third is a bitmap image directly taken from the screen. This routine is common to all plot windows.
- **Help** gives access to the help file. This routine is common to all plot windows.
- **Colour** allows the user to independently control the colours used on the screen, the colours used by the printer and the colours used for the “eps” files, via the dialogue boxes shown in Figure 8.2. Select the item to be changed in the corresponding list box. The current colour of this item will then appear in the colour sample box on the right hand side level with the list box. Double click the colour sample box to open the second dialogue box from which a new colour can be chosen. This routine is common to all plot windows.
- **Back F9** returns the control to the prior window.

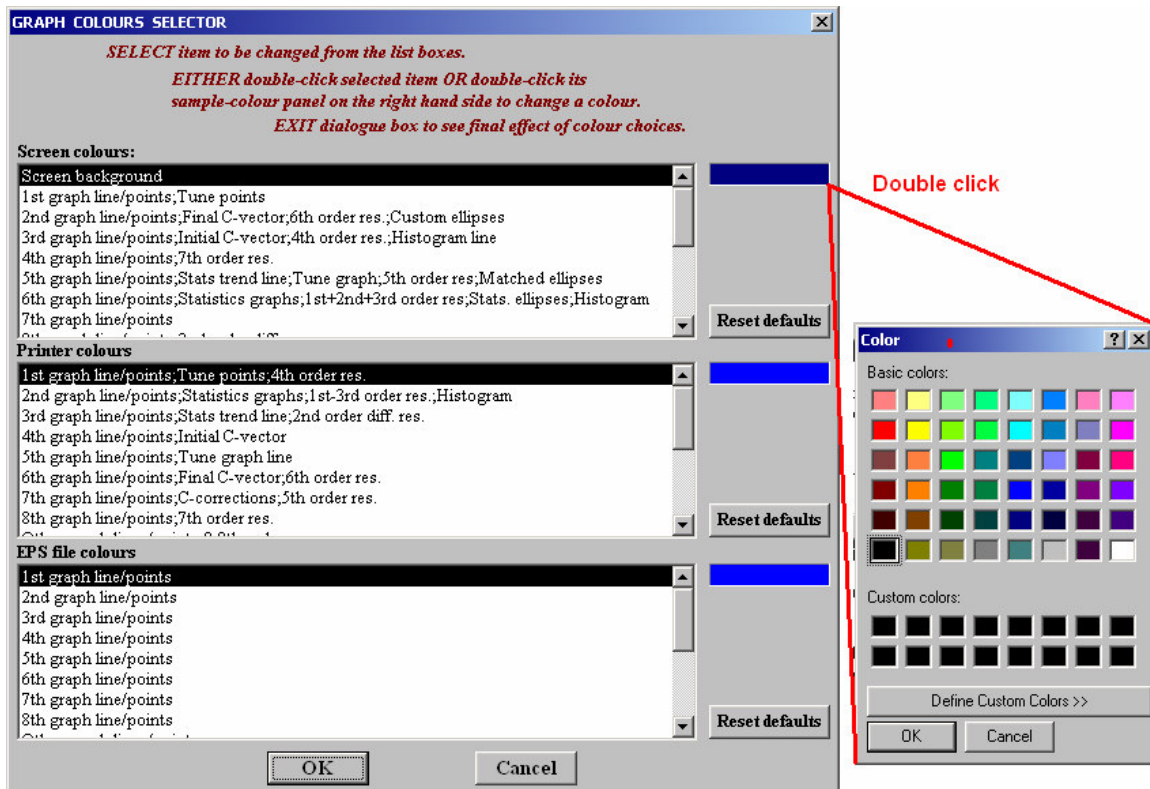


Figure 8.2 Dialogue box for selecting colours for the screen, printer and “eps” files

8.1.2 Twiss-styled, transverse, bunched-beam parameters

The transverse Twiss plot window is also used for the Twiss-styled, bunched-beam parameters. This is a rather special use of the Twiss formalism that can lead to confusion and apparent conflict with the normal transverse Twiss parameters. It is

included here for comparison with programs such as Trace-3D and is only available when RF is applied to the beam and has to be specifically selected by the user from the Tables and Graphs menus. Some additional explanation is given in Section 5.5.5, but for a full derivation please refer to the WinAGILE physics manual.

When showing the bunched parameters, the control button for toggling the dispersion graphs is removed and only the upper graph for the betatron amplitude functions is shown.

8.1.3 Twiss parameters with scattering

The transverse Twiss plot window is also used for the Twiss parameters in lines with scattering (scatterers are excluded from rings). This is a rather special use of the Twiss formalism that is valid for beam envelopes, but not for tracking individual particles. Please refer to the WinAGILE physics manual for a fuller description and derivation. The plot window is standard, but the betatron amplitude functions may look a little strange, see for example Figure 8.3.

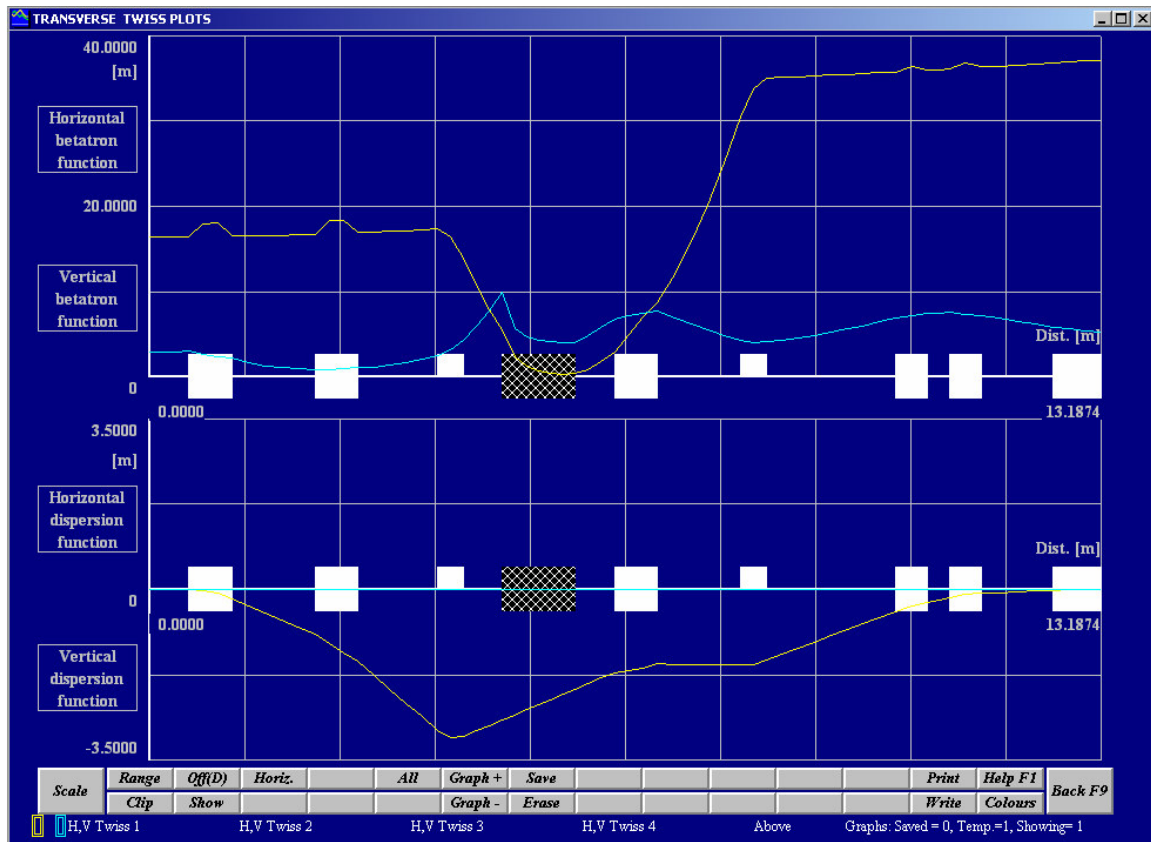


Figure 8.3 Twiss functions through a scatterer

When a scatterer is traversed, the emittance grows sharply due to the increased divergence of the beam, but the short distance in the scatterer does not allow the particles to change position significantly. The combination of the sharp increase in emittance with a quasi-constant beam size means that the betatron amplitude functions in both planes must decrease sharply, which may appear strange at first. Further along the lattice, the

betatron amplitudes increase as the divergence of the beam is focused and the beam size increases.

8.2 Longitudinal Twiss Plot Window

The longitudinal Twiss functions are rarely used in contrast to the transverse Twiss functions. They appear in programs such as Trace-3D, but they are only valid for the relatively small linear region within the RF bucket. An example is shown in Figure 8.4. The top graph shows the longitudinal betatron amplitude and the lower half of the screen shows the beam ellipses in longitudinal phase space at three positions: the entry, a movable position along the lattice and the exit. In the example, the adiabatic damping has reduced the longitudinal emittance. The cavity is the brown-coloured icon.

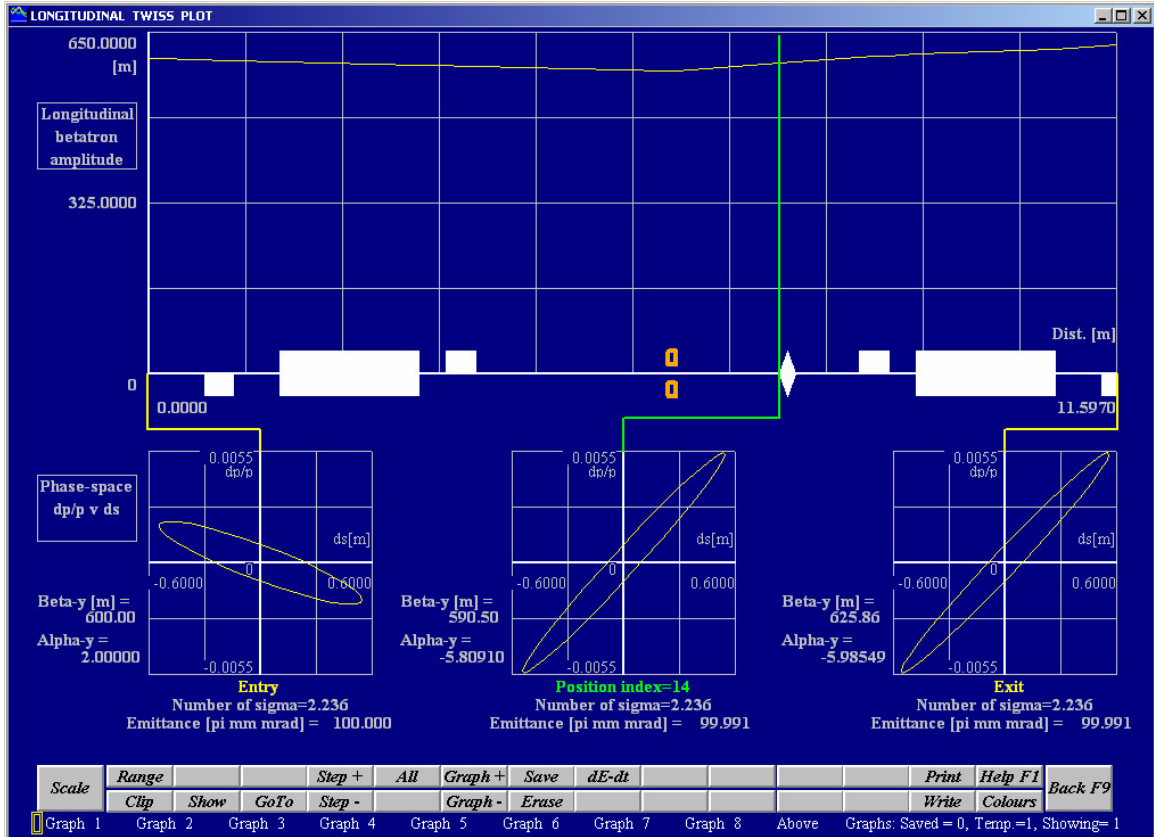


Figure 8.4 The longitudinal Twiss plot window

Most of the control buttons have already been described in the previous sections. The new buttons are:

- **GoTo** allows the user to move the central phase-space graph directly to any point along the lattice. The phase-space graph shows the beam ellipse, the emittance and the α and β values.
- **Step +** and **Step -** allows the user to move through the lattice looking at the longitudinal Twiss functions.
- **dE-dt/dE-deg/dp/p-ds** toggles the phase-space graphs through three pairs of variables: energy-time/energy-phase/relative momentum deviation-lag.

8.3 Dispersion Plot Window

The dispersion functions are tabulated and displayed graphically in the transverse Twiss plot window, but some further insight into the construction of dispersion bumps and achromatic lattices can be obtained by looking at the normalised functions and displaying the data in phase space. Figure 8.5 shows an example of a “missing-magnet” dispersion suppressor. The main loop at large amplitudes is due to the dispersion vector in the FODO lattice of the arc, the smaller loop at half the amplitude of the main loop is due to the missing magnet and the small excursions around the zero are due to the residual dispersion error in the straight section.

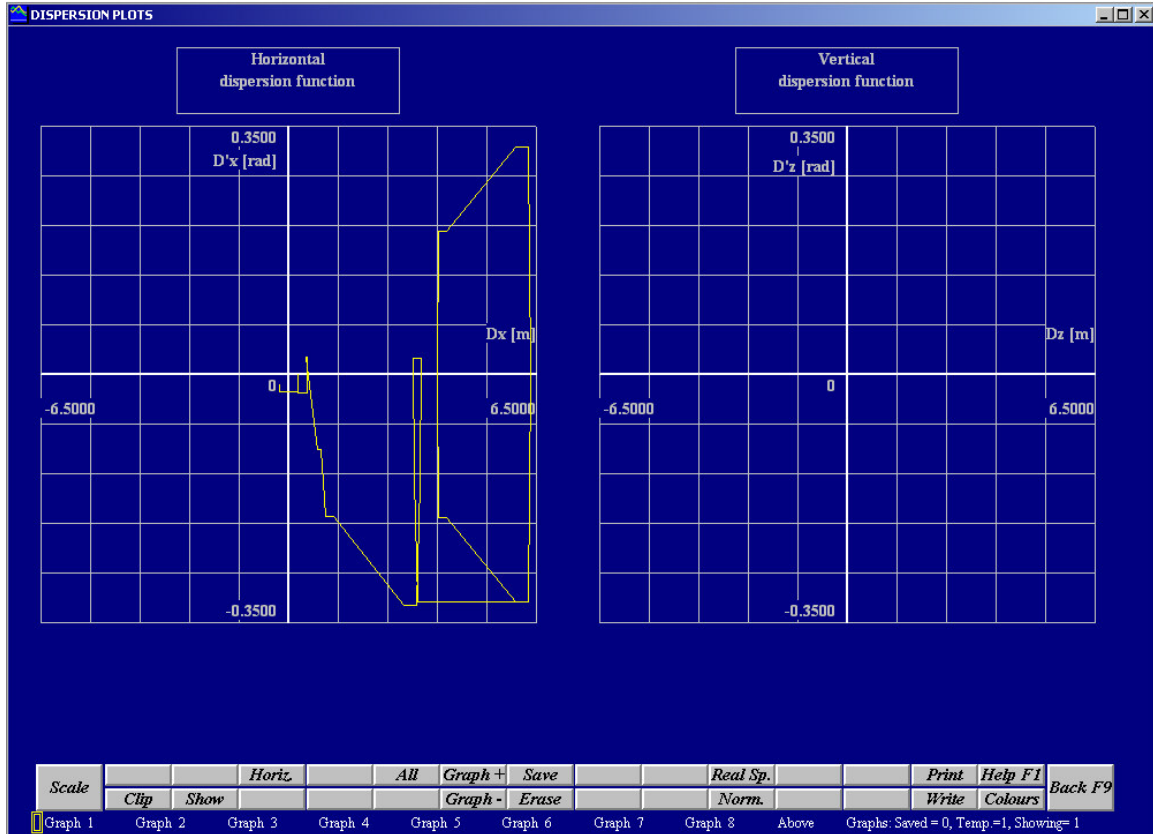


Figure 8.5 Dispersion plot window

8.4 Geometry Plot Window

8.4.1 2D geometry (X,Y) ; (Z,X); (Y,Z)

The 2D geometry plots of the lattice are drawn on a square grid of 10 steps by 10 steps. The step size of the grid is shown in metres at the top left corner of the grid. Figure 8.6 shows an example of a ring. The lengths of the elements correspond to the effective magnetic lengths given in the lattice definition. The widths can be set by the user (see below).

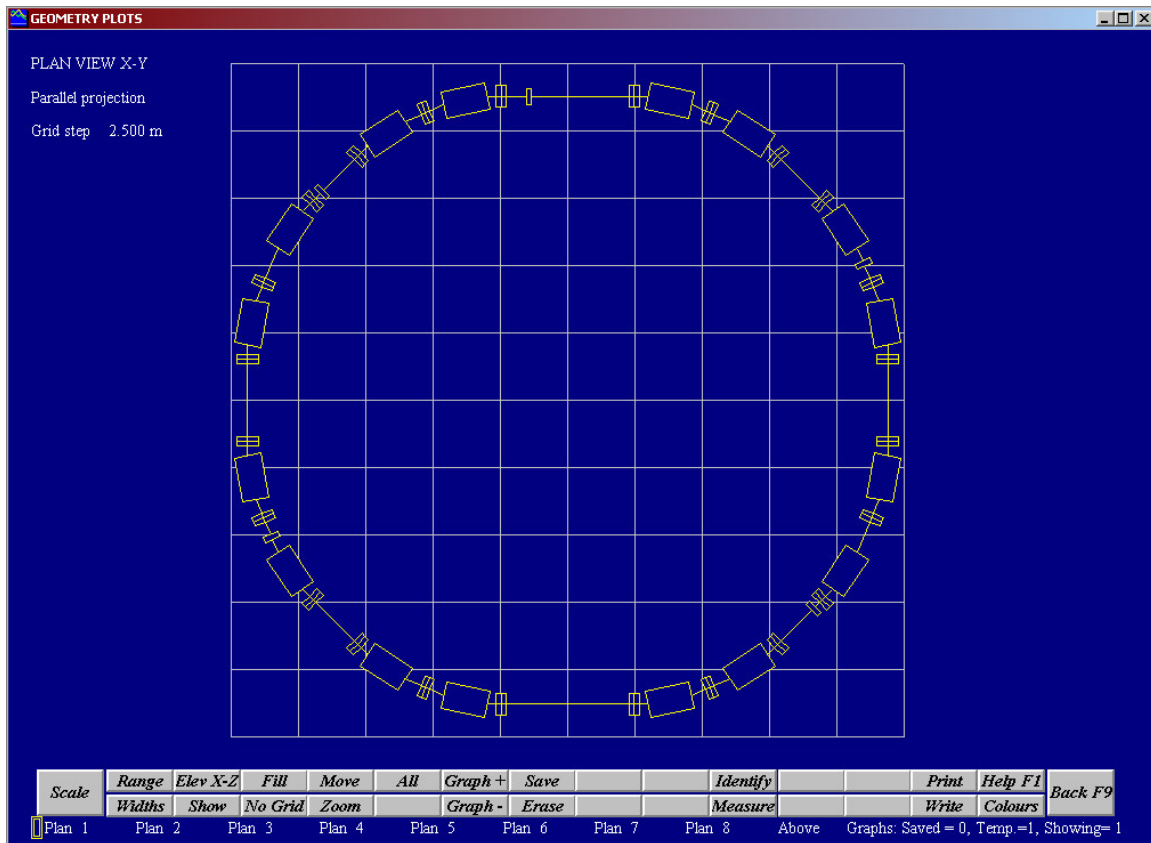


Figure 8.6 2D geometry plan view of a ring

Many of the control buttons have already been described in the previous sections. The new or modified buttons are:

- **Scale** opens the dialogue box shown in Figure 8.7. The grid can be customised, the grid size can be set to a precise value and the image can be shifted by precise offsets. The defaults settings can also be reinstated.
- **Widths** allows the user to set the half-widths in metres of the elements appearing in the lattice plot. The elements are represented by quadrilaterals centred on the beam path. Edge angles are included. For units with a strong curvature, it is better to subdivide the element which will then appear curved in the plot. A zero half-beam width leaves only the beam track visible.
- **Elev X-Z/Side Y-Z/Plan X-Y** allows the user to toggle between the three orthogonal 2D views of the lattice.
- **Hidden/Fill/Wire** toggles the plots through three views: a “wire-framework”, hidden lines removed and solid volumes filled.
- **No Grid /Grid** toggles the grid on and off.
- **Move** changes the cursor to a hand with an extended finger. Use this icon to push the lattice in any direction desired.
- **Zoom** allows the user to select a part of the image and to enlarge it or to return to the default settings. To facilitate zooming, it is possible to select the area to be viewed by dragging the left mouse button across the region. A box is ‘rubber-banded’ to indicate the area selected. When the mouse button is released a dialogue box appears with the three choices: ‘Zoom to selected box’,

'Reset defaults' and 'Cancel'. To set the view to a specific scale size and position use the **Button Scale** mentioned above.

- **Identitfy** use to reset the default cursor after using **Move, Measure or Zoom**. When the default cursor is active, the elements and straight sections can be identified by double-clicking close to the boundaries of the object, see Figure 8.8.
- **Measure** changes the mouse cursor to 'cross-wires' and allows the user to measure distances directly on the geometry plot by dragging the mouse cursor with the left hand button depressed.

ZOOM, SCALE AND POSITION GRAPH

ZOOM :
Grid spacing upon which lattice to be drawn [m]

POSITION :
Shift grid in X-direction by [m]
Shift grid in Y-direction by [m]
NOTE: Shifts are w.r.t. default grid position

LINES :
Set 1 to 10 pixels for graph line(s) [dflt = 1]
Set 1 to 5 pixels for grid line thickness [dflt = 1]
Set number of intervals in graph grid (same in both directions)

RESET DEFAULTS :
☐ Reset lattice grid size and position (overrides Zoom and Position).

OK **Cancel**

NOTES:
- The printer and post script outputs are correspondingly scaled.
- Setting the # of grid intervals to unity produces a bounding box.
- Setting the # of grid intervals to zero removes the grid.
[If #=0, then effective grid size is that for #=1.]

Figure 8.7 Scaling of 2D geometry plots

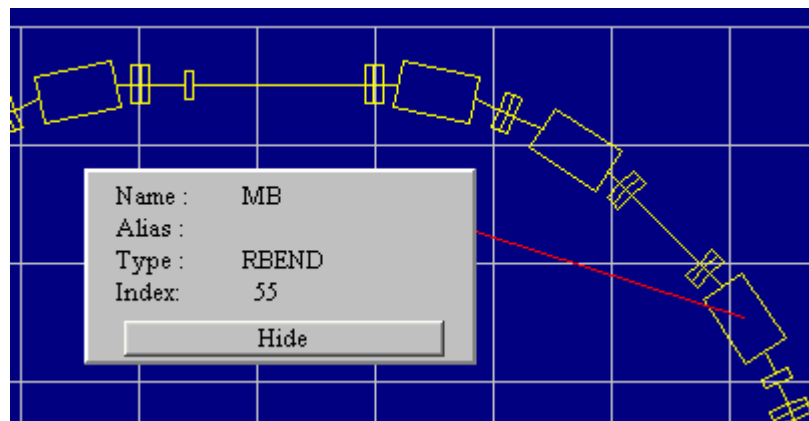


Figure 8.8 Use of the "Identify" button

8.4.2 3D geometry

This window provides a 3D view of the lattice, see Figure 8.9, that can be customised in a number of ways and then stored or printed directly. See Section 8.15 for a key to the symbols used to represent the elements.

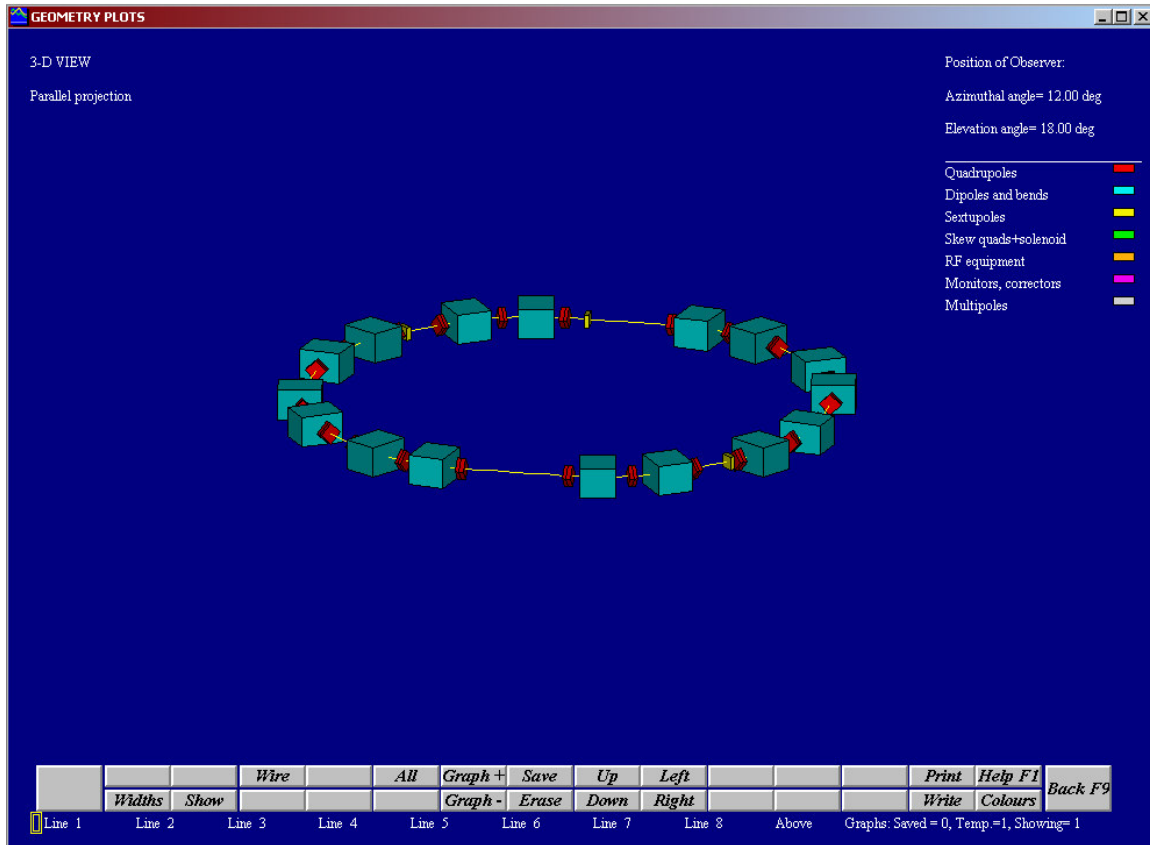


Figure 8.9 3D geometry plot window

Many of the control buttons have already been described in the previous sections. The new buttons are:

- **Up and Down** move the position of the observer upwards (and over) or downwards (and under) respectively, or alternatively, one can interpret the actions as the lattice tipping in the opposite sense.
- **Left and Right** move the observer round the outside of the lattice in the leftwards (clockwise) or rightwards (anticlockwise) directions respectively, or alternatively, one can interpret the actions as the lattice turning in the opposite sense.

8.4.3 Trajectories in 2D geometry (X,Y) ; (Z,X); (Y,Z)

This plot window is also used for plotting particle tracks in the real-space survey co-ordinates system. This is restricted to viewing the tracks in the drift spaces. Since the trajectories are essentially close to the axis, it is usually necessary to zoom in on a particular straight section to see the physical positions of the trajectories, see Figure 8.10. This feature is intended for looking at injection or extraction trajectories.

To show several tracks, calculate and store the tracks either directly in the 2D geometry window, or store the tracks in the particle trajectory window and then move to the 2D geometry window once all the tracks have been saved. Because of the dual use of arrays, it is possible (but not intended) to show the current closed orbit in the 2D geometry window as well.

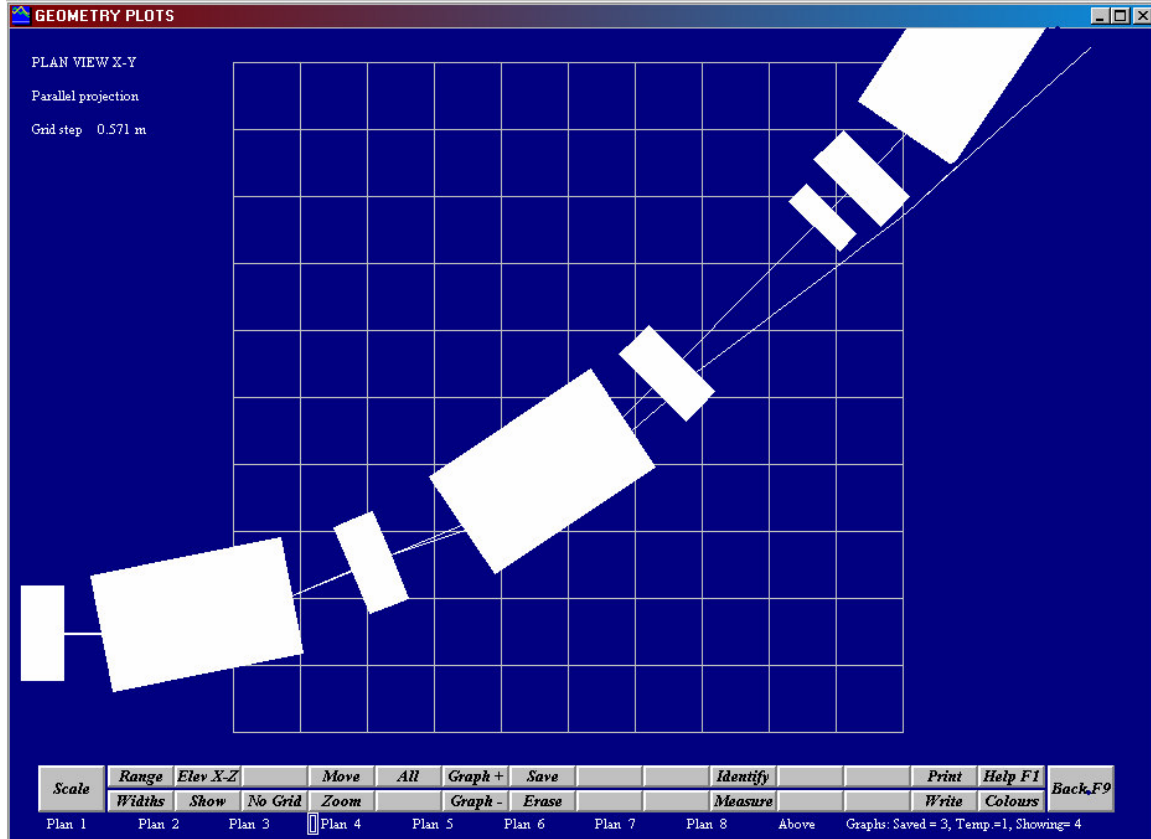


Figure 8.10 An extraction trajectory shown in the 2D geometry plot window

8.5 Closed-orbit Plot Window

8.5.1 Machine generated closed orbits

Figure 8.11 shows an example of a closed-orbit generated internally by WinAGILE by randomly moving all the quadrupoles in a ring lattice according to a Gaussian distribution with a sigma of 2 mm. The plot window not only presents the data graphically, but it also allows the user to experiment with closed-orbit corrections. Most of the control buttons have already been described in the previous sections. The two new buttons are:

- **Button Norm./Unnorm.** toggles the display between real-space, orbit excursions and normalised excursions. In some cases, it is possible to identify single, large field errors, which appear as a cusp in the normalised curve.
- **Button COCorr** allows the user to experiment with the correction of the closed-orbit. Since this is an extremely important task it is described separately in Section 8.5.3.

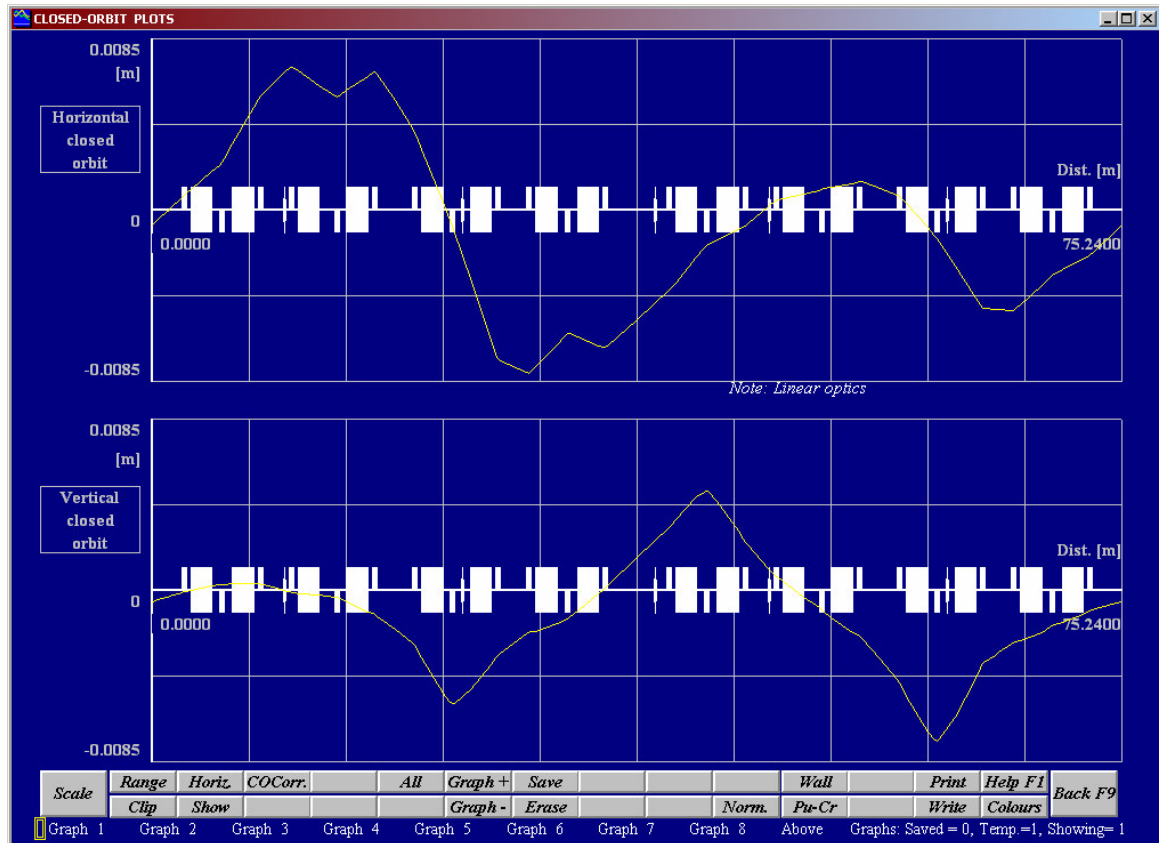


Figure 8.11 Closed-orbit plot window with an example orbit generated by WinAGILE from random errors

8.5.2 Measured closed orbits

Although WinAGILE is primarily meant for studying lattices, it can also be used to correct measured closed orbits. First, the measured closed orbit must be loaded as described in Chapter 5 in Section 5.2.6. After loading, the program will automatically open the Closed-orbit Plot Window and exhibit the measured orbit. An example is shown in Figure 8.12.

Note that measured closed orbits can only be treated in one plane at a time and for this reason the **Button Horiz./Vert./Horiz.+Vert.** is disabled. Note also that the beam excursions are represented by a bar chart with the bars positioned at the beam position monitors. Unlike the machine-generated orbit in Figure 8.11, a measured closed orbit is only known at the monitors and elsewhere it is only possible to estimate the beam excursion. It can also be seen in Figure 8.12 that the **Button Pu-Cr** is disabled for measured closed orbits.

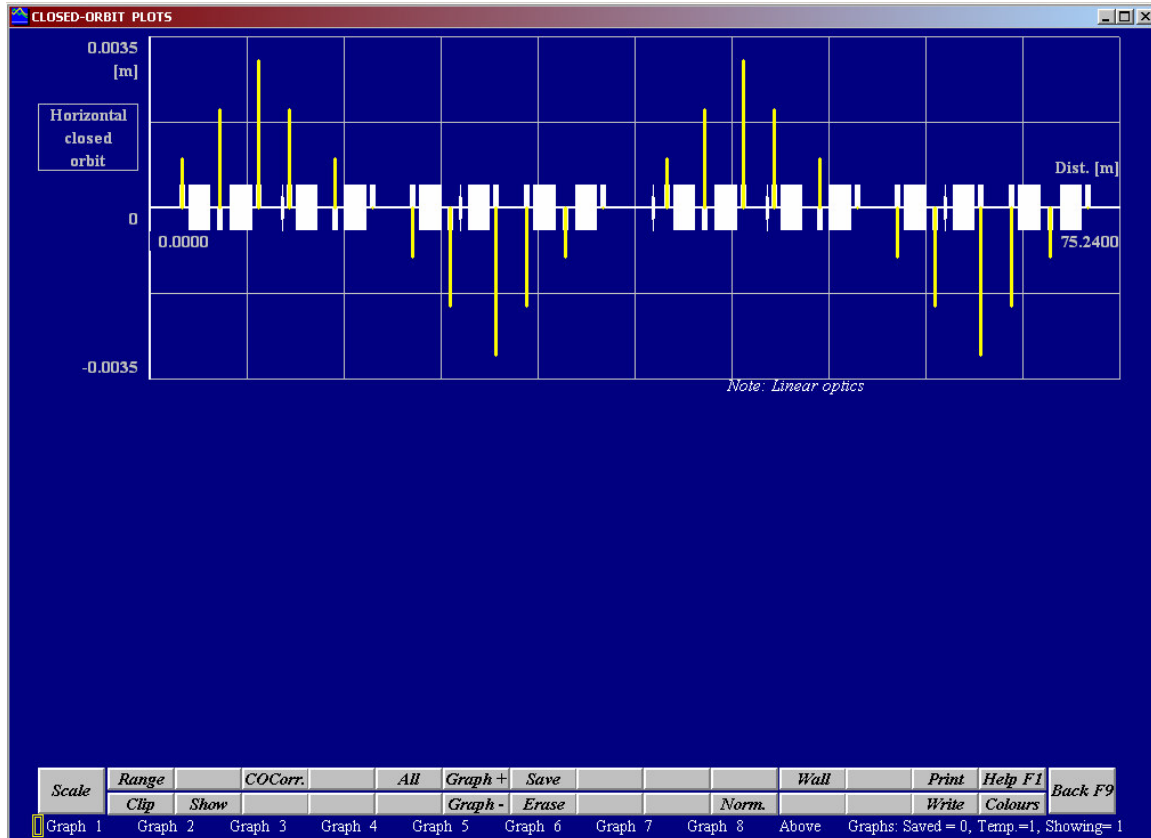


Figure 8.12 Closed-orbit plot window with an example of a measured closed orbit

8.5.3 Correction of closed orbits

On clicking the **Button COCCorr**, the dialogue box shown in Figure 8.13 is opened. To correct an orbit, the following steps must be completed, although there is some flexibility in the order chosen.

- **Step 1. Choice of the plane to be corrected.** For measured orbits, this is determined in advance when the measured data is entered.
- **Step 2. Choice of the monitor option.** In accordance with the plane chosen for correction, the program will check if suitable monitors have been placed in the lattice (types HPU and HVPU for the horizontal plane and VPU and HVPU for the vertical plane). Since the monitors are not always known at an early stage of a study, there is a backup mode that uses the entry points to all lattice quadrupoles as monitor positions. According to what is available, it is necessary to choose either the monitors in the lattice or the backup mode using the quadrupole positions. Do not subdivide the quadrupoles as this introduces a lot of redundant monitors.
- **Step 3. Choice of the corrector option.** The correctors are treated in the same way as the monitors. In Figure 8.13, for example, the program has not found any correctors in the lattice and only the quadrupole backup mode is possible.
- **Step 4. Editing monitors and correctors.** There are many reasons why it may be necessary to remove monitors and/or correctors from the correction routine. This function is described later in Section 8.5.4.
- **Step 5. Number of correctors to be used.** Although there may be a large number of correctors available, it is usually advantageous to check if a small number of correctors will correct the orbit.

This increases operational reliability because there are fewer power converters working and it increases the currents delivered by those that are working, helping to prevent instabilities in converters that are required to control close to zero. In the example in Figure 8.13, the routine has been asked to tabulate the expected result for 1 to 5 correctors.

- **Step 6 Computational method.** The user has three inversion techniques available for experimentation.
- **Step 7. Compute.** Once all the preparatory steps have been completed the routine can calculate the estimated reductions in the closed-orbit distortion. The top line in the list box shows the initial uncorrected orbit. The subsequent lines show what can be expected as the number of correctors increases one by one.
- **Step 8 Add a corrector.** This button allows the user to add another corrector if he feels that there is still some potential for improving the correction. At a certain point however, numerical errors, measurement errors and setting errors will be more influential than adding more correctors.
- **Step 9 OK.** To select a correction, highlight the correction in the list and click OK. This will apply the correction and display the closed orbit. It is usual to choose the lowest number of correctors that delivers an acceptable solution. Judging what is acceptable is based on the three parameters: the peak-to-peak distortion, the average distortion and the rms distortion.

CORRECTION OF A CLOSED ORBIT

CHOOSE PLANE :

☒ Correct HORIZONTAL plane
☐ Correct VERTICAL plane

CHOOSE MONITOR OPTION :

☒ Use elements of types HPU, VPU, HVPU
☐ Use entries to quadrupoles for positions (backup mode)

CHOOSE CORRECTOR OPTION :

☐ Use elements of types HCORR, VCORR, HVCOR
☒ Use entries to quadrupoles for positions (backup mode)

Edit monitor and/or corrector lists

NO. OF CORRECTORS TO BE USED :

Enter min. no. of correctors to be used
Enter max. no. of correctors to be used
Total no. of monitors available =
Total no. of correctors available =

COMPUTATIONAL METHOD :

☒ Least Squares Fit + Gauss-Jordan (fast)
☐ Singular Value Decomposition Fit (safe)
☐ Least Squares Fit + Householder Trans. + Tri-diag.

NOTES :

Max. size of "monitor" array =
Max. size of "corrector" array =

Compute **Add a corrector**

# corr	Pk-Pk [m]	Mean [m]	RMS [m]
0	0.019597	0.001271	0.005624
1	0.007029	0.000970	0.001953
2	0.005292	0.000957	0.001245
3	0.005260	0.000952	0.001244
4	0.005298	0.000953	0.001245
5	0.005213	0.000855	0.001230

SELECT ONE SOLUTION IN THE LIST BOX THEN CLICK OK

OK **Cancel**

- When using the quadrupole backup mode avoid sub-dividing quadrupoles as this leads to an unnecessarily large no. of points.

Figure 8.13 Closed-orbit correction dialogue box

8.5.4 Editing monitor and corrector lists

On clicking the **Button Editing monitor and/or corrector lists**, the dialogue box shown in Figure 8.14 is opened. In the example shown, the monitors in the list box on the left are elements in the lattice, whereas the correctors in the right hand list box are taken from the backup mode that uses the entry points to the quadrupoles. To edit a list, highlight an element and click one of the buttons at the side of the list. To undo the effect of a button, click the button for a second time.

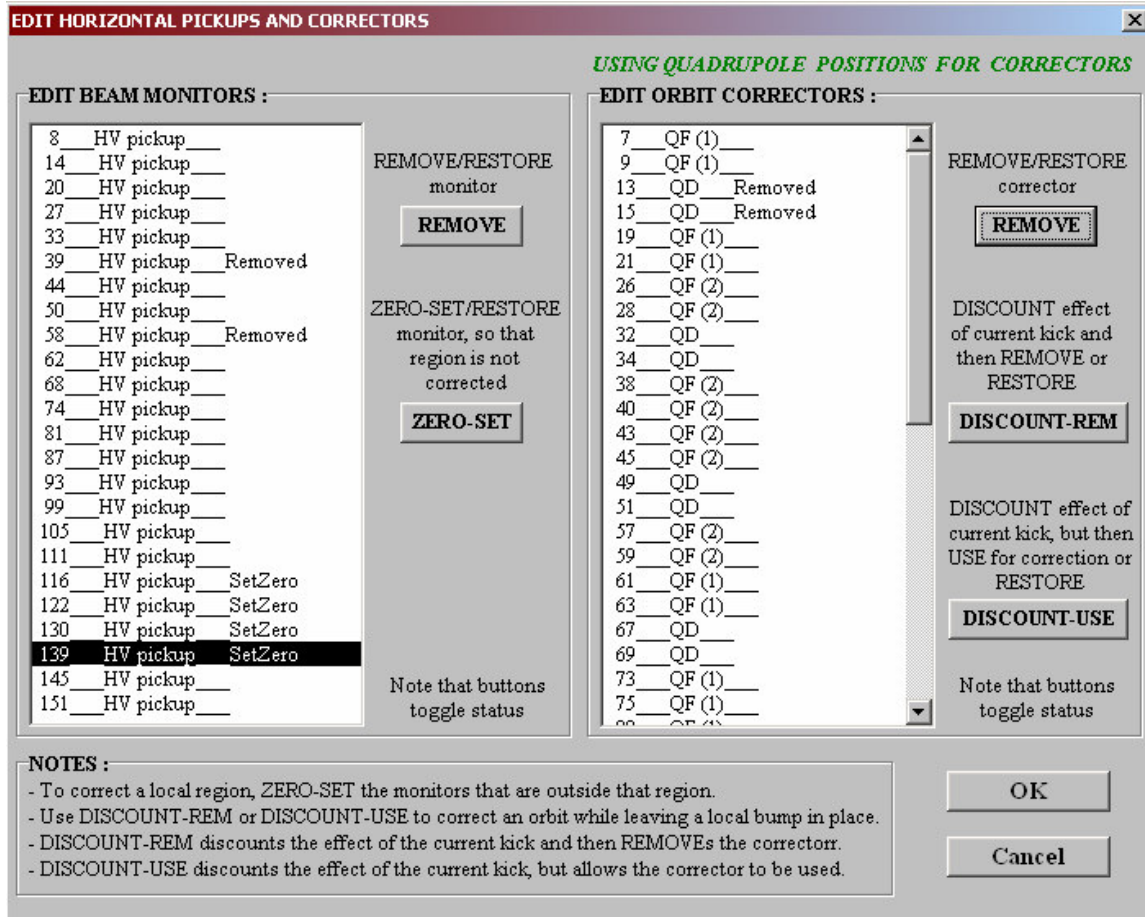


Figure 8.14 Editing monitor and/or corrector lists before correcting an orbit

In the left hand list box for monitors, units 39 and 58 have been “Removed”. This would be done if the monitors had known electrical faults, for example. In addition, units 116 up to 139 have been marked “SetZero”. This means that the correction routine will make no effort to change the orbit in this region, since no distortion is recorded. This would be done, for example, if a large orbit bump had been applied and one wanted to correct the basic orbit without affecting the bump.

In the right hand list box for correctors, units 13 and 155 have been removed in the same way as monitors are removed. For correctors there are two more functions. The “Discount and Remove” function subtracts from the measured orbit the effect of any kick currently given by the corrector and then removes the corrector from the correction array. The “Discount and Use” function subtracts from the measured orbit the effect of any kick

currently given by the corrector, but then leaves the corrector in the array for possible future use.

8.6 Transverse Track Plot Window

Figure 8.15 shows an example of a particle trajectory in the transverse plot window. The trajectory starts at a point inside the lattice. Most of the control buttons in this plot window have already been described in previous sections. The only new button is:

- **Phase Space/ Real Space** toggles the graphs between the real-space presentation shown in Figure 8.15 and a phase-space presentation. Both the phase-space and real-space graphs can be normalised or unnormalised and shown with or without the chamber wall.

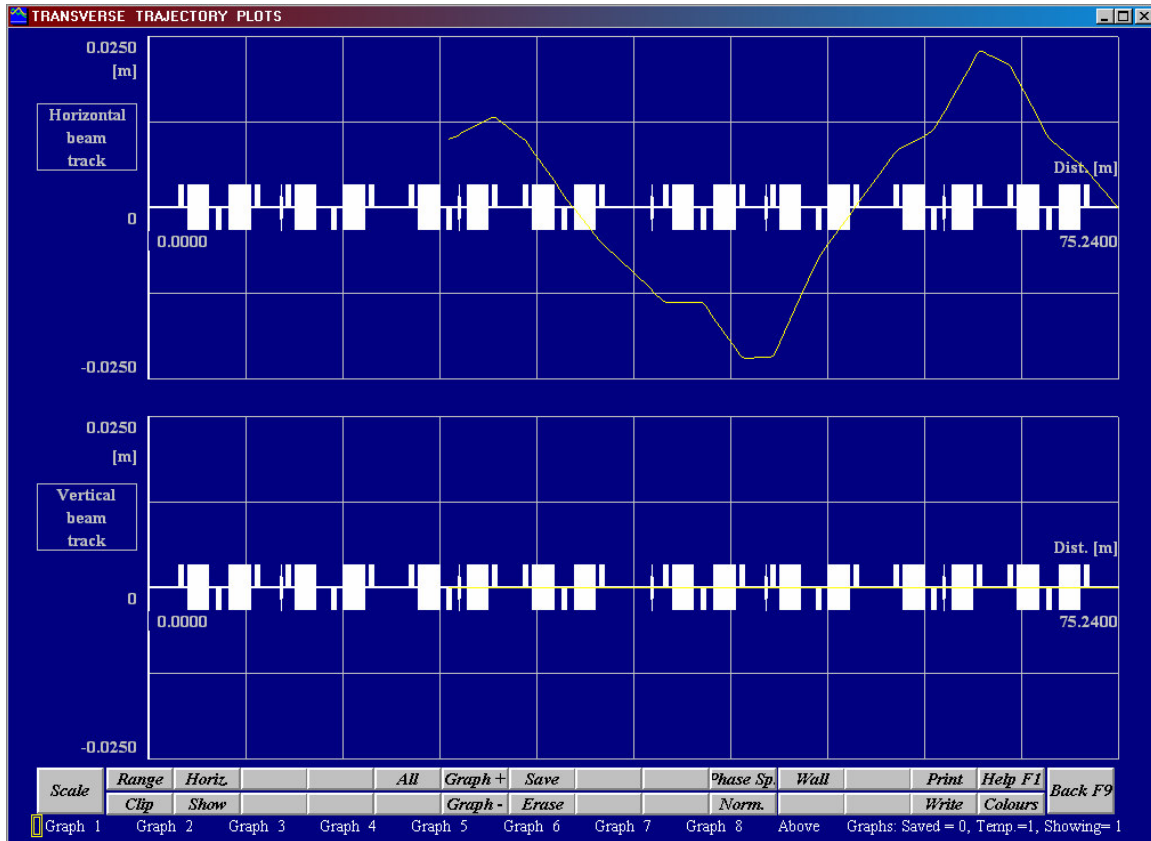


Figure 8.15 Transverse track plot window

8.7 Longitudinal Track Plot Window

Figure 8.16 shows an example of a particle trajectory in the longitudinal plot window. The example shows the tracked particle increasingly leading the synchronous particle in the upper graph and a reduction in the relative momentum deviation from the synchronous particle when crossing the cavity in the lower graph.

This window is only available when the synchronous particle is defined (i.e. when RF is on and the beam is bunched). Most of the control buttons in this plot window have already been described in previous sections. The only new buttons is:

- **Upper/ Lower / Both** toggles the graphs that are shown between the upper graph alone, the lower graph alone and both graphs being shown together.

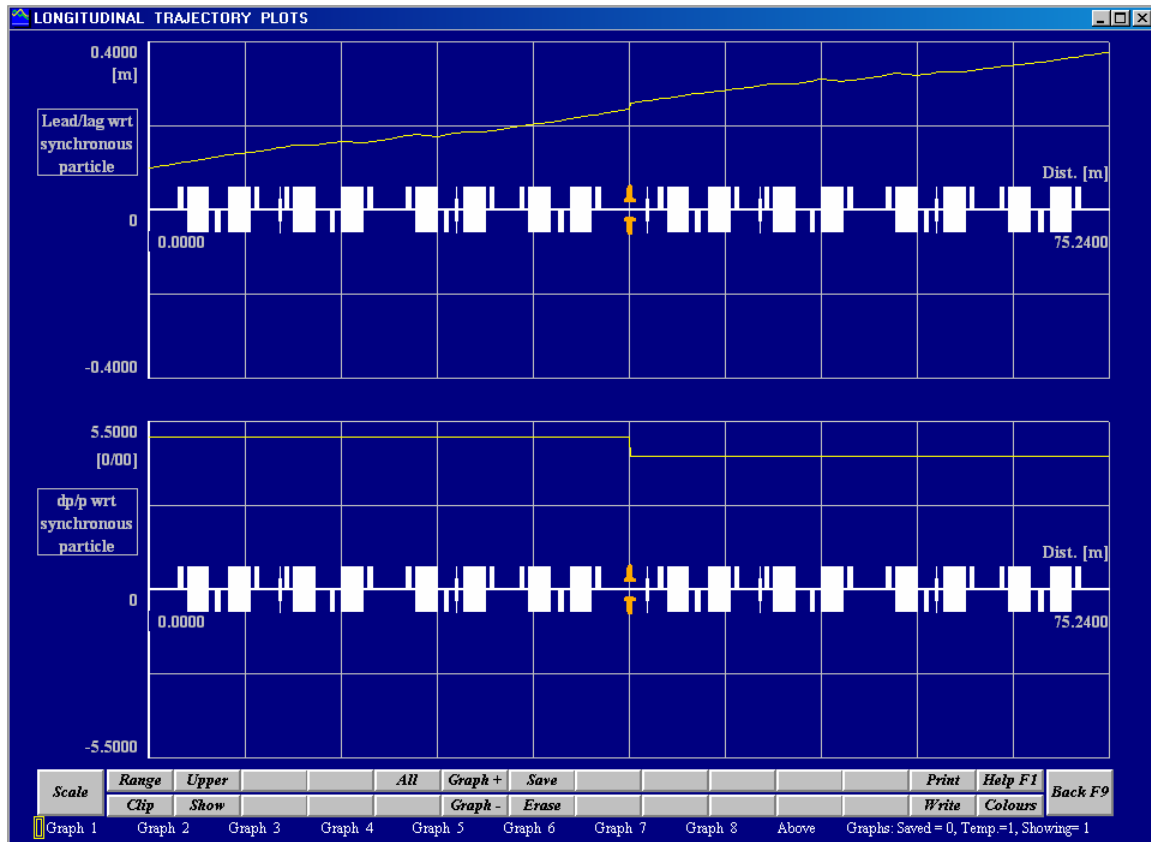


Figure 8.16 Longitudinal track plot window

8.8 Transverse Envelope Plot Window

This window shows the horizontal and vertical transverse beam envelopes along the lattice, see Figure 8.17. The yellow traces indicate the beam envelopes and the light blue traces indicate the vacuum chamber wall. In the horizontal plane, the inner yellow traces indicate the extent of the envelope due to the momentum spread in the beam. The buttons below the graphs provide a number of additional features and controls.

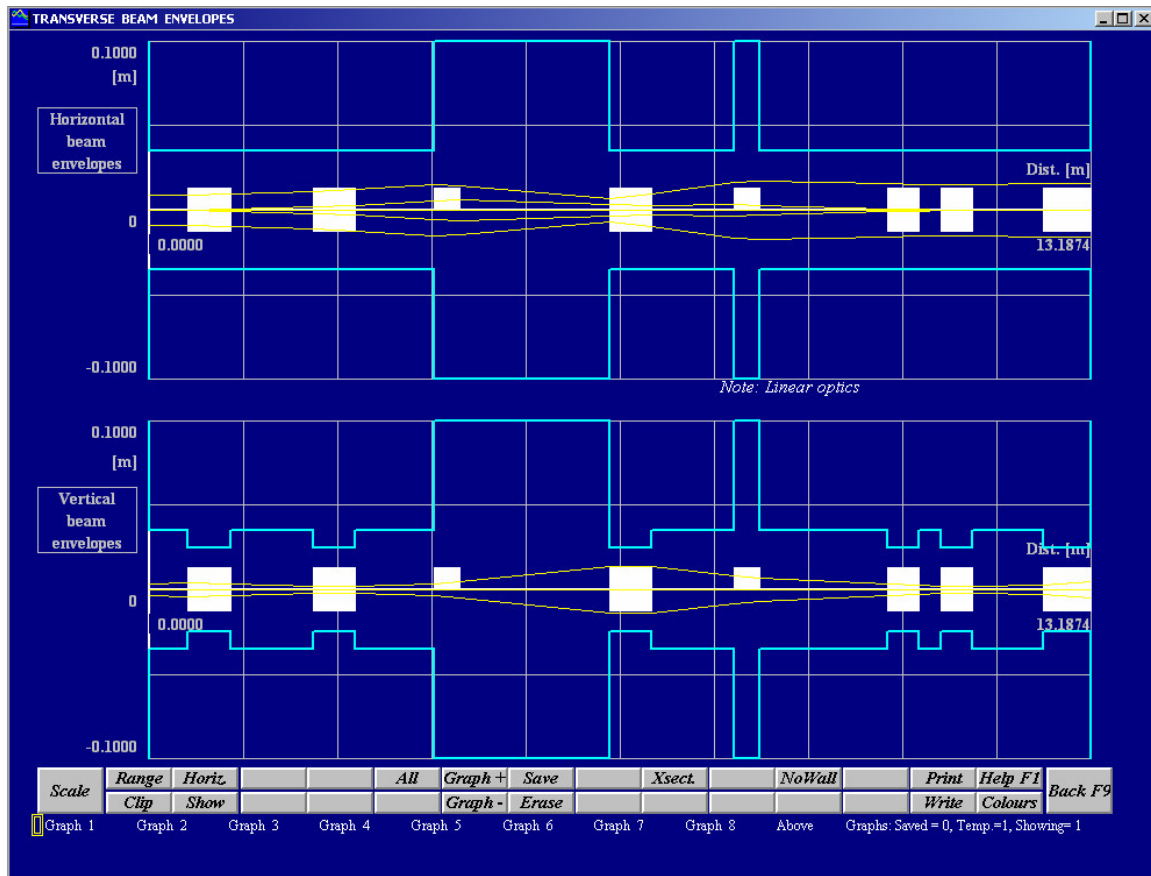


Figure 8.17 Transverse beam envelopes with vacuum chamber

- **Xsect/Envel** switches between showing the envelope along the lattice (Figure 8.17) and showing the beam cross-section either in real space or phase space. The former is a static display, whereas the latter gives an animated picture of the beam cross-section as it moves through the lattice. The screen showing the beam cross-section is effectively a new plot window and is described separately below.
- **No Wall/Wall** toggles the display between showing and not showing the boundary wall. The “wall” is the boundary against which the beam will be lost. In most cases, this is the vacuum chamber, but it can also be the wires of an electric septum or a dump block for example.

As mentioned above, the cross-section/envelope button (Xsect/Envel) opens what is effectively a new plot window, see Figure 8.18. In this example, the magenta line in the left hand graph (horizontal phase space) indicates the space occupied by the momentum spread in the beam and the three ellipses represent the betatron ellipses of the low momentum particles, the central momentum particles and the high momentum particles, from left to right respectively. In the right hand graph (vertical phase space), there is no dispersion so there is only one beam ellipse and no visible effect due to the momentum spread. The miniature plot in the bottom right hand corner of the screen shows the lattice and the red spot shows the current position of the observer.

In this plot window, the following control buttons are changed compared to the previous window.

- **Run** starts the routine stepping through the lattice effectively showing a film of the beam shape as the observer moves with the beam through the lattice. The miniature lattice in the bottom right

hand corner of the screen keeps track of where the observer is positioned. While the routine is stepping through the lattice, the “Run” button changes its label to “Stop”. In rings, the routine continues circulating until the user stops the calculation. In transfer lines, the observer is returned to the start of the line and the calculation is recycled.

- **Go To** allows the user to move directly to a specific point in the lattice.
- **Step+ and Step –** allow the user to move slowly in the lattice and to search for a special position.
- **Envel/Xsect** toggles the display between the static envelope view and the dynamic cross-section view.
- **Real Sp./Phase Sp.** toggles the display from the phase space shown in Figure 8.3 to an equivalent real space view.
- All other buttons have already been described.

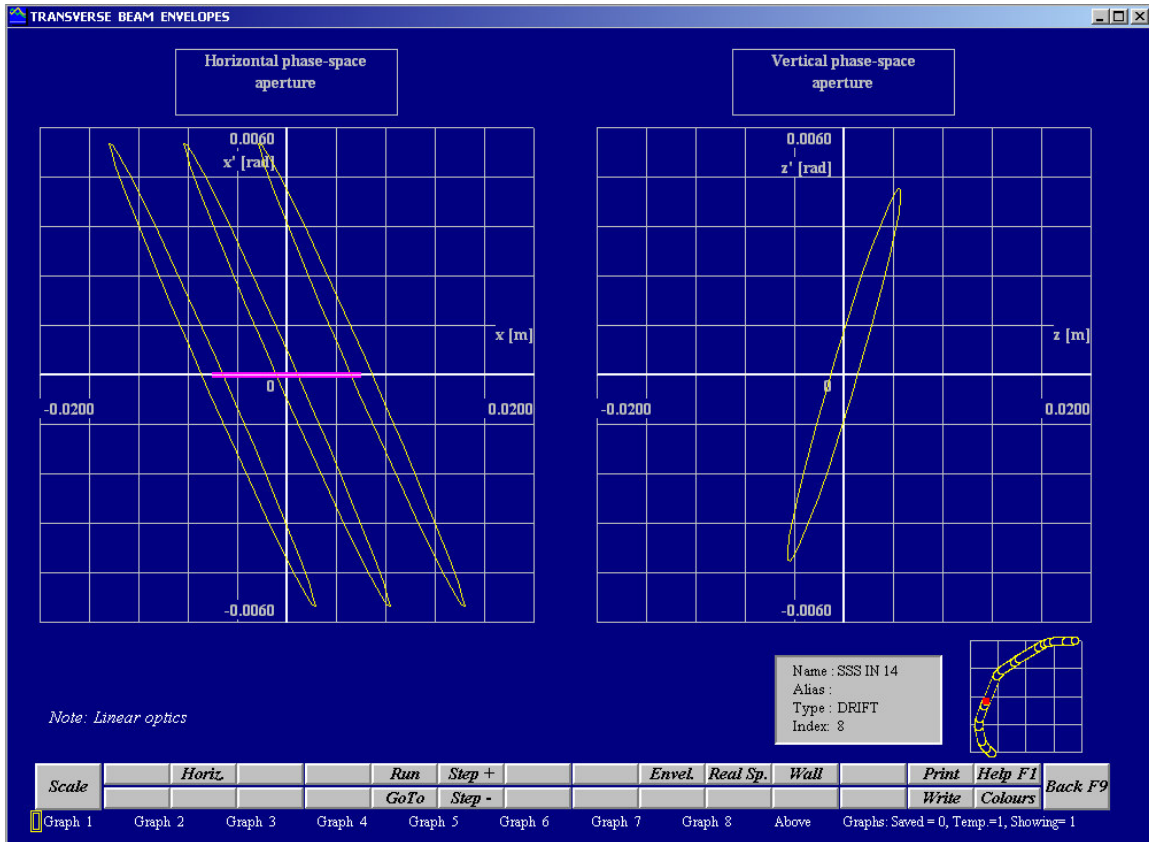


Figure 8.18 Beam cross-sections in phase space

8.9 Longitudinal Envelope Plot Window

This window shows the longitudinal beam envelope along the lattice, see Figure 8.19. Because the synchrotron frequency is much slower than the betatron oscillation frequency, it is necessary to track many turns to see oscillations in the longitudinal envelope. The example shows only the detail for one cavity crossing. The envelope can be viewed in five ways against the distance along the lattice. It can be viewed as the RF bunch length in metres, in nanoseconds and in degrees of RF phase and the orthogonal component can be viewed as the energy spread in keV or the relative momentum

deviation. Unlike the transverse case, the longitudinal envelope cannot be superimposed on a trajectory. The buttons below the graphs provide most of the usual features and controls. There is one new button that controls the form the envelope is viewed in:

- *s-dE / s-dp/p / s-ds / s-dt / s-deg.* toggles the display of the longitudinal envelope between different forms with the bunch length in metres, in nanoseconds and in degrees of rf phase and the orthogonal component as the energy spread in keV or the relative momentum deviation.

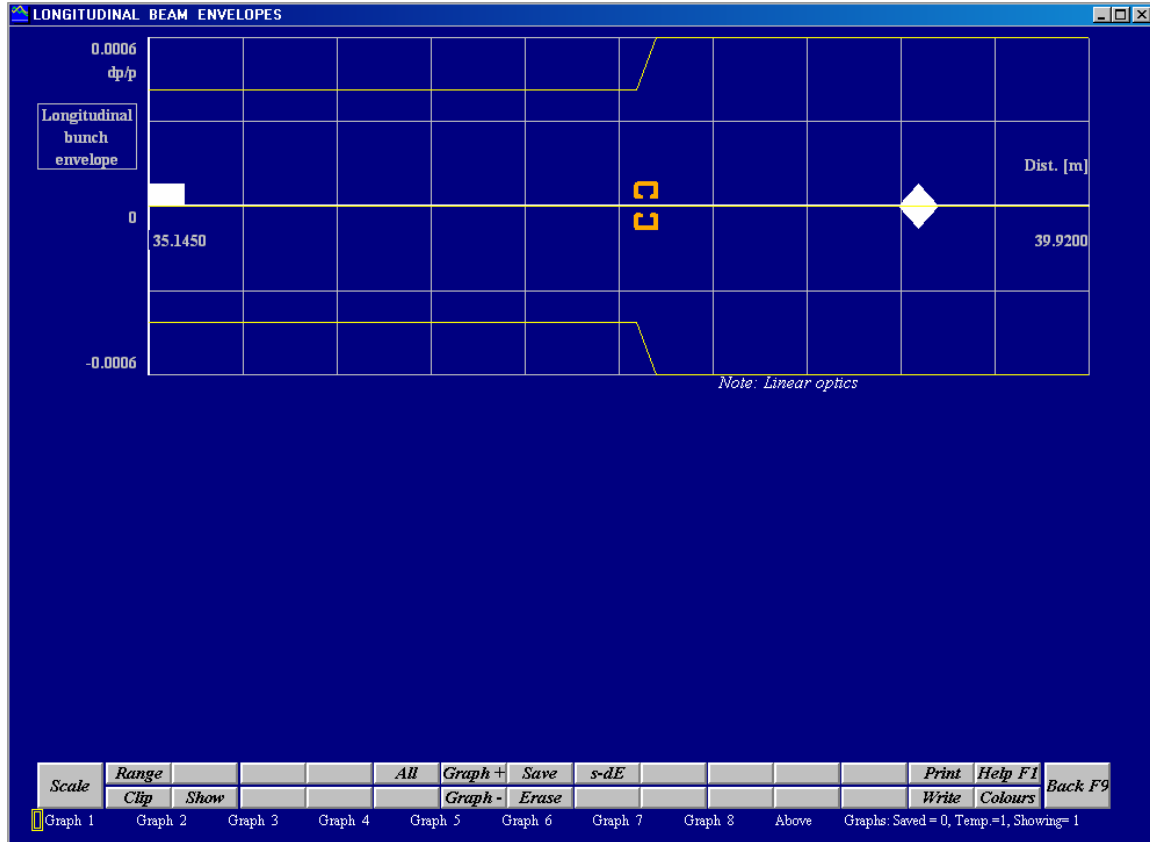


Figure 8.19 Longitudinal beam envelopes

8.10 Transverse Distribution Plot Window

Before the Distribution Plot Window opens, a small dialogue box is presented to set the environment for the tracking. The user is required to click radio buttons according to whether he wishes to include, or not include in the tracking, the current dipole kicks, momentum kicks and misalignments stored in memory. If the user wishes to include such kicks and/or misalignments, then he must create the files in readiness before entering the routine.

Figure 8.20 shows the transverse distribution plot window. This example is for a beam with a momentum spread shown in real space with the cross-section of the vacuum chamber. The yellow dots represent the low-momentum particles, the brown dots the on-momentum particles and the red dots the high-momentum particles in the beam. The beam is traversing a short transfer line seen in the miniature geometry plot in the bottom right hand corner of the screen. The red dot in the miniature geometry plot

shows the current position of the observer. Beams can be constructed from a maximum of 100 beamlets of 100 particles each. Each beamlet can be ascribed a momentum, horizontal and vertical emittances, Twiss parameters, a starting position and one of 9 colours. The beam can be viewed as it moves continuously through the lattice, or it can be moved step by step. The display can be changed freely between real space and phase space and while in phase space the co-ordinates can be toggled between real and normalised. There is sub-panel that shows the initial number of beam particles and the surviving number. Losses on the vacuum chamber walls are recorded and can be viewed in a separate plot window. If scattering is present, a transmission efficiency (or survival probability) is calculated and the particle number that has survived collision with the walls is modified by the fraction that represents this survival probability. In this case, 'particles' should be viewed as 'macro-particles' comprising many ordinary particles. Similarly, for space charge calculations, the 'particles' are interpreted as 'macro-particles'. For improved precision when tracking off-momentum particles, create the off-axis lattice and using this for the tracking calculation.

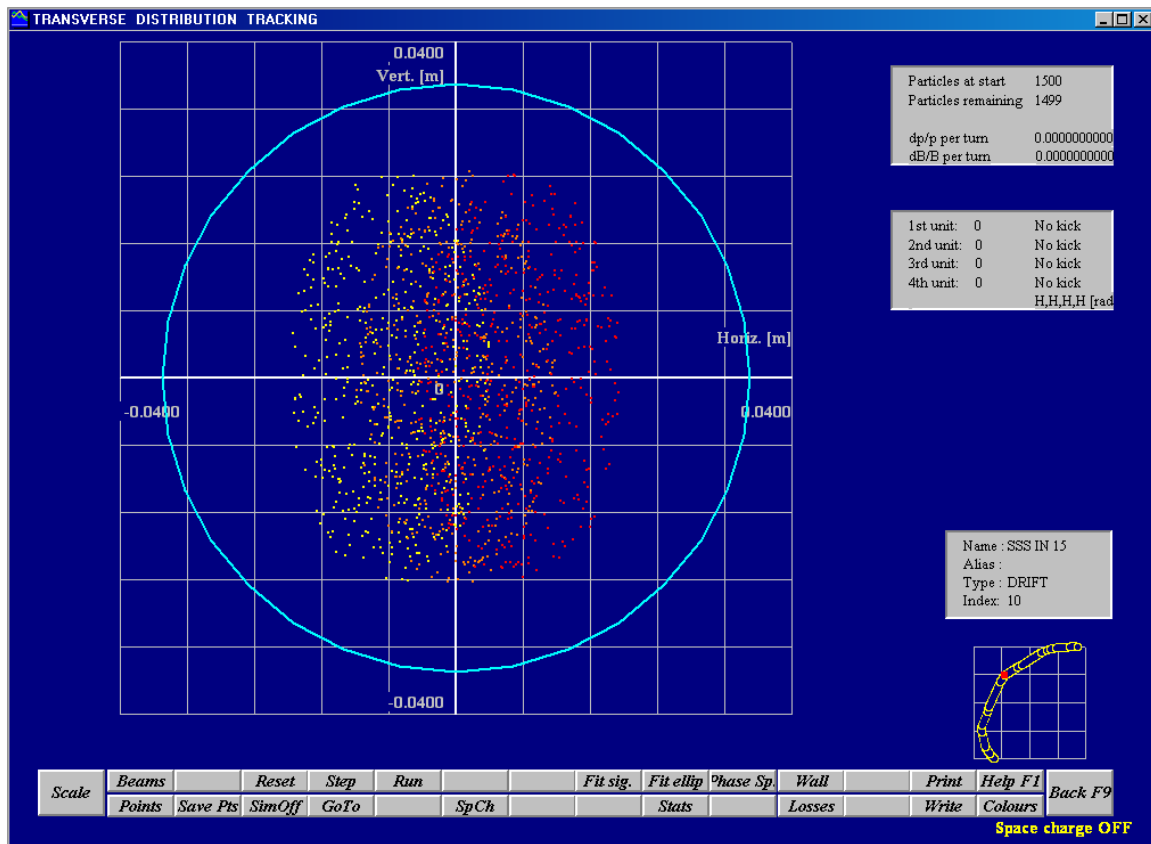


Figure 8.20 Tracking a transverse distribution of particles in a cylindrical vacuum chamber.

Many of the control buttons have been described in earlier sections. The new or modified buttons are listed below:

- **Scale** allows the user to change the horizontal and vertical scale of the graphs, as well as the thicknesses of lines, sizes of points and numbers of grid lines. Note that it is possible to distort the real-space graphs with unequal scales in the horizontal and vertical directions.

- **Beams** opens the dialogue box shown in Figure 8.21 for the introduction of particle distributions. On the left-hand side, there are the default, matched, Twiss parameters that the user can edit to introduce a mismatched beam. Below the Twiss parameters, there are the options for different beam distributions. The options comprise uniform and gaussian distributions in phase space generated as correlated or uncorrelated distributions and two special distributions. The first is known as the ‘bar of charge’ for simulating slow-extracted beams and the second is the phase space occupation of the separatrices for the final turn in a slow extraction scheme. The latter is explained more fully in Section 8.11 under *Sptrx II*. On the right-hand side at the top, the user should enter the 1-sigma geometric emittances, the number of sigma defining the envelope of the beamlet(s), the momentum deviation and the number of ‘beamlets’. A ‘beamlet’ contains 100 particles (macro-particles). The colour of the particles in the beamlet(s) can be chosen before moving to the next dialogue box, shown in Figure 8.22. This dialogue shows as default the matched transverse positions and angles of the beamlet(s). The positions and angles can be freely edited. Note that there is also a button for setting the last-used values. The greyed section of the dialogue box is not relevant to this window and will be used later in the Map Plot Window for an expert routine for evaluating a multi-turn injection. When this last dialogue box is closed the beam will appear on the screen. More beamlets with different parameters can be added up to a maximum of 100 (i.e. 10’000 macro-particles).

Note that truncated gaussian distributions are no longer strictly gaussian and when fitting an ellipse to the distribution one will not obtain the same 1-sigma emittance that was used to create the distribution. The finite number of particles in a beam also means that the halo region tends to be under-populated, so again fitting an ellipse to the distribution will usually produce an emittance that is slightly smaller than the one used to create the distribution.

INPUT OF MONO-ENERGETIC BEAMLETS AND SPECIAL BEAMS

TWISS PARAMETERS :
Defaults are the matched values.
Edit TWISS values if a mismatched beam is wanted.

Beta-x [m]
Alpha-x
Beta-z [m]
Alpha-z
Horizontal dispersion [m]
Derivative of the horizontal dispersion
Vertical dispersion [m]
Derivative of the vertical dispersion

BEAMLET PARAMETERS :

Horiz. 1 sigma geom. emitt. [Pi mm mrad] =
Vert. 1 sigma geom. emitt. [Pi mm mrad] =
Relative momentum deviation [dp/p] =
Number of beamlets to be added =

DISTRIBUTIONS :

Correlated distributions $E_x + E_z = \text{const.}$:

☒ Uniform 2D (see Notes).
☐ Gaussian 2D.

Uncorrelated or factorised distributions :

☐ Uniform 2D (see Notes).
☐ Gaussian 2D.
☐ Simulation of the beam from a slow extraction.
☐ Steinbach separatrix beam (see Manual)

SELECT PARTICLE COLOUR :

☒ (graph 1 colour) ☐ (graph 4 colour) ☐ (graph 7 colour)
☐ (graph 2 colour) ☐ (graph 5 colour) ☐ (graph 8 colour)
☐ (graph 3 colour) ☐ (graph 6 colour) ☐ (graphs 9+)

Colour scheme can be changed by clicking "Colours" in the parent Graph Window. Colours correspond to the Graph colours.

NOTES :

A beamlet contains 100 monoenergetic particles.
-The beamlet will be constructed from its 1 sigma emittances.
A total of 10000 particles can be stored.
-A beam with a uniformly filled 2-D cross-section has an elliptic 1-D cross-section with edges at 2 sigma.

Figure 8.21 Dialogue box for generating particle distributions using ‘beamlets’

POSITION OF BEAMLET(S)

EDIT/ENTER BEAM POSITION :

PLEASE ENTER REAL-SPACE VALUES

Beamlet centre on x-axis at start =

Beamlet centre on dx/ds-axis at start =

Beamlet centre on z-axis at start =

Beamlet centre on dz/ds-axis at start =

PAINTING. (FOR MULTI-TURN INJECTION ONLY)

Beamlet centre on x-axis at end of horizontal painting =

Beamlet centre on dx/ds-axis at end of horizontal painting =

Beamlet centre on z-axis at end of vertical painting =

Beamlet centre on dz/ds-axis at end of vertical painting =

RESET LAST USED VALUES

OK **Cancel**

NOTES :

- The default values are the expected beam positions on the basis of dispersion and dipole kicks (if any).
- Closed-orbit distortion is not considered for maps or lines or matched sections.
- **To stop horizontal and/or vertical painting set start and end values equal.**

Figure 8.22 Dialogue box editing the positions and angles for ‘beamlets’

- **Points** allows the user to enter particles in one of three ways:
 - **Manual introduction of the coordinates of a single point,**
 - **Reading a point file.** A point file (*.pnt) can have up to 10'000 particles and can be produced by saving an earlier calculation in WinAGILE, or writing a file according to the standard spreadsheet format (ascii) using commas as separators and carriage returns to separate one particle from the next. The file has the form;
 - x, dx/ds, z, dz/ds, dp/p, RGBcolour, Carriage return
 - x, dx/ds, z, dz/ds, dp/p, RGBcolour, Carriage return
 - x, dx/ds, z, dz/ds, dp/p, RGBcolour, Carriage return etc.
 - The RGB colour is a long integer.
 - **Reading a map file.** A map file (*.map) can have up to 10'000 entries and can be produced by WinAGILE in the Map Plot Window. . A map is generated by a single particle sampled on consecutive turns. Effectively, it draws a shell distribution in 4-D space, which can be used as particle distributions for tracking. Normally, this file would only be produced by WinAGILE. The file has the standard spreadsheet format (ascii) using commas as separators and carriage returns to separate one particle from the next. The file has the form;
 - x, dx/ds, z, dz/ds, Xn, dXn/dmux, Zn, dZn/dmuz, H-invariant, V-Invariant, Carriage return

- $x, dx/ds, z, dz/ds, X_n, dX_n/d\mu_x, Z_n, dZ_n/d\mu_z, H\text{-invariant}, V\text{-Invariant},$ Carriage return
 - $x, dx/ds, z, dz/ds, X_n, dX_n/d\mu_x, Z_n, dZ_n/d\mu_z, H\text{-invariant}, V\text{-Invariant},$ Carriage return etc.
 - Since this file does not contain the momentum deviation, the dialogue box asks the user to specify this parameter. The file also does not specify the colour, so for simplicity, all map file points are automatically made yellow.
- **Save Points** opens a standard dialogue box that allows the user to write a disk file containing the current beam distribution. This file can be loaded at any later time using the **Button Points**, see above.
 - **Reset** deletes settings, purges storage arrays and re-sets the observer to the start of the lattice.
 - **SimOff / SimOn** The screen display takes a significant fraction of the processor time and for production work it is not strictly necessary. This button allows the user to turn the screen simulation off and to benefit from the full processor power.
 - **Step** transfers the beam distribution forward one element in the lattice. In rings, the beam can be made to turn indefinitely, but in transfer lines and matched sections the beam stops at the end of the lattice.
 - **Goto** transfers the beam to the specified position if that is downstream of the current position. If the specified position is upstream in an open lattice an error message is shown saying that back-tracking is not possible. When there is no beam there is no restriction on the movement.
 - **Run / Stop** starts a continuous calculation that steps the beam forward in the lattice until stopped either by the user re-clicking the Run/Stop button, the beam being lost, or the beam arriving at the end of an open lattice.
 - **Run # / Stop** this button only appears in closed rings and can be used to move the beam forward a finite number of turns from the current position.
 - **Proj / Proj Off** controls the drawing of the projected beam distributions on the horizontal and vertical axes.
 - **MT-Inj** is an expert routine for simulating a conventional multi-turn injection. The routine is only available for closed rings in the Ring Window. It works in real or phase space, but NOT with normalised co-ordinates. If the display is in normalised co-ordinates it is automatically switched to the unnormalised mode.

The routine opens with the screen shown in Figure 8.23 that has a dialogue box for the main parameters, some additional explanation and a schematic view of the injection scheme. The schematic view shows the injected beam as a red ellipse entering the main ring from behind the septum wall. The red ellipse is drawn with two vertical positions to symbolise vertical ‘painting’ controlled by kickers in the injection line. The injection is made horizontally onto the central orbit that has been moved towards the septum by a closed-orbit bump inside the machine, symbolised by the white lines. As the injection progresses, the closed-orbit bump is collapsed and the beam enters further and further from the central orbit, thus providing the horizontal ‘painting’. It is usual to arrange the horizontal and vertical painting such that the large amplitude particles in one plane are correlated with the small amplitude particles in the other. In more advanced schemes, the horizontal bump may be combined with a vertical bump to ‘paint’ both planes simultaneously. In some cases, the main machine may also be ramped during the injection.

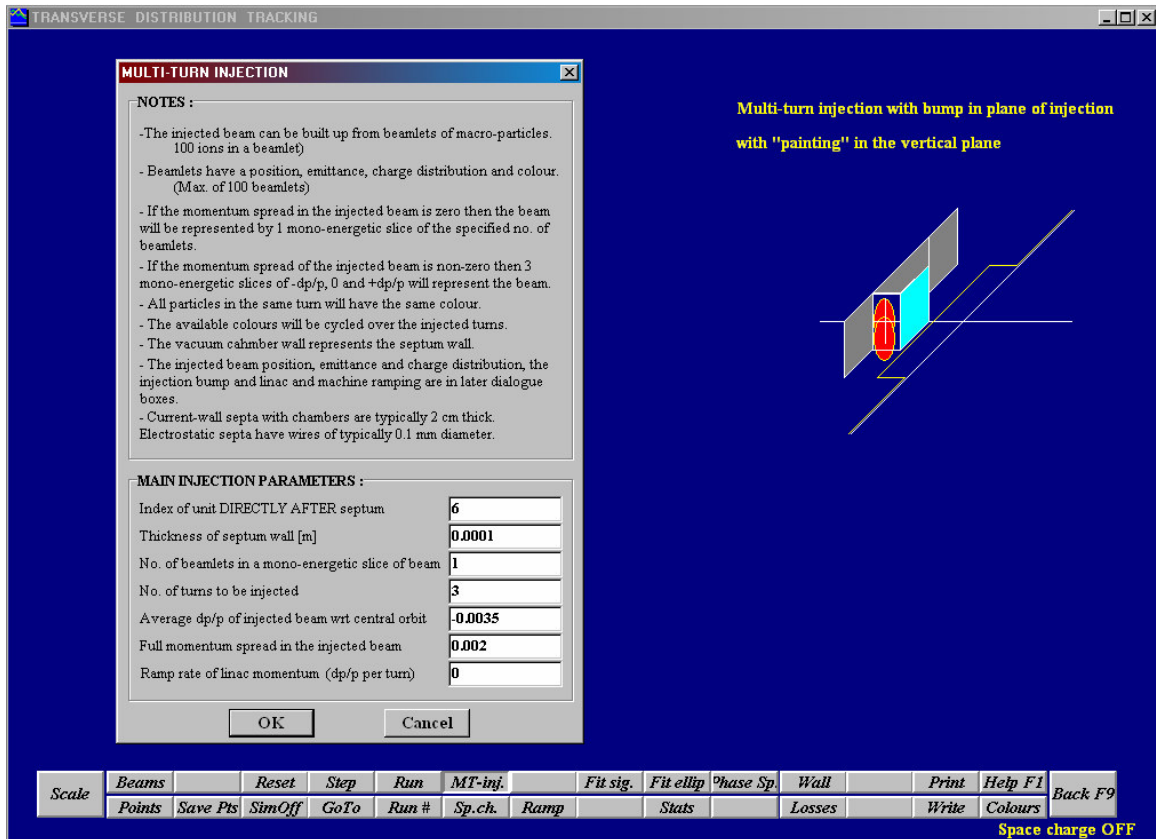


Figure 8.23 Opening screen and dialogue box for the multi-turn injection routine

When the opening screen and dialogue box is closed, a second dialogue box, as shown in Figure 8.24, is opened. This dialogue box allows the user to define a ramp rate in the main ring (note that the first dialogue box in Figure 9.23 allows the user to set the ramp rate of the injected beam) and the injection bump, which is created by point dipole kicks. The dialogue box contains a number of notes to remind the user of limitations and possibilities. One important limitation is that the bump kicks must be entered in the dialogue box in beam order, since the routine uses the order to position the starting point, the bump and the injection point for the synchronisation of the kicks.

The routine can be confusing to use. If problems are encountered check carefully that the vacuum chamber wall is correctly entered and that in the first dialogue box the index is the first unit AFTER the septum. Check the bump is moving the beam to the correct side and that the number of turns to be injected is consisted with the number of turns during which the bump collapses.

RAMPING

RAMPING GLOBAL PARAMETERS :

Relative field ramp of the machine per turn =

0.0000000000

[(dB/B)/dt, where dt = 1 turn]

RAMPING UP TO 4 POINT DIPOLE KICKS :

Index #	H	V	Initial kick [rad]	Final kick [rad]	# of turns
0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	0.0000000000	0.0000000000	0
0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	0.0000000000	0.0000000000	0
0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	0.0000000000	0.0000000000	0
0	<input checked="" type="checkbox"/>	<input type="checkbox"/>	0.0000000000	0.0000000000	0

Copy data to clipboard

Paste time stamp

OK

Copy data to notebook

Load from notebook

Cancel

NOTES :

- The momentum ramp can be used to simulated a betatron core or cavity.
- The field ramp can be used for coasting beams in ramping machine.
- A unit index of "0" gives no kick.
- Zero turns applies only the initial kick.
- Any existing kicks in the specified units are overwritten.
- Kicks are applied at the entry to the unit specified by the index #.
- When ramping only dipole kicks can be non-zero. When not ramping quadrupole or skew quadrupole kicks can be added. Misalignments can be added in both cases.
- Enter kicks in beam order starting in top edit box. The routine relies on this ordering to detect if the starting point is inside or outside bump in order to synchronise the changes in the kicks.

Figure 8.24 Opening screen and dialogue box for the multi-turn injection routine

When the dialogue box in Figure 8.24 is closed there are two more dialogue boxes to define the Twiss parameters of the injected beam that need not be matched to the circulating beam and the positions of the ‘beamlets’ as they are injected. These latter two dialogue boxes have already been described and are shown in Figures 8.21 and 8.22. However, certain features are greyed, or no longer greyed, in order to conform to the special needs of the injection scheme. Note that the injected ‘beamlets’ have envelopes preset to 2 sigma and the colours are also preset.

Figure 8.25 shows an example calculation in phase space with the beam injected over 6 turns and a bump collapsing over 8 turns. The six ‘beamlets’ are clearly visible and are placed at increasing distance from the central equilibrium orbit in the horizontal plane. Since the plot is made in a region of zero dispersion it is not possible to see that each ‘beamlet’ is in fact three ‘beamlets’ representing the momentum spread in the injected beam. In the vertical plane, no ‘painting’ was applied, so all the ‘beamlets’ are superimposed. When designing an injection scheme, the aim would be to pack the ‘beamlets’ closer together to create a denser beam.

Note that display boxes show the main parameters as the injection progresses. The simulation is limited to 100 ‘beamlets’ of 100 macro-particles each, as in the main routine. Space charge can be included, but this is expensive in computer time.

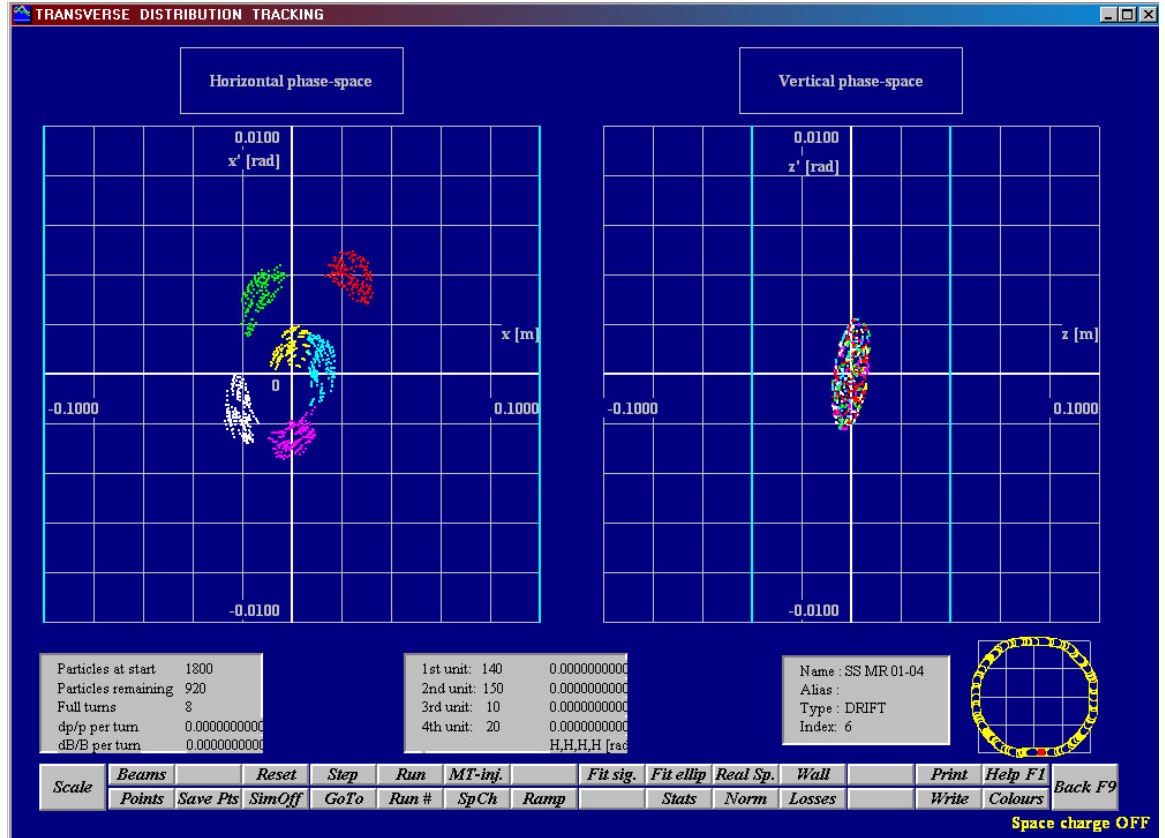


Figure 8.25 Example multi-turn injection

- **Sp. Ch. / No Sp. Ch.** is an experimental routine that attempts to calculate the self space charge forces (no images) using the direct interaction between the macro-particles. This routine can be used to include space charge in any of the calculations performed in this window. However, in view of the experimental nature of the routine, the results should be treated with some care.

The opening dialogue box is shown in Figure 8.26. The first group box shows the existing particle and energy per nucleon. To change these parameters, it is necessary to go all the way back to the Main Window. The second group box deals with the calibration of the macro-particles. By default, all 'beamlets' have 100 macro-particles and there can be a maximum of 100 'beamlets'. The user is required to specify how much current is to be attributed per 'beamlet'. Once this calibration is determined, the routine will attribute the current according to the number of 'beamlets' created. In this way, for example, the current will increase as a multi-turn injection progresses. In the third group box, there is a calculator for converting particle numbers into current. Finally, in the fourth group box, the user is advised on whether the lattice needs sub-dividing and is offered this possibility. It is recommended that this is only done once the user is sure the calculation is correctly set up.

Whether this routine can simulate a beam of say 10^{13} particles can be questioned. Trying different numbers of macro-particles is not so much help, because the gap between 10^4 particles and 10^{13} is still very large. The sub-division of the lattice is also a problem. The built-in routine tries to optimise the subdivision by grading the step size to the rate of the change in the distribution. This is done using the non-space charge lattice functions and hence, in extreme cases, it may not be optimum and the user may wish to perform the sub-division. Note that once the lattice has been sub-divided it remains sub-divided while in the Distribution Plot Window, but on returning to the Prior Window the original lattice is reloaded. Note that when the lattice is sub-divided, the indices of the kickers and bump kicks etc. are changed.

TRACKING WITH DIRECT SPACE CHARGE FORCES

1. BEAM PARTICLE :

Current ion is a12C 6+

K.E./nucleon [GeV/u] =1.00000000

Momentum/nucleon [GeV/c/u] =1.69187618

To change these parameters please return to the Main Window

2. CALIBRATE MACRO PARTICLES :

Beam current [A] =1.0000000

[If beam is bunched use peak]

EQUIVALENT TO:

Number of beamlets of 100 macro particles that are equivalent to the current declared above.1

This calibration will be used to evaluate the beam current for whatever number of beamlets are used in the lattice.

3. CALCULATOR :

Optional aid to convert particle number to beam current

Enter total number of particles in the closed ring e.g. 2.5E102.980E+0011

Compute beam current

Beam current [A] =1.0000000

4. SUB-DIVISION OF LATTICE:

SHOULD THE LATTICE BE SUB-DIVIDED?

Accuracy depends on step size, but smaller step sizes require more computation time.

The program has a criterion based on the rate of change of lattice functions, which it uses to sub-divide only those elements that are sensitive.

Experience shows that the figure of merit given below should be less than 0.25. The figure of merit can be edited.

Current figure of merit.1.2993

Should be < 0.25

Edit the figure of merit and subdivide lattice.1.2993

(Re)-subdivide lattice ?

Number of elements on entry =158

Present number of elements =158

OKCancel

NOTES :

Increasing the figure of merit does NOT recombine the elements.

Cancel reverses subdivision to the condition on entry.

Figure 8.26 Dialogue box for adding space charge

- **Ramp** this routine, which is only available for closed rings, allows the user to add a bump to the lattice that will rise or fall in a specified number of turns and to ramp either the beam momentum, the ring momentum, or both.. The dialogue box is the same as shown in Figure 8.24, except that the ramping of the beam energy is included. This routine is intended for testing beam dumping systems.
- **Fit sig** this routine analyses the 2nd moments of the particle distribution and displays the sigma matrix, the centroids and the total, 1-sigma, statistical Ex-z emittance.
- **Fit ellip** this routine fits and draws ellipses for the phase-space particle distributions in unnormalised co-ordinates. If the current mode is real space or phase space with normalised co-ordinates, then the routine automatically switches the display to unnormalised phase space. Likewise, switching from phase space in unnormalised co-ordinates to one of the other choices will remove the ellipses. There are three types of ellipse that can be drawn in any combinations:
 - (a) The routine fits ellipses using the statistical emittances.
 - (b) The routine uses the matched Twiss functions and then centres theoretical ellipses on the average position of the distribution. In this case, the user can specify the emittances. The aim is to compare the distribution obtained with a theoretical distribution.
 - (c) The user can specify all parameters for his own ellipses and their positions.
- **Stats** this routine analyses all the ellipses that have been created and lists their parameters.

SIGMA MATRIX CALCULATED FROM DISTRIBUTION

SIGMA MATRIX AT ENTRY TO:

Index number = **1** Name = **SS MR 01-03**
Alias =

$\langle x x \rangle$ 34.134901	$\langle x x' \rangle$ 0.321380	$\langle x z \rangle$ 0.228599	$\langle x z' \rangle$ 0.102106
$\langle x' x \rangle$ 0.321380	$\langle x' x' \rangle$ 0.845716	$\langle x' z \rangle$ -0.002584	$\langle x' z' \rangle$ -0.034876
$\langle z x \rangle$ 0.228599	$\langle z x' \rangle$ -0.002584	$\langle z z \rangle$ 17.560601	$\langle z z' \rangle$ 0.099132
$\langle z' x \rangle$ 0.102106	$\langle z' x' \rangle$ -0.034876	$\langle z' z \rangle$ 0.099132	$\langle z' z' \rangle$ 1.630032

Note that matrix elements have dimensions: [mm]², [mm mrad]² and [mrad]²

CENTROIDS AND TOTAL EMITTANCE:

Average x-position [m] = **-0.0003**
Average dx/ds-slope [rad] = **-0.0001**
Average z-position [m] = **-0.0000**
Average dz/ds-slope [rad] = **-0.0001**
Total 1-sigma emitt. Ex-z [Pi mm mrad]² = **28.6724**

OK
Copy data to notebook
Copy data to clipboard

Figure 8.27 Dialogue box showing the statistical fit leading to the sigma matrix

- **Losses** This routine opens a new plot window that shows the magnitude and positions of the beam losses in the most recent calculation. Figure 8.27 shows an example of the Loss Plot Window in which all the beam losses occur in the vertical plane and quasi-equally on the upper and lower chamber walls. The various control buttons in this window are standard and have been described previously.

When the lattice contains scatterers, accounting for the losses is a little more complicated. In this case the particles are considered as macro-particles. When a scatterer is crossed, a probability is calculated for the particle survival and this number is used to modify the losses. Firstly, the scatterer itself is accredited with 25% of the probable loss on each edge (up, down, right, left). Secondly, the particles that are lost by collision with the vacuum chamber wall downstream of the scatterers are only counted according to the percentage survival rate calculated from the scatterers in the line. The net result is a total loss equal to the number of macro particles, but at any one position there may be fractional particle (macro-particle) losses.

Note that when a particle is lost from a distribution, it is in fact kept in the memory at its last position, but its colour is changed to the background colour. Similarly, it appears in the data file written by the program and the lost particles can be identified by their colour. If the default colour is kept this is dark blue = \$00800000. If all the particles are lost then the file contains only lost particles.

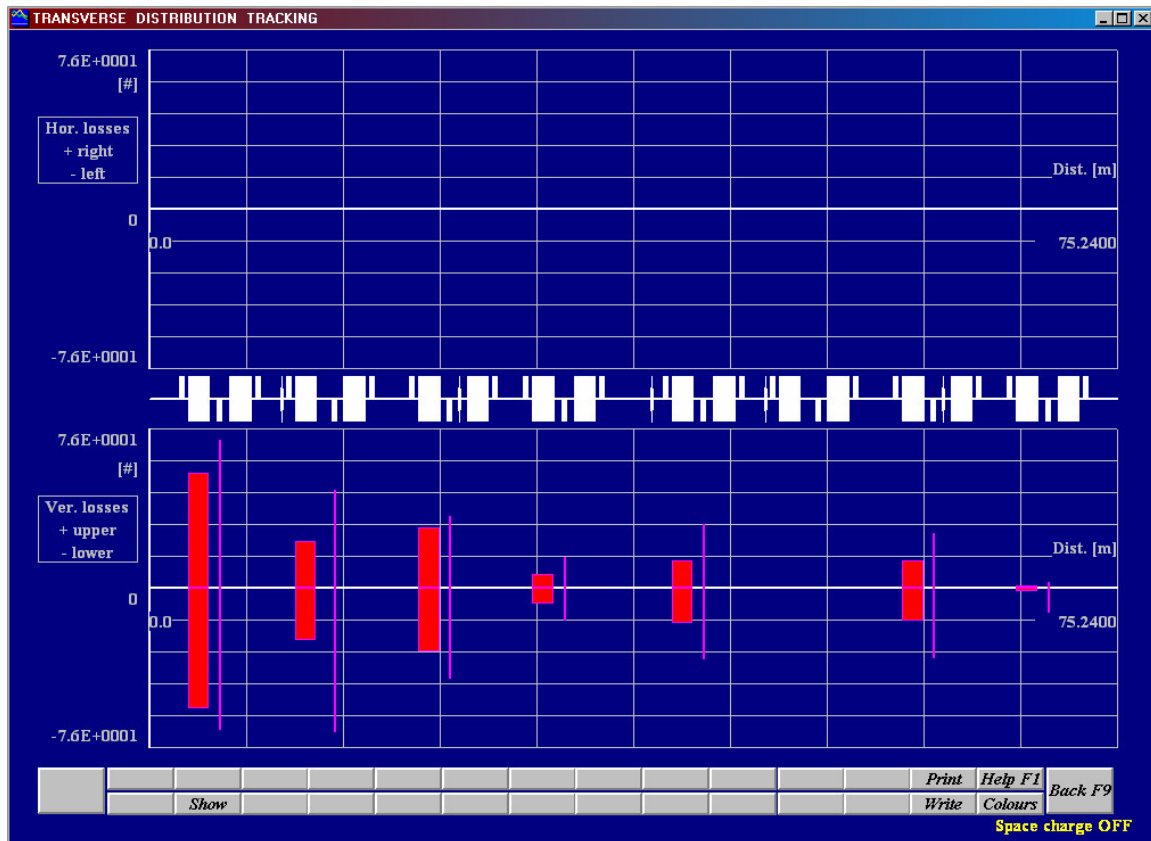


Figure 8.28 The loss plot window

8.11 Transverse Map Plot Window

The previous routine for tracking particle distributions through a lattice is relatively slow because it takes care of all the structure, apertures, kicks and so on. This approach is ideal for beam loss and injection studies, but it would be useful to have a quicker way of tracking much larger numbers of turns. One approach is to compress the linear sections of the lattice into single matrices and then to record the particle positions only once per turn at a specified position along the lattice. The resulting plot is called a "map". This throws away some of the detail, such as where particles are lost, but it increases the speed and allows the user to 'see' many more turns for the same level of numerical error.

The map routine only applies to closed rings. It tracks up to 10'000 particles around the machine and plots them after each turn at a single specified position along the lattice. The opening dialogue box requests the user for the position at which the map is to be made and whether he wishes to apply high or normal precision. Normal precision is faster and therefore good for tests and it is the only option if the lattice is coupled.

The Map Plot Window opens in phase space, as shown in Figure 8.29, but can be toggled to and from real space at will. While in phase space the display can be toggled between normalised and unnormalised co-ordinates. Beams and/or particles can be added in exactly the same way as in the Transverse Distribution Plot Window,

described in the previous section. Similarly, the display can be animated by tracking turn after turn. In addition, this window has an expert routine for studying slow extraction on a third-integer resonance. The image in Figure 8.29 is explained under Button **Trace** below,

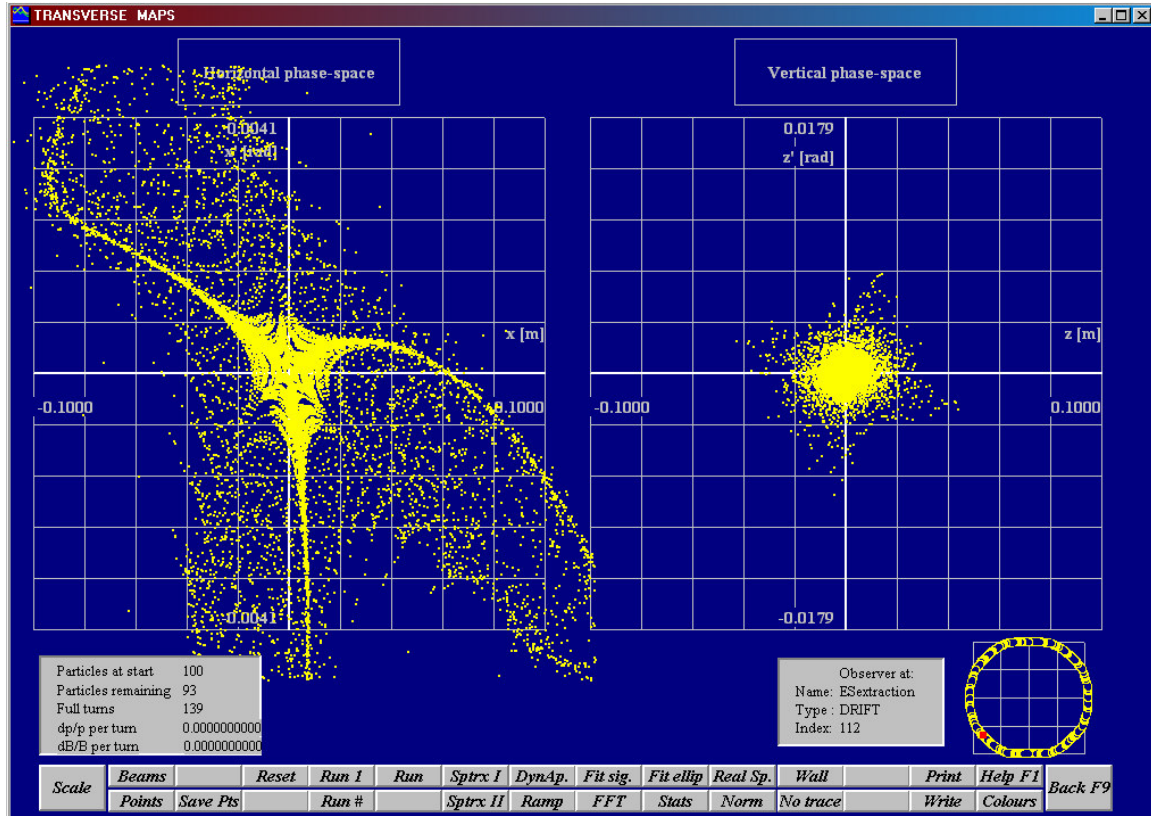


Figure 8.29 The map plot window

Most of the control buttons have been described in earlier sections. The new buttons are listed below:

- **Run 1 / Run # / Run** trigger the tracking routine to make 1 turn or # turns or continuous turns until stopped, respectively.
- **Sptrx I** is an expert routine for studying slow extractions. It locates the separatrices for on- and off-resonance particles and deduces from the data the precision positions, angles, spiral step etc. needed for an extraction system. Make sure that the observer's position corresponds to the entry of the electrostatic extraction septum and that the position of the wires of the septum are correctly entered as the position of the aperture/chamber wall in the lattice database.

The routine is specially adapted for the PIMMS-styled extraction that uses an electrostatic septum at the position of the observer and a magnetic septum downstream. The PIMMS-styled extraction accelerates the beam into a stationary resonance using a betatron core. The two key parameters are the emittance of the incoming beam and the resulting momentum spread in the extracted beam. For computational reasons, it is easier to calculate the emittance for a specified momentum spread and to iterate until the desired combination is found.

The routine opens with the dialogue box shown in Figure 8.30. The user inputs the expected momentum spread for the extracted beam, in addition he can introduce a non-zero motion in the vertical plane, which will perturb the extraction and he can place a limit on the aperture to prevent particles re-entering the aperture. The routine proceeds automatically switching to unnormalised

phase space with the chamber wall switched on. First, it locates the resonance, then plots the on-resonance separatrices and then plots the separatrices for off-momentum particles up to the requested value. The final display is shown in Figure 8.31. The yellow traces correspond to the on-resonance separatrices and the white traces correspond to the last stable triangle and the separatrices for the requested momentum spread. The numerical results are displayed in the dialogue box on the right in Figure 8.31. The graphical part of the display clearly shows if the Hardt Condition is respected and enables the user to optimise by changing the chromaticity and re-running the routine. Note that the numerical results can be copied directly to the Windows clipboard or the internal notebook for the lattice and the graphical result can be saved in a disk file.

CALCULATION OF SEPARATRICES

MOMENTUM SPREAD/DEVIATION :
 Enter momentum spread to be extracted or momentum deviation from resonance [dp/p] =
 - Positive dp/p means beam is above resonance in mom. space
 negative dp/p means beam is below resonance in mom. space.

COUPLING :
 To test the effect of coupling, a vertical emittance can be simulated by entering non-zero vertical coordinates.
At first, leave both the vertical position and angle ZERO.
 Vertical position of the test particle [m] =
 Vertical angle of the test particle [rad] =

APERTURE :
The particles must be intercepted or they return like comets after passing round a distant stable island.
 Horizontal half-aperture at map point [m] =

NOTES :
 - This routine works only in the horizontal plane.
 - The beam has to be close to the resonance with a non-zero chromaticity so that the routine can find the resonance and move the beam into it.
 - The interpretation of the momentum spread/deviation depends on the type of extraction to be used.
 - Field and momentum ramping are switched off.

Figure 8.30 Dialogue box for launching the slow extraction evaluation

- Sptrx II** prepares the data from *Sptrx I* for a further evaluation of the slow extraction scheme and in particular the positioning of the magnetic septum. This graphical analysis was first demonstrated by Charles Steinbach and was used in his program FERMEX. Basically, the display shows the three wing-shaped areas in phase space occupied by all the separatrices (i.e. from on-resonance up to the maximum off-resonance case). There is also a fourth block that represents the beam that is deviated by the electrostatic septum. Enter the requested septum parameters, Click OK to save the data and then go to the Distribution Plot Window and load the data using the **Beams** control button, see Section 8.10 and Figure 8.21. It is convenient to transfer this calculation to the Distribution Plot Window because one needs to see what happens in detail at each point in the lattice and the Map Plot Window is not equipped to do this. A dialogue box will request the kick of the electrostatic septum and the index of the first element directly after the septum. Note that it is possible to quit the dialogue box and drag the top window aside and look in the previous window to check the lattice numbering.

Figure 8.32 shows some plots made in this way in the Distribution Plot Window. They show how well the separatrices turn in phase space and how well the gap in the separatrices is adjusted at the magnetic septum.

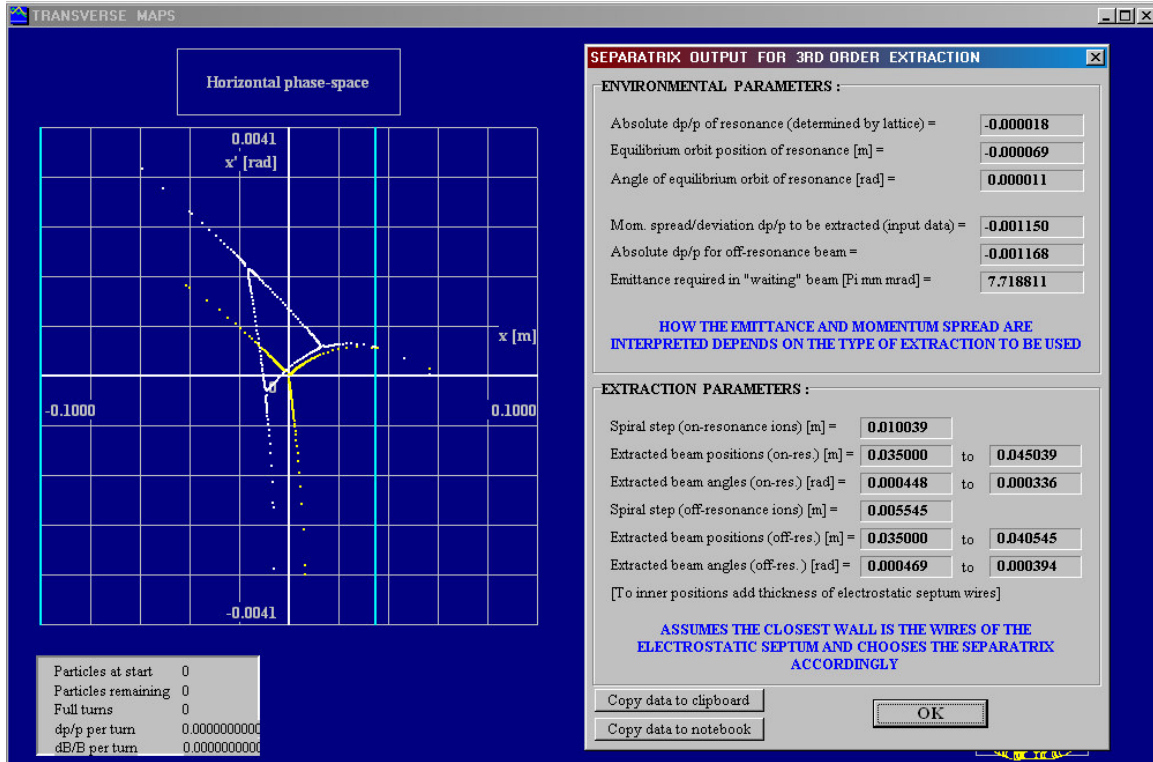
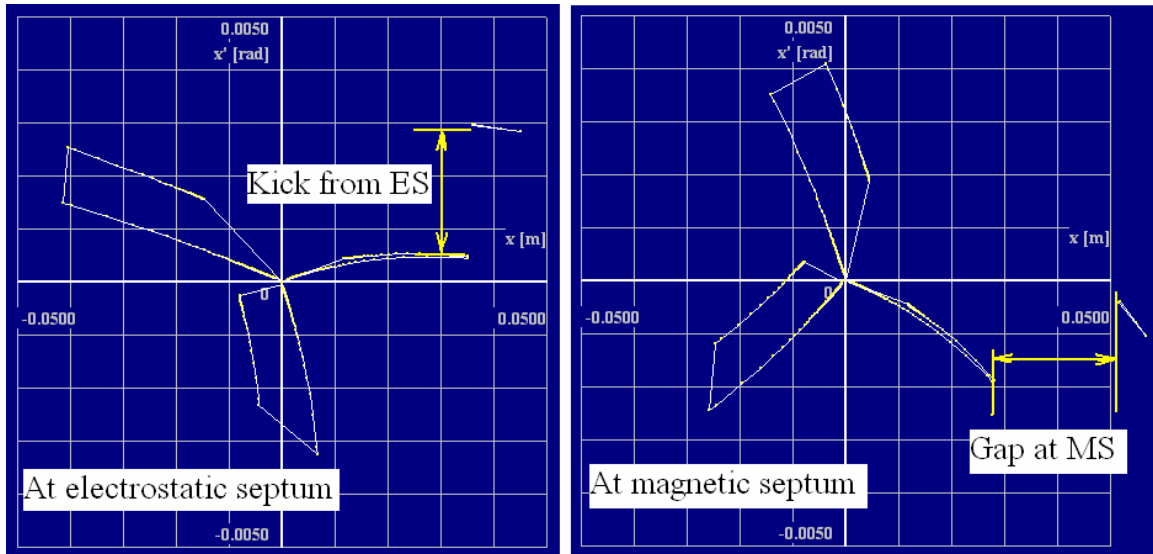


Figure 8.31 Graphical and numerical results from the slow extraction calculation (Map Plot Window)



8.32 "Steinbach" graphical representation of a slow extraction (Distribution Plot Window)

- **Dyn. Ap.** Not coded in this version.
- **FFT** Not coded in this version.

- **Trace / No Trace** is a feature that only appears in this window. The normal situation is with Trace off, in which case the old positions of particles from earlier turns are removed and only the current positions are shown on the screen. With Trace active, the old positions are retained so that one sees the “trace” of the motion of the particle from turn to turn. This can sometimes show some interesting phase space resonance patterns. Figure 8.29 shows a beam of 100 particles tracked over 139 turns while sitting on a third-order resonance. The three stable islands are clearly visible, as well as the traces going out, round the islands and back into the central aperture. Some coupling is also causing growth in the vertical plane with four separatrices.

8.12 *Coupling Plot Window*

This plot window opens with two polar graphs showing the difference (left) and sum (right) coupling-vectors. The same window is used for coupling compensation calculations and the driving terms of other non-linear resonances. The routine automatically adapts the titles to suit single mode and summed mode coupling vectors according to the calculation choices made by the user. For an explanation and definitions of the various coupling vectors see the WinAGILE physics manual. The only new control button is described below.

- **Tag / No tag** toggles the addition/removal of labels to the vectors in a compensation scheme.

8.13 *W-vector Plot Window*

The W-vector is used for studying chromaticity and building chromaticity compensation schemes. This plot window has basically the same design as the coupling plot window. For an explanation and definition of the W-vector see the WinAGILE physics manual.

8.14 *Tune Diagram*

The basic tune diagram for closed rings shows the vertical tune values against the horizontal tune values. The central orbit tunes define the *working point* and a momentum spread creates the *working line*. An optional addition is to add the resonance pattern for any order resonance up to 9th order. An alternative form is two diagrams for the horizontal and vertical tune values shown separately against the relative momentum deviation.

The equivalent diagrams for transfer lines replace the tunes by the betatron phase advances, but without the resonance pattern.

Tune diagrams are extremely useful aids for recording measurements, shaping working lines, avoiding resonances, etc. when running machine development sessions. For this reason, ‘empty’ tune diagrams can be printed out when in the Main Window.

When in the Ring and Line Windows, the tune diagram routine automatically adapts to the program status. First the positions of the central orbit tunes (phase advances) of the current lattice are displayed (i.e. the *working point*). If the user has calculated off-axis orbits then these are also displayed giving the *working line*. If the central orbit is active and incoherent or coherent space charge has been applied then the space charge tune shift is shown. If a tune shift is made on central orbit, then the off-axis

orbits are invalidated and no longer appear in the display. In this case, they must be re-calculated.

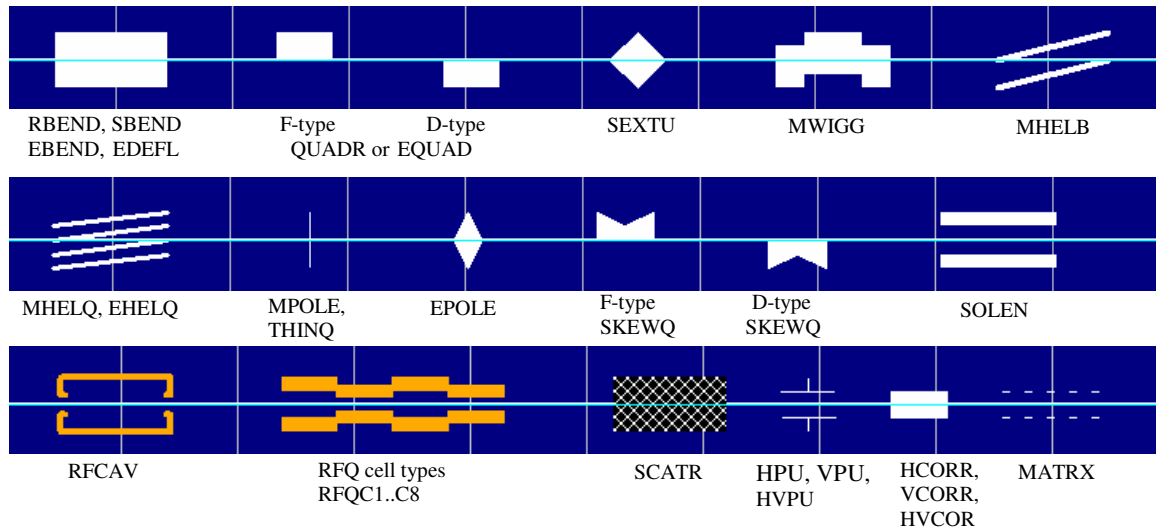
The control buttons available in the Tune Diagram Window are:

- **Scale:** Adjusts the graph scales.
- **$Q\text{-}dp/p$ / $Qh\text{-}Qv$:** Toggles between displaying the tunes (or the phase advances) plotted against each other, or plotted separately against the momentum deviation. If incoherent space charge has been applied, then the space-charge "necktie" will be plotted in the $Qh\text{-}Qv$ mode. If off-axis orbits have been calculated, then the working line will be plotted.
- **Horiz / Vert / Hor+Ver.** When the tunes (or phase advances) are plotted separately against the momentum deviation, this button is active and toggles between showing only the horizontal tune graph, only the vertical tune graph, or both the horizontal and vertical tunes graphs. This command also affects the printout and postscript files.
- **Res +:** When the display shows the tunes plotted against each other this button will be active. This button increases the order of the resonance pattern in steps up to a maximum of the 9th order. The order of the resonance pattern is shown under the buttons.
- **Res-:** When the display shows the tunes plotted against each other this button will be active. This button decreases the order of the resonance pattern in steps down to zero. The order of the resonance pattern is shown under the buttons.

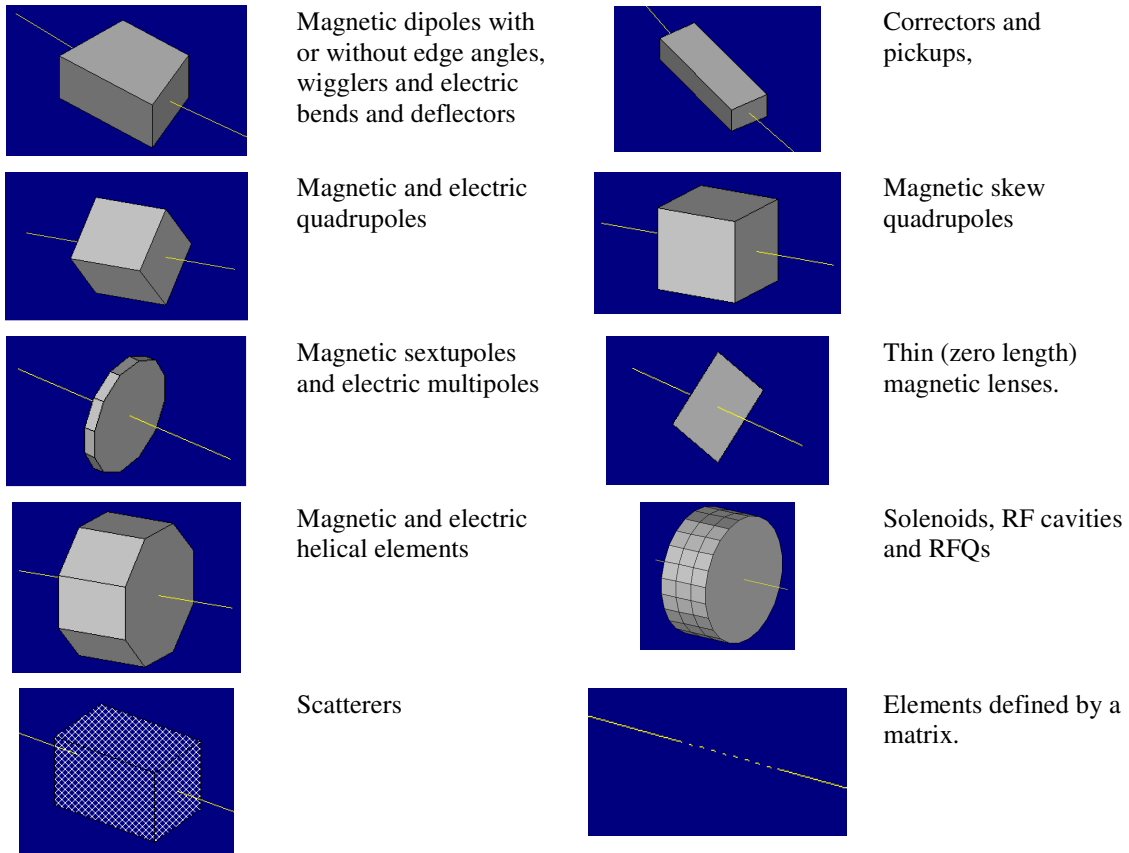
The remaining buttons are standard and have already been described.

8.15 Element icons

Graphs having the distance along the lattice as abscissa are shown with the following icons for the elements. Not all elements have icons and some elements share.



2D and 3D survey drawings represent the elements with a smaller range of icons. The following icons are used:



* * *

Chapter 9 Kick and Misalignment Windows

The Kick Window is used for creating and editing files of dipole, quadrupole, skew-quadrupole and momentum kicks. These data sets can be used to perturb the lattice, for example for the creation of closed-orbit distortion, betatron modulation and coupling. The Misalignment Window performs a similar job for misalignments. Both windows are accessible from the Main, Line and Ring Windows as well as many dialogue boxes, but only when there is a valid lattice.

9.1 Kick Window

Figure 9.1 shows the Kick Window with inserts for the column headings for the different kicks: dipole [rad], quadrupole [m^{-1}], skew quadrupole [m^{-1}] and momentum [dp/p]. As in the Edit Window, double clicking a cell in a kick column opens an edit box on the left at the top of the spreadsheet for data entry.

KICK FILE EDITING
Options KickFile Distributions Output Help

Selection: **No Data** Status: **Lattice OK**
Quad fringe-field Off
HJM chromaticity eqn **Non-space charge optics**

File name: OPT_EXTR **DIPOLE KICKS** Date of run: 8/27/2010
Treated as: User title: PIMMS-Generic lattice for extraction 18/100 Time of run: 8:36:22 AM

Unit no.	Name	Type	Kick-x [rad]	Kick-z [rad]	End
1	SS MR 01-03	DRIFT	0.000000	0.000000	"
2	ES injection	DRIFT	0.000000	0.000000	"
3	ES injection	DRIFT	0.000000	0.000000	"

File name: OPT_EXTR **QUADRUPOLE KICKS** Date of run: 8/27/2010
Treated as: User title: PIMMS-Generic lattice for extraction 18/100 Time of run: 9:08:49 AM

Unit no.	Name	Type	Quad-kick-x [m^{-1}]	Quad-kick-z [m^{-1}]	End
1	SS MR 01-03	DRIFT	0.000000	0.000000	"
2	ES injection	DRIFT	0.000000	0.000000	"
3	ES injection	DRIFT	0.000000	0.000000	"

File name: OPT_EXTR **SKEW QUADRUPOLE KICKS** Date of run: 8/27/2010
Treated as: User title: PIMMS-Generic lattice for extraction 18/100 Time of run: 9:11:29 AM

Unit no.	Name	Type	Quad-kick-x [m^{-1}]	Quad-kick-z [m^{-1}]	End
1	SS MR 01-03	DRIFT	0.000000	0.000000	"
2	ES injection	DRIFT	0.000000	0.000000	"
3	ES injection	DRIFT	0.000000	0.000000	"

File name: OPT_EXTR **MOMENTUM KICKS** Date of run: 8/27/2010
Treated as: User title: PIMMS-Generic lattice for extraction 18/100 Time of run: 9:12:54 AM

Unit no.	Name	Type	Kick [dp/p]	End
1	SS MR 01-03	DRIFT	0.000000	"
2	ES injection	DRIFT	0.000000	"
3	ES injection	DRIFT	0.000000	"

Figure 9.1 Kick Window

When entering a quadrupole kick, the program automatically enters the equal and opposite value in the other plane. Skew quadrupole kicks behave similarly, except they have the same sign in the two planes. Momentum kicks have only a single column for the longitudinal plane. Kicks are applied at the entries to elements and an extra line is

added at the end of the lattice. In order to distribute a kick between the entry and exit of a given element, enter the second half of the kick at the entry to the next element.

Kick files can be saved and loaded in the usual way. When loading a file, the program checks that the number of elements corresponds to the current lattice and that the kick type is what is requested. However, these simple checks cannot guarantee that the kick file truly belongs with the lattice and the user must make the final check.

A historical limitation in the program is that the kick memory can only hold either dipole or quadrupole or skew quadrupole kicks at any one time. Momentum kicks and misalignments, being later additions to the program, have their own separate memories and are therefore free of all restrictions.

In principle, all kick files can be created by hand, but in practice the advantage of the Kick Window lies in the routines described below for simulating systematic and random errors of various types.

9.1.1 Dipole kick distributions for random errors

Figure 9.2 shows the dialogue box available in the Kick Window for generating dipole kick distributions corresponding to various random alignment and excitation errors.

- **Section 1** Errors can be generated with a truncated gaussian distribution or a uniform distribution in a band around the nominal value.
- **Section 2** generates dipole kicks to simulate three typical types of random alignment errors in synchrotrons. These are: Axial shift of all dipoles, Tilt of all dipoles, and Shifts of all quadrupoles.
- **Section 3** generates dipole kicks to simulate random powering (excitation) errors in all dipoles.
- **Section 4** generates random dipole kicks for up to 3 named series of units. These units need not be dipoles.

The entries in the dialogue box can be saved to the internal notebook and reloaded at a later date.

RANDOM DIPOLE KICKS

1. Choose type of distribution for all cases:

☒ Truncated gaussian: Number of sigma = 2.236

☐ Symmetric, uniform band :

2. Set 1 sigma or half-width of band for ALIGNMENT errors :

Axial shifts of ALL dipoles [m] = 0.0

Tilts of ALL dipoles [rad] = 0.0

Trans. shifts of ALL quads. [m] = Horizontal 0.0 Vertical 0.0

3. Set 1 sigma or half-width of band for EXCITATION errors :

For ALL dipoles. Express error as fraction of excitation = Horizontal 0.0 Vertical 0.0

4. Set 1 sigma or half-width of band for KICKS applied to any series of units :

Name of 1st series of units =

Name of 2nd series of units =

Name of 3rd series of units =

Apply kicks for 1st series [rad] = Horizontal 0.0 Vertical 0.0

Apply kicks for 2nd series [rad] = Horizontal 0.0 Vertical 0.0

Apply kicks for 3rd series [rad] = Horizontal 0.0 Vertical 0.0

NOTES:

- Dipole kicks are expressed in radian in the data file.
- Half the kick will be attributed to each end of the element (except axial shifts).
- Overlapping kick distributions will be added linearly.
- 'Dipole' and 'Quadrupole' categories both include combined-function units.
- Note that if random errors are applied to a unit that has been subdivided, then each slice gets a different random error.

Load from notebook Paste time stamp Add errors Cancel Copy data to clipboard Copy data to notebook

Figure 9.2 Dipole kick distributions simulating random alignment and excitation errors

9.1.2 Dipole kick distributions for systematic errors

Figure 9.3 shows the dialogue box available in the Kick Window for generating dipole kick distributions corresponding to various systematic alignment and excitation errors.

- **Section 1** generates dipole kicks to simulate three typical types of systematic alignment errors in synchrotrons. These are: Axial shift of all dipoles, Tilt of all dipoles, and Shifts of all quadrupoles. Note that although the shifts and tilts are systematic, the kicks will vary from unit to unit.
- **Section 2** generates dipole kicks to simulate systematic powering (excitation) errors in all dipoles.
- **Section 4** generates systematic dipole kicks for up to 3 named series of units. These units need not be dipoles.

The entries in the dialogue box can be saved to the internal notebook and reloaded at a later date.

SYSTEMATIC DIPOLE KICKS

1. Set ALIGNMENT errors :

Axial shift of ALL dipoles [m] =

Tilt of ALL dipoles [rad] =

Trans. shift of ALL quads. [m] =

Horizontal Vertical

2. Set EXCITATION errors in dipoles :

For ALL dipoles. Express error as fraction of excitation =

Horizontal Vertical

3. Set KICKS applied to any series of units :

Name of 1st series of units =

Name of 2nd series of units =

Name of 3rd series of units =

Horizontal Vertical

Apply kicks for 1st series [rad] =

Apply kicks for 2nd series [rad] =

Apply kicks for 3rd series [rad] =

Add errors **Cancel**

NOTES:

- Dipole kicks are expressed in radian in the data file.
- Overlapping kick distributions will be added linearly.
- Half the kick will be attributed to each end of the element (except axial shifts).
- 'Dipole' and 'Quadrupole' categories both include combined-function units.

Copy data to clipboard **Copy data to notebook** **Load from notebook** **Paste time stamp**

Figure 9.3 Dipole kick distributions simulating systematic alignment and excitation errors

9.1.3 Quadrupole kick distributions for random errors

Figure 9.4 shows the dialogue box available in the Kick Window for generating quadrupole kick distributions corresponding to various random alignment and excitation errors.

- **Section 1** Errors can be generated with a truncated gaussian distribution or a uniform distribution in a band around the nominal value.
- **Section 2** generates quadrupole kicks to simulate random alignment errors in the horizontal position of all units with a sextupole component.
- **Section 3** generates quadrupole kicks to simulate random powering (excitation) errors in all units with a quadrupole component.
- **Section 4** generates random quadrupole kicks for up to 3 named series of units. These units need not be quadrupoles.

The entries in the dialogue box can be saved to the internal notebook and reloaded at a later date.

RANDOM QUADRUPOLE KICKS

1. Choose type of distribution for all cases:
☒ Truncated gaussian: Number of sigma = 2.236
☐ Symmetric, uniform band :

2. Set 1 sigma or half-width of band for ALIGNMENT errors :
 Horiz. shifts of ALL units with a sextupole comp. [m] = 0.0
 [Nb. Vertical shift gives a skew quadrupole component.]

3. Set 1 sigma or half-width of band for EXCITATION errors :
 For ALL units with a quadrupole component.
 Express error as fraction of excitation = 0.0

4. Set 1 sigma or half-width of band for KICKS applied to :
 Name of 1st series of units =
 Name of 2nd series of units =
 Name of 3rd series of units =
 Apply kicks for 1st series [rad] = 0.0
 Apply kicks for 2nd series [rad] = 0.0
 Apply kicks for 3rd series [rad] = 0.0

Buttons: Add errors, Cancel, Copy data to clipboard, Copy data to notebook, Load from notebook, Paste time stamp

NOTES:
 - Half the kick will be attributed to each end of the element.
 - Quadrupole kicks are stored in normalised form in [m-1] in the data file.
 - Overlapping kick distributions will be added linearly.
 - If random errors are applied to a unit that has been subdivided, then each slice gets a different random error.

Figure 9.4 Quadrupole kick distributions simulating random alignment and excitation errors

9.1.4 Quadrupole kick distributions for systematic errors

Figure 9.5 shows the dialogue box available in the Kick Window for generating quadrupole kick distributions corresponding to various systematic alignment and excitation errors.

- **Section 1** generates quadrupole kicks to simulate systematic alignment errors in the horizontal position of all units with a sextupole component.
- **Section 2** generates quadrupole kicks to simulate systematic powering (excitation) errors in all units with a quadrupole component.
- **Section 4** generates systematic quadrupole kicks for up to 3 named series of units. These units need not be quadrupoles.

The entries in the dialogue box can be saved to the internal notebook and reloaded at a later date.

SYSTEMATIC QUADRUPOLE KICKS

1. Set ALIGNMENT errors :
 Horiz. shift of ALL units with a sextupole comp. [m] = 0.0
 [Nb. Vertical shift gives a skew quadrupole component.]

2. Set EXCITATION errors :
 For ALL units with a quadrupole component.
 Express error as fraction of excitation = 0.0

3. Set KICKS applied to :
 Name of 1st series of units =
 Name of 2nd series of units =
 Name of 3rd series of units =
 Apply kicks for 1st series [rad] = 0.0
 Apply kicks for 2nd series [rad] = 0.0
 Apply kicks for 3rd series [rad] = 0.0

Buttons: Add errors, Cancel, Copy data to clipboard, Copy data to notebook, Load from notebook, Paste time stamp

NOTES:
 - Quadrupole kicks are stored in normalised form in [m-1] in the data file.
 - Half the kick will be attributed to each end of the element.
 - Overlapping kick distributions will be added linearly.
 - If random errors are applied to a unit that has been subdivided, then each slice gets a different random error.

Figure 9.5 Quadrupole kick distributions simulating systematic alignment and excitation errors

9.1.5 Skew quadrupole kick distributions for random errors

Figure 9.6 shows the dialogue box available in the Kick Window for generating skew quadrupole kick distributions corresponding to various random alignment and excitation errors.

- **Section 1** Errors can be generated with a truncated gaussian distribution or a uniform distribution in a band around the nominal value.
- **Section 2** generates skew quadrupole kicks to simulate random alignment errors in the vertical position of all units with a sextupole component and tilts in all units with a quadrupole component.
- **Section 3** generates skew quadrupole kicks to simulate random powering (excitation) errors in all units with a skew quadrupole component.
- **Section 4** generates random skew quadrupole kicks for up to 3 named series of units. These units need not be skew quadrupoles.

The entries in the dialogue box can be saved to the internal notebook and reloaded at a later date.

RANDOM SKEW QUADRUPOLE KICKS

1. Choose type of distribution for all cases:
☒ Truncated gaussian: Number of sigma = 2.236
☐ Symmetric, uniform band :

2. Set 1 sigma or half-width of band for ALIGNMENT errors :
 Vert. shifts of ALL units with a sextupole comp. [m] = 0.0
 [Nb. Horizontal shift gives a normal quadrupole component.]
 Tilts of ALL units with a quadrupole comp. [rad] = 0.0

3. Set 1 sigma or half-width of band for EXCITATION errors :
 For ALL units with a skew quadrupole component.
 Express error as fraction of excitation = 0.0

4. Set 1 sigma or half-width of band for KICKS applied to :
 Name of 1st series of units =
 Name of 2nd series of units =
 Name of 3rd series of units =
 Apply kicks for 1st series [rad] = 0.0
 Apply kicks for 2nd series [rad] = 0.0
 Apply kicks for 3rd series [rad] = 0.0

Buttons: Add errors, Cancel, Copy data to clipboard, Copy data to notebook, Load from notebook, Paste time stamp

NOTES:
 - Skew quad. kicks are stored in normalised form in [m-1] in the data file.
 - Half the kick will be attributed to each end of the element.
 - Overlapping kick distributions will be added linearly.
 - If random errors are applied to a unit that has been subdivided, then each slice gets a different random error.

Figure 9.6 Skew quadrupole kick distributions simulating random alignment and excitation errors

9.1.6 Skew quadrupole kick distributions for systematic errors

Figure 9.7 shows the dialogue box available in the Kick Window for generating skew quadrupole kick distributions corresponding to various systematic alignment and excitation errors.

- **Section 1** generates skew quadrupole kicks to simulate systematic alignment errors in the vertical position of all units with a sextupole component and the tilts of all units with a quadrupole component.
- **Section 2** generates skew quadrupole kicks to simulate systematic powering (excitation) errors in all units with a skew quadrupole component.
- **Section 4** generates systematic skew quadrupole kicks for up to 3 named series of units. These units need not be skew quadrupoles.

The entries in the dialogue box can be saved to the internal notebook and reloaded at a later date.

SYSTEMATIC SKEW QUADRUPOLE KICKS

1. Set ALIGNMENT errors :

Vert. shift of ALL units with a sextupole comp. [m] =

[Nb. Horizontal shift gives a normal quadrupole component.]

Tilt of ALL units with a quadrupole comp. [rad] =

2. Set EXCITATION errors :

For ALL units with a skew quadrupole component.
Express error as fraction of excitation =

3. Set KICKS applied to :

Name of 1st series of units =

Name of 2nd series of units =

Name of 3rd series of units =

Apply kicks for 1st series [rad] =

Apply kicks for 2nd series [rad] =

Apply kicks for 3rd series [rad] =

NOTES:

- Skew quad. kicks are stored in normalised form in [m-1] in the data file.
- Overlapping kick distributions will be added linearly.
- Half the kick will be attributed to each end of the element.

Add errors **Cancel** **Copy data to clipboard**
Copy data to notebook
Paste time stamp **Load from notebook**

Figure 9.7 Skew quadrupole kick distributions simulating systematic alignment and excitation errors

9.1.7 Momentum kicks

The momentum kicks can only be created and edited by hand. There are no routines for generating data for simulating common error distributions.

9.2 Misalignment Window

Figure 9.8 shows the Misalignment Window. The columns from left to right refer to:

- Misalign-x Horizontal shifts (Δx)
- Misalign-xx Longitudinal slope ($\Delta x/\Delta s$)
- Misalign-z Vertical shifts (Δz)
- Misalign-zz Longitudinal slope ($\Delta z/\Delta s$)
- Misalign-tilt Rotation about longitudinal axis ($\Delta \tau$).

EDITING MISALIGNMENT FILES

Options: MisalignmentFile Distributions Output Help

Selection: **Status:** Lattice OK with ion **Beam ion:** proton

Quad fringe-field Off **1.000000 [GeV/n](entry)**

HJM chromaticity eqn **Non-space charge optics**

File name: DEMO_A **MISALIGNMENTS** **Date of run:** 8/27/2010

Treated as: circular machine **Time of run:** 12:02:46 PM

User title: Demonstration FODO cell with 60deg phase advance, no bending.

Unit no.	Name	Type	Misalign-x [m]	Misalign-z [m]	Misalign-xx [rad]	Misalign-zz [rad]	Misalign-Tilt [rad]	End
1	FQH1	QUADR	0.0000	0.0000	0.000000	0.000000	0.000000	"
2	ss1	DRIFT	0.0000	0.0000	0.000000	0.000000	0.000000	"
3	DQH1	QUADR	0.0000	0.0000	0.000000	0.000000	0.000000	"
4	DQH1	QUADR	0.0000	0.0000	0.000000	0.000000	0.000000	"
5	ss1	DRIFT	0.0000	0.0000	0.000000	0.000000	0.000000	"
6	FQH1	QUADR	0.0000	0.0000	0.000000	0.000000	0.000000	"
7	End		0.0000	0.0000	0.000000	0.000000	0.000000	"

Figure 9.8 Misalignment Window

As in the Edit Window, double clicking a cell in a misalignment column opens an edit box on the left at the top of the spreadsheet. In principle, all misalignment files

can be created by hand, but in practice the advantage of the Misalignment Window lies in the routines described below for simulating systematic and random errors of various types.

9.2.1 Random misalignments

Figure 9.9 shows the dialogue box available in the Misalignment Window for generating random misalignment distributions.

- **Section 1** Errors can be generated with a truncated gaussian distribution or a uniform distribution in a band around the nominal value.
- **Section 2** generates random misalignments in x , x' , z , z' and tilt for a named series of units. Enter “ALL” or “all” to generate errors for all units except drift spaces.

The entries in the dialogue box can be saved to the internal notebook and reloaded at a later date.

Figure 9.9 Random misalignment distributions

9.2.2 Systematic misalignments

Figure 9.10 shows the dialogue box available in the Misalignment Window for generating systematic misalignment distributions.

- **Section 1** generates random misalignments in x , x' , z , z' and tilt for a named series of units. Enter “ALL” or “all” to generate errors for all units except drift spaces.

The entries in the dialogue box can be saved to the internal notebook and reloaded at a later date.

Figure 9.10 Systematic misalignment distributions

* * *

Chapter 10 How to do...

10.1 Load a lattice file

From the Main Window select “Open...” in the File Menu.

A dialogue box will appear requesting the file name and path. If the demonstration files are present, load the file DEMO-002.LAT. The name of the file DEMO-002 will appear in the titles window at the top left of the screen and will be printed on any data listing made by the program. At the same time, the title window will also show that this file has been treated as a “matched section” in the past and the user title will describe it as “a 60 degree phase advance FODO cell with sector bends”. The status panel on the top right of the screen will indicate that the lattice data is okay. If a different file is loaded, then similar information should also be displayed, unless it is an unchecked and previously unused lattice.

To load another file, repeat the above. If the current lattice has not been saved recently or not purged, then the program will first ask if the user wishes to save his lattice before opening (or creating) a new one.

If a lattice is in compressed form a warning message will be given. In this case, select "Edit" in the File Menu. This transfers the focus to the Edit Window. After being offered the opportunity to edit the title of the lattice, which is optional, select "Check data with decompression" in the Check Data Menu. Once the lattice is decompressed and checked return to the Main Window. If the lattice has errors, the program will signal each error one by one and prompt the user to make corrections. Once the errors are corrected, re-try the "Check data with decompression" in the Check Data Menu.

Checking the data in the Edit Window is one of the fundamental requirements in WinAGILE.

10.2 Create and run a lattice

From the Main Window select “New...” in the File Menu. First a dialogue box requests an optional user title (69 char. max.). This is included in the print out of data displays along with the date and the time of the calculation and the lattice file name. Note that the user title can be edited at any time when in a data display window by clicking on the title. The user title appears in dark green.

Directly the user title dialogue box is closed the Edit Window is opened. This is essentially a spreadsheet with 5'000 rows and 45 columns. When creating a new lattice there are only the titles of the columns. It is possible to enter anything anywhere apart from some reserved symbols and from limitations on numbers of characters. However, whatever is entered has to be checked by the program before it can be executed.

Try entering a FODO cell. Position the cursor on column 2, line 1 and click the left-hand mouse button to highlight that cell. Now, either press Enter or double click the left mouse button. Either action will open the Edit Dialogue Box in the top left corner of the screen. Type “ss1”. This will be name of the first drift space. The name can be up to

12 characters long. Click OK or press Enter again. Either action will put the data into the memory and into the highlighted cell.

Next position the highlight box on column 3 in line 1 and press Enter and type “drift”. In practice “dr” or “ss” is enough for the program to recognise a drift space or straight section type.

Next position the highlight box on column 4 in line 1 and enter “9” for the length in [m]. This is the minimum acceptable lattice - just one element.

To check the data click “Check Data with decompression” in the Check Data Menu. The program will check and if all is correct it will offer to return to the Main Window. Click “No” for the moment. Experiment by re-entering -9 for the length and then checking the data again. The program will give a warning message that negative lengths are not allowed.

Complete the FODO cell by entering the full structure:

Col.1	Col.2	Col.3	Col.4		Col.9
Index	Name	Type	Length [m]		k [m-2]
1	ss1	drift	9		
2	FQH1	quadr	0.5		-.103513
3	FQH1	quadr	0.5		-.103513
4	ss1	drift	9		
5	DQH1	quadr	0.5		0.103513
6	DQH1	quadr	0.5		0.103513

Note that if an element already exists, it is sufficient to retype its name on a new line and all the attributes will be copied automatically (except the vacuum chamber which will have default values). Notice also that the normalised gradient for F-quadrupoles is negative and for D-quadrupoles positive. This is an old sign convention that comes from the weak focusing era and differs from some other programs.

If the data is not checked a "datagood" flag is set to false. It is still possible to return to the Main Window and save the data in a file, but it will not be possible to execute the file before it has been checked. So to check the data, click “Check Data with decompression” in the Check Data Menu and this time return to the Main Window when asked. It will now be possible to perform several calculations. For example, click “Calculate ring or matched section...” in the Calculations menu. A dialogue box will appear giving information about the lattice and asking for some data. For the moment, the defaults will be sufficient, so click “OK” and the Twiss function display will appear. In fact, to get to this stage the transfer matrices and the geometry have also been calculated and can be accessed via the Tables Menu and the relevant graphs can be accessed via the Graphs Menu. If the current data display has a preferred graph, this can be accessed quickly via the “Graph” button on the right hand side of the screen.

Note that it is possible to select areas in the spreadsheet displays and to copy them to the Windows clip board of the internal notebook for the lattice. It is also possible

to drag the columns to new positions to group together the data that you want to see and perhaps copy or print.

10.3 Adjust the tunes values of a lattice

Open a lattice file from the File Menu in the Main Window. Note that it may be quicker to select the file from the recently-used files listed at the bottom of the File Menu or to simply double-click a lattice file providing the computer is set up to recognize the extension “lat”.

If the lattice is compressed or unchecked, a warning message will be given, in which case it will be necessary to go to the Edit Window to check the file. Once the file is checked, calculate the lattice as a ring or as a transfer line. For rings, go to the "Fit tunes.." in the Calculations-I Menu in the Ring Window and for transfer lines go to "Fit phase advances.." in the Calculation Menu in the Line Window.

In either case, basically the same dialogue box will appear. At the top, enter the desired values for the tunes or for the phase advances. In the list box, choose the units that will control the focusing. If there is an un-named unit that can have a quadrupole gradient its index number can be entered directly. Click “Compute” and in most cases the routine will be able to re-adjust the lattice to you requested values.

Fitting tunes or phases advances is a non-linear process and some interaction with the fitting routine may be needed. If a ring goes unstable, try reducing the step sizes and/or their signs. It may also work better to progress to the desired values in say two or three jumps. It may be necessary to use one pair of quadrupoles for half of the shift and a second pair for the rest of the shift. The routine works with space charge, but it is slower because it cycles the lattice several times to try and stabilise the space-charge forces. To test if the result has indeed stabilised, try introducing very small steps and asking the routine to re-compute. If the two units chosen for the fitting are degenerate (i.e. have the same effect) then the routine proposes a "best" fit with the first unit. A "best" fit can also be forced by introducing the same unit twice.

There are several fitting routines of this type. Some examples are for setting the chromaticity, adjusting the geometry, decoupling matrices and so on. Some routines fit up to six parameters and therefore need six variables. However, the principles remain the same.

10.4 Export data and graphics

The export of data and graphics is possible in many ways. All spreadsheet displays can be printed directly via “Print display” in the Output Menu. There is also, in the same menu, the routine “Write export file...” that allows the user to select the data he wants. In addition, blocks of lines can be selected by dragging the mouse in column 1, or blocks of columns by dragging the mouse in the headers, or any sub-block by dragging the mouse across the block. Ctrl+C then copies the selection to the Windows clip board and Ctrl+K copies to the internal notebook.

All plot windows have “Print” and “Write” buttons. The write button offers different options for “encapsulated post script” and “bitmap” files.

10.5 *Combining graphs*

Most plot windows allow the user to save the current graph in a temporary file. If for example, the Twiss functions are displayed for a lattice, then clicking the control button “Save” below the graph will make a temporary data file for that graph. If now the “Fit tunes” routine is used to change the tunes a new graph can be displayed and saved. By clicking “All” the two graphs can be displayed superimposed. This can be done for up to 24 graphs. For exporting, the “Print” and “Write” buttons will treat combined and single graphs equally.

Normally, the program will erase the temporary graph files when a new lattice is opened in order to avoid inconsistencies. However, the exception to this rule is that geometry plots are not erased. It is therefore possible to open a series of lattices, plot their geometries, save their graphs and then to display them all together as an accelerator complex. In this way, complex layouts can be built up to show the relative positions of elements and potential clashes for space between say an extraction line and a ring. When combining lattices in a geometric layout, be sure to have the same survey origin for all plots.

10.6 *On-line Help*

The on-line Help contains many more “How to do...” topics.

* * *