TraceWin

D. Uriot

CEA Saclay
DSM/Irfu/SACM/LEDA
CEN Saclay 91191 Gif sur Yvette

Updated, April 07th 2011

Abstract

The TraceWin code is used for linear (matrix) and non linear calculations (macroparticle transport with Partran library) for 2D or 3D electrons or ions beams. It permits fast beam envelop computations or/and can be used as a GUI for Toutatis (link). The code is capable to match the beam to a channel. The different elements of a linac can be tuned (EM fields amplitude, length, phase for cavity) to obtain phase advance law or to set the beam size or Twiss parameters at different locations. Multiparticle transport can be executed transparently compared to envelop computations; this allows to easily study the impact of non linear effects. The user can change any parameter and observe the effect very easily due to the very powerful graphics display which allows to visualize most of the useful parameters of the simulation (envelopes, beam ellipses, emittances, phase advances…). All these outputs can be easily stored on disk, save in several image formats and inserted into reports (using copy and paste tools). Sensitivity to element errors in a linac can be studied with or without diagnostic and corrections by using a Monte Carlo approach. A huge number of cases can be simulated remotely via a home made client/server architecture. A heterogeneous array of machines can be used (window, linux, MacOS). It has been mainly written in C++ and Qt4.6 for Windows, Linux and MacOS operating system. Its development started in 1998 with **URIOT Didier** and **PICHOFF Nicolas** from CEA, since 2009 it is distributed under CEA license.

Technical support & maintenance services

Updates of the software

The CEA agrees to notify the Licensee of any Updates and Corrected Versions of the Software. The content of the Updates shall be decided by the CEA in its entire discretion, and shall comprise fixes for anomalies, improvements and upgrades to make the Software correspond to the most recent version of the Software. However, Updates may not include: versions of the Software that are compatibles with new operating systems, improvements, updates or new versions that are sold separately.
Updates and Corrected Versions shall be made available to the Licensee, in executable form, in the manner indicated by the CEA by email.

Corrective maintenance

The CEA agrees to correct reproducible errors in software programming. The Licensee shall notify the CEA of any unidentified error and provide it with all information necessary for error reproduction. The resources implemented by the CEA to correct an error shall depend on its seriousness.
If the error stems from a modification that was not made by the CEA or expressly authorized by it, the Licensee agrees to compensate the CEA for the time spent on correcting the error and restoring the Software to good operating condition, at the CEA rates then applicable.

Technical support

The CEA shall provide technical support to the Licensee during its normal office hours, Mondays to Fridays (excluding bank holidays and annual weeks of closure). The CEA shall use its best efforts to provide a detailed response within a maximum period of thirty (30) business days depending on its human resources available at the time of the request.
Requests should be sent to the CEA by email to contact address defined below. The code can be downloaded here: http://irfu.cea.fr/Sacm/logiciels/ .

Evolutive maintenance

The CEA shall add new functions to the Software where required by a change in the legal or regulatory requirements or in the software or hardware environment. New Versions shall be made available to the Licensee, in executable form, in the manner indicated by the CEA by email

Contact

This version of **TraceWin** is supported by **Didier URIOT**. We would appreciate hearing from you if you found a bug. Your questions or remarks can be sent at the following Emails address:

didier.uriot@cea.fr

Very important: How report a bug: Please use the button "*Send project*" in main page and include the specified files. By this way we are able to know which system you used and which version produced the bug and finally with the include files we can easily reproduce it, thus fix it

Installation

Minimum configuration required

1 gigahertz (GHz) or faster 32-bit (x86) or 64-bit (x64) processor
1 gigabyte (GB) RAM (32-bit or 64-bits)
30 megabyte (MB) available hard disk space (32-bit or 64-bit)

Operating system**:**

Window (32-bit or-64-bit), version equal or bigger then WinXp
Linux (32-bit or-64-bit) with gcc library version equal or bigger 3.3.0
MacOS with system version equal or bigger 10.6

Installation:

No installation is necessary; all the extra files used by TraceWin are directly extracted from main code and installed if a process requires it.
The downlowdable version is not a full capability version. To activate its full capabilities you have to include in the TraceWin executable directory a key file "*TraceWin.key*".

- PlotWin is based in part on the work on ***Qt*** project (http://qt.nokia.com) with Qt commercial license agreement verion: 3.8
- PlotWin is based in part on the work of the ***Qwt*** project (http://qwt.sf.net)

Main features

- A wide range elements including RFQ with Toutatis module,
- 2D and 3D space-charge,
- 1D, 2D and 3D static and RF magnetic or/and electric field maps (with superposition capability),
- envelope simulation,
- macro-particle tracking simulations (number of particle depending of your free memory),
- each particle has an can be detailed analysis of its trajectory is available,
- start-to-end simulations from source to target,
- transport of two beams in the same structure,
- gas stripping and scattering analysis,
- automatic transverse and longitudinal beam tuning in envelope or/and tracking mode,
- beam tuning in period structure based on smoothing phase advances,
- correction procedure based on diagnostics,
- static and dynamic error simulations for all elements,
- simulations with large number of particles for large scale computations (Monte Carlo) based on a client/server architecture,
- statistic analysis including beam loss location,
- GUI and various help tools,
- Windows/Linux/MACOS versions,
- reference code for IFMIF, LINAC4, SPIRAL2, EUROTRANS, EURISOL and SPL projects.

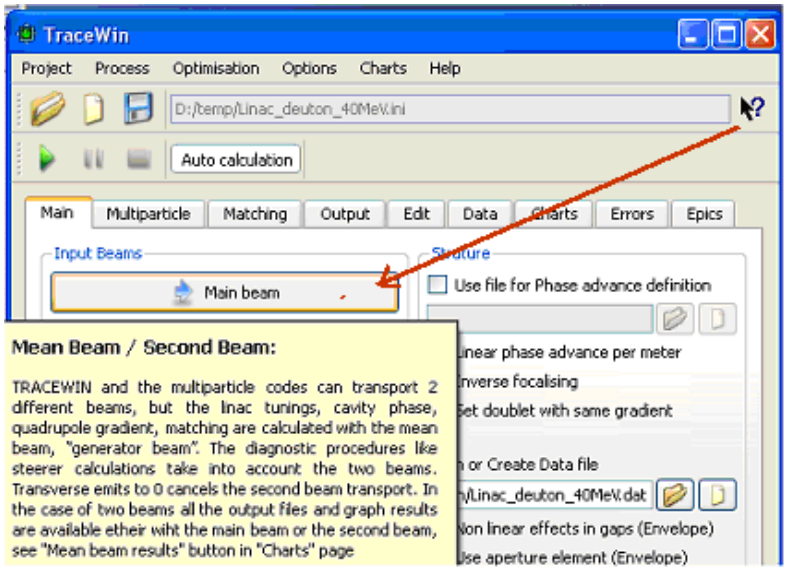
Way of using

General description

TraceWin's program is organized in 8 pages and 3 toolbars. You can find more details about these pages below.

Menu	: Shows the 20 last current opened projects.
First ToolBar	: Open save or create a project file (configuration file, *.ini), show the current open project.
Second ToolBar	: To launch the process, to have a break or stop it and set "***auto_calculation***".
Third ToolBar	: Visible only during matching, to stop or visualize criteria variation.
Main page	: To set input beam parameters, structure options and calculation options.
Multiparticle page	: To configure multiparticle code options.
Matching page	: To configure beam matching options
Output page	: To visualize the calculation stages
Edit page	: To modify or visualize the main input and output files.
Data page	: To visualize the elements and commands list from the data file
Charts page	: To visualize the results with plots.
Errors page	: To parameterize the error study and visualize results.
Epics page	: For EPICS virtual machine configuration

Each input or widget item of TraceWin GUI owns explanatory text. The default way for users to view the help is to move the focus to the relevant widget and press Shift+F1. The help text appears immediately. A second ways is to use the help button, see following picture.

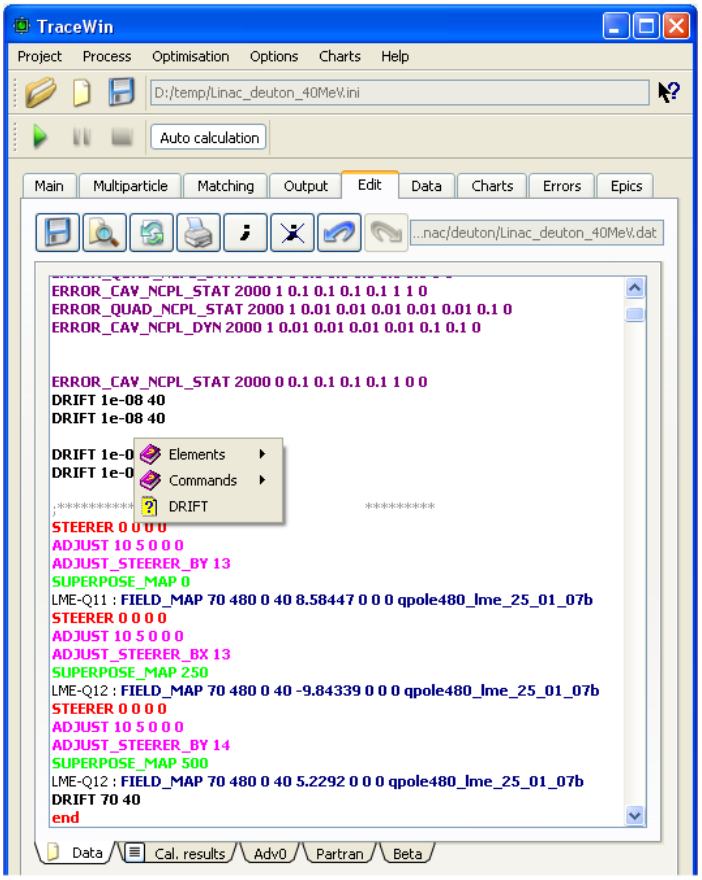


TraceWin process is organized in several stages. The stages can automatically run one behind the other or not ("***Auto calculation***" button of ToolBar). Some of them can be disabled according to some options and commands.

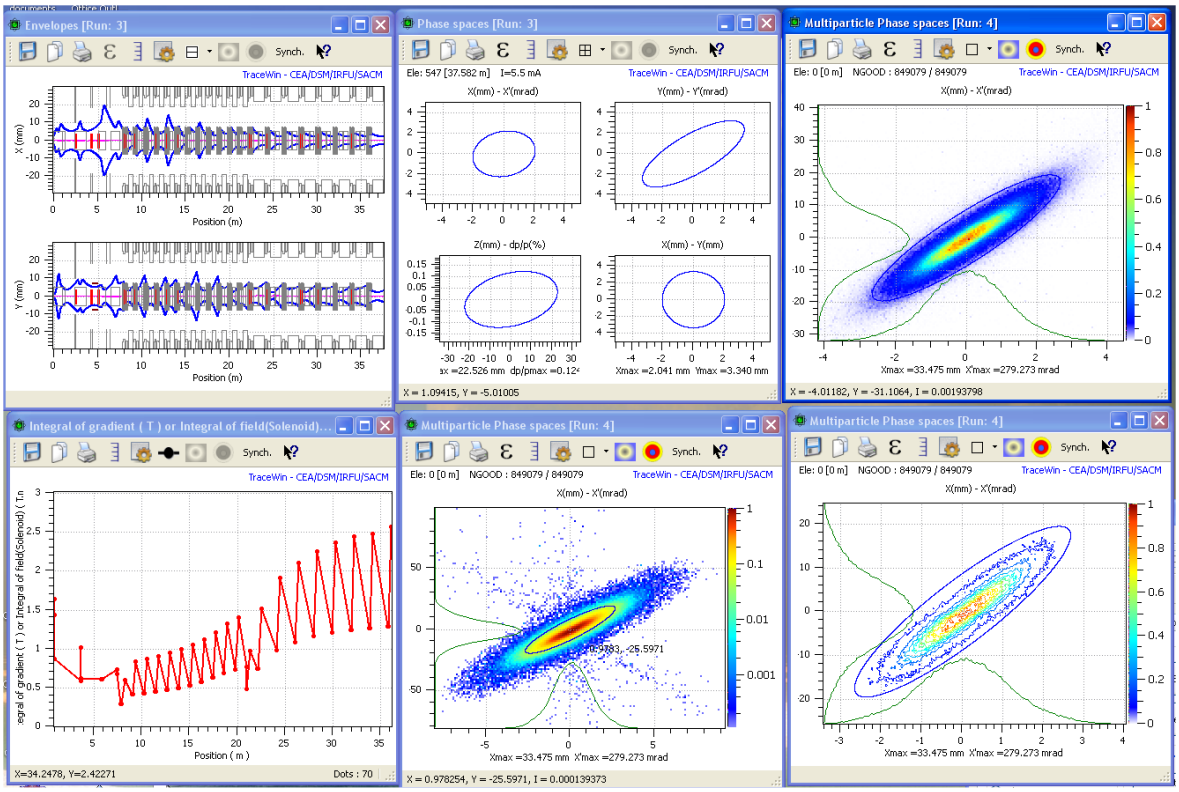
			Results saved	Affected or used beam	
	Different stages	Needed Condition(s)		Main	Secon d
1	Read input data file and set tab in "***Data***" page	Data file			
2	Transport of the reference particle			X	
3	Set Phase advance law (set Quadrupole, Solenoid, Field map strengths)	SET_ADV commands in data file or "***Use file for phase adv definition***" checked LATTICE commands in data file		X	
4	Read particle files	Particle file defined in "***Main***" page		X	X
5	Calculate the input matched beam	"Calculate match beam" checked in "Matching" page		X	X
6	Set quads or cavity strengths to match the beam through the different linac sections or set Twiss parameters (In order of theirs positions)	"Matching with family" checked in page "Matching" MATCH_FAM commands and (LATTICE or SET_TWISS command) in data file	X	X	
7	Diagnostics (Example: Steerers) calculations (In the order of theirs numbers)	"Match with diagnostics" checked in page "Matching" Diagnostic elements and Adjust commands in data file	X	X	X
8	Input beam distribution (*.dst) is adjusted in order to fit the input beam defined in "***Main***" page or to fit the input matched beam calculated in (5)	"Calculate match beam" -> "With partran" checked in "Matching" page "Use particle file" in "Input distribution type" in "Multipart" page Particle file defined in "***Main***" page			
9	Repetition of the preceding stages (5)(6)(7) using mutliparticle code	On or several options "***With partran***" checked in "***Match***" page	X	X	X
10	Random errors generator initialised	"Reinitialize random generator" checked in "Main" page.			
11	Apply Static errors	"Include error defined in…" checked in "Main" page "***..Data No***" in "***Main***" page set to errors defined in "***Errors setup***" of page "***Error***" "***ERROR_xxx_STAT_xxxx***" commands in data file		X	X
12	Diagnostics (Example: Steerers) calculations	"Match wiht diagnostics" in "Match" page Diagnostic elements and ADJUST commands in data file		X	X

13	Repetition of the preceding stages (10 to 13) using mutliparticle code	"Launch Partran" checked in "Multiparticle" page		X	X
14	Apply dynamic errors	"Include error defined in…" checked in "Main" page "***..Data No***" in "***Main***" page set to errors defined in "***Errors setup***" of page "***Error***" "***ERROR_xxx_DYN_xxxx***" commands in data file		X	X
15	Calculates the transport line envelope	Always		X	X
	Losses and beam parameters variations estimated in envelope transport	"***Nbr of particles***" in "***Main***" page greater then 10 "***Use aperture element***" checked in "Main" page		X	X
16	Write new data file in "*calculation directory"*	Always			
17	Make the Error studies (envelope). N linacs, Loop with stage (10,11,12,14,15)	*"**Study Envelope**"* checked in "***Errors***" page and error selection done		X	X
18	Make the Error studies (Particles). N linacs, Loop with stage (10,11,12,13,14,15,19)	*"**Study Multipartilce**"* checked in "***Errors***" page and error selection done		X	X
19	Write input files of multiparticle codes PARTRAN and TOUTATIS and launch them	"Launch Partran or Toutatis" checked in "Multiparticle" page		X	X

Somme help about element or command in the data file editor can be obtained by using the right mouse button. I you don't release it; the element number is shown before.



Different king of plot, **explore and test** all the top buttons to well understand all available options.



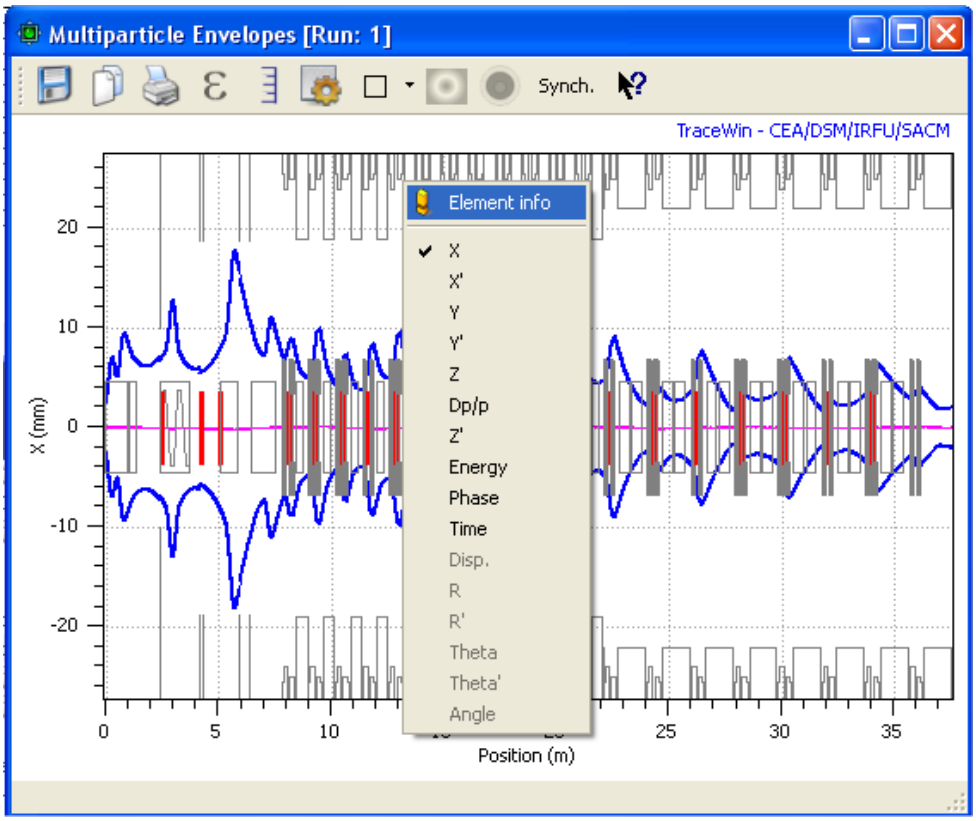
Zoome with left button mouse.
Zoome back with double click.
Move, only after zooming, with right button mouse.
Save, in several picture formats, pdf, ps, data ASCII file.
Copy in png format.
Plot options: Color, size, font, dots type… are configurable.
Synch.: All chart can be synchronize with current process.
Envelope plot:
Element info can be obtain by right button menu on a element, the envelope types (*X, Y, X', Y', Z, Z', Phase, Energy, Z, dp/p,…*) can be selected by using this menu.
The plot area contains at most 6 plots.
Phase space plot:
The selection of the **plot phase-space coordinates** is obtained by clicking right on the chart. In this case, one can either select one of the 8 proposed 2D phase-space or chose the variable for the horizontal and vertical axis.
On can "**Select"** or "**Unselect"** the visible particles on the chart area. The sizes, colors of the selected or non selected particles are configurable by using the option button (very convenient to localize in 6D phase-space some particles and study their behavior in a line).
The plot area contains at most 6 plots.

The Emittance button "**ε**" calculates statistic information about plotted distribution. The code does the same calculation in the total number of phase-space plotted whatever they are.

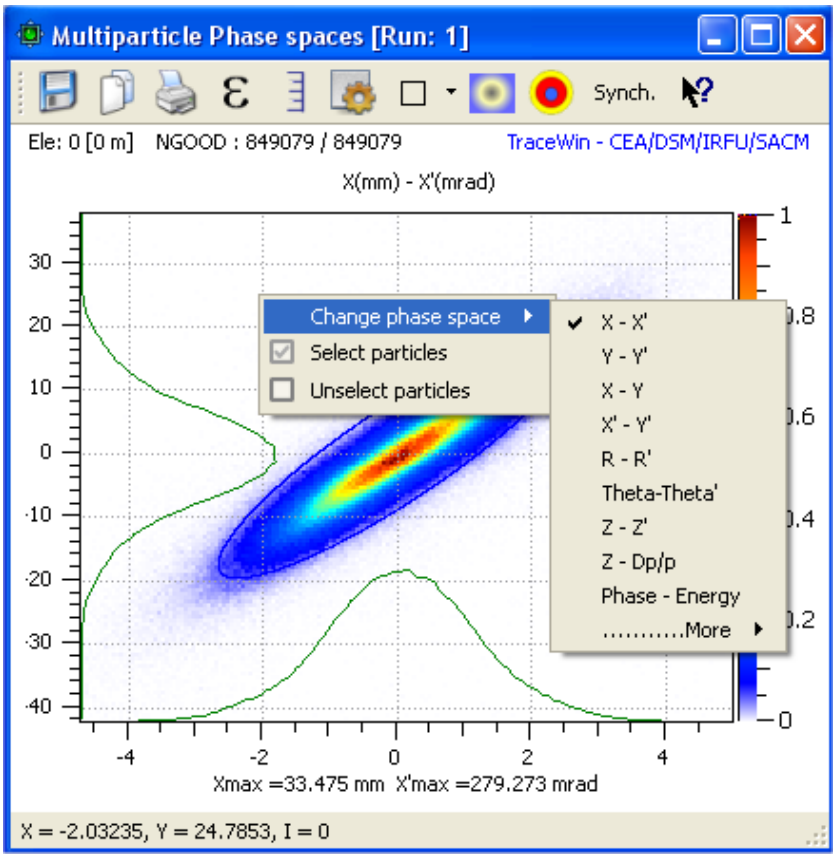
- Emit [rms] = $\sqrt{\left\langle (x-\langle x\rangle)^2 \right\rangle \cdot \left\langle (y-\langle y\rangle)^2 \right\rangle - \left\langle (x-\langle x\rangle)\cdot(y-\langle y\rangle) \right\rangle^2}$,
- Bet = $\left\langle (x-\langle x\rangle)^2 \right\rangle / \text{Emit [rms]}$,
- Alp = $-\left\langle (x-\langle x\rangle)\cdot(y-\langle y\rangle) \right\rangle / \text{Emit [rms]}$,

Emit [xx%] gives the ellipsis surface divided by homothetic to the rms ellipsis containing xx% of the beam particles. The xx is either the calculated faction of particles inside an homothetic ellipses

whose area is N times the rms ellipses, or a given fraction of the beam. The plotted ellipses are those last ellipses. Position of the c.o.g. and size of the beam can be also calculated. Finally, one can plot 2 graphs, the first showing the evolution of the number of particles outside a given emittance (scaled to the rms emittance calculated above), the second showing the evolution of the number of particles outside a given size (scaled to the rms size calculated above). The last button (σ beam) shows 6x6 the beam matrix.
The emittances are calculated according to "***Energy and Phase limit***" defined in "***Multipartcle***" page.



Envelope plot example



Multiparticle phase space plot

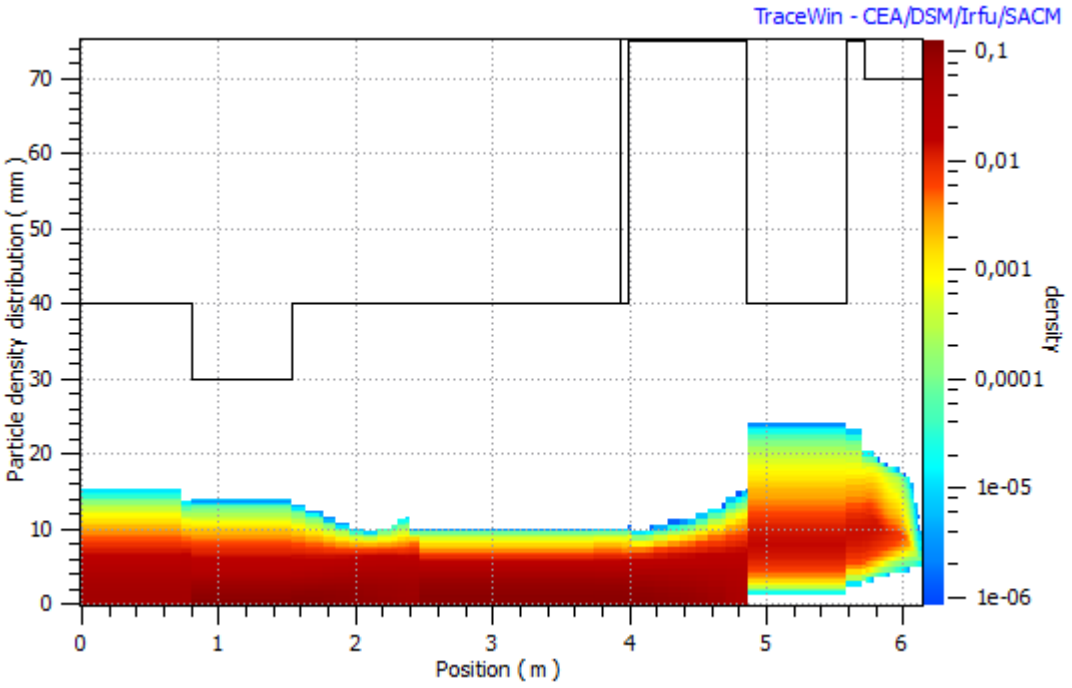
PlotWin code

PlotWin is a post-processing tool allowing to project and plot a 6D beam distribution in 2D sub-phase-spaces, to plot beam density profiles. As much as 6.phase-spaces can be plotted on the same chart. The number of phase-spaces and the plot distribution can be chosen. The beam is represented by a set of particles with the same weight. This tool allows to observe each particle transport individually.

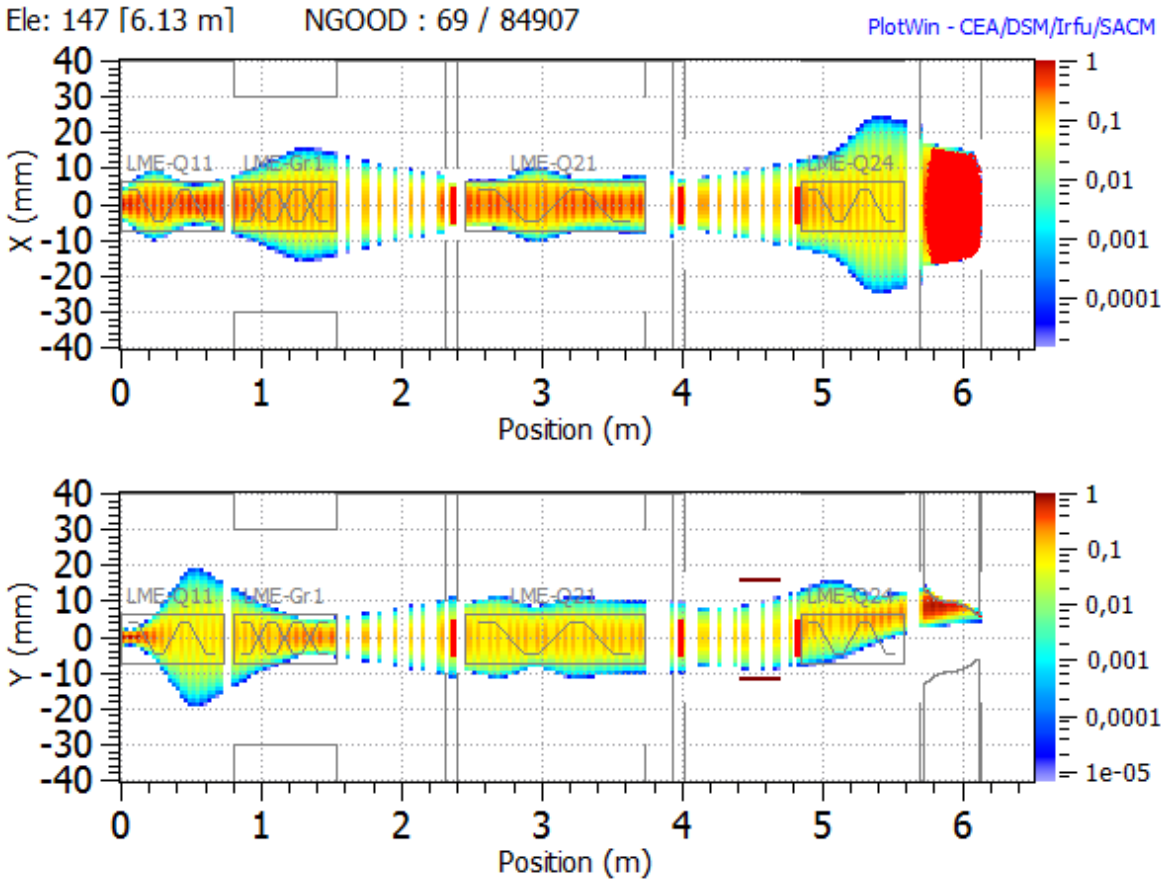
You can find the code at this location: http://irfu.cea.fr/Sacm/logiciels/

The beam density profile performed by TraceWin with "*Density*" button is just a roughly view of the beam density in *R* space. The aperture of element is divided in 100 rings, and for each of them the number of particles inside is counted in order to generate the density plot.

PlotWin provides much better quality density plots, like the example in the following page.



Distribution plot from **TraceWin**



The same distribution plot from **PlotWin**

Files

Data file (*.dat)
Init project file (*.ini)
Results file (*.cal)
Adjusted value file (*.txt)
Steerer strength file (*.txt)
Sigma0 file (*.sig0)
Input file for multiparticle program (*.par, *.dat)
Particle distribution (*.dat)
Particle losses distribution (*.loss)
Error file results (*.txt)
Input & Output particle distribution (*.dst, *.plt)
Partran or Toutatis output (*.out)
Electric or magnetic field map
Current or space charge compensation map (*.scc)
Aperture map (*.ouv)
Magnetic stripping file (*.los)
Gas stripping file (*.los)

Data file

The data file *("*.dat*")* contains the list of elements and commands. It must be ended by the command "END". The elements and the commands syntax are described in the rubric "Element definitions" and "Command definitions". Comments line begins by the character ';'.
A name for each element can be specified with 8 characters max, (See example below).
Result files are automatically created at the first use of the data file. At the end of a run, TraceWin creates another data file with the same name but located in the calculation directory, which contains the new element list. With, for example, the quadrupole values calculated to have the wanted phase advance law. If calculation directory is the same than the data file directory, the name of the new data file start with "*new_...*"
Warning:
Each command concerns the following element, par example "SET_TWISS" will impose some Twiss parameters at the output of the following element.
Two identical commands cannot be consecutive.

Example 1:

```
; ************************************************
DRIFT 1e-08 100
SPACE_CHARGE_COMP 0.7
DRIFT 350 100
DRIFT 60 100
DRIFT 192 100
MATCH_FAM_GRAD 1 1
ADJUST 1 2 1 0 0
SOLENOID 410 0.25 100
DRIFT 100 100
MATCH_FAM_GRAD 1 2
STEERER 0 0 100 0
ADJUST_STEERER 2
ADJUST 1 2 2 0 0
QUAD 200 0.18 100 0
DRIFT 150 100
END
;************************************************
```

Example 2:

```
; ************************************************
DR1    : DRIFT 1e-08 100
SP1    :  SPACE_CHARGE_COMP 0.7
DR2    : DRIFT 350 100
DR3    : DRIFT 60 100
SOL 1 : SOLENOID 410 0.25 100
DR5    : DRIFT 100 100
QPF1   : QUAD 200 0.18 100 0
DR6    : DRIFT 150 100
;************************************************
```

Init project file

The init file *"project_name.ini*" contains all the TraceWin project parameters. It can be loaded, saved, copied by using the TraceWin menu.

Results file

Created by TraceWin, his name is "*Data_file_name.cal*" and it is located in the data file directory and contains the results of the already done matching calculations, to avoid redundant calculations. See the following example.

Twiss_parameters_of_matched_beam
0.3167415265 0.1850852302 0.5246751875
-0.0938830920 0.0822867115 -0.0875778140

Matching_Between_Section_1_to_2
-8.11004 8.16711 -8.19803 8.21871
-3.8207 -2.8207 -4.1476 -1.1476
0.00834559 0.000887792
BEAM_FAM_69_0.DST

The three first lines are written after a matching beam calculation. The second line contain the Twiss parameters $\beta_{xx'}$, $\beta_{yy'}$, $\beta_{zz'}$, and the last $\alpha_{xx'}$, $\alpha_{yy'}$, $\alpha_{zz'}$.
The five following lines are written after a matching calculation, it contains the result of a matching between two sections. The first line contains the quadrupole gradients witch have been adjusted ("MATCH_FAM_GRAD" command), the second line is either the phase shift or the field factor correction or both, of the accelerator elements witch have been adjusted ("MATCH_FAM_PHASE", "MATCH_FAM_FIELD" or "MATCH_FAM_LFOC" command). The third line correspond to the element length witch have been adjusted ("MATCH_FAM_LENGTH" command) and the last is the name of a beam distribution file (located in the file data path), witch is save when the matching family calculated is done with the option "*With beam from Partran*". All of these lines are optional and depend from the "MATCH_FAM…"command in your data file.
For more details, see the matching commands and its examples. You can also force the optimization process of calculation with starting values by the following commands

Init_Matching_Between_Section_1_to_2
-8.11004 8.16711 -8.19803 8.21871
-2.8207 -2.8207 -3.1476 -3.1476

To put in comment a result, simply add the char **";"** as a first character.
This file contains also all the diagnostic results like the following example. For more details, see the adjust commands and its examples. You can also force the optimization process of diagnostic calculation with starting values by using "***Init_***" syntax

Diagnostic_10
10.7992 -10.5893 5.81701

Init_Diagnostic_10
10.7992 -10.5893 5.81701

For all these result, an extension "*_PAR*" is added when the result comes from a multiparticle optimization '***With Partran***" is checked

Sigma0 file

Created by TraceWin or No, his name is "**.sig*". It's located in the data file directory and contains the transverse phase advances law with no current, one value per lattice. See following example, where red values correspond to optional vertical pahse advance, by default sigy=sigx.

60 40
60 41
61 41
62 42
…
..
.

Magnetic or electric Field map

Input field for "FIELD_MAP" element, see also FIELD_MAP details.
In "Chart" page a tool allows to visualize (1D or 2D) the field maps from elements defined in data file. This tool also allows to convert the field ASCII format to Binary format. That allows code to be faster when the field maps size are too big.

The field map file syntax is the following in the ASCII format:

Fz are in MV/m for electric field or in T for magnetic field.
The dimensions are in meter.

- Dimension 1 :

nz zmax
Norm
for k=0 to nz
    Fz(k.zmax/nz)
    Return

- Dimension 2 :

nz zmax
nr rmax
Norm
for k=0 to nz
    for i=0 to nr
        Fz(k·z_{max}/nz, i·r_{max}/nr)
        Return

or

nx xmin xmax
ny ymin ymax
Norm
for k=0 to ny
    for i=0 to nx
        Fz(k·x_{max}/nx, i·y_{max}/ny)
        Return

- Dimension 3 :

nz zmax
nx xmin xmax
ny ymin ymax
Norm
for k=0 to nz
    for j=0 to ny
        for i=0 to nx
            Fz(k·z_{max}/nz, y_{min}+j·(y_{max}-y_{min})/ny, x_{min}+i·(x_{max}-x_{min})/nx)
            Return

The field map file syntax is the following in the BINARY format:

- Dimension 1 :
nz (integer 4 bytes) zmax (double 8 bytes)
Norm (double 8 bytes)
for k=0 to nz
    Fz(k.zmax/nz) ***(float 4 bytes)***

- Dimension 2 :
nz (integer 4 bytes) zmax (double 8 bytes)
nr (integer 4 bytes) rmax (double 8 bytes)
Norm (double 8 bytes)
for k=0 to nz
    for i=0 to nr
        Fz(k·z_{max}/nz, i·r_{max}/nr) ***(float 4 bytes)***

- Dimension 3 :
nz (integer 4 bytes) zmax (double 8 bytes)
nx (integer 4 bytes)xmin (double 8 bytes) xmax (double 8 bytes)
ny (integer 4 bytes) ymin (double 8 bytes) ymax (double 8 bytes)
Norm (double 8 bytes)
for k=0 to nz
    for j=0 to ny
        for i=0 to nx
            Fz(k·z_{max}/nz, y_{min}+j·(y_{max}-y_{min})/ny, x_{min}+i·(x_{max}-x_{min})/nx) ***(float 4 bytes)***

Warning: The lattice has to be regular.
The normalization factor is equal to k_e/Norm or k_b/Norm.

Fz are in MV/m for electric field or in T for magnetic field.
The dimensions are in meter.

Current or space charge compensation map

"FileMapName.scc"
A flag in "FIELD_MAP" element syntax allow to include it.
The space charge compensation or current file syntax is like following:

- Space charge compensation according to *Z* format:
0 N
for i=0 to N-1
    Z_i Scc_i

- Current evolution according to *Z* file format:
1 N
for i=0 to N-1
    Z_i I_i

- *Zi* is the position (m)
- *Scci* is the space charge compensation at the *Zi* position, (1 for 100%)
- *Ii* is the current (mA) at the *Zi* position

Partran and TraceWin codes make an interpolation in between this figure.

Aperture map

"FileMapName.ouv"
A flag in FIELD_MAP element syntax allow to include it.
For the field map elements, sometime we need to define a beam pipe radius geometry according to z axis. The file syntax is the following:
Warning in case of superposed field map these aperture map have to be defined in the first FIELD_MAP element and have to get a length equivalent to all field_map.

- Aperture according to *Z* format:
N
for i=0 to N-1
Z_i Ouv_i

- *Zi* is the position (m)
- Ouv*i* is the aperture radius(m) at *Zi*.

The first location Zi has to be 0.

Input files for multiparticle programs

At the end of a calculation TraceWin creates the input files for multiparticle library, PARTRAN (*Data_file_name.par*), and TOUTATIS (*toutatis.dat*).

Particle distribution

Dist_Error_Env.dat
Contain the beam distribution at the end of each element after an envelope calculation. This file is created if "***nbr of particles"*** is greater than 10 and "***Use aperture element"*** of ***"Main"*** page is selected. During an error study the condition "***nbr of particles"*** is sufficient. You can visualize these results in the "***Error***" page by setting the "***Distribution file***" and using the right buttons

Dist_Error_PAR.dat
Contain the beam distribution at the end of each element after a multiparticle calculation. This file is created either by PARTRAN or TOUTATIS. You can visualize these results in the "***Error***" page by setting the "***Distribution file***" and using the right buttons

- N: Number of linac = 1
- Element number
- Element aperture (cm)
- Element aperture (cm)
- $\sum_{1}^{N} \sqrt{x^2 + y^2}$ (cm)

- $\sum_{1}^{N} x^2 + y^2$ (cm^2)
- $\sum_{1}^{N} x$ (cm)
- $\sum_{1}^{N} x^2$ (cm^2)
- $\sum_{1}^{N} y$ (cm)
- $\sum_{1}^{N} y^2$ (cm^2)
- 100 integers corresponding to the particle distribution along the aperture divided in 100 steps.
- 100 Doubles corresponding to the power distribution along the aperture divided in 100 steps.
- $\sum_{1}^{N} particle\_lost$
- $\sum_{1}^{N} particle\_lost^2$
- Max particle lost
- Min particle lost
- $\sum_{1}^{N} power\_lost$ (w)
- $\sum_{1}^{N} power\_lost^2$ (w)
- Max power lost (w)
- Min power lost (w)

In case of statistical error study 2 new files are created name **Dist_Error_tot_ENV.dat** and **Dist_Error_tot_PAR.dat** witch contain the sum of the 2 preceding files **(N>1).**
If the "*Nbr of Step*" parameter of the tab-sheet "*Error*" is bigger than 1 the name of the 2 files become for example for 5 steps

"Dist_Error_Tot_Env_0.2000.dat" for 20%
"Dist_Error_Tot_Env_0.4000.dat" for 40%,
…
"Dist_Error_Tot_Env_1.0000.dat" for 100%,

Steerer strength file

The file "*Steerer_Values.txt*" is created after diagnostic position calculation. Its syntax is number of diagnostic follows by all the steerer strengths associated in T.m (Plane X and Y).

In case of statistical error study a new file named "*Steerer_Values_Tot.txt"* is written including all the steerer strength of the whole linac simulated.

Adjusted values file

The file "*Adjusted_Values.txt*" is created after each diagnostic optimization adjusting some elements parameters.

In case of statistical error study a new file named "*Adjusted_value_Tot.txt"* is written including all the element parameters adjusted of the whole linac simulated.

Magnetic stripping file

The file "*MAGSTRIP1.LOS*" is created only in multiparticle mode if option "*Magnetic stripping*" is selected in "*Option*" of multiparticle codes. You can directly exploit these results using "*Stripping*" button in tab-sheet "*Graphs*". It contains the probability losses due to Lorentz magnetic stripping. The syntax is the following. (One line per element):

Number of element, Position (m), Energy (MeV), Losses probability

In case of statistical error study a new file named "*MAGSTRIP1_TOT.LOS"* is written including the probability sum of the whole linacs simulated. You can directly exploit these results using "***Stripping losses probability results"*** button in "***Errors***" page. The syntax is the following (One line per element):

Number of element, Number of linac simulated, Sum of losses probability

The probability corresponding to Sum of Magnetic stripping probability divided by Number of linac simulated.

Gas stripping file

The file "*GASSTRIP1.LOS*" is created only in multiparticle mode if option "Gas *stripping*" is selected in "*Option*" of multiparticle codes and if command Gas pressure is included in the data file. It contains the probability losses due to Gas stripping. The syntax is the following. (One line per element):

Number of element, Position (m), Energy (MeV), Losses probability

In case of statistical error study a new file named "GAS*STRIP1_TOT.LOS"* is written including the probability sum of the whole linacs simulated. You can directly exploit these results using "***Stripping losses probability results"*** button in "***Errors***" page. The syntax is the following (One line per element):

Number of element, Number of linac simulated, Sum of losses probability

The probability corresponding to Sum of Gas stripping probability divided by Number of linac simulated.

Particle losses distribution

The following file are created only while statistical error study of *k* linacs
Dist_Error_Tot_Env.loss
Contain the number of losses at the end of each element after an envelope calculation. This file is created if "***nbr of particles"*** is greater than 10. You can visualize these results in the "***Error***" page by setting the "***Distribution file***" and using the right buttons

Dist_Error_Tot_PAR.loss
Contain the number of losses at the end of each element after a multiparticle calculation. This file is created either by PARTRAN. You can visualize these results in the "***Error***" page by setting the "***Distribution file***" and using the right buttons

- Number of elements
- Number of particles

- Linac 1 (Losses element 1, Lost power element 1, losses element 2, Lost power element 2,… losses element N, Lost power element N)

- Linac 2 (Losses element 1, Lost power element 1, losses element 2, Lost power element 2,… losses element N, Lost power element N)

….

- Linac k (Losses element 1, Lost power element 1, losses element 2, Lost power element 2,… losses element N, Lost power element N)

Input & Output particle distribution

These following files are created while multiparticle simulation
part_dtl1.dst: Binary file containing the output beam distribution at the end of the linac.
part_rfq1.dst: Binary file containing the beam distribution at the entrance of the linac.

A .dst file use a binary format. It contains informations of a beam at a given longitudinal position: number of particles, beam current, repetition frequency and rest mass as well as the 6D particles coordinates. The format is the following:

CHAR(125)+CHAR(100)+INT(Np)+DOUBLE(Ib(A))+DOUBLE(freq(MHz))+CHAR(0)+
Np×[6×DOUBLE(x(cm),x'(rad),y(cm),y'(rad),phi(rad),Energie(MeV))]+
DOUBLE(mc2(MeV))

Comments:
- CHAR is 1 byte long ,
- INT is 4 bytes long,
- DOUBLE is 8 bytes long.
- Np is the number of particles,
- Ib is the beam current,
- freq is the bunch frequency,
- mc2 is the particle rest mass.

dtl1.plt: Binary file containing the beam distribution at the end of each element.

A .plt file use a binary format. It contains information of a beam at many longitudinal positions: longitudinal position, number of particles, beam current, repetition frequency and rest mass as well as the 6D particles coordinates. The format is the following:

CHAR(125)+CHAR(100)+INT(Ne)+INT(Np)+DOUBLE(Ib(A))+DOUBLE(freq(MHz))+
DOUBLE(mc2(MeV))+
Ne×[CHAR(0)+INT(Nelp)+DOUBLE(Zgen)+DOUBLE(phase0(deg))+DOUBLE(wgen(MeV))+
Np×[7×FLOAT(x(cm),x'(rad),y(cm),y'(rad),phi(rad),Energie(MeV),Loss)]]

Comments:

- CHAR is 1 byte long,
- INT is 4 bytes long,
- FLOAT is a Real 4 bytes long.
- DOUBLE is a Real 8 bytes long.
- Ne is the number of different positions,
- Np is the number of particles,
- Ib is the beam current,
- freq is the bunch frequency,
- mc2 is the particle rest mass,
- Nelp is the longitudinal element position,
- Zgen is the longitudinal position in meter,
- Phase0 & wgen are the phase and energy references of the beam,

Compressions format

Error file result

The final results can be found on the calculation directory. Named "*Error_study_Name_TRA.txt*" when the result comes from an envelope calculation and "*Error_study_Name_PAR.txt*" when it is a PARTRAN study. The format of these files is the same for each kind of error. For each step of calculation one line of 11 parameters is written with the following format.

- Step of error (0->1)
- 1-(Nbr of particles)/(Nbr of particles, reference case)
- (Emittance rms xx',yy',zz')/ (Reference rms emittance xx',yy',zz')-1
- X beam center (m)
- Y beam center (m)
- X' beam center (rad)
- Y' beam center (rad)
- Energy beam center (keV)

- Phase beam center (deg)
- RMS Beam X size (m)
- RMS Beam Y size (m)
- RMS Beam X' size (m)
- RMS Beam Y' size (m)
- RMS Beam Energy size (keV)
- RMS Beam Phase size (deg)
- Halo parameter xx'
- Halo parameter yy'
- Halo parameter zz'

Halo definition.

All these values are relative to the output beam without errors.

In case of statistical study, where each step of calculation contains several runs, the format becomes:

- Step of error (0->1)
- AVERAGE(1-(Nbr of particles)/(Nbr of particles, reference case))
- (AVERAGE(Emittance rms xx',yy',zz'))/ (Reference rms emittance xx',yy',zz')-1
- RMS(X beam center (m))
- RMS(Y beam center (m))
- RMS(X' beam center (rad))
- RMS(Y' beam center (rad))
- RMS(Energy beam center (keV))
- RMS(Phase beam center (deg))
- AVERAGE(RMS Beam X size (m))
- AVERAGE(RMS Beam Y size (m))
- AVERAGE(RMS Beam X' size (m))
- AVERAGE(RMS Beam Y' size (m))
- AVERAGE(RMS Beam Energy size (keV))
- AVERAGE(RMS Beam Phase size (deg))
- AVERAGE(Halo parameter xx')
- AVERAGE(Halo parameter yy')
- AVERAGE(Halo parameter zz')

And a file call "*Error_study_Name_TRA_tot.txt*" or "*Error_study_Name_PAR_tot.txt*" is written containing all run results.

Partran and Toutatis output

The final multiparticle results contain one line by element output, the first line being the input beam parameters. The format is like following.

- Element number
- Element position (m)
- Relativistic parameters: (γ-1)
- Centroid position: x(mm), y(mm), Phase(°), x'(mrad), y'(mrad), W(MeV)
- RMS_SIZE(x(mm), y(mm), Phase(deg))
- RMS (xx'(mm.mrad), yy'(mm), Phase.Energy(deg.MeV))
- Normalized rms emit: xx'(mm.mrad), yy'(mm.mrad), PW (Deg.MeV).

- Halo parameters: ($H_{xx'}$, $H_{yy'}$, $H_{z.dp/p}$)
- Number of particles
- Phase advance with space charge (deg): σ_x, σ_y, σ_z.
- Emittance at 99%: $\varepsilon_{xx'}$, $\varepsilon_{yy'}$, $\varepsilon_{z.dp/p}$
- Beam currant (mA) used for space charge calculation
- Aperture (mm)
- Normalized 4D transverse emittance $E_{xx'yy'}$ $(mm.mrad)^2$
- Normalized rms emit (mm.mrad): $\varepsilon_{rr'}$.
- Phase advance with space charge (deg): σ_r
- Lost power (w)
- Maximum excursion particle : Xmax(mm), Ymax(mm),
- Normalized long. rms emit: $\varepsilon_{z.dp/p}$ (mm.mrad) [replace PW(Deg.Mev)]
- Dispersion: Dh (mm), Dv(mm)
- Derivative dispersion: Dh' (mrad), Dv'(mrad)

Since TraceWin version 2.1.0.0 longitudinal rms emittance PW is set to zero and has been replaced by $\varepsilon_{z.dp/p}$

Elements

Drift

Mnemonic	Parameter	Definition
DRIFT	*L*	Length (mm)
	R	Aperture (mm)
	Ry	Aperture (mm)

If *Ry* equal 0 aperture is circular with *R* radius.
If *Ry* not equal 0 aperture is rectangular with *R* for x plane and *Ry* for Y plane.

Drift matrix

Quadrupole

Mnemonic	Parameter	Definition
QUAD	*L*	Length (mm)
	G	Magnetic field gradient (T/m)
	R	Aperture (mm)
	Θ	Skew Angle (°)
	G_3	Sextupole gradient (T/m^2)
	G_4	Octupole gradient (T/m^3)
	G_5	Decapole gradient (T/m^4)
	G_6	Dodecapole gradient (T/m^5)

Red value are optionnal
If L equal 0, quadrupole element are simulated as thin lens and all gradient components have to be replace by the gradient integral values.
If L not equal to 0, multipole kicks are applied at the half of the quadrupole.
Remark: Short MULTIPOLE elements (based on field map) have to be replaced by QUAD element with multipole component. Quad element is much more efficient in terms of speed.

Quadrupole matrix

Beam Rotation

Mnemonic	Parameter	Definition
BEAM_ROT	*θxy*	Angle (°) in the XY space around Z
	θxz	Angle (°) in the XZ space around Y
	θyz	Angle (°) in the YZ space around X
	dx	X shift (mm)
	dy	Y shift (mm)
	dxp	Xp shift (mrad)
	Dyp	Yp shift (mrad)
	dz	Z (mm)

Rotation are performed on after the other (in the order: XY, XZ, YZ) and finally the shifts.
BEAM_ROT is an element

Beam rotation matrix

Thin Lens

Mnemonic	Parameter	Definition
THIN_LENS	*fx* *fy* *R*	Focal Length (mm) Focal Length (mm) Aperture (mm)

Thin lens matrix

Thin Matrix

Mnemonic	Parameter	Definition
THIN_MATRIX	*lg* a_{00} *to* a_{55}	Length (mm) Matrix terms, row per row

The 36 terms of matrix have to be set row by row from a00 to a55. Length, Lg, is just used in graphic view.
See matrix format *R*
The units are respectively: x (m), x' (rad), y (m), y' (rad), z (m), dp/p (rad)

Quadrupole Example: THIN_MATRIX 10.0 1 0 0 0 0 0 -2.5 1 0 0 0 0 0 0 1. 0 0 0 0 0 2.5 1 0 0 0 0 0 0 1 0 0 0 0 0 0 1

Electrostatic Quadrupole

Mnemonic	Parameter	Definition
QUAD_ELE	*L* *Vo* *R* *Θ* V_3 V_4 V_5 V_6	Length (mm) Voltage between electrodes (V) Aperture (mm) Skew Angle (°) Sextupole voltage component (V/m) Octupole voltage component (V/m^2) Decapole voltage component (V/m^3) Dodecapole voltage component (V/m^4)

Vo between electrodes means +Vo/2 on one electrode and –Vo/2 on the near electrode for a total voltage of Vo.
In order to keep coherence with old TraceWin version, all electrostatic components are in voltage: (V) and not in field: E (V/m)
If L equal 0, electrostatic quadrupole element are simulated as thin lens and all voltage components have to be replace by the voltage integral values.
If L not equal to 0, multipole kicks are applied at the half of the electrostatic quadrupole.

Quadrupole matrix

Bunched cavity or thin gap

Mnemonic	Parameter	Definition
GAP	E_0TL	Effective gap voltage (V)
	Θ_S	RF phase (deg) (absolute or relative)
	R	Aperture (mm)
	P	1: θ_s is absolute phase, 0: θ_s is relative
	β_S	Particle reduced velocity
	T_S	Transit time factor
	kT'_S	(*)
	$k^2T''_S$	(*)
	kS'	(*)
	k^2S''	(*)

Red parameters are optional

(*) See Transit time factor definition

Bunched cavity or thin gap matrix

Sinus cavity or CCL

Mnemonic	Parameter	Definition
CAVSIN	L	Length (mm)
	N	Cell number
	E_oT	Average accelerating field (V/m)
	Θ_S	Phase of the synchronous particle at the entrance(deg)(*)
	R	Aperture (mm)
	P	1: θ_s is absolute phase, 0: θ_s is relative

(*) Use SET_SYNC_PHASE command in order to change this phase as the synchronous phase

Sinus cavity or CCL matrix

Bending magnet

Mnemonic	Parameter	Definition
BEND	α	Bend angle in the rotation plane (deg)
	$\|\rho\|$	Curvature radius of central trajectory (mm)
	N	Field gradient index
	R	Aperture (mm)
	HV	0 : horizontal, 1 : vertical

Warning: A bend always have to be preceded and followed by edge element, even if edge is set to zero.
A positive bend (denoted $\alpha>0$) bends the particles to the right in the horizontal plane, regardless of the sign of the charge.
The bend with field gradient index not equal to zero are only treated in the first order and for horizontal bending

Bending magnet matrix

Edge angle on bending magnet

Mnemonic	Parameter	Definition
EDGE	B	Pole face rotation angle (deg)
	$\|\rho\|$	Curvature radius of bend (mm)
	G	Total gap of magnet (mm)
	K_1	Fringe-field factor (default = 0.45)
	$K2$	Fringe-field factor (default = 2.80)
	R	Aperture (mm)
	HV	0 : horizontal, 1 : vertical

By definition: An edge focalize the transverse x plane if the pole $\beta < 0$, whatever the curvature radius sign or the bend angle sign and the particle charge.
Set *K1* and *K2* to a very small value to disable fringe field estimation.
The gap of magnet is used for particle loss estimation in bend.

Edge angle on bending magnet matrix

Electrostatic bend

Mnemonic	Parameter	Definition
BEND_ELE	α	Bend angle in the rotation plane (deg)
	$\|\rho\|$	Curvature radius of central trajectory (mm)
	ne	1:Cylindrical, 2:Spherical, 3:Toroidal
	R	Aperture (mm)
	HV	0 : horizontal, 1 : vertical

A positive bend (denoted $\alpha>0$) bends the particles to the right in the horizontal plane, regardless of the sign of the charge.
Only Cylindrical bend is available in multiparticle mode

Circular or rectangular aperture

Mnemonic	Parameter	Definition
APERTURE	d_x	X half width (mm) or radius hole
	d_y	Y half width (mm) or distance between hole
	n	0 : Rectangular aperture 1 : Circular aperture 2 : Perperpot mode 3 : Rectangular aperture with dx & dy corresponding to a beam fraction. (aperture is adjusted with 0.1 mm step) 4 : Horizontal finger with dx=total finger width and dy=finger position. 5 : vertical finger with dx=total finger width and dy=finger position. 6 : Ring aperture if particle radius, $r>dy$ or $r<dx$ with ($dx<dy$) particle is lost

Space charge compensation

Mnemonic	Parameter	Definition
SPACE_CHARGE_COMP	*k*	Beam current is compensated by a factor *k*

Beam current

Mnemonic	Parameter	Definition
CURRENT	*Ib*	Beam current (mA)

Ib (mA) is the new current beam.

Solenoid

Mnemonic	Parameter	Definition
SOLENOID	*L*	Length (mm)
	B	Magnetic field (T)
	R	Aperture (mm)

Solenoid matrix

Thin steering

Mnemonic	Parameter	Definition
THIN_STEERING	*BLx or ELx*	x-component (T.m or V)
	Bly or ELy	y-component (T.m or V)
	r	Aperture (mm)
	Elec	If =1 electric deviation is applied

Thin steering matrix

DTL cell

Mnemonic	Parameter	Definition
DTL_CEL	*L*	Cell length (mm)
	Lq_1	First ½ quadruploe length (mm)
	Lq_2	Second ½ quadruploe length (mm)
	g	Cell center (mm)
	B_1'	First magnetic field gradient (T/m)
	B_2'	Second magnetic field gradient (T/m)
	E_0TL	Effective gap voltage (V)
	Θs	RF phase (deg)
	R	Aperture (mm)
	p	1: θ_s is absolute phase, 0: θ_s is relative
	β_s	Particle reduced velocity
	T_s	Transit time factor
	kT'_s	(*)
	$k^2T''_s$	(*)

Red parameters are optional
If β_s is not equal to *0 Ts* become needed

(*) See Transit time factor definition

DTL cell matrix

Copy following examples:
examples\DTL_1.ini
examples\DTL_1.dat
examples\DTL_1.cal

examples\DTL_2.ini
examples\DTL_2.dat
examples\DTL_2.cal

examples\DTL_3.ini
examples\DTL_3.dat
examples\DTL_3.cal

examples\DTL_4.ini
examples\DTL_4.dat
examples\DTL_4.cal

Cavity multi-gap

Mnemonic	Parameter	Definition
NCELLS	*Mode*	(0)2π, (1)π, (2)π & 2π
	N_c	Number of cell
	β_g	Geometric β
	E_oT	Effective gap voltage (V/m)
	θ	RF phase at the first gap position (deg) (**)
	R	Aperture (mm)
	P	1: θ is absolute phase, 0: θ is relative
	kE_oT_i	Input field correction, $E_o = E_o*(1+k)$
	kE_oT_o	Output field correction, $E_o = E_o*(1+k)$
	dz_i	First gap displacement (mm)
	dz_o	Last gap displacement (mm)
	β_s	Particle reduced velocity
	T_s	Transit time factor of the middle gaps
	kT'_s	(*)
	$k^2T''_s$	(*)
	T_i	Transit time factor of the input gaps
	kT'_i	(*)
	$k^2T''_i$	(*)
	T_o	Transit time factor of the output gaps
	kT'_o	(*)
	$k^2T''_o$	(*)

Red parameters are optional
If β_s is not equal to *0 Ts* become needed

(*) See Transit time factor definition
(**) Use SET_SYNC_PHASE command in order to change this phase as the synchronous phase

Cavity multi-gap matrix

Copy following examples:
examples\Supra_1.ini
examples\Supra_1.dat
examples\Supra_1.cal

examples\Supra_2.ini
examples\Supra_2.dat
examples\Supra_2.cal

examples\Supra_3.ini
examples\Supra_3.dat
examples\Supra_3.cal

examples\Supra_4.ini
examples\Supra_4.dat
examples\Supra_4.cal

examples\Supra_5.ini
examples\Supra_5.dat
examples\Supra_5.cal

RFQ cell

Mnemonic	Parameter	Definition
RFQ_CELL	*V*	Effective gap voltage (V)
	R_0	Vane radius (mm)
	A10	Acceleration parameter
	m	Modulation
	L	Length (mm)
	θ_s	RF phase (deg)
	Type	
	Tc	Transverse curvature (mm)
	A01	Transverse focusing parameter

RFQ cell matrix

See following examples:
examples\RFQ_1.ini
examples\RFQ_1.dat
examples\RFQ_1.cal

examples\RFQ_2.ini
examples\RFQ_2.dat
examples\RFQ_2.cal

Diagnostic elements

Diagnostic type	Mnemonic	Parameter	Definition
Current measurement	**DIAG_CURRENT**	*N*	Diagnostic number
		Ib	Wanted beam current (mA)
Positions measurement	**DIAG_POSITION**	*N*	Diagnostic number
		X	Wanted X beam position (mm), if *X*<1e50
		Y	Wanted Y beam position (mm), if *Y*<1e50
		dm	Diagnostic Accuracy (mm)
Divergences measurement	**DIAG_DIVERGENCE**	*N*	Diagnostic number
		X'	Wanted X beam divergence (mrad) if X'<1e50

		Y' *dm*	Wanted Y beam divergence (mrad), if Y'<1e50 Diagnostic Accuracy (mrad)
Size measurement	**DIAG_SIZE**	*N* *Sx* *Sy* *ΔP* *Dm* *dΔP* *fo*	Diagnostic number Wanted X rms beam size (mm) Wanted Y rms beam size (mm) Wanted rms Phase spread (°) Size Accuracy (mm) Phase spread Accuracy (°) *ΔP* Low-pass filter frequency (MHz)
Divergence measurement (*) if S<0 Twiss parameter alpha will set <0	**DIAG_SIZEP**	*N* *Sx' (*)* *Sy' (*)* *ΔW* *Dm* *dW*	Diagnostic number Wanted X' rms beam divergence (mrad) Wanted Y' rms beam divergence (mrad) Wanted rms Energy spread (MeV) Divergence Accuracy (mrad) Energy spread Accuracy (%)
Delta size Measurement	**DIAG_DSIZE**	*N* x_n-y_n *dm*	Diagnostic number Wanted x-y rms beam delta size (mm) dx-y size Accuracy (mm)
Delta size Measurement 2 (At least 2 are needed)	**DIAG_DSIZE2**	*N* x_n-x_{n-1} y_n-y_{n-1} *dm*	Diagnostic number Wanted x rms beam delta size (mm) Wanted y rms beam delta size (mm) dx & dy size Accuracy (mm)
Delta phase spread Measurement 2 (At least 2 are Needed)	**DIAG_DSIZE3**	*N* ΔP_n-ΔP_{n-1} *dΔP* *fo*	Diagnostic number Wanted rms delta phase spread (°) Phase spread accuracy (°) *dΔP* Low-pass filter frequency (MHz)
Phase measurement	**DIAG_PHASE**	*N* *Θ*	Diagnostic number Wanted centroïd Phase (°)
Beam energy - Perfect linac energy measurement	**DIAG_DENERGY**	*N* *W*	Diagnostic number Wanted Energy (MeV)
Beam phase - Perfect linac phase measurement	**DIAG_DPHASE**	*N* *Θ*	Diagnostic number Wanted centroïd Phase (°)
Luminosity	**DIAG_LUMINOSITY**	*N* *Lu* *Dlu*	Diagnostic number Wanted luminosity (mm^{-2}) Luminosity accuracy (mm^{-2})
Waist setting	**DIAG_WAIST**	*N* *fx* *fy* *dxy*	Diagnostic number XX' waist asked (for *fx* not equal to *0*) YY' waist asked (for *fy* not equal to *0*) Transverse waist Accuracy

Achromat setting	**DIAG_ACHROMAT**	*N* *n* *f1* *f2*	Diagnostic number First element number If =1 set achromatic position If =1 set achromatic angle
Emittance setting (rms values)	**DIAG_EMIT**	*N* *Exx'* *Eyy'* *Epw*	Diagnostic number Wanted emittance if greater than 0 Wanted emittance if greater than 0 Wanted emittance if greater than 0
Emittance setting (99% value)	**DIAG_EMIT_99**	*N* *Exx'* *Eyy'* *Epw*	Diagnostic number Wanted emittance if greater than 0 Wanted emittance if greater than 0 Wanted emittance if greater than 0
Transfer matrix setting	**DIAG_SET_MATRIX**	*N* *N°ele* *Row(i)* *Column(j)* *k* *Mij*	Diagnostic number Transfer matrix form N°ele to diag. position Row transfer matrix term Column transfer matrix term Corrector coefficient Wanted transfer matrix term value
Beam Twiss parameters setting Parameter is used if it is not equal to 0, if you want $\alpha=0$, set very small α	**DIAG_TWISS**	$\alpha_{xx'}$ $\beta_{xx'}$ $\alpha_{yy'}$ $\beta_{yy'}$ $\alpha_{zdp/p}$ $\beta_{zdp/p'}$	Wanted alpXX' Wanted betXX' (mm/mrad) Wanted alpYY' Wanted betYY' (mm/mrad) Wanted alpZdp/p Wanted betZdp/p (mm/mrad)

TraceWin is able to verify if a beam line contains enough diagnostic elements to control the beam and correct the errors coming from the input beam or from the different element errors. In order to put one diagnostics in the line use the elements "*DIAG_...*" followed by the diagnostic number and the parameters which have to be imposed at the diagnostic location (current, position, size or emittance). To choose the different elements, which have to be adjusted to control the beam, use the adjust commands followed by the number of the associated diagnostic.
There are two ways of using these diagnostics. The first one is for example to see if your design is able to control the beam position at a given location when you input beam is not at the centre or if a few elements, like steerers induce misalignment. The second way is to use diagnostics associated with a error study in order to see if you scheme of misalignment correction for example is good or not.
The diagnostics adjustments are independent process, which occurs after the matching process. If any diagnostic element is present any adjustment is started. Finally, at the end of a line design process, only diagnostic elements should be used to tune it.
Warning: They are treated in the order of the diagnostic number.

Adjust and diagniostic examples

The diagnostic influence can be weighted by putting a correcting *factor* between brackets like following:

DIAG_EMIT(1e-4) 1 0.1 0.1 0.2 , here this diagnostic emittance criteria is reduced by a factor 10000.

By default, without bracket, the correction is set to 1.

Copy following examples:
examples\Supra_2.ini
examples\Supra_2.dat
examples\Supra_2.cal

Funneling gap

	Mnemonic	Parameter	Definition
Funneling Gap	**FUNNEL_GAP**	E_0TL Θ_s R p	Effective gap voltage (V) RF phase (deg) (absolute or relative) Aperture (mm) 1: θ_s is absolute phase, 0: θ_s is relative
Frame change	**CHFRAME**	X_0 X'_0	(mm) (deg)

Funneling gap matrix

Alpha magnet

Mnemonic	Parameter	Definition
ALPHA_MAGNET	Θ K R *plan*	Entrance angle (°) (T/m) Aperture (mm) 0 (x)/1 (y)

Alpha magnet matrix

Electrostatic Acceleration

Mnemonic	Parameter	Definition
ELECTROSTA_ACC	*Vo* *L* *K* *R*	Voltage (V) Length (mm) Transverse defocal (eV/ mm²) Aperture (mm)

Electrostatic Acceleration matrix

Field Map

Mnemonic	Parameter	Definition
FIELD_MAP	*geom*	Field map type
	L	Field map length (mm)
	θ_i	RF input field phase (°) (*)
	R	Aperture (mm)
	k_b	Magnetic field intensity factor
	ke	Electric field intensity factor
	Ki	Space charge compensation factor
	Ka	Aperture flag
	FielName	File name without extension

(*) Use SET_SYNC_PHASE command in order to change this phase as the synchronous phase.

File name is the field map name without extension and without path if files is en the data file (*.dat) path, otherwise put the full file name with path without extension

Particle motion in electromagnetic field

"geom" parameter defines the field map type as following:

Unit figure : Static electric field
Tens figure : Static magnetic field
Hundreds figure : RF electric field
Thousands figure : RF magnetic field
10 Thousands figure : 3D aperture map ---------- NEW since 2.4.0.1 version

0 - No field
1 - 1D : Fz(z)
2 - Not available
3 - Not available
4 - 2D cylindrical static or RF electric field : Ez(r,z), Er(r,z) and Bθ(r,z) for RF
5 - 2D cylindrical static or RF magnetic field : Bz(r,z), Br(r,z) and Eθ(r,z) for RF
6 - 2D Cartesian field: Fx(x,y), Fy(x,y)
7 - 3D Cartesian field: Fx(x,y,z), Fy(x,y,z), Fz(x,y,z)
8 - 3D cylindrical field : Fr(r, θ,z), F θ (r,θ,z), Fz(r,θ,z) Not implemented yet
9 - 1D : G(z), only use for magnetic quadrupole, thus *geom* parameter has to be set to *0090* (see 3D field development for more details)

New (v:2.4.0.1)
If *10 thousand figure* of *geom* parameter is not egal to zero, an aperture field map can be defined. The finame extension is "*ouv*" and the file syntax is similar to 3D field map except than the field value is set to 1 or 0 (1 for material) .

θi: RF phase when the generator particle enters the cavity.
The RF field is E=E0.cos(θ), B=-B0.sin(θ)

FileName: is the file name without extension where are localized the field maps (.bsz, .bsr, .edx …), the space charge compensation map (.scc) and the beam pipe radius evolution with z (.ouv). See the file formats description

Field in 1D: The field can be described in 1D, Field according to z position
- 1 file contains the static electric field : filename.esz
- 1 file contains the static magnetic field : filename.bsz
- 1 file contains the RF electric field : filename.edz
- 1 file contains the RF magnetic field : filename.bdz

New (v:2.3.1.8): The development of the field off-axis is make at the first order, if the *geom* parameter value is set negative, this development is performed up to the second order.

Field in 2D: The fields can be described in 2D Cartesian (invariant through translation on z axis) or in cylindrical (invariant through rotation around the z axis) coordinates.

In Cartesian coordinates (x,y)
- 2 files contain the static electric field : filename.esx, filename.esy
- 2 files contain the static magnetic field : filename.bsx, filename.bsy
- 2 files contain the RF electric field : filename.edx, filename.edy
- 2 files contain the RF magnetic field : filename.bdx, filename.bdy

In cylindrical coordinates (r,z,θ)
- 2 files contain the static electric field : filename.esr, filename.esz
- 2 files contain the static magnetic field : filename.bsr, filename.bsz
- 3 files contain the RF fields (TM) : filename.edr, filename.edz, filename.bdq
- 3 files contain the RF fields (TE) : filename.bdr, filename.bdz, filename.edq

Field in 3D:
The fields can be described in 3D either Cartesian or cylindrical frame.

In Cartesian coordinates (x,y,z)
- 3 files contain the static electric field : filename.esx, filename.esy, filename.esz
- 3 files contain the static magnetic field : filename.bsx, filename.bsy, filename.bsz
- 3 files contain the RF electric field : filename.edx, filename.edy, filename.edz
- 3 files contain the RF magnetic field : filename.bdx, filename.bdy, filename.bdz

In cylindrical coordinates (r, θ,z)
- 3 files contain the static electric field : filename.esr, filename.esq, filename.esz
- 3 files contain the static magnetic field : filename.bsr, filename.bsq, filename.bsz
- 3 files contain the RF electric field : filename.edr, filename.edq, filename.edz
- 3 files contain the RF magnetic field : filename.bdr, filename.bdq, filename.bdz

ki: If it is greater than 0, a space charge compensation map or current map has to be created in "FileName.scc" and ki is the normlization factor.
Space charge compensation file syntax

ka: If it is equal to 1, a beam pipe radius map has to be created in "FileName.ouv".

Aperture map file syntax

Losses consideration:
Particle is lost in FIELD_MAP when it comes out of aperture defined in element syntax or field_map frame size defined in field_map files. This behavior is true except when aperture flag is set to 1 in order to use beam pipe radius file map. Now, if its flag is set to 2, no pipe file is required, but particle

will not be lost when it comes out field map frame. That allows to superpose small size field map inside bigger one.
All these files have to be located in the directory of the main data file.

Copy following example:
examples\LBE_1.ini
examples\LBE_1.dat
examples\LBE_1.cal
examples\solenoid1.bsz
examples\solenoid1.ouv
examples\solenoid2.bsz
examples\solenoid2.ouv

Multipole Field Map

Mnemonic	Parameter	Definition
MULTIPOLE	*Order*	Multipole order
	L	Field map length (mm)
	step	Number of step along x & y direction
	B	Magnetic field on pole (T)
	R	Aperture (mm)
	Lsol	Physical length (mm) of solenoid (*Order=0*)
	Zstep	Step number for solenoid case (*Order=0*)

Mutipole element generates a static magnetic field map file where the step along x & y direction is defined as 2.R/step. Warning, 2.R/step must be much lower than the beam size. But, the field map size depends on *step* parameter.

And the step along z direction (beam) is defined by TraceWin calculation step.
Order = 0: (Special mode) for Solenoid field map (Br(r,z) & Bz(r,z))
Order = 1: for dipole
Order = 2: for quadrupole
Order = 3: for sextupole
Order = 4: for octupole
…

Commands

Change element parameters:
- Chopper
- Change structure frequency
- Duplicate element
- Steerer
- Set RFQ coupling gap
- Set RFQ electrode type
- Set RFQ vane geometry
- Shift
- Structure file end

Change frequency

FREQ *f*(Mhz)

Change the R.F. frequency of the following structure, the beam frequency is not affected.

Begin of lattice

LATTICE *n1, n2*

Define the periodic focusing lattices, *n1* is the number of element per lattice, *n2* is the number of lattice per macro-lattice (generally 1). You don't have to repeat this command; all following elements are considered as part of the lattice structure until reaching the commands LATTICE_END or END.

Copy following examples:
examples\DTL_1.ini
examples\DTL_1.dat
examples\DTL_1.cal

End of lattice

LATTICE_END

End the periodic focusing lattices.

Structure file end

END

A structure file description must finish by the word "END".

Set phase advance

SET_ADV kx_{0b} ky_{0t}

Set the horizontal transverse phase advance with no current to kx_{0t}
Set the vertical transverse phase advance with no current to ky_{0t}
By default without ky_{0t} value $ky_{0t} = kx_{0t}$

See the phase advance definition

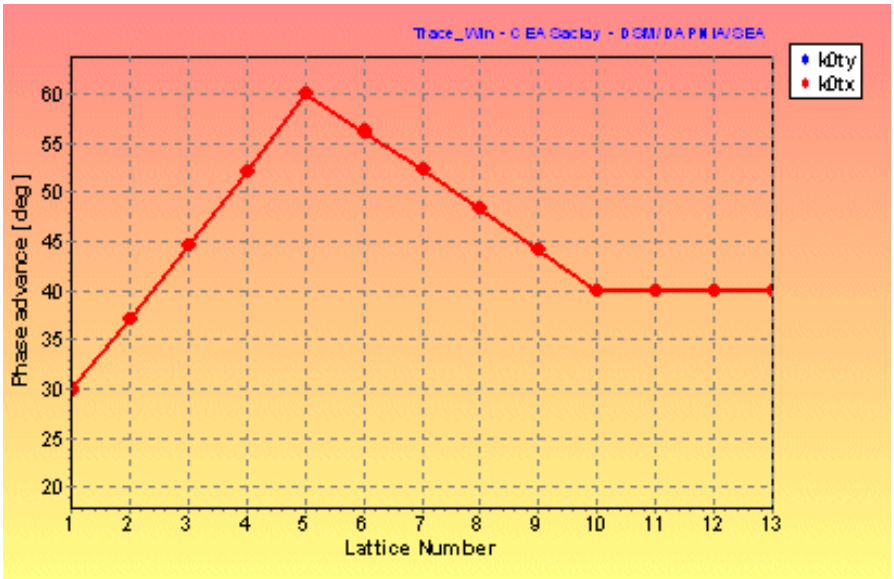
To set a transverse phase advance law

The periodic focusing lattices structure has to be defined by using the command LATTICE and LATTICE_END, in order to indicate the number of lattice and the number of element per lattice.
The zero-current transverse phase advance law can be imposed in two ways:
When the option "***Use phase advance definition***" of "***Main***" page is checked, TraceWin imposes the zero-current transverse phase advance law describe in the Sigma0 file. This file is editable in the "***Main***" page
If the option "***Use phase advance definition***" is unchecked that means you have to use the commands SET_ADV in you data file in order to describe your phase advance law.

Phase advance example

```
;FODO lattices in a DTL tank
;Tank 1
LATTICE 2 1
SET_ADV 30
DTL_CEL 87.5277 28 28 0.00548673 71.3977 -71.6524 74699.4 -50 10 0 0.102751 0.723252 -0.475485 -0.194532
DTL_CEL 87.9518 28 28 0.0080072 -71.6524 71.9632 76721.8 -49.5 10 0 0.103248 0.726628 -0.470582 -0.195991
DTL_CEL 88.3919 28 28 0.0112292 71.9632 -72.2903 79282.7 -49 10 0 0.103764 0.734598 -0.458632 -0.197949
DTL_CEL 88.8487 28 28 0.0144758 -72.2903 72.5931 81864.3 -48.5 10 0 0.104299 0.742159 -0.447189 -0.199371
DTL_CEL 89.3222 28 28 0.0177645 72.5931 -72.9101 84484.9 -48 10 0 0.104853 0.74948 -0.436027 -0.200384
DTL_CEL 89.8133 28 28 0.0212224 -72.9101 73.1977 87251.7 -47.5 10 0 0.105428 0.757488 -0.423767 -0.201319
DTL_CEL 90.3217 28 28 0.0245936 73.1977 -73.4992 89953.5 -47 10 0 0.106024 0.764331 -0.413161 -0.201532
DTL_CEL 90.8475 28 28 0.0280032 -73.4992 73.7703 92694.2 -46.5 10 0 0.10664 0.770931 -0.402823 -0.201301
DTL_CEL 91.3909 28 28 0.0314415 73.7703 -74.0554 95477.4 -46 10 0 0.107276 0.777317 -0.392802 -0.200986
SET_ADV 60
DTL_CEL 91.952 28 28 0.0349097 -74.0554 74.3058 98302.2 -45.5 10 0 0.107934 0.783481 -0.38306 -0.200411
DTL_CEL 92.5298 28 28 0.0379774 74.3058 -74.5691 100888 -45 10 0 0.108611 0.787243 -0.377176 -0.2003
DTL_CEL 93.124 28 28 0.0410704 -74.5691 74.8056 103520 -44.5 10 0 0.109308 0.790922 -0.371402 -0.200107
DTL_CEL 93.7349 28 28 0.0442596 74.8056 -75.056 106233 -44 10 0 0.110024 0.794763 -0.365289 -0.199539
DTL_CEL 94.3626 28 28 0.0473999 -75.056 75.2835 108961 -43.5 10 0 0.11076 0.798277 -0.35973 -0.199184
DTL_CEL 95.0078 28 28 0.0507833 75.2835 -75.5255 111880 -43 10 0 0.111516 0.802715 -0.352626 -0.198381
DTL_CEL 95.6703 28 28 0.0539705 -75.5255 75.7413 114710 -42.5 10 0 0.112293 0.80606 -0.347306 -0.197902
DTL_CEL 96.3499 28 28 0.0571744 75.7413 -75.9704 117592 -42 10 0 0.11309 0.809328 -0.342081 -0.197341
DTL_CEL 97.0469 28 28 0.0603973 -75.9704 76.1784 120527 -41.5 10 0 0.113907 0.812524 -0.336963 -0.196748
DTL_CEL 97.7612 28 28 0.0636044 76.1784 -76.3982 123492 -41 10 0 0.114744 0.815493 -0.332338 -0.196611
SET_ADV 40
DTL_CEL 98.4937 28 28 0.0670821 -76.3982 76.5998 126672 -40.5 10 0 0.115603 0.819426 -0.325941 -0.195469
DTL_CEL 99.244 28 28 0.0703571 76.5998 -76.8124 129777 -40 10 0 0.116483 0.822424 -0.321067 -0.19464
DTL_CEL 100.013 28 28 0.0738459 -76.8124 77.0048 133061 -39.5 10 0 0.117384 0.826103 -0.315053 -0.193443
DTL_CEL 100.8 28 28 0.0771374 77.0048 -77.209 136276 -39 10 0 0.118307 0.828909 -0.31049 -0.192635
DTL_CEL 101.604 28 28 0.0804404 -77.209 77.395 139551 -38.5 10 0 0.11925 0.831652 -0.306016 -0.191796
DTL_CEL 102.428 28 28 0.0839794 77.395 -77.5934 143023 -38 10 0 0.120216 0.835125 -0.30028 -0.190411
DTL_CEL 103.271 28 28 0.0875319 -77.5934 77.7738 146561 -37.5 10 0 0.121204 0.838521 -0.29462 -0.188871
DTL_CEL 104.132 28 28 0.0908581 77.7738 -77.9644 150025 -37 10 0 0.122213 0.841052 -0.290457 -0.187938
LATTICE_END
END
```

"*LATTICE 2 1*" defines the periodic focusing lattices, 2 is the number of element per lattice (2 DTL cell), 1 is the number of lattice per macro-lattice (generally 1). And TraceWin imposes a phase advance linear continuity between each "SET_ADV" command. This linear continuity can be per lattice or per meter according to the option "***Linear phase advance per meter***" of the "***Main***" page. This option is very useful, if you have lattice length discontinuity in order to keep continuity in the phase advance law per meter. TraceWin calculates the quadrupole gradients to obtain the asked zero-current phase advance law, using the lattice transfer matrix.
Warning: it is not always possible to find the required quadrupole gradients.



Copy following examples:
examples\DTL_1.ini
examples\DTL_1.dat
examples\DTL_1.cal

Matching commands

MATCH_FAM_GRAD f_n, n
MATCH_FAM_FIELD f_n, n
MATCH_FAM_PHASE f_n, n
MATCH_FAM_LFOC f_n, n
MATCH_FAM_LENGTH f_n, n

f_n is the family or section number and n is the matching element number. When one of these commands precedes an element, it is used (modified) to match the section f_n. For example, if two independent cavities are needed to match the beam at the entrance of a section two commands "*MATCH_FAM_FIELD*" or "*MATCH_FAM_PHASE*" or "*MATCH_FAM_LFOC*", with different matching element numbers n, have to be placed before these cavity entries. But if these two cavities must have the same field, they must have the same matching element number n. If you need for the matching five different quadrupoles and 2 different cavities, then the last command "*MATCH_FAM...*" must have n = 7. All combinations are possible. You can use the number of quadrupole, the number of cavity and the number of quadrupole or cavity coupled, you want. The coupled elements are not necessary consecutive. Two examples are presented below. These commands can also be used to match the beam at the entrance of the linac. The only constraint about the family numbers f_n is to have a different number per matching family, you can use for example 5,4,8,1...If you set $n=0$ in each comment of a matching, TraceWin sets automatically a different number for each command, but no coupling are possible between elements.

MATCH_FAM_GRAD: the quadrupole gradient is adjusted.
MATCH_FAM_FIELD: the field is adjusted.

MATCH_FAM_PHASE: the synchronous phase is adjusted.
MATCH_FAM_LFOC: the longitudinal focalization is adjusted by moving the field and the synchronous phase and the energy gain is kept.
MATCH_FAM_LENGTH: the element length is adjusted.

Matching Way

TraceWin is able to match the beam at the entrance of the linac or between the different sections. In these two cases the criterion for a good matching is either keep the longitudinal and transverse phase advances as smooth as possible or have at the input and the output lattice the same Twiss parameters (taking into account the acceleration parameters.
TraceWin is also able to impose Twiss parameters at a position in the linac. In this case you have to insert a SET_TWISS command behind the matching commands.
These three processes are named "*Optimization*", and can be stop by using the menu "*Stop*".

Optimization

To make the optimization of the beam at the entrance of the linac or between the different sections, TraceWin calculates the phase advance on *N* lattices, using the option "***Nbr of phase advance period to smooth***" of "***Matching***" page ". The choice of the criterion (Twiss parameter or smooth phase advances) depends of this option. If the number of lattice is too small (below 6) or if the number of lattice to optimize is set to 0, the Twiss criterion is used. You can watch the optimization process by plotting the phase advance chart, "***Beam***" and starting the "***Synch.***" option in the chart. The results of the optimizations can be found in the Results file. The optimization automatically stops when the criterion reaches "***Max. Number of iteration***" defined in the "***Matching***" page, but it can be stopped before in the menu "***Stop/Optimize***". During a matching some other criterions can be included in order to control the beam size, the beam separation, the beam transverse position or the beam emittance growth...(See matching commands) In this case all the different criterions are added.

Phase advance criterion: $vcr = \frac{1}{NM}\sum_{i=2}^{N-2}\sum_{j=1}^{M}(d^2\sigma_{i,j})$ With $d^2\sigma_{i,j} = \frac{\sigma_{i+1,j} + \sigma_{i-1,j} - 2\sigma_{i,j}}{\sigma_{i,j}}$

With *M=2* for DC beam (*x*,*y*) or *M=3* for bunched beam (*x*,*y*,*z*), and N is the number of lattice to optimize. $\sigma_{t,\varphi}$ is in °/m.

See also the phase advance definition and setting

Matching section or family

If the option "***Matching with family & Twiss commands***" of page "***Matching***" is checked, TraceWin changes some quads and cavities strength or element lengths pointed by "*MATCH_FAM...*" commands, in order to match the beam between two sections. The syntax is not always very easy to use because it has been defined in order to be able to represent most of the cases. Some examples are shown below to help you and if you meet some difficulties yet, send an Email with your data file attached to the authors.

Matching example 1
Matching example 2
Matching example 3
Matching example 4
Matching example 5

Copy following examples:
examples\Supra_1.ini
examples\Supra_1.dat
examples\Supra_1.cal

examples\DTL_2.ini
examples\DTL_2.dat
examples\DTL_2.cal

Matching example 3

Matching line calculation between a RFQ and a 352 MHz DTL. Two 704MHz buncher cavities and 4 quadrupoles are used in order to match the beam to the structure. A command SET_SIZE_MAX has been include in order to try to reduce the beam size. To help the matching optimization two drift lengths are adjusted by using the command "*MATCH_FAM_LENGTH*".

```
; *****   RFQ-DTL Matching line    ******
DRIFT 1e-05 10
FREQ 704
SET_SIZE_MAX 0.025 25 4.5 4.5 25 1
DRIFT 20 10
MATCH_FAM_GRAD 1 0
QUAD 56 55.9938 10
MATCH_FAM_LENGTH 1 0
DRIFT 58.85 10
MATCH_FAM_FIELD 1 0
GAP 251000 -90 10 0 0 0 0 0 0 0
DRIFT 58.85 10
MATCH_FAM_GRAD 1 0
QUAD 56 -56.1374 10
MATCH_FAM_LENGTH 1 0
DRIFT 50.0 10
MATCH_FAM_GRAD 1 0
QUAD 56 62.7152 10
DRIFT 58.85 10
MATCH_FAM_FIELD 1 0
GAP 282000 -90 10 0 0 0 0 0 0 0
DRIFT 58.85 10
MATCH_FAM_GRAD 1 0
QUAD 56 -56.5712 10
DRIFT 70 10
FREQ 352
QUAD 28 71.3977 10

; *****************   DTL    *****************
LATTICE 2 1
SET_ADV 46
DTL_CEL 87.5277 28 28 0.00548673 71.3977 -71.6524 74699.4 -50 10 0 0.102751 0.723252 -0.475485 -0.194532
DTL_CEL 87.9518 28 28 0.0080072 -71.6524 71.9632 76721.8 -49.5 10 0 0.103248 0.726628 -0.470582 -0.195991
DTL_CEL 88.3919 28 28 0.0112292 71.9632 -72.2903 79282.7 -49 10 0 0.103764 0.734598 -0.458632 -0.197949
DTL_CEL 88.8487 28 28 0.0144758 -72.2903 72.5931 81864.3 -48.5 10 0 0.104299 0.742159 -0.447189 -0.99371
DTL_CEL 89.3222 28 28 0.0177645 72.5931 -72.9101 84484.9 -48 10 0 0.104853 0.74948 -0.436027 -0.200384
DTL_CEL 89.8133 28 28 0.0212224 -72.9101 73.1977 87251.7 -47.5 10 0 0.105428 0.757488 -0.423767 -0.201319
DTL_CEL 90.3217 28 28 0.0245936 73.1977 -73.4992 89953.5 -47 10 0 0.106024 0.764331 -0.413161 -0.201532
…
..
.
```

The calculation result is the following:
The first line result contains the 4 quadrupole gradients in T/m, the second is the cavity field corrections and the last the drift lengths in mm.

```
Matching_Between_Section_0_to_1
56.3399 -56.1173 62.8806 -56.4843
0.934963 0.991466
0.00834559 0.000887792 0.0294733 0.0874924
```

Matching example 4

Matching line calculation between a two super conducting cavity families by adjusting the synchronous phases of the last cavities of the first family and the first cavities of the second family. Four quadrupoles are also adjusted in order to match the beam.

```
.....
DRIFT 325 100
QUAD 400 10 100
DRIFT 400 100
QUAD 400 -10 100
DRIFT 950 100
NCELLS 1 5 0.6579000 1.11349e+07 -12.331 100 0 0.1801715 0.2934260 -10.0848222 14.7245834 …
DRIFT 475 100
NCELLS 1 5 0.6579000 1.11616e+07 -10.584 100 0 0.1801715 0.2934260 -10.1057623 14.7737516 …
DRIFT 475 100
NCELLS 1 5 0.6579000 1.11873e+07 -8.890 100 0 0.1801715 0.2934260 -10.1259198 14.8211363 …
DRIFT 625 100
SET_ADV 70
DRIFT 325 100
MATCH_FAM_GRAD 8 1
QUAD 400 10 100
DRIFT 400 100
MATCH_FAM_GRAD 8 2
QUAD 400 -10 100
DRIFT 950 100
MATCH_FAM_PHASE 8 3
NCELLS 1 5 0.6579000 1.12121e+07 -7.248 100 0 0.1801715 0.2934260 -10.1453288 14.8668118 …
DRIFT 475 100
MATCH_FAM_PHASE 8 3
NCELLS 1 5 0.6579000 1.12361e+07 -5.657 100 0 0.1801715 0.2934260 -10.1640222 14.9108503 …
DRIFT 475 100
MATCH_FAM_PHASE 8 3
NCELLS 1 5 0.6579000 1.12593e+07 -4.114 100 0 0.1801715 0.2934260 -10.1820316 14.9533216 …
DRIFT 625 100
LATTICE_END
DRIFT 1187.35 100

; Second superconducting family
LATTICE 13 1
SET_ADV 85
DRIFT 325 100
MATCH_FAM_GRAD 8 4
QUAD 500 10 100
DRIFT 400 100
MATCH_FAM_GRAD 8 5
QUAD 500 -10 100
DRIFT 975 100
MATCH_FAM_PHASE 8 6
NCELLS 1 5 0.8458000 1.14453e+07 -94.435 100 0 0.1560511 0.2270898 -10.0611253 13.2916014 …
DRIFT 525 100
MATCH_FAM_PHASE 8 6
NCELLS 1 5 0.8458000 1.14816e+07 -92.773 100 0 0.1560511 0.2270898 -10.0883587 13.3415267 …
DRIFT 525 100
MATCH_FAM_PHASE 8 6
NCELLS 1 5 0.8458000 1.15177e+07 -91.116 100 0 0.1560511 0.2270898 -10.1153599 13.3910179 …
DRIFT 525 100
```

```
MATCH_FAM_PHASE 8 6
NCELLS 1 5 0.8458000 1.15536e+07 -89.464 100 0 0.1560511 0.2270898 -10.1421056 13.4400346 …
DRIFT 650 100
DRIFT 325 100
QUAD 500 10 100
DRIFT 400 100
QUAD 500 -10 100
DRIFT 975 100
…
```

Matching example 5

TraceWin sets Twiss parameter to the entrance of a RFQ. To impose Twiss parameter in your structure by adjusting elements you have to use the same "*MATCH_FAM…*" commands, but a SET_TWISS command has been put in front of the element where you want to impose Twiss parameters at its output. The first parameter of SET_TWISS and "*MATCH_FAM…*" commands is the family or the section number.

```
..
…
DRIFT 50.1 100
MATCH_FAM_FIELD 1 0
GAP 154335 -90 100 0 0
DRIFT 50.1 100
MATCH_FAM_LENGTH 1 0
DRIFT 60 100
MATCH_FAM_GRAD 1 0
QUAD 60 16.9089 100
MATCH_FAM_LENGTH 1 0
DRIFT 40 100
DRIFT 20 100
MATCH_FAM_GRAD 1 0
QUAD 60 -16.9089 100
DRIFT 50.1 100
MATCH_FAM_FIELD 1 0
GAP 154335 -90 100 0 0
DRIFT 50.1 100
DRIFT 20 100
MATCH_FAM_GRAD 1 0
QUAD 60 16.9089 100
MATCH_FAM_LENGTH 1 0
DRIFT 40 100
DRIFT 20 100
DIAPHRAGME 2.7 2.9 1
FREQ 352.21
DRIFT 0.00001 10
RFQ_CELL 120000 4.96362 0 1 109.954 -60 3
RFQ_CELL 120000 4.96362 0.114946 1.13305 27.4884 -60 4
SET_TWISS  1 -1.5440 0.2462 0.7924 0.1898 -0.0754 0.9523 0 0 0 0 0 0
RFQ_CELL 120000 4.96362 0.114946 1.13305 27.4884 -60 2
RFQ_CELL 120000 4.96362 0.115356 1.13354 27.5261 -60 -2
RFQ_CELL 120000 4.96362 0.115799 1.13406 27.567 -59.9802 2
RFQ_CELL 120000 4.96362 0.116304 1.13465 27.611 -59.9208 -2
RFQ_CELL 120000 4.96362 0.116874 1.13532 27.6552 -59.822 2
RFQ_CELL 120000 4.96362 0.117509 1.13609 27.6995 -59.6843 -2
RFQ_CELL 120000 4.96362 0.118213 1.13692 27.7441 -59.5081 2
RFQ_CELL 120000 4.96362 0.118987 1.13786 27.789 -59.2943 -2
RFQ_CELL 120000 4.96362 0.119833 1.13888 27.8343 -59.0435 2
RFQ_CELL 120000 4.96362 0.120755 1.13999 27.8799 -58.7569 –2
…
..
.
```

Matching example 1

4 quadrupoles and 2 cavities are used to match the beam between section 1 and 2. The longitudinal matching is done by a cavity field modification. This is the simplest case.

```
LATTICE 7 1                          «section 1»
DRIFT 100 100
MATCH_FAM_GRAD 1 1
QUAD 150 10 100                      «first quadrupole used for matching »
DRIFT 178.56 100
MATCH_FAM_GRAD 1 2
QUAD 150 -10 100                     «second quadrupole used for matching»
DRIFT 488.64 100
MATCH_FAM_FIELD 1 3
GAP_M 2.94577e+06 -30.5122 100       «first cavity used for matching, the field is adjusted»
DRIFT 722.08 100

LATTICE 9 1                          «section 2»
DRIFT 150 100
MATCH_FAM_GRAD 1 4
QUAD 350 10 100                      «third quadrupole used for matching»
DRIFT 100 100
MATCH_FAM_GRAD 1 5
QUAD 350 -10 100                     «fourth quadrupole used for matching»
DRIFT 935.126 100
MATCH_FAM_FIELD 1 6
GAP_M 3.04084e+06 -30.5122 100       «second cavity used for matching, the field is  adjusted»
DRIFT 1385.81 100
GAP_M 3.07847e+06 -30.5122 100
DRIFT 729.063 100
```

Matching example 2

4 quadrupoles and 4 cavities are used for the matching between section 1 and 2. The longitudinal matching is done by a cavity phase adjustment. Here, the cavities are coupled 2 by 2.

```
LATTICE 8 1                          «section 1»
…
DRIFT 100 100
MATCH_FAM_GRAD 1 1
QUAD 150 10 100                      «first quadrupole used for matching »
DRIFT 178.56 100
MATCH_FAM_GRAD 1 2
QUAD 150 -10 100                     «second quadrupole used for matching»
DRIFT 488.64 100
MATCH_FAM_PHASE 1 3
GAP_M 2.94577e+06 -30.5122 100       «first cavity used for matching, the phase is adjusted. n=3 ».
MATCH_FAM_PHASE 1 3
GAP_M 2.94577e+06 -30.5122 100       «second cavity used for matching. It will conserve the same phase as
the first cavity. n=3 »
DRIFT 722.08 100
LATTICE 10 1                         «section 2»
DRIFT 150 100
MATCH_FAM_GRAD 1 4
QUAD 350 10 100                      «third quadrupole used for matching»
DRIFT 100 100
MATCH_FAM_GRAD 1 5
QUAD 350 -10 100                     «fourth quadrupole used for matching»
DRIFT 935.126 100
MATCH_FAM_PHASE 1 6
GAP_M 3.04084e+06 -30.5122 100       «third cavity used for matching, the phase is adjusted. n=6 ».
DRIFT 1385.81 100
MATCH_FAM_PHASE 1 6
```

GAP_M 3.04084e+06 -30.5122 100 **«fourth cavity used for matching, It will conserve the same phase as the third cavity. *n*=6 »**
DRIFT 729.063 100

Set Twiss parameters

SET_TWISS f_n, α_x, β_x*(mm/mrad)*, α_y, β_y*(mm/mrad)*, α_z, β_z*(mm/mrad)*,k_{ax},k_{bx},k_{ay},k_{by},k_{az},k_{bz}

The forces of elements pointed with the matching commands "*MATCH_FAM_XXX*" are adjusted to impose the Twiss parameters given by the "*SET_TWISS*" command at the output element following the command. More than one command "*SET_TWISS*" can be used in the same optimization. In this case, use the same f_n parameter.

Twiss criterion:

$$vcr = \frac{1}{2M}\sum_{j=1}^{M}\left(\sqrt{\frac{R_j + \sqrt{R_j^2 - 4}}{2}}\right) - 1$$

With $R_j = \beta_j\gamma_{j0} + \beta_{j0}\gamma_j - 2\alpha_j\alpha_{j0}$

f_n is the section or family number.
β_j, γ_j, α_j being the beam Twiss parameters of the space phase *xx', yy', zz'*.
The 6 following parameters "*k*" are optional, set one to "*1*" allows to not taking account of the corresponding Twiss parameters
Write "SET_TWISS f_n" corresponds to write "SET_TWISS f_n *0 0 0 0 0 0 1 1 1 1 1 1*". That disables the SET_TWISS command and allows different other optimisations by inserting another commands like for example "SET_SIZE".

Copy following example:
examples\LBE_1.ini
examples\LBE_1.dat
examples\LBE_1.cal
examples\solenoid1.bsz
examples\solenoid1.ouv
examples\solenoid2.bsz
examples\solenoid2.ouv

Set beam centroid position

SET_POSITION *k(m^{-1}), x(mm), x'(mrad), y(mm), y'(mrad)*

x, x', y, y' are the centroid beam positions imposed at the point where this command appears, *k* is used in the criterion calculation.

Set position criterion:
$$vcr = \frac{k}{2}\cdot\sqrt{(x-x_0)^2 + (y-y_0)^2 + (x'-x'_0)^2 + (y'-y'_0)^2}$$

With *x(mm), x'(mrad), y(mm), y'(mrad)* being the beam centroid transverse positions at the place where the command "*SET_POSITION*" appears and x_o *(mm)*, x_o*'(mrad)*, y_o *(mm)*, y_o*'(mrad)* being the beam imposed positions.

Set achromatic line

SET_ACHROMAT *k f1 f2*

Located after a deviation for instance this command allows to make achromatic the preceding line. *k* is used to balance the criterion calculation
(*f1=1* means set achromatic position).
(*f2=1* means set achromatic angle).
(*f1=f2* means set achromatic position and angle).

Example:

```
DRIFT 300 100
START_ACHROMAT
EDGE 22.5 500 100 0.45 2.8 50 0
BEND 45 500 0 50 0
EDGE 22.5 500 100 0.45 2.8 50 0
DRIFT 200 100
MATCH_FAM_GRAD 4
QUAD 200 2
DRIFT 200 100
EDGE 22.5 500 100 0.45 2.8 50 0
BEND 45 500 0 50 0
EDGE 22.5 500 100 0.45 2.8 50 0
DRIFT 107 100
SET_ACHROMAT 1.0 1 1
DRIFT 100 100
SET_TWISS 4
DRIFT 100 100
END
```

The command SET_TWISS is always obligatory to allow the matching of the gradrupole. And the option "***Matching with family & Twiss commands***" of page "***Matching***" has to be checked

Copy following example:
examples\Achromat_2.ini
examples\Achromat_2.dat
examples\Achromat_2.cal

examples\Achromat_1.ini
examples\Achromat_1.dat

Set maximum beam max

SET_SIZE_MAX *k, N, x(mm), y(mm), φ(°)/z(mm), k2 (0/1)*

x, y, φ are the imposed beam size max in the N elements following this command, *N, k* and *k2* is used in the criterion calculation.

- Set size max criterion: $vcr = \frac{k}{M} \cdot \left(\exp\left(-\left(\frac{x_0}{x} \right)^4 \right) + \exp\left(-\left(\frac{y_0}{y} \right)^4 \right) + \exp\left(-\left(\frac{\varphi_0}{\varphi} \right)^4 \right) \right)$

With *x(mm), y(mm), φ(°)* being the beam maximum sizes in the *N* elements following the command "SET_SIZE_MAX", and x_0*(mm),* y_0*(mm),* φ_0*(°)* being the beam imposed sizes. If *φ* is lower than 0 the longitudinal size used is *z(mm).* If one of these last parameters is set to 0, no optimization is done on this size and *M* is reduced by one. If *k2=0*, the transverses sizes are is calculated without taking account the beam centroid position. If one of the parameters *x*, *y* or *φ* is equal to 0, no optimization is done on this size and *M* is reduced by one.
The sizes are the effective beam sizes (rms*sqrt(5) for bunched beam or rms*sqrt(4) for CW beam)

Set beam size

SET_SIZE *k, x(mm), y(mm), φ(°)/z(mm), k2 (0/1)*

x, y, φ are the imposed beam size in the output element following this command, *N*, *k* is used in the criterion calculation. If *k2=0*, the transverses sizes are calculated without taking account the beam centroid position.

Set size criterion: $vcr = \frac{k}{M} \cdot \sqrt{\left(\frac{x - x_0}{x_0} \right)^2 + \left(\frac{y - y_0}{y_0} \right)^2 + \left(\frac{\varphi - \varphi_0}{\varphi_0} \right)^2}$

With *x(mm), y(mm), φ(°)* being the beam sizes in the output elements following the command "SET_SIZE", and x_0*(mm),* y_0*(mm),),* φ_0*(°)* being the beam imposed sizes. If *φ* is lower than 0 the longitudinal size used is *z(mm).* If one of these last parameters is equal to 0, no optimization is done on this size and *M* is reduced by one.
The sizes are the effective beam sizes (rms*sqrt(5) for bunched beam or rms*sqrt(4) for CW beam)

Set beam separation

SET_SEPARATION *k,* S_x*,* S_y

S_x, S_y are the centroid beam positions divided by the imposed beam size max at the point where this command appears, *k* is used in the criterion calculation.

Set separation criterion: $vcr3 = k \cdot \sqrt{(Sx_0 - Sx)^2 + (Sy_0 - Sy)^2}$

Minimize emittance growth

MIN_EMIT_GROW *k, N, ex, ey, ez, f*

k and *N* are used in the criterion calculation.
ex, ey, ez and *f* are optional: if *ex* equal to 1 the criterion doesn't take into account ε_x beam emittance. That is the same way for *ey* and *ez*. f parameter define if the criterion calculation is relative or not (see formula below).

The emittance growth criterion:

- If (f=0) $vcr = k \cdot \sum_N \sqrt{\left(\frac{\varepsilon_x - \varepsilon_{x0}}{\varepsilon_{x0}}\right)^2 + \left(\frac{\varepsilon_y - \varepsilon_{y0}}{\varepsilon_{y0}}\right)^2 + \left(\frac{\varepsilon_z - \varepsilon_{z0}}{\varepsilon_{z0}}\right)^2}$
- If (f=1) $vcr = k \cdot \left(|\varepsilon_x| + |\varepsilon_y| + |\varepsilon_z|\right)$ at the command position

With ε_{xo}, ε_{yo}, ε_{zo} being the beam emittances where this command appears, and ε_x, ε_y, ε_z being the beam emittances after *N* elements.

Steerer

STEERER *Bx(T), By (T), Bmax, 0*

STEERER *Ex(V/m), Ey (V/m), Emax, 1*

Its a command not a element (Corresponding element: *THIN_STERRING*). The magnetic steerer is inserted in the element (magnetic or electric quadrupole, solenoid and field map) placed just after the instruction "Steerer" (keeping the same length).

$$x' = x' - \frac{q \cdot \Delta s \cdot B_y c}{mc^2 \beta\gamma} \quad \text{And} \quad y' = y' + \frac{q \cdot \Delta s \cdot B_x c}{mc^2 \beta\gamma}$$

$$x' = x' + \frac{\Delta s \cdot E_x}{E\rho} \quad \text{And} \quad y' = y' + \frac{\cdot \Delta s \cdot E_y}{E\rho}$$

Where *x'* and *y'* being respectively the horizontal and vertical beam centroid slope.
Used in diagnostic optimization (with *ADJUST_STERRER), Bmax* and *Emax* is the maximum limit for *Bx, By* and *Ex, Ey*, if it's greater than zero.

Chopper

CHOPPER *N, U(V), D(mm), C(mm), p(0/1)*

The chopper is inserted in the *N* elements placed just after the instruction "Chopper" (keeping the same length). *U* is the voltage between axis and plates and *C* is the chopper transverse position. ±D is the distance between axis and plates

If p=0 $x' = x' + \frac{q \cdot \Delta s \cdot U}{mc^2 \beta^2 \gamma \cdot D}$ and if p=1 $y' = y' + \frac{q \cdot \Delta s \cdot U}{mc^2 \beta^2 \gamma \cdot D}$

Where x' and y' are respectively the horizontal and vertical beam centroid slope.

See following example:
examples\Chopper_1.ini
examples\chopper_1.dat

Input beam errors

ERROR_BEAM_STAT *r(0/1), dx(mm), dy(mm), dφ(°), dxp(mrad), dyp(mrad), de(MeV),dEx(%), dEy(%), dEz(%), mx(%), my(%), mz(%), dib(mA), αxx'_min, αxx'_max, βxx'_min(mm/mrad), βxx'_max(mm/mrad), αyy'_min, αyy'_max, βyy'_min(mm/mrad), βyy'_max(mm/mrad), αzdp_min, αzdp_max, βzdp_min(mm/mrad), βzdp_max(mm/mrad)*

ERROR_BEAM_DYN

Five kind of error can be set:
Beam displacement: (*dx*, *dy*, *dρ*, *dxp*, *dyp*, *de*) The beam input position is not centered.
Emittance growth: (*dEx*, *dEy*, *dEz*) The input beam emittance is increased by a percentage.
Beam mismatch: (*mx*, *my*, *mz*) The input beam is mismatched by a percentage. A 20 % mismatch in x plane means α_x and β_x are multiplied by 1.2^2.
Beam Current error: (*dib*) Allows to study the effect of the input beam current variation.
Twiss parameter range: (*α_min*, *α_max*, *β_min*, *β_max*) The input beam is randomly select in the twiss parameter range. The 4 parameters, *α_min*, *α_max*, *β_min*, *β_max*, must be different of 0 and mishmash parameter *m* must be equal to 0.

This command concerns only the input beam. If *r* is equal to 0, the errors are constant and equal to each value of the command line. If *r* is equal to 1, the errors are distributed randomly; each value of the command line is then the maximum error. The errors are uniformly distributed between +/- the maximum error.

NCP CPL STAT DYN meaning

Quadrupole errors

ERROR_QUAD_NCPL_STAT *N r(0/1), dx(mm), dy(mm), φx(°), φy(°), φz(°), dg(%), dz(mm)*

ERROR_QUAD_NCPL_DYN

ERROR_QUAD_CPL_STAT

ERROR_QUAD_CPL_DYN

dx and dy being respectively the horizontal and vertical magnetic element displacement. *dz* being the longitudinal shift. φ_x ,φ_y, φ_z being respectively the quadrupole rotation around *x*, *y*, *z*-axis and *dg* being the gradient amplitude error. These errors are applied in the *N* elements following this command, excepted, if a new error command appears. If *r* is equal to 0, the errors are constant and equal to each value of the command line. If *r* is equal to 1, the errors are distributed randomly, each value of the command line is then the maximum error. The errors are uniformly distributed between +/- the maximum error.

This error set affects:
- Quadrupole
- Solenoid
- Quadrupole of DTL
- Field map if defined as static magnetic field

NCP CPL STAT DYN meaning

Copy following examples:
examples\Supra_4.ini
examples\Supra_4.dat
examples\Supra_4.cal

examples\Supra_5.ini
examples\Supra_5.dat
examples\Supra_5.cal

Cavity errors

ERROR_CAV_NCPL_STAT *N r(0/1/2/3), dx(mm), dy(mm), φ_x(°), φ_y(°), E(%), φ(°), dz(mm)*

ERROR_CAV_NCPL_DYN

ERROR_CAV_CPL_STAT

ERROR_CAV_CPL_DYN

dx and *dy* being respectively the horizontal and vertical electric element displacement. *dz* being the longitudinal shift, φ_x, φ_y being respectively the cavity rotation around x, y-axis. *E* being the field amplitude error. φ being the field phase error. These errors are applied in the *N* elements following this command, excepted, if a new error command appears.

- If *r* is equal to 0, the errors are constant and equal to each value of the command line.
- If *r* is equal to 1, the errors are distributed randomly, each value of the command line is then the maximum error. The errors are uniformly distributed between +/- the maximum error.

Specail feature for *r=2* or *r=3*
If *r* is equal to 2, the errors are constant and equal to each value of the command line.
If *r* is equal to 3, the errors are distributed randomly.
And the command syntax becomes :
ERROR_CAV_XXX_XXX *N r(0/1/2/3), dx(mm), dy(mm), φ_x(°), φ_y(°), E(%), kφ(deg/%), dz(mm)*
The phase error applied on the cavities is $\boldsymbol{\varphi_{err} = k\varphi * E_{rr}}$

In case of coupled error (_CPL_) the field phase and the field amplitude error sign **are reversed** each new cavity.

This error set affects:
- Bunched cavity
- Cavity multi-gap
- DTL (Warning the DTL are only concerns by the field errors)
- Sinus cavity
- Field map if not defined as static magnetic field

NCP CPL STAT DYN meaning

Bend errors

ERROR_BEND_NCPL_STAT *N, r(0/1), dx(mm), dy(mm), φ_x(°), φ_y(°), φ_z(°), dg(%), dz(mm)*

ERROR_BEND_NCPL_DYN

ERROR_BEND_CPL_STAT

ERROR_BEND_CPL_DYN

dx and dy being respectively the horizontal and vertical magnetic element displacement. *dz* being the longitudinal shift. φ_x ,φ_y, φ_z being respectively the bend rotation around *x*, *y*, *z*-axis and *dg* being the magnetic field amplitude error. These errors are applied in the *N* elements following this command, excepted, if a new error command appears. If *r* is equal to 0, the errors are constant and equal to each value of the command line. If *r* is equal to 1, the errors are distributed randomly, each value of the command line is then the maximum error. The errors are uniformly distributed between +/- the maximum error.

This error set affects only magnet bend and edge
Valid only if the BEND element is not cut in several parts.

Warning, the "ERROR_BEND_CPL_X" commands concern only the magnetic field amplitude errors, all other errors (shift, rotation,…) are not couple.

NCP CPL STAT DYN meaning

See detail about bend error treatment.

Adjust commands

ADJUST *N, v, n, min, max, first_step*

Red parameters are optional.
This command has to be associated with a Diagnostic elements number *N*. *v*, an integer, is the v-th variable to be adjust in the next element list of variable. As an exemple if v=2 and the next element is a quad then the gradient will be adjusted. n (if it's different from zero) allows to link ADJUST commands each other, therefore, two ADJUST commands with the same *n* will give the same value to the *v* variable point. *v* lower than zero allows to replace both command "ADJUST_DPHASE" and "ADJUST_DFIELD" when *v* points to phase or field variable.

New in version 2.2.0.16 : *first_step* parameter has been added in order to choose the first step of optimization.

New in version 2.2.1.9: The *n* parameter allow to link two variables: Now if you set *n* parameter to opposite sign you'll get opposite variation for the linked variables. For example two drifts coupled with (*n*=1 & *n*=-1) will keep the same total sum length.

5 consecutives ADJUST command can be set.
(See Adjust and diagnostic examples)

ADJUST_STEERER *N, max, first_step*

ADJUST_STEERER_BX *N, max, first_step*

ADJUST_STEERER_BY *N, max, first_step*

These commands have to be associated with Diagnostic elements number *N* and should be placed before a STEERER command. It allows to adjust the horizontal and/or vertical magnetic field steering.

Particular case of steerer adjustment: When the number of steerers corresponds to the number of BMP, no optimisation is performed and the resolution of the system is directly made by a matrix inversion

New in version 2.2.0.20: Input beam parameters can be adjusted in order to fit with a set of diagnostic values at different positions of a line
(Example adjusted beam emittances and Twiss parameters at the input of a simulated line in order to obtain the beam sizes measured at different positions with real beam)

ADJUST_BEAM_TWISS *N, AlpX_flag, betX_flag, AlpY_flag, BetY_flag, AlpZ_flag, betZ_flag*
If flag is set to 1 the selected Twiss parameter will be adjusted, if you want to have alpX=AlpY set alpX_flag=1 and alpY_flag=2, same way for Bet_flag)

ADJUST_BEAM_EMIT *N, Ex_flag, Ey_flag, Ez_flag*
If flag is set to 1 the selected emittance will be adjusted, if you want to have Ex=Ey set Ex_flag=1 and Ey_flag=2)

ADJUST_BEAM_CENTROID *N, X_flag, Xp_flag, Y_flag, Yp_flag, Z_flag, Zp_flag*
If flag is set to 1 the selected beam centroid parameter will be adjusted, if you want to have X=Y set X_flag=1 and Y_flag=2, same way for Xp_flag)

ADJUST_BEAM_CURRENT *N, I_flag*

All "*ADJUST_BEAM_XXX*" commands must be located in front of the first element. Some elements can be also adjusted in the same matching process.

Adjust and diagnostic examples1

Beam alignment from a RFQ to a DTL. Two ADJUST commands are associated with two STEERER and with two Diagnostic elements.

```
DRIFT 0.00001 6.5
DRIFT 56.0 6.5
ADJUST_STEERER 1                    «Adjust the steerer inside the following quadrupole »
STEERER -.01 0.01
QUAD_ERROR 1 0 0.1 0.1 0.3 0.3 0.5 0    «Include 0.1mm misalignment and 0.3° rotation errors in the quad »
QUAD 56.0 55.0 6.5
DRIFT 2.15 6.5
DRIFT 58.85 6.5
GAP_M 180000 -90 6.5
DRIFT 58.85 6.5
ADJUST_STEERER 1                    «Adjust the steerer inside the following quadrupole »
STEERER .01 0.01
QUAD 56.0 -55.0 6.5
DRIFT 82.9 6.5
DIAG_POSITION 1 0 0                 « Position monitor with x and y position imposed to 0 »
DRIFT 20 6.5
QUAD 56.0 55.0 6.5
DRIFT 58.85 6.5
GAP_M 210000 -90 6.5
DRIFT 38.85 6.5
QUAD 56.0 -55.0 6.5
DRIFT 54.3 6.5
DIAG_POSITION 1 0 0                 « Position monitor with x and y position imposed to 0 »
DRIFT 20.0 6.5
…
```

You can see the result in the result file. This result is not used in a statistical error study where an adjustment is calculated for each random error distribution.

```
Diagnostic_1
-0.0661744 -0.0663664 0.0554  0.0458
```

Adjust and diagnostic examples2

Triplet adjusting in order to match a beam size

```
ADJUST 1 2 1                  « Quadrupole 1 and 3 are linked, the point variable '2' is the gradient »
QUAD 152 2.14 52
DRIFT 119 52
ADJUST 1 1 0 200 300          « Here, the quadrupole length is adjusted, but the length will keep between (200 & 300) »
QUAD 281 -2.13 52
DRIFT 119 52
ADJUST 1 2 1
QUAD 152 2.0 52
DRIFT 3190 100
APERTURE 2 2 0
DIAG_SIZE 1 2 2
DRIFT 3590 100
DIAG_SIZE 1 4 4
DRIFT 100 100
…
```

Adjust and diagnostic examples3

Set achromatic line part from element 3 to achromatic diagnostic position

```
DRIFT 30 100
DRIFT 30 100
DRIFT 30 100
```

```
DRIFT 30 100
EDGE 22.5 500 100 0.45 2.8 50 0
BEND 45 500 0 50 0
EDGE 22.5 500 100 0.45 2.8 50 0
DRIFT 200 100
ADJUST 11 2 1
QUAD 100 -200 20
DRIFT 200 100
ADJUST 11 2 1
QUAD 100 200 20
DRIFT 200 100
ADJUST 11 2 1
QUAD 100 -200 20
DRIFT 200 100
EDGE 22.5 500 100 0.45 2.8 50 0
BEND 45 500 0 50 0
EDGE 22.5 500 100 0.45 2.8 50 0
DRIFT 107 100
DIAG_ACHROMAT 11 3 1 1
DRIFT 100 100
DRIFT 100 100
End
```

Adjust and diagnostic examples4

Find the inout beam Twiss parameters and emittances given the rms sizes measured at the end of the line.

```
ADJUST_BEAM_CURRENT 99 1
ADJUST_BEAM_EMIT 99 1 2 0  ; Ex=Ey
ADJUST_BEAM_TWISS 99 1 1 2 2 0 0  ; Rond beam
DRIFT 0 100
DRIFT 200 100
DRIFT 200 100
QUAD 100 -200 20
DRIFT 200 100
ADJUST 11 2 1
QUAD 100 200 20
DRIFT 200 100
DIAG_SIZE 99 2.5. 3.6
DRIFT 200 100
DIAG_SIZE 99 4.2. 1.6
DRIFT 0 100
END
```

RFQ errors

ERROR_RFQ_CEL_NCPL_STAT *N, r(0/1), dR(mm), d (mm), E(%),φ(°), TEpe(mm), TEpa(mm), DEpe(mm), DEpa(mm), DELong(mm) TSVerti(mm), TSHori(mm), DSVerti(mm), DSHori(mm), DSLong(m)*

ERROR_RFQ_CEL _NCPL_DYN *(not implemented yet)*

dR is the error for the longitudinal profile, d is the error for the transverse curvature of the electrode, E is the voltage amplitude error. φ being the field phase error (uneffective).

For the transverse plane, TEpe is the perpendicular tilt error by electrode, TEpa is the parallel tilt error by electrode, DEpe is the perpendicular displacement error by electrode, DEpa is the parallel displacement error by electrode, DELong is the a longitudinal displacement error by electrode.
For each segment (block of four electrodes defined with the RFQ_GAP command), TSVerti is the vertical tilt error, TSHori is the horizontal tilt error, DSVerti is the vertical displacement error, DSHori is the horizontal displacement error and DSLong is the longitudinal displacement error.
These errors are applied in the N elements following this command, excepted, if a new error command appears. If r is equal to 0, the errors are constant and equal to each value of the command line. If r is equal to 1, the errors are distributed randomly, each value of the command line is then the maximum error. RFQ is a special element in TraceWin and all these errors are no effect in a envelope calculation except the phase and filed errors.

NCP CPL STAT DYN meaning

Plot distribution

PLOT_DST

Indicate to the multiparticle code to store beam distribution (in the *.plt file) at the command location

Gas pressure

GAS *C, N, P*

Only used in multiparticle simulation, this command set the gas pressure parameter until a new command *GAS*
C is the cross section at 1 MeV (m²), *N* is the atomic number and *P* is the pressure (hPa).
You have to select '*gas stripping*' or '*gas scattering*" in the '*Multiparticle*' page.
The cross section values are used for gas scattering calculation.
Several "gas" commands can be set simultaneously (Max. 9).

Minimize enveloppe variation

MIN_ENV_VARIATION *k, N*

Minimize the variation of the maximum and minimum transverse beam envelope in the quadrupoles included in *N* elements following the command. For bunched beam, the phase-spread variation is also minimizing in the accelerating elements. *k* is the pondering factor of this criterion.

Minimize phase variation

MIN_PHASE_VARIATION *k, N, θ*

During a matching procedure, where some synchronous phases of cavities are adjusted in order to match a beam, it's often useful to limit the maximum phase amplitude variation.
k is used in the criterion calculation. θ is the max angle variation wanted during a matching, , N is the number of elements where this command is applied but only cavities are concerned.

NCPL CPL STAT DYN meaning

NCPL: means: No Coupled. The errors are individually applied on each element
CPL: means: Coupled. The errors are coupled on the N elements. In other words, a rotation error corresponds to an N elements block rotation.
STAT: means: Static. The effect of these errors can be detected and corrected with appropriate diagnostic and correctors. For example, beam position measurement coupled with steerers can compensate the quadrupole or cavities misalignments. Correction strategy should be known to be able to estimate their impact on beam dynamics
DYN: means: Dynamic. The effect of these errors cannot be measured and then corrected. Fortunately, they have usually lower amplitude than static errors. They are, for example, the vibrations of the elements or the RF field control errors (in phase or amplitude). The knowledge of the correction scheme is not needed to study their statistic impact. They are responsible of orbit oscillations around the corrected orbit (this notion of orbit is also extended in the longitudinal motion).

To read a particle file

The particle file from the page "***Main***" allows to define the input particle file, but if you want to force the envelope or multiparticle calculation to load a new particle file characteristics (Twiss parameters, current, emittances, centroid) at a given position in the linac, you have to include the command below followed by the full name of the particle file.

READ_DST *full_particle_file_name*

Warning: This command is read at its position and not like other commands at the end of the following element.

Set RFQ vane geometry

RFQ_GEOM *type N dz*

Put this command just before the first RFQ cell.
According to *type* parameter:

If *type* = 0: Toutatis generates vane geometry file with 50steps/cell. It's the default case.

If *type* = 1: Toutatis reads vane geometry file
Example: RFQ_GEOM 1 c:\my_project\rfq\My_rfq.vane

If *type* = 2: and *N* = 0: Toutatis generates vane geometry file with a cell step = *dz*
Example: **RFQ_GEOM 2 0 0.001** for a 1 mm step

If *type* = 2: and $N \neq 0$: Toutatis generates vane geometry file with *N step*/cell
Example: RFQ_GEOM 2 20

For statistical error study case: (for X cases)
Example: RFQ_GEOM 1 c:\my_project\rfq\My_rfq.vane

Toutatis, for each case, will kook for a RFQ geometry vane file called:
c:\my_project\rfq\My_rfq000001.vane for first run
c:\my_project\rfq\My_rfq000002.vane for second one
…
..
.
c:\my_project\rfq\My_rfq00000X.vane for last one

Set RFQ electrode type

FOUR_RODS

This option generates electrode profile with varying transverse curvature. By default the transverse curvature is proportional to *Ro* (four vane type). Put this command just before the first RFQ cell element.

TWOTERMS

This command allows to generate a longitudinal profile (way to modulate) which is governed by the classical 2 terms potential. In case this option is used, only *m* and *a* are taken into account, *Ro* is recomputed. The sinus modulation is the default in case nothing is specified. Put this command just before the first RFQ cell.

Set RFQ coupling gap

RFQ_GAP *Lp(m), Lg(m), Sl(m), St(m)*

This command defines in a RFQ structure a resonant coupling gap. The position of the command in the data file doesn't matter, if you respect 2 rules: place it before a RFQ cell element, and if you need more than one coupling gap you must avoid to put the new command before the same RFQ cell element. *Lp* is the longitudinal position of the center of the gap. *Lg* is gap width. *Sl* is the half-ellipse size in the beam direction and *St* is the half-ellipse height in the perpendicular beam direction.

To read a multiparticle output file

If you choose to run multiparticle code (Partran or Toutatis), you can avoid to run a linac part or even all the linac which has been already computed. Then, you have to indicate to TraceWin, which part doesn't have to be run, by insert at the beginning of this part the command "*READ_OUT"* followed by the full name of the Partran or Toutatis output file

READ_OUT *full_output_multiparticle_file_name*

Warning: This command is read at its position and not like other commands at the end of the following element.

Shift

SHIFT d_x*(mm),* d_y *(mm)*

Move the following element

PARTRAN step calculation

PARTRAN_STEP *step1, step2*

Step1 is the new step of calculation per metre; *step2* is the new step of space-charge calculation per meter until reaching a new "*PARTRAN_STEP*" command. The default *step1* and *step2* value is put in the "***Multiparticle***" page, see "***Partran step of calcul***".
These two step concern only DRIFT and FIELD_MAP elements, all other elements are treated in 2 steps.

Magnetic or electric static field

FIELD B_x*(T),* B_y *(T),* B_z *(T)* E_x *(V/m)* E_y *(V/m)* E_z *(V/m)*

Add to the following elements a magnetic or electric force until a new command "*FIELD*" canceling the preceding. In PARTRAN simulation, only the field command of the first element is considering.

$$x' = x' - \frac{q \cdot \Delta s \cdot B_y c}{mc^2 \beta\gamma} \qquad y' = y' + \frac{q \cdot \Delta s \cdot B_x c}{mc^2 \beta\gamma}$$

$$x' = x' + \frac{q \cdot \Delta s \cdot E_x}{mc^2 \beta^2 \gamma} \qquad y' = y' + \frac{q \cdot \Delta s \cdot E_y}{mc^2 \beta^2 \gamma} \qquad \gamma = \gamma + \frac{qE_z}{mc^2}$$

Where *x'* , *y'* and g being respectively the horizontal and vertical beam centroid slope and the the reduced energy

Set beam phase advance

SET_BEAM_PHASE_ADV *k, N,* φ_x*(°),* φ_y*(°),* φ_z*(°)*

φ_x, φ_y, φ_z, are the imposed beam phase advance in space charge during *N* elements from command. *k* is used in the criterion calculation.

Set beam adv criterion: $vcr = \frac{k}{M} \cdot \sqrt{\left(\frac{\varphi_x - \varphi_{x0}}{\varphi_{x0}}\right)^2 + \left(\frac{\varphi_y - \varphi_{y0}}{\varphi_{y0}}\right)^2 + \left(\frac{\varphi_z - \varphi_{z0}}{\varphi_{z0}}\right)^2}$

With $\varphi_x(°)$, $\varphi_y(°)$, $\varphi_z(°)$ being the beam phase advances in the output elements following the command "SET_BEAM_PHASE_ADV", and $\varphi_{x0}(°)$, $\varphi_{y0}(°)$, $\varphi_{z0}(°)$, being the imposed beam phase advances. If one of these last parameters is equal to 0, no optimization is done on this phase and *M* is reduced by one.

Set beam energy and phase

SET_BEAM_E0_P0 *k, ΔE, Δφ, ke, kp*

ΔE and *Δφ,* are the imposed delta beam energy and phase.

Set beam energy and phase criterion: $vcr = k \cdot \sqrt{k_e \left(\Delta\varphi - \Delta\varphi_0\right)^2 + k_p \left(\Delta E - \Delta E_0\right)^2}$

With ΔE_0 being the difference between the beam energy and the linac energy (close to 0 in the ideal case without error). And $\Delta\varphi_0$ being the difference between the beam phase and the linac phase (close to 0 in the ideal case without error)
ke, kp allow to take into account or no one of both.

Set beam energy

SET_BEAM_ENERGY *k*, E_i

E_i is the imposed beam energy (MeV).

Set beam energy criterion: $vcr = k \cdot (E - E_i)^2$

With *E* being the the beam energy.

Set synchronous phase

SET_SYNC_PHASE

Allows fixing a synchronous phase into an accelerating element where the phase parameter is defined as the input phase or the first gap phase. Concerns the following element: (So, replace the input phase value by the wanted synchronous phase value in the element parameter).

FIELD_MAP
RFQ_CEL
CAVSIN
NCELLS

Minimize field variation

MIN_FIELD_VARIATION *k, N, ke*

During a matching procedure, where some electric fields of cavities are adjusted in order to match a beam, it's often useful to limit the maximum field amplitude variation.
k is used in the criterion calculation. *ke* is the max field variation (initial field amplitude divided by new field amplitude) wanted during a matching, *N* is the number of elements where this command is applied but only cavities are concerned.

Duplicate elements

REPEAT_ELE *k, n*

Allows to duplicate the following *n* elements *k* times. Caution: The associated commands are also repeated except: '*LATTICE*', '*LATTICE_END*' and '*SET_ADV*' one. A '*REPEAT_ELE'* command cannot include another one.

Example: 100 times a period.

```
DRIFT 5 30
DRIFT 5 30
REPEAT_ELE 100 4
LATTICE 4 1
SET_ADV 20
QUAD 100 1530 0
DRIFT 100 30
QUAD 100 -15 30 0
DRIFT 100 30
LATTICE_END
DRIFT 5 30
DRIFT 5 30
END
```

Example : Repeat 10 times a quadrupole

```
DRIFT 5 30
REPEAT_ELE 10 1
QUAD 10 15 30 0
DRIFT 5 30
END
```

Example : Divide in 10 steps a dipole with Edge and fringe-field.

```
EDGE 20 600 40 0 0 25 0
BEND -90 600 0 20 0
EDGE 20 600 40 0 0 25 0
```

Become

```
EDGE 20 600 40 0 0 25 0
BEND -9 600 0 20 0
EDGE 0 600 40 1e-12 1e-12 25 0
REPEAT_ELE 8 3
EDGE 0 600 40 1e-12 1e-12 25 0
BEND -9 600 0 20 0
EDGE 0 600 40 1e-12 1e-12 25 0
EDGE 0 600 40 1e-12 1e-12 25 0
BEND -9 600 0 20 0
EDGE 20 600 40 0 0 25 0
```

Superpose field maps

SUPERPOSE_MAP *shift (mm)*

A FIELD_MAP element already allows to superpose 4 field map types (electrostatic, electrodynamic, magnetic or electric field map). With the "SUPERPOSE_MAP command, different FIELD_MAP elements can be superposed and shifted. There are no limitations excepted than only one RF field map is allowed among all field maps.

If you want to use field map aperture or current in case of superposed field map, you have to include a empty field map element a the position 0 including aperture and/or current field map.

Example: quadrupole inside a solenoid.

```
DRIFT 5 30
SUPERPOSE_MAP 400
FIELD_MAP  70 100 0 42 -0.3 0 0 0 qpole_field_map_file
SUPERPOSE_MAP 0
FIELD_MAP  70 1000 0 100 -1.3 0 0 0 solenoid_fiedl_map_file
DRIFT 5 30
End
```

Set beam phase error

SET_BEAM_PHASE_ERROR *Dp(°) RamdomFlag(0/1)*

If *Dp* is equal to zero, this command allows to cancel, phase error coming from preceding elements. The beam phase is not affected by this command, the strategy is to shift all the following cavity RF phases, by the difference between reference design and beam phase observe at the command position. By this way, you have the possibility to uncouple the RF phase of some part of the machine and start a linac part with a new RF phase.
This command cannot correct to phase errors coming from dynamic errors, only static or input beam error can be corrected.
If *Dp* not equal to zero and *RamdomFlag* is equel to zero, the RF shift will be increased of *Dp* value.
If *Dp* not equal to zero and *RamdomFlag* is not equal to zero, the RF shift will be increased of a ramdom value between +/-*Dp*.

Develop its own elements or diagnostics

This feature allows to each user to develop its own elements, diagnostics or commands. A detailed example following explains how to peform it. Use the following '*main.cpp*' file and compile it as a dynamic library (read following explanations). Its feature is experimental and could evolve according to user remarks.

```
/*************************************************************************
main.cpp - description
-------------------
begin : Wed Dec 1 2010
copyright : (C) 2010 by URIOT Didier
email : rdidier@cea.fr
*************************************************************************/

#ifdef _WIN32
#include <windows.h>
  #define DLL_EXPORT __declspec(dllexport)
#else
  #define DLL_EXPORT
#endif
#include <cmath>
#include <cstring>

#ifdef __cplusplus
  extern "C" {
#endif

//------------------------------------------------------------------
//------------------------------------------------------------------
//------------------------------------------------------------------
//-- MY_ELE - MY_ELE - MY_ELE - MY_ELE - MY_ELE - MY_ELE - MY_ELE
//-- MY_ELE - MY_ELE - MY_ELE - MY_ELE - MY_ELE - MY_ELE - MY_ELE
//-- MY_ELE - MY_ELE - MY_ELE - MY_ELE - MY_ELE - MY_ELE - MY_ELE
//------------------------------------------------------------------
//------------------------------------------------------------------
//------------------------------------------------------------------
// MY_ELE
// MY_ELE
// MY_ELE
//
// Syntax in struture.dat file
//
// MY_ELE L(mm),R(mm),Nstep,arg3,arg4...arg20
//
// Settings (L, R, Nstep) must be defined
// - Element length (mm)
// - Aperture R (mm): This parameter are only used for plotting, you have to manage
yourself the
particle losses.
// - Nstep : the element will cut in Nstep part with a space-charge kick each 2
steps
// If Nstep = 0 the number of space-charge kick will be defined by TraceWin
according to
"step of space-charge" parameter of "Multiparticle" page.
// If Nstep = 1 the element will not be cut in and no space-charge kick will be
applied
// - arg3 : could be used to make difference between several elements
//
// below you can find an example of a Drift :
// Below you can find an example (Drift element)
// 2 functions are defined
// - my_ele_partran : for the beam tracking treatment
// - my_ele_envelope : for the beam envelope treatment
//
// You have to compile this example as an dll
// - my_element.dll (for windows)
// - my_element.so (for linux)
// - my_element.dylib (for MacOS)
// Copy it in TraceWin executable or *.dat directory
//
// Commands to compil and link GNU gcc compiler:
```

```
// g++.exe -Wall -c main.cpp -o main.o
// g++.exe -shared -Wl,--dll main.o -o my_elements.dll
//
// MY_ELE Multiparticle (Drift example) [ MY_ELE L(mm), R(mm), Nstep ]
// MY_ELE Multiparticle (Drift example) [ MY_ELE L(mm), R(mm), Nstep ]
// MY_ELE Multiparticle (Drift example) [ MY_ELE L(mm), R(mm), Nstep ]
// MY_ELE Multiparticle (Drift example) [ MY_ELE L(mm), R(mm), Nstep ]
int DLL_EXPORT my_ele_partran(double Zs,double *param,int Nele,int npart,double
*cord,double *loss,double freq,double mass0,int q,double *ws,double *Ibeam,double
*extra_param,char *error_mess)
{
//
// Zs : Current position in the element (from 0 to Length-Length/Nstep)
// param[1] : Length/Nstep (m)
// param[2] : Aperture (m)
// param[3] : Nstep
// cord : See example
// loss : see example
// freq : beam frequency (Hz)
// mass : Particle mass (eV)
// q : particle charge
// *ws : reference kinetic energy (eV) (can be modified)
// *Ibeam : Beam current (A) (can be modified)
// error_mess : TraceWin stop and show this error message if this function return 0
// if error_mess!="" and function return 1, this message is print to the standart
console without stop TraceWin
  double x,y,xp,yp,z,w,dpsp,r,zzs,bgs,gamma,gams,betas,ds,Aperture;

  ds=param[1];
  Aperture=param[2];
  strcpy(error_mess,"");
  zzs=(*ws)/mass0;
  bgs=sqrt(zzs*(2.+zzs));
  gams=1.0+zzs;
  betas=bgs/gams;
  for (int i=0;i<npart;i++) {
    if ((int)loss[i]==0) {
      x=cord[i*6]; // m
      xp=cord[i*6+1]; // rad
      y=cord[i*6+2]; // m
      yp=cord[i*6+3]; // rad
      z=cord[i*6+4]; // m
      dpsp=cord[i*6+5]; // dp/p
      r=sqrt(x*x+y*y);
      if (r>Aperture) loss[i]=Nele; // A particle lost has to be set to Element
number
      else {
        x=x+ds*xp;
        y=y+ds*yp;
        w=dpsp*betas*betas*gams*mass0+(*ws);
        gamma = 1+w/mass0;
        z=z+dpsp*ds/(gamma*gamma);
      }
      cord[i*6]=x;
      yp=cord[i*6+2]=y;
      cord[i*6+4]=z;
    }
  }
  return(1);
}

//---------------------------------------------------------------------
//---------------------------------------------------------------------
//---------------------------------------------------------------------
// MY_ELE Envelope (Drift example)
// MY_ELE Envelope (Drift example)
// MY_ELE Envelope (Drift example)
// MY_ELE Envelope (Drift example)
int DLL_EXPORT my_ele_envelope(double Zs,double *param,int Nele,double
Tmat[][6],double *Bcent,double freq,double mass0,int q,double *ws,double
*ibeam,double *extra_param,char *error_mess)
{
//
// Zs : Current position in the element (from 0 to Length-Length/Istep)
// my_ele_envelope is first time called wiht parma[1]=Length (in this case Zs=-10)
```

```
// the following calls is performed Istep times for Zs from 0 to Length-Length/Istep
(Istep is defined by TraceWin according to "step of calcul" parameter of "Main"
page.
// param[1] : Length(m) or Length/Istep (m)
// param[2] : Aperture (m)
// param[3] : Nstep
// cord : See example
// loss : see example
// freq : beam frequency (Hz)
// mass : Particle mass (eV)
// q : particle charge
// *ws : reference kinetic energy (eV) (can be modified)
// *Ibeam : Beam current (A) (can be modified)
// error_mess : TraceWin stop and show this error message if this function return 0
// if error_mess!="" and function return 1, this message is print to the standart
console without stop TraceWin

  double gams;
  double ds=param[1];

  strcpy(error_mess,"");
  gams = 1+(*ws)/(mass0);
  Tmat[0][0]=Tmat[1][1]=1;
  Tmat[0][1]=ds;
  Tmat[2][2]=Tmat[3][3]=1;
  Tmat[2][3]=ds;
  Tmat[4][4]=Tmat[5][5]=1;
  Tmat[4][5]=ds/(gams*gams);
  return(1);
}

//-----------------------------------------------------------------
//-----------------------------------------------------------------
//-----------------------------------------------------------------
//-----------------------------------------------------------------
//-- MY_DIAG - MY_DIAG - MY_DIAG - MY_DIAG - MY_DIAG - MY_DIAG - MY_DIAG
//-- MY_DIAG - MY_DIAG - MY_DIAG - MY_DIAG - MY_DIAG - MY_DIAG - MY_DIAG
//-- MY_DIAG - MY_DIAG - MY_DIAG - MY_DIAG - MY_DIAG - MY_DIAG - MY_DIAG
//-----------------------------------------------------------------
//-----------------------------------------------------------------
-----------------------------------------------------------------
// MY_DIAG
// MY_DIAG
// MY_DIAG
//
// Syntax in struture.dat file
//
// MY_DIAG(weight) target arg2,arg3...arg10
//
// TraceWin Criteria = pow(target_value - diag_value),2)
// below you can find an example of a beam position or beam size measurment :
// 2 functions are defined
// - my_daig_partran : for the beam tracking treatment
// - my_diag_envelope : for the beam envelope treatment
// MY_DIAG Multiparticle
// MY_DIAG Multiparticle
// MY_DIAG Multiparticle
// MY_DIAG Multiparticle
int DLL_EXPORT my_diag_partran(double *diag_value,double *param,int Nele,int
npart,double *cord,double *loss,double freq,double mass0,int q,double ws,double
Ibeam,double *extra_param,char *error_mess)
{
//
// param[1] : target_value
// param[2] : you can use this second parameter to define to diagnostic types
// param[3->8] : free
// diag_value : value of your diagnostic
// cord : See example
// loss : see example
// freq : beam frequency (Hz)
// mass : Particle mass (eV)
// q : particle charge
// ws : reference kinetic energy (eV) (can be modified)
// Ibeam : Beam current (A) (can be modified)
// error_mess : TraceWin stop and show this error message if this function return 0
// if error_mess!="" and function return 1, this message is print to the standart
console without stop TraceWin
```

```
  double x2,x,y,xp,yp,z,dpsp,xmoy;
  int ng;

  strcpy(error_mess,"");
  xmoy=x2=0;
  ng=0;
  for (int i=0;i<npart;i++) {
    if ((int)loss[i]==0) {
      x=cord[i*6]; // m
      xp=cord[i*6+1]; // rad
      y=cord[i*6+2]; // m
      yp=cord[i*6+3]; // rad
      z=cord[i*6+4]; // m
      dpsp=cord[i*6+5]; // dp/p
      x2+=x*x;
      xmoy+=x;
      ng++;
    }
  }
  if (param[2]==0) {
    xmoy/=ng;
   *diag_value=xmoy; // X position (m)
  }
  if (param[2]==1) {
    x2=sqrt(x2/ng);
   *diag_value=x2; // X size (m)
  }
  return(1);
}
//---------------------------------------------------------------------
//---------------------------------------------------------------------
// MY_DIAG Envelope
// MY_DIAG Envelope
// MY_DIAG Envelope
// MY_DIAG Envelope
int DLL_EXPORT my_diag_envelope(double *diag_value,double *param,int Nele,double
Bmat[][6],double *Bcent,double freq,double mass0,int q,double ws,double ibeam,double
*extra_param,char *error_mess)
{
//
// param[1] : target_value
// param[2] : you can use this second parameter to define to diagnostic types
// param[3->8] : free
// diag_value : value of your diagnostic
// Nele : Element number
// Bmat : Beam matrix 6x6 (x(m), xp(rad, y(m), yp(rad), z(m), dp/p)
// Bcent : beam centroid vector (x(m), xp(rad, y(m), yp(rad), z(m), dp/p)
// freq : beam frequency (Hz)
// mass : Particle mass (eV)
// q : particle charge
// ws : reference kinetic energy (eV) (can be modified)
// Ibeam : Beam current (A) (can be modified)
// error_mess : TraceWin stop and show this error message if this function return 0
// if error_mess!="" and function return 1, this message is print to the standart
console without stop TraceWin

  strcpy(error_mess,"");
  if (param[2]==0) {
    *diag_value=Bcent[1]; // X beam position (m)
  }
  if (param[2]==1) {
    *diag_value=sqrt(Bmat[0][0]); // beam size (X); (m)
  }
  return(1);
}


#ifdef __cplusplus
} // "C"
#endif
```

Transfer matrices

Funneling gap (*EoTL*, *φs*)

EoTL (V) is the maximum energy gain, *φs* (°) is the synchronous phase.

$$\beta_z = \sqrt{\frac{1-\frac{1}{\gamma_i^2}}{1+x'^2+y'^2}}$$

$$K = |q|E_0TL \cdot \cos\varphi$$

$$x' = x' + \frac{K}{\beta_z^2\sqrt{(\gamma_i \cdot mc^2)^2 + \frac{K^2}{\beta_z^2}}}$$

Where *x'* being the horizontal beam centroid slope.

$$R_{xx} = R_{yy} = R_{zz} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \qquad R_{xz} = \begin{bmatrix} 0 & 0 \\ \frac{2\pi|q|E_0TL \cdot \sin\varphi_s}{\gamma_i\beta_z^3 \cdot \lambda \cdot mc^2} & 0 \end{bmatrix}$$

Drift (*Δs*)

Δs (mm) is the drift length:
The non null 2×2 transfer sub-matrixes are:

$$R_{xx} = R_{yy} = \begin{bmatrix} 1 & \Delta s \\ 0 & 1 \end{bmatrix}, \text{ and } \quad R_{zz} = \begin{bmatrix} 1 & \frac{\Delta s}{\gamma^2} \\ 0 & 1 \end{bmatrix}.$$

Thin lens (f_x, f_y)

f_x, f_y are focal length in meter
The non null 2×2 transfer sub-matrixes are:

$$R_{xx} = \begin{bmatrix} 1 & 0 \\ \frac{-1}{f_x} & 1 \end{bmatrix}, \quad R_{yy} = \begin{bmatrix} 1 & 0 \\ \frac{-1}{f_y} & 1 \end{bmatrix}, \quad \text{and} \quad R_{zz} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

Beam rotation (*θxy, θxz, θzy, dx, dy, dxp, dyp*)

θ_{xy} are the beam rotation XY angle in degre
The ellipsoid can be brought upright by rotations $-\theta_{xy}$ accomplished by applying the transfer matrixes:

$$R_{xy} = \left[\begin{array}{cc|cc|cc} \cos(\theta_{xy}) & 0 & \sin(\theta_{xy}) & 0 & 0 & 0 \\ 0 & \cos(\theta_{xy}) & 0 & \sin(\theta_{xy}) & 0 & 0 \\ \hline -\sin(\theta_{xy}) & 0 & \cos(\theta_{xy}) & 0 & 0 & 0 \\ 0 & -\sin(\theta_{xy}) & 0 & \cos(\theta_{xy}) & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{array}\right],$$

The rotation is then applied:

$$[\sigma] = R_{xy} \cdot [\sigma] \cdot R_{xy}^{T},$$

Quadrupole (*Δs, G*)

Δs (mm) is the quadrupole length, G (T/m) its gradient.

For electrostatic quadrupole $G = \frac{V_0}{\beta_c R^2}$, where *Vo* is the voltage between electrode, *βc* beam speed and *R* is the half distance between electrode.

Lets use $k = \sqrt{\left|\frac{G}{B\rho}\right|}$, with $B\rho = \frac{m_0 c \beta \gamma}{q}$, the magnetic rigidity of the particle.

If q·G is positive, the quadrupole is focusing in the horizontal direction, else it's defocusing.
The non null 2×2 transfer sub-matrixes are:

In the longitudinal direction, we have: $R_{zz} = \begin{bmatrix} 1 & \frac{\Delta s}{\gamma^2} \\ 0 & 1 \end{bmatrix}$.

In the transverse direction, two possibilities:

Focusing quadrupole (in horizontal direction, *q·G* ≠ 0)

$$R_{xx} = \begin{bmatrix} \cos(k\Delta s) & \frac{\sin(k\Delta s)}{k} \\ -k\sin(k\Delta s) & \cos(k\Delta s) \end{bmatrix}, \quad R_{yy} = \begin{bmatrix} \cosh(k\Delta s) & \frac{\sinh(k\Delta s)}{k} \\ k\sinh(k\Delta s) & \cosh(k\Delta s) \end{bmatrix}.$$

Defocusing quadrupole (in horizontal direction, *q·G* ≠ 0)

$$R_{xx} = \begin{bmatrix} \cosh(k\Delta s) & \dfrac{\sinh(k\Delta s)}{k} \\ k\sinh(k\Delta s) & \cosh(k\Delta s) \end{bmatrix}, \qquad R_{yy} = \begin{bmatrix} \cos(k\Delta s) & \dfrac{\sin(k\Delta s)}{k} \\ -k\sin(k\Delta s) & \cos(k\Delta s) \end{bmatrix}.$$

Bunched cavity or thin gap (*EoTL, φs, p, βs, Ts, kT's,k²T''s,kS', k²S''*)

EoTL (eV) is the maximum energy gain, *φs* (°) is the synchronous phase.
The reduced energy change in the gap is:

$$\gamma_s = \gamma_e + \frac{|q|E_0 TL\cos(\varphi_s)}{mc^2}.$$

The phase shift in the gap is:

If $\beta s = 0$: $\Delta\varphi = 0$ else $\Delta\phi = \dfrac{qE_0 TL\cdot\sin(\varphi_s)}{mc^2\cdot\overline{\gamma^2}\cdot\overline{\beta}}\left(\dfrac{kT'}{T}\right)$

The changes in the normalized momentum caused by the gap are given by:

$$k_{xy} = \frac{-q\pi E_0 TL\sin(\varphi_s)}{mc^2\overline{\beta^2}\cdot\overline{\gamma^2}\,\lambda}, \qquad \text{With } \bar{\gamma} = \frac{\gamma_e+\gamma_s}{2} \text{ and } \bar{\beta} = \frac{\beta_e+\beta_s}{2}.$$

$$k_z = \frac{2q\pi E_0 TL\sin(\varphi_s)}{mc^2\overline{\beta^2}\,\lambda}.$$

The non null 2×2 transfer sub-matrixes are:

$$R_{xx} = \begin{bmatrix} k_x.C & 0 \\ \dfrac{k_{xy}}{(\beta\gamma)_s} & \dfrac{(\beta\gamma)_e}{(\beta\gamma)_s} \end{bmatrix}, \quad R_{yy} = \begin{bmatrix} k_y C & 0 \\ \dfrac{k_{xy}}{(\beta\gamma)_s} & \dfrac{(\beta\gamma)_e}{(\beta\gamma)_s} \end{bmatrix}, \quad R_{zz} = \begin{bmatrix} 1 & 0 \\ \dfrac{k_z}{(\beta\gamma)_s} & \dfrac{(\beta\gamma)_e}{(\beta\gamma)_s} \end{bmatrix}.$$

- If $\beta s = 0$: $k_x = k_y = 1 - \dfrac{qE_0 TL\cos(\varphi_s)}{2mc^2\overline{\beta^2}\cdot\bar{\gamma}}$

- If $\beta s \neq 0$: $k_x = \dfrac{-qE_0 T\,L\cos(\varphi_s)}{2mc^2\cdot\overline{\gamma^3}\cdot\overline{\beta^2}}\left(\bar{\gamma}^2 + \dfrac{kT'}{T}\right)$, $\quad k_y = \dfrac{-qE_0 T\,L\cos(\varphi_s)}{2mc^2\cdot\overline{\gamma^3}\cdot\overline{\beta^2}}\left(\bar{\gamma}^2 - \dfrac{kT'}{T}\right)$

(*) See Transit time factor definition

C: is a coefficient allowing to keep the matrix determinant equal to:

$\dfrac{(\beta\gamma)_e}{(\beta\gamma)_s}$: $C = \sqrt{\dfrac{\dfrac{(\beta\gamma)_e}{(\beta\gamma)_s}}{k_x k_y}}$

Solenoid (*Δs, B*)

Δs (mm) is the solenoid length, *B (T)* its axis magnetic field.

Let's use $k = \frac{B}{2B\rho}$, with $B\rho = \frac{m_0 c\beta\gamma}{q}$, the magnetic rigidity of the particle.

The non null 2×2 transfer sub-matrixes are:

$$R_{xx} = R_{yy} = \begin{bmatrix} \cos^2(k\Delta s) & \frac{\sin(k\Delta s)\cos(k\Delta s)}{k} \\ -k\sin(k\Delta s)\cos(k\Delta s) & \cos^2(k\Delta s) \end{bmatrix},$$

$$R_{xy} = -R_{yx} = \begin{bmatrix} \sin(k\Delta s)\cos(k\Delta s) & \frac{\sin^2(k\Delta s)}{k} \\ -k\sin^2(k\Delta s) & \sin(k\Delta s)\cos(k\Delta s) \end{bmatrix},$$

$$R_{zz} = \begin{bmatrix} 1 & \frac{\Delta s}{\gamma^2} \\ 0 & 1 \end{bmatrix}.$$

Bending magnet (*Δα, |ρ|, n, ouv, HV*)

Δα (°) is the rotation angle, *ρ (mm)* is the curvature radius, *n* is the field index, *ouv(mm)* the aperture and *HV* means: horizontal (=0) or vertical bend (=1).

A positive bend (*α>0*) bends the particles to the right in the horizontal plane, regardless of the sign of the charge on the particle. A negative α bends particles to the left.

$$h = \frac{1}{|\rho|}\frac{\Delta\alpha}{|\Delta\alpha|}, \quad k_x = \sqrt{(1-n)\cdot h^2}, \quad k_y = \sqrt{n\cdot h^2}, \quad \Delta s = |\rho|\Delta\alpha.$$

The non null 2×2 transfer sub-matrixes are:

$$R_{xx} = \begin{bmatrix} \cos(k_x\Delta s) & \frac{\sin(k_x\Delta s)}{k_x} \\ -k_x\sin(k_x\Delta s) & \cos(k_x\Delta s) \end{bmatrix}, R_{yy} = \begin{bmatrix} \cos(k_y\Delta s) & \frac{\sin(k_y\Delta s)}{k_y} \\ -k_y\sin(k_y\Delta s) & \cos(k_y\Delta s) \end{bmatrix},$$

$$R_{zz} = \begin{bmatrix} 1 & \frac{-h^2(k_x\Delta s\beta^2 - \sin(k_x\Delta s))}{k_x^3} + \frac{\Delta s}{\gamma^2}(1 - \frac{h^2}{k_x^2}) \\ 0 & 1 \end{bmatrix},$$

$$R_{zx} = \begin{bmatrix} \dfrac{-h\sin(k_x\Delta s)}{k_x} & \dfrac{-h(1-\cos k_x\Delta s))}{k_x^2} \\ 0 & 0 \end{bmatrix}. \quad R_{xz} = \begin{bmatrix} 0 & \dfrac{h(1-\cos k_x\Delta s))}{k_x^2} \\ 0 & \dfrac{h\sin(k_x\Delta s)}{k_x} \end{bmatrix}$$

Edge angle on bending magnet (β , $|\rho|$, g, K1, K2, ouv, HV)

β (°) is the edge angle, *ρ (mm)* is the curvature radius in the bending magnet, *g (mm)* is the gap between the poles of the bending magnet, K_1 and K_2 are used in a development for the fringe-field correction. If they are equal to zero, K_1 = 0.45 and K_2 = 2.8. Set small values to cancel fringe-field correction.
ouv (mm) is the aperture and *HV* means: horizontal (=0) or vertical bend (=1).
The edge angle is treated as a thin lens. Ψ is the fringe-field correction.

$$\Psi = K_1 \frac{g}{|\rho|}\left(\frac{1+\sin^2(\beta)}{\cos(\beta)}\right)\cdot\left(1-K_1 K_2 \frac{g}{|\rho|}\tan(\beta)\right).$$

The non null 2×2 transfer sub-matrixes are:

$$R_{xx} = \begin{bmatrix} 1 & 0 \\ \dfrac{\tan(\beta)}{|\rho|} & 1 \end{bmatrix}, \quad R_{yy} = \begin{bmatrix} 1 & 0 \\ \dfrac{-\tan(\beta-\Psi)}{|\rho|} & 1 \end{bmatrix} \text{ And } \quad R_{zz} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

Thin steering (BLx or ELx, Bly or ELy, r, Elec_flag)

$$x' = x' - \frac{qBL_y \cdot c}{mc^2\beta\gamma} \text{ And } y' = y' + \frac{qBL_x \cdot c}{mc^2\beta\gamma}$$

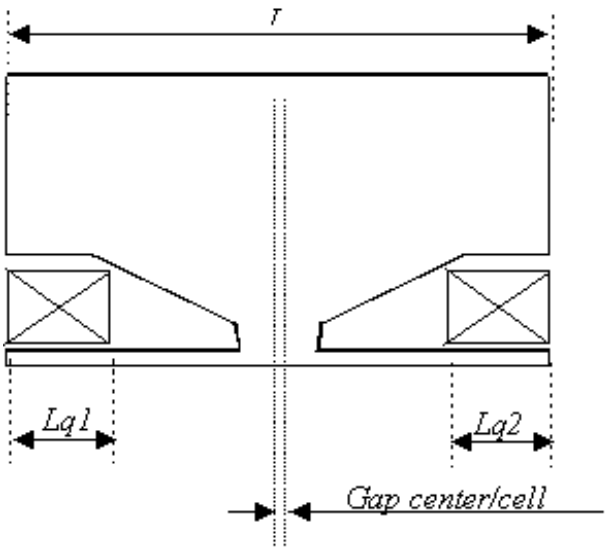
If *Elec_flag* equal 1

$$x' = x' + \frac{EL_x}{E\rho} \text{ and } y' = y' + \frac{\cdot EL_y}{E\rho}$$

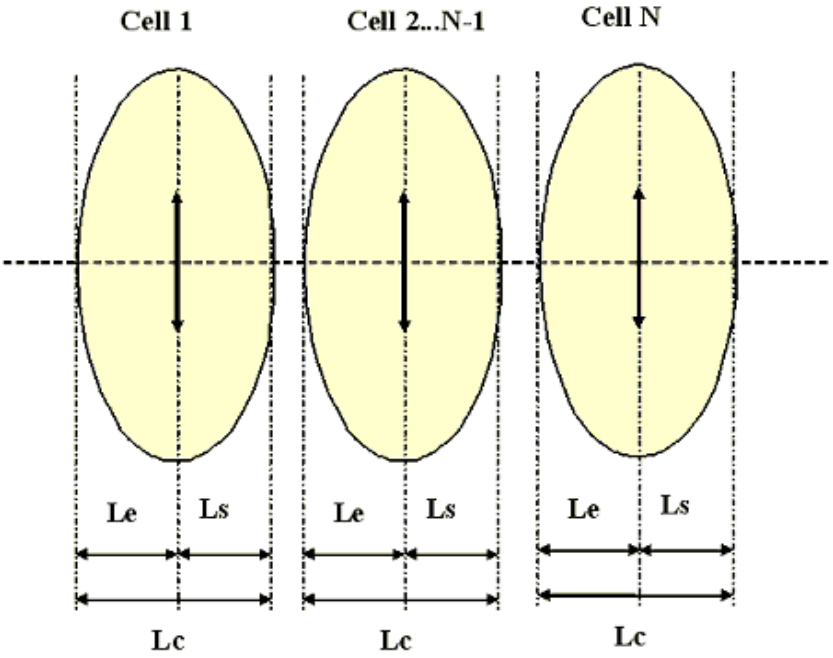
Where *x'* and *y'* being respectively the horizontal and vertical beam centroid slope.

DTL cell (L, Lq1, Lq2, g, B1', B2', EoTL, θs, r, p, βs, Ts, kT's, k²T''s)

The dimension *g(mm)* is defined as: *gap position* $= \frac{L}{2} - g - Lq_1$.



Cavity multi-gap (*m, N, βg, EoT, θs, r, p, kEoTi, kEoTo, Dzi¸Dzo, βs, Ts, kT's,k²T''i, Ts, kT'i,k²T''i, To, kT'o,k²T''o*)



If m=0:
For cell1 : Le = ½ βλ +dzi, Ls = ½ βλ, Lc = βλ, EoT(1) = EoT (1+kEoTi).(Ti/Ts)
For cell2 to N-1 : Le = Ls = ½ βλ, Lc = βλ, EoT(2..N-1) = EoT
For cellN : Le = ½ βλ , Ls = ½ βλ + dzo, Lc = βλ, EoT(N) = EoT.(1+kEoTo).(To/Ts)
If m=1:
For cell1 : Le=1/4 βλ +dzi, Ls=1/4 βλ, Lc= ½ βλ, EoT(1)= EoT.(1+kEoTi).(Ti/Ts)
For cell2 to N-1 : Le=Ls=1/4 βλ, Lc = ½ βλ, EoT(2..N-1)= EoT.(1+kEoTi).(Ti/Ts)
For cellN : Le=1/4 βλ , Ls=1/4 βλ+dzo, Lc= ½ βλ, EoT(N)= EoT.(1+kEoTo).(To/Ts)
If m=2:
For cell1 : Le=1/4 βλ + dzi, Ls=½ βλ, Lc= ¾ βλ, EoT(1)= EoT.(1+kEoTi).(Ti/Ts)
For cell2 to N-1 : Le=Ls= ½ βλ, Lc = βλ, EoT(2..N-1)= EoT.(1+kEoTi).(Ti/Ts)
For cellN : Le=½ βλ, Ls=1/2 βλ + dzo, Lc= ¾ βλ, EoT(N)= EoT.(1+kEoTo).(To/Ts)

For all cases : EoTL= EoT(x).(Ls + Le)

Electrostatic Acceleration (*Vo, Δs, K*)

Vo (V) is the voltage, *Δs* (mm) is the step length, *L* (mm) is the element length, and *K* (eV/m²) is the transverse defocalisation contribution.
The reduced energy change in the gap is:

$$\gamma_s = \gamma_e + \frac{|q|\frac{V_0}{L}\Delta_s}{mc^2}.$$

Let's use:

$$K^* = \frac{qK\Delta_s}{m_0c^2\bar{\beta}^2\bar{\gamma}} \text{ and } \delta = \sqrt{\frac{(\beta\gamma)_e}{(\beta\gamma)_s}} \text{ with } \bar{\gamma} = \frac{\gamma_e + \gamma_s}{2} \text{ and } \bar{\beta} = \frac{\beta_e + \beta_s}{2}.$$

The non null 2×2 transfer sub-matrixes are:

$$R_{xx} = R_{yy} = \begin{bmatrix} 1 - K^*\frac{\Delta_s}{2} & \frac{\Delta_s}{2}\left(1 - K^*\frac{\Delta_s}{2} + \delta\right) \\ -K^*\delta & -K^*\delta\frac{\Delta_s}{2} + \delta^2 \end{bmatrix}$$

$$R_{zz} = \begin{bmatrix} 1 & \Delta_s \\ 0 & \delta^2 \end{bmatrix}$$

Sinus cavity or CCL (*L, N, EoT, θs*)

L (mm) is the cavity length, *N* is the number of cells, *EoT (eV/m)* is the mean electric field of the cavity, *θs (°)* is the phase of the synchronous particle at the entrance of the cavity (relative to the R.F. phase).

Fields

Let's assume the electric field on the axis:

$$E_z(z,t) = E_0\sin(\omega t + \varphi_0)\sin\left(\frac{K}{\beta_c}z\right), \qquad \text{With } \beta_c = \frac{2L}{N\lambda}.$$

The transverse electric field component can be deduced from Maxwell equations with a first order expansion:

$$E_r(z,t,r) = -\frac{r}{2}\frac{dE_z}{dz} = -\frac{KE_0}{2\beta_c}\sin(\omega t + \varphi_0)\cos\left(\frac{K}{\beta_c}z\right)\cdot r,$$

$$\Rightarrow \begin{cases} E_x(z,t,x) = -\dfrac{KE_0}{2\beta_c}\sin(\omega t+\varphi_0)\cos\left(\dfrac{K}{\beta_c}z\right)\cdot x \\ E_y(z,t,y) = -\dfrac{KE_0}{2\beta_c}\sin(\omega t+\varphi_0)\cos\left(\dfrac{K}{\beta_c}z\right)\cdot y \end{cases}$$

The same way, the transverse magnetic field component can be deduced:

$$B_\theta(z,t,r) = \frac{r}{2c^2}\frac{dE_z}{dt} = \frac{KE_0}{2c}\cos(\omega t+\varphi_0)\sin\left(\frac{K}{\beta_c}z\right)\cdot r$$

,

$$\Rightarrow \begin{cases} B_x(z,t,y) = -\dfrac{KE_0}{2c}\cos(\omega t+\varphi_0)\sin\left(\dfrac{K}{\beta_c}z\right)\cdot y \\ B_y(z,t,x) = \dfrac{KE_0}{2c}\cos(\omega t+\varphi_0)\sin\left(\dfrac{K}{\beta_c}z\right)\cdot x \end{cases}$$

With $K = \dfrac{2\pi}{\lambda} = \dfrac{\omega}{c}$, $\omega = 2\pi f$, f the R.F frequency and c the speed of light in vacuum.

Longitudinal motion

Let's p_s be the synchronous particle momentum at a given z, its evolution is given by:

$$\dot{p}_s = qE_z(z,t_s) = qE_{zs}.$$

The particle momentum is: $p = \sqrt{p_x^2+p_y^2+p_z^2} = p_z\sqrt{1+x'^2+y'^2}$,

With $\begin{cases} \dot{p}_x = q\left(E_x - v_z B_y + v_y B_z\right) \\ \dot{p}_y = q\left(E_y + v_z B_x - v_x B_z\right) \\ \dot{p}_z = q\left(E_z + v_x B_y - v_y B_x\right) \end{cases}$

The evolution with time of the particle momentum p is given by:

$$\dot{p} = \frac{p_z}{p}\left(\dot{p}_z + x'\dot{p}_x + y'\dot{p}_y\right).$$

Let's δ be the momentum of the particle relative to that of the synchronous particle:

$$\delta = \frac{p-p_s}{p_s}.$$

We have: $p = (1+\delta)p_s = p_z\sqrt{1+x'^2+y'^2}$.

The evolution of δ is given by the equation:

$$\dot{\delta} = \frac{1}{p_s^2}\left(p_s\dot{p} - \dot{p}_s p\right),$$

$$\dot{\delta} = \frac{1}{p_s^2}\left(\frac{p_s p_z}{p}\left(\dot{p}_z + x'\dot{p}_x + y'\dot{p}_y\right) - \dot{p}_s p\right),$$

$$\dot{\delta} = \frac{q}{p_s}\left[\left(1+x'^2+y'^2\right)^{1/2}\cdot\left(E_z + x'E_x + y'E_y\right) - (1+\delta)E_{zs}\right].$$

With $E_z = E_{zs} + \left.\frac{dE_z}{d\varphi}\right]_{\varphi=\varphi_s} \cdot \phi$, $\phi = \varphi - \varphi_s$ and $\frac{d}{dz} = \frac{1}{\beta_s c} \cdot \frac{d}{dt}$,

We finally find at first order:

$$\boxed{\frac{d\delta}{dz} = \delta' = \frac{qE_0}{\gamma_s \beta_s^2 mc^2} \sin\left(\frac{Kz}{\beta_c}\right) \left[\cos(\omega t_s + \varphi_0) \cdot \phi - \sin(\omega t_s + \varphi_0) \cdot \delta\right]}.$$

Both focusing and damping effects can be observed.

Thin lens approximation

Using: $\phi = -\frac{K}{\beta_s} \cdot \delta z$ the matrix transport can then be written over a small step *dz*:

$$\begin{pmatrix} \delta z \\ \delta \end{pmatrix}_{z+dz/2} = \begin{pmatrix} 1 & dz/2\gamma_I^{*2} \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ K_1 & K_2 \end{pmatrix} \begin{pmatrix} 1 & dz/2\gamma_O^{*2} \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} \delta z \\ \delta \end{pmatrix}_{z-dz/2} = M_z \cdot \begin{pmatrix} \delta z \\ \delta \end{pmatrix}_{z-dz/2},$$

With:

γ_s and γ_e being the synchronous normalized energy before and after the "gap",

$$\boxed{K_1 = -\frac{qE_0 K}{\gamma_s \beta_s^3 mc^2} \sin\left(\frac{Kz}{\beta_c}\right) \cos(\omega t_s + \varphi_0)}$$ And

$$\boxed{K_2 = 1 - \frac{qE_0}{\gamma_s \beta_s^2 mc^2} \sin\left(\frac{Kz}{\beta_c}\right) \sin(\omega t_s + \varphi_0)}.$$

Transverse motion

The evolution of *x'* with time is driven by the equation:

$$\dot{x}' = \frac{d}{dt}\left(\frac{p_x}{p_z}\right) = \frac{1}{p_z^2}\left(\dot{p}_x p_z - \dot{p}_z p_x\right)$$

$$\dot{x}' = \frac{q}{p_s} \cdot \frac{\left(1 + x'^2 + y'^2\right)^{1/2}}{1+\delta} \cdot \left(E_x - v_z B_y - x'\left(E_z + v_x B_y - v_y B_x\right)\right).$$

At first order, we finally have:

$$\boxed{x'' = -\frac{qE_0}{\gamma_s \beta_s^2 mc^2} \left[\begin{array}{l} \frac{K}{2}\left(\frac{1}{\beta_c} \sin(\omega t_s + \varphi_0) \cos\left(\frac{Kz}{\beta_c}\right) + \beta_s \cos(\omega t_s + \varphi_0) \sin\left(\frac{Kz}{\beta_c}\right)\right) \cdot x \\ + \sin(\omega t_s + \varphi_0) \sin\left(\frac{Kz}{\beta_c}\right) \cdot x' \end{array}\right]}.$$

Both focusing and damping effects can be observed.

Thin lens approximation

The matrix transport over a small step *dz* can then be written:

$$\begin{pmatrix} x \\ x' \end{pmatrix}_{z+dz/2} = \begin{pmatrix} 1 & dz/2 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ k_1 & k_2 \end{pmatrix} \begin{pmatrix} 1 & dz/2 \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x \\ x' \end{pmatrix}_{z-dz/2} = M_x \cdot \begin{pmatrix} x \\ x' \end{pmatrix}_{z-dz/2},$$

With: $\boxed{k_1 = -\frac{qE_0K}{2\gamma_s\beta_s^2mc^2}\left[\frac{1}{\beta_c}\sin(\omega t_s+\varphi_0)\cos\left(\frac{Kz}{\beta_c}\right)+\beta_s\cos(\omega t_s+\varphi_0)\sin\left(\frac{Kz}{\beta_c}\right)\right]}$,

And $\boxed{k_2 = K_2}$.

Thick lens approximation

The differential equation can be written:

$x'' + 2\cdot\alpha\cdot x' + \beta\cdot x = 0$,

With: $\beta = \frac{qE_0K}{2\gamma_s\beta_s^2mc^2}\left[\frac{1}{\beta_c}\sin(\omega t_s+\varphi_0)\cos\left(\frac{Kz}{\beta_c}\right)+\beta_s\cos(\omega t_s+\varphi_0)\sin\left(\frac{Kz}{\beta_c}\right)\right]$

And: $\alpha = \frac{qE_0}{2\gamma_s\beta_s^2mc^2}\cdot\sin(\omega t_s+\varphi_0)\cdot\sin\left(\frac{Kz}{\beta_c}\right)$,

And: $\omega = \sqrt{\beta-\alpha^2}$,

The solution of this differential equation gives:

$\boxed{M_x = e^{-\alpha\cdot dz}\cdot\begin{pmatrix}\cos(\omega\,dz)+\frac{\alpha}{\omega}\cdot\sin(\omega\,dz) & \frac{\sin(\omega\,dz)}{\omega}\\ -\frac{\beta\sin(\omega\,dz)}{\omega} & \cos(\omega\,dz)-\frac{\alpha}{\omega}\cdot\sin(\omega\,dz)\end{pmatrix}}$, If ω is real,

And

$\boxed{M_x = e^{-\alpha\cdot dz}\cdot\begin{pmatrix}ch(|\omega|\,dz)+\frac{\alpha}{|\omega|}\cdot sh(|\omega|\,dz) & \frac{sh(|\omega|\,dz)}{|\omega|}\\ \frac{\beta\,sh(|\omega|\,dz)}{|\omega|} & ch(|\omega|\,dz)-\frac{\alpha}{|\omega|}\cdot sh(|\omega|\,dz)\end{pmatrix}}$, If $j\,\omega$ is real.

Transport through a sin-like cavity

The N_c-cells cavity is divided in n·N_c steps of length: $dz = \frac{\beta_c\lambda}{2n}$.

As input, we have:

$\gamma_o = \gamma_{in}$; $t_s = \frac{dz}{2\beta_{in}c}$; $z_s = \frac{dz}{2}$; $\begin{pmatrix}x\\x'\end{pmatrix}_0$, $\begin{pmatrix}\delta z\\ \delta\end{pmatrix}_0$: Particle co-ordinates at cavity input.

Then, we loop until reaching the end of the cavity:

For i from 0 to n·Nc-1 do

$\gamma_I^* = \gamma_O^*$, $\gamma_O^* = \gamma_{i+1} = \gamma_i + \frac{qE_0}{mc^2}\sin(\omega t_s+\varphi_0)\sin\left(\frac{K}{\beta_c}\cdot z_s\right)\cdot dz$, $\beta_{i+1} = \sqrt{1-\gamma_{i+1}^{-2}}$,

$\gamma_s = \frac{\gamma_{i+1}+\gamma_i}{2}$, $\beta_s = \sqrt{1-\gamma_s^{-2}}$,

$\begin{pmatrix}x\\x'\end{pmatrix}_{i+1} = M_x\cdot\begin{pmatrix}x\\x'\end{pmatrix}_i$, $\begin{pmatrix}\delta z\\ \delta\end{pmatrix}_{i+1} = M_z\cdot\begin{pmatrix}\delta z\\ \delta\end{pmatrix}_i$,

$$t_s = t_s + \frac{dz}{\beta_{i+1} c}, \quad z_s = z_s + dz .$$

RFQ cell (*V, ro, A10, m, L, θs, Type ,ρ, A01*)

L (mm) is the RFQ cell length, V*(V)* is the mean voltage of the cell, *θs(°)* is the phase of the synchronous particle, *ro* is…,*m* is … and *type*...
Cell *type*:

±2: Accelerating cell.
±3: Front-end cell.
±4: Transcell.

The sign of type being…..

$$W_{i+1} = W_i + |q| dz E_z \text{ And } \overline{W} = \frac{1}{2}(W_i + W_{i+1})$$

$$\Phi_{i+1} = \Phi_i + dz \frac{2\pi}{\overline{\beta}\lambda}$$

$$E_z = \frac{\pi A_{10} V}{2L} \sin\left(\frac{\pi}{L} z\right) \sin(\omega t_s + \varphi_0)$$

Thin lens approximation (Longitudinal)

$\omega = 2\pi f$, f the R.F frequency and *c* the speed of light in vacuum.
The matrix transport can then be written over a small step *dz*:

$$\begin{pmatrix} \delta z \\ \delta \end{pmatrix}_{z+dz/2} = \begin{pmatrix} 1 & dz/2\gamma_I^{*2} \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ K_1 & K_2 \end{pmatrix} \begin{pmatrix} 1 & dz/2\gamma_O^{*2} \\ 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} \delta z \\ \delta \end{pmatrix}_{z-dz/2} = M_z \cdot \begin{pmatrix} \delta z \\ \delta \end{pmatrix}_{z-dz/2},$$

With:

γs and γe being the synchronous normalized energy before and after the "gap",

$$K_1 = -\frac{|q| d_z A_{10} V}{\gamma_s \beta_s^2 2mc^2} \left(\frac{\pi}{L}\right)^2 \cdot C_3 \cos(\omega t_s + \varphi_0)$$

$$K_2 = 1 - \frac{|q| d_z A_{10} V}{\gamma_s \beta_s^2 2mc^2} \left(\frac{\pi}{L}\right) \cdot C_3 \cos(\omega t_s + \varphi_0)$$

C3 depend of the cell *type*.

±2 or ±3: $C_3 = \sin\left(\frac{\pi}{L} z\right)$

±4: $C_3 = \frac{1}{2} \sin\left(\frac{\pi}{L} z\right)$

Thin lens approximation (Transverse)

The matrix transport over a small step *dz* can then be written:

$$\begin{pmatrix} x \\ x' \end{pmatrix}_{z+dz/2} = \begin{pmatrix} 1 & dz/2 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} 1 & 0 \\ k_{x1} & k_{x2} \end{pmatrix}\begin{pmatrix} 1 & dz/2 \\ 0 & 1 \end{pmatrix}\cdot\begin{pmatrix} x \\ x' \end{pmatrix}_{z-dz/2} = M_x \cdot \begin{pmatrix} x \\ x' \end{pmatrix}_{z-dz/2},$$

$$\begin{pmatrix} y \\ y \end{pmatrix}_{z+dz/2} = \begin{pmatrix} 1 & dz/2 \\ 0 & 1 \end{pmatrix}\begin{pmatrix} 1 & 0 \\ k_{y1} & k_{y2} \end{pmatrix}\begin{pmatrix} 1 & dz/2 \\ 0 & 1 \end{pmatrix}\cdot\begin{pmatrix} y \\ yx \end{pmatrix}_{z-dz/2} = M_y \cdot \begin{pmatrix} y \\ y' \end{pmatrix}_{z-dz/2}$$

With: $\boxed{k_{x1} = -\frac{|q|d_z}{\gamma_s \beta_s^2 2mc^2}\cos(\omega t_s + \varphi_0)\left(S\frac{V}{r_0^2}A_{01}C_1 - \left(\frac{\pi}{L}\right)^2 \frac{A_{10}V}{4}C_2\right)}$,

$$\boxed{k_{y1} = -\frac{|q|d_z}{\gamma_s \beta_s^2 2mc^2}\cos(\omega t_s + \varphi_0)\left(-S\frac{V}{r_0^2}A_{01}C_1 - \left(\frac{\pi}{L}\right)^2 \frac{A_{10}V}{4}C_2\right)}$$

And $\boxed{k_{x2} = k_{y2} = K_2}$.

$$\begin{pmatrix} x \\ x' \end{pmatrix}_{i+1} = M_x \cdot \begin{pmatrix} x \\ x' \end{pmatrix}_i \text{ And } \begin{pmatrix} y \\ y' \end{pmatrix}_{i+1} = M_y \cdot \begin{pmatrix} y \\ y' \end{pmatrix}_i$$

C1 , *C2* and *S* depend of the cell *type*.

±2: $C_1 = 1, C_2 = \sin\left(\frac{\pi}{L}z\right)$, $S = -sign(type)$

+3: $C_1 = \frac{1}{4}\left(3\cos(\frac{1}{2}\frac{\pi}{L}z - \frac{\pi}{2}) + \cos\left(3\cdot\left(\frac{1}{2}\frac{\pi}{L}z - \frac{\pi}{2}\right)\right)\right), C_2 = 0$, $S = -sign(type[n+1])$

-3: $C_1 = \frac{3}{4}\left(\cos(\frac{1}{2}\frac{\pi}{L}z) - \frac{1}{3}\cos(\frac{3}{2}\frac{\pi}{L}z)\right)$, $C_2 = 0$, $S = -sign(type[n-1])$

+4: $C_1 = 1, C_2 = \frac{1}{2}\left(\cos\left(\frac{\pi}{L}z\right) + 1\right)$, $S = -sign(type[n+1])$

-4: $C_1 = 1, C_2 = -\frac{1}{2}\left(\cos\left(\frac{\pi}{L}z\right) - 1\right)$, $S = -sign(type[n-1])$

With *type[n+1]* being the type from the next cell and *type[n-1]* the type from preceding cell.

Transport through a rfq cell

The rfq cell is divided in ·*N* steps of length: $dz = \frac{L}{N}$.

As input, we have:

$\gamma_o = \gamma_{in}$; $t_s = \frac{dz}{2\beta_{in}c}$; $z_s = \frac{dz}{2}$; $\begin{pmatrix} x \\ x' \end{pmatrix}_0$, $\begin{pmatrix} \delta z \\ \delta \end{pmatrix}_0$: Particle co-ordinates at cavity input.

Then, we loop until reaching the end of the cavity:
For i from 0 to N-1 do

$$\gamma_I^* = \gamma_O^*,\ \gamma_O^* = \gamma_{i+1} = \gamma_i + \frac{|q|\pi AV}{2Lmc^2}\sin(\omega t_s + \varphi_0)\sin\left(\frac{\pi}{L}\cdot z_s\right)\cdot dz,\ \beta_{i+1} = \sqrt{1-\gamma_{i+1}^{-2}},$$

$$\gamma_s = \frac{\gamma_{i+1}+\gamma_i}{2},\ \beta_s = \sqrt{1-\gamma_s^{-2}},$$

$$\begin{pmatrix} x \\ x' \end{pmatrix}_{i+1} = M_x \cdot \begin{pmatrix} x \\ x' \end{pmatrix}_i,\ \begin{pmatrix} \delta z \\ \delta \end{pmatrix}_{i+1} = M_z \cdot \begin{pmatrix} \delta z \\ \delta \end{pmatrix}_i,$$

$$t_s = t_s + \frac{dz}{\beta_{i+1}c},\ z_s = z_s + dz.$$

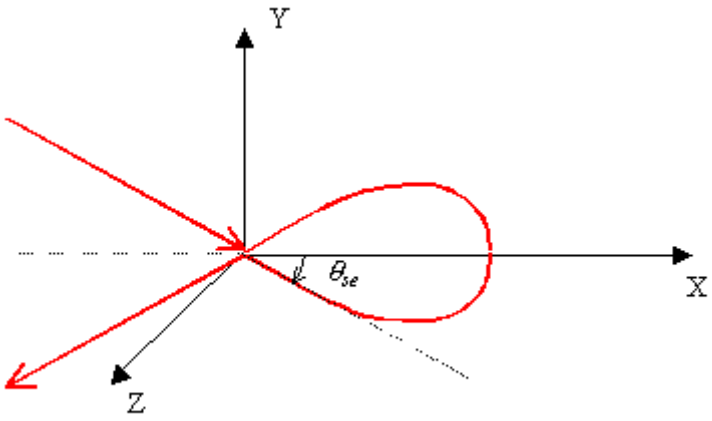
Transfer matrix of alpha magnet (*Θ, K, R, plan*)

The following paper come from a CEA report named:
"Transfer matrix of a constant gradient alpha magnet for ELSA extension"
Ref: DSM/DAPNIA/SACM/2002/13
N. Pichoff

In this paper, the motion equation of a particle in such a device is linearised to get the transfer matrix. The matrix has been validated with a step-by-step integration of the motion of a particle in the magnetic field.

Magnetic field and fixed coordinate system

The trajectory of the synchronous particle (ideal trajectory) is, by definition, in the plan (X, Y). The frame origin is the point where the particle enters the magnet. The X direction is the direction of the magnetic field gradient. At any position, the main trajectory makes an angle θ_s with the X axis. The entrance angle for negative particle θ_{se} is: -40.71°. With this particular angle, the synchronous particle exits the magnet at the same position as the entering particle (and with the opposite angle).



In this frame, the magnetic field is:

$$\vec{B} = \begin{pmatrix} B_X = k\cdot Z \\ B_Y = 0 \\ B_Z = k\cdot X \end{pmatrix}$$

Trajectory of the synchronous particle

The synchronous particle with charge *q* moves in the (X, Y) plan.
Its motion equations with time *t* are:

$$\begin{cases} \dfrac{dp_X}{dt} = q \cdot v_Y \cdot B_Z \\ \dfrac{dp_Y}{dt} = -q \cdot v_X \cdot B_Z \end{cases}$$

p is the particle momentum, *v* its velocity,

giving:

$$\begin{cases} \dfrac{dp_X}{ds} = q \cdot \dfrac{p_Y}{p_0} \cdot k \cdot X \\ \dfrac{dp_Y}{ds} = -q \cdot \dfrac{p_X}{p_0} \cdot k \cdot X \\ \dfrac{dX}{ds} = \dfrac{p_X}{p_0} \\ \theta_s = \arcsin\left(\dfrac{p_Y}{p_0}\right) = \arccos\left(\dfrac{p_X}{p_0}\right) \end{cases}$$

$p_0 = \beta_0 \gamma_0 m_0 c$ is the particle momentum modulus,

s is the curvilign abscissa, with $ds = v_0 \cdot dt$.

These equations have to be solved using as initial conditions:

$$\begin{cases} p_X = p_0 \cos\theta_{se} \\ p_Y = p_0 \sin\theta_{se} \\ X = 0 \\ \theta_s = \theta_{se} \end{cases}$$

The maximum *penetration of the particle* X_M in the magnet can be calculated the following way:

$$\frac{dp_Y}{dX} = -q \cdot k \cdot X \qquad \Rightarrow \qquad p_Y = p_{Y0} - \frac{1}{2} \cdot q \cdot k \cdot X^2 .$$

At maximum penetration, one has: $p_Y = p_0$, giving:

$$X_M = \sqrt{\frac{2 \cdot (p_{Y0} - p_0)}{q \cdot k}} .$$

The length *L* of the trajectory is obtained from the integration of:

$$dt = \frac{\gamma_0 m_0 \cdot dX}{p_X} .$$

One obtains:

$$L = 2 \cdot p_0 \cdot \int_0^{X_M} \left(p_{X0}^2 + q \cdot k \cdot p_{Y0} \cdot X^2 - 0.25 \cdot q^2 k^2 \cdot X^4\right)^{-\frac{1}{2}} \cdot dX .$$

This integral cannot be solved analytically but can be easily calculated numerically.

The moving coordinate system

In the transfer matrix formalism, a beam particle is referenced, at a given curvilign abscissa ***s***, to the synchronous particle in a 6D phase-space, with a 6-coordinates vector:

$$\vec{r} = \begin{pmatrix} x \\ x' \\ y \\ y' \\ \varphi \\ \delta \end{pmatrix},$$

with:

- ***x*** is the particle transverse position in the deviation plan,

- ***x'*** is the particle transverse slope in the x direction. $x' = \frac{p_x}{p_s} = \frac{dx}{ds}$, $\boldsymbol{p_x}$ the x-component of the particle momentum, and $\boldsymbol{p_s}$ the s-component of the particle momentum,
- **y** is the particle transverse position orthogonel to the x direction,

- ***y'*** is the particle transverse slope in the y direction. $y' = \frac{p_y}{p_s} = \frac{dy}{ds}$, $\boldsymbol{p_y}$ the y-component of the particle momentum,
- φ is the time difference between the particle arriving in s (***t***) and the synchronous particle arriving in s ($\boldsymbol{t^{(S)}}$). This time is normalized with the RF frequency $\boldsymbol{f_{RF}}$.

$$\varphi = 2\pi \cdot f_{RF} \cdot \left(t - t^{(S)}\right),$$

- δ is the particle momentum ***p*** relative to the synchronous particle momentum $\boldsymbol{p^{(S)}}$.

$$\delta = \frac{p - p^{(S)}}{p^{(S)}}.$$

Matrix calculation

In the matrix formalism, the particle vector change from a point (1) to a point (2) is given by:

$$(\vec{r})_{(2)} = [T] \cdot (\vec{r})_{(1)},$$

where [*T*] is the transfer matrix from (1) to (2). The coefficients of [*T*] are:

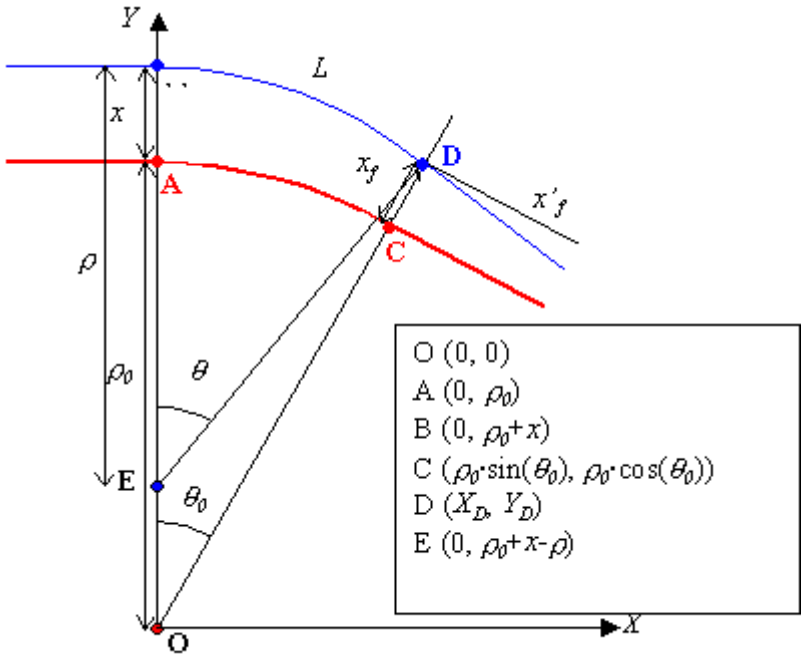
$$T_{i,j} = \frac{\partial r_i)_{(2)}}{\partial r_j)_{(1)}}.$$

i is the line index, *j* is the column index (between 1 and 6).

The matrix coefficients can be calculated by varying the input particle coordinate along each direction independently and looking at the output coordinates.

Matrix first column
Matrix second column
Matrix sixth column
Matrix fifth column
Matrix third and fourth columns

Matrix first column: variation with x



Calculus of $T_{1,1}$

D is at the intersection of (O, C) and the circle with centre E and radius ρ. Its coordinates satisfy:

$$\begin{cases} X_D = \tan(\theta_0)\cdot Y_D \\ X_D^2 + \left(Y_D - (\rho_0 + x - \rho)\right)^2 = \rho^2 \end{cases}.$$

Giving:b $$\left(1+\tan^2\theta_0\right)\cdot Y_D^2 - 2\cdot(x-d\rho)\cdot Y_D + (\rho_0+x)\cdot(x-2\cdot d\rho-\rho_0) = 0,$$

with: $$\rho = \rho_0 + d\rho = \rho_0\cdot\left(1+\frac{d\rho}{\rho_0}\right).$$

As: $$\rho = \frac{p}{q\cdot B},$$ one has: $$\frac{d\rho}{\rho_0} = -\frac{dB}{B_0} = \sin\theta_s\cdot\frac{x}{X_s}.$$

The solution of (X) is obtained giving a reduced discriminator:

$$\Delta' = (x-d\rho)^2 - \frac{(\rho_0+x)\cdot(x-d\rho-\rho_0)}{\cos^2\theta_0},$$

giving:

$$\Delta' = \frac{\rho_0^2}{\cos^2\theta_0}\cdot\left[\left(\frac{x}{\rho_0}-\frac{d\rho}{\rho_0}\right)^2\cdot\cos^2\theta_0 - \left(1+\frac{x}{\rho_0}\right)\cdot\left(\frac{x}{\rho_0}-2\cdot\frac{d\rho}{\rho_0}-1\right)\right].$$

A first order development in $\frac{x}{\rho_0} << 1$ and $\frac{d\rho}{\rho_0} << 1$ gives:

$$\Delta' = \frac{\rho_0^2}{\cos^2\theta_0} \cdot \left[1 + 2 \cdot \frac{d\rho}{\rho_0}\right].$$

This gives the coordinates if D:

$$\begin{cases} X_D = \rho_0 \cdot \sin\theta_0 \cdot \left(1 + \cos\theta_0 \cdot \frac{x}{\rho_0} + (1 - \cos\theta_0) \cdot \frac{d\rho}{\rho_0}\right) \\ Y_D = \rho_0 \cdot \cos\theta_0 \cdot \left(1 + \cos\theta_0 \cdot \frac{x}{\rho_0} + (1 - \cos\theta_0) \cdot \frac{d\rho}{\rho_0}\right) \end{cases}$$

The final position of the particle in the moving frame is then:

$$\boxed{x_f = \frac{Y_D - Y_C}{\cos\theta_0} = \left(\cos\theta_0 + (1 - \cos\theta_0) \cdot \frac{\rho_0 \cdot \sin\theta_s}{X_s}\right) \cdot x = T_{1,1} \cdot x}$$

Calculus of $T_{2,1}$

$$x'_f = \tan(\theta_0 - \theta) = \frac{\tan\theta_0 - \tan\theta}{1 + \tan\theta_0 \cdot \tan\theta},$$

with: $\tan\theta = \frac{X_D - X_E}{Y_D - Y_E}$.

At first order, one has:

$$\tan\theta = \tan\theta_0 \cdot \left(1 + \frac{x - d\rho}{\rho_0 \cos\theta_0}\right),$$

giving :

$$\boxed{x'_f = -\sin\theta_0 \cdot \left(1 - \frac{\rho_0 \cdot \sin\theta_s}{X_s}\right) \cdot \frac{x}{\rho_0} = T_{2,1} \cdot x}$$

Calculus of $T_{5,1}$

$$\varphi = \frac{2\pi \cdot f_{RF}}{\beta_0 c}(L - \rho_0\theta_0) = \frac{2\pi \cdot f_{RF}}{\beta_0 c}\rho_0\theta_0 \cdot \left(\frac{d\rho}{\rho_0} + \frac{d\theta}{\theta_0}\right).$$

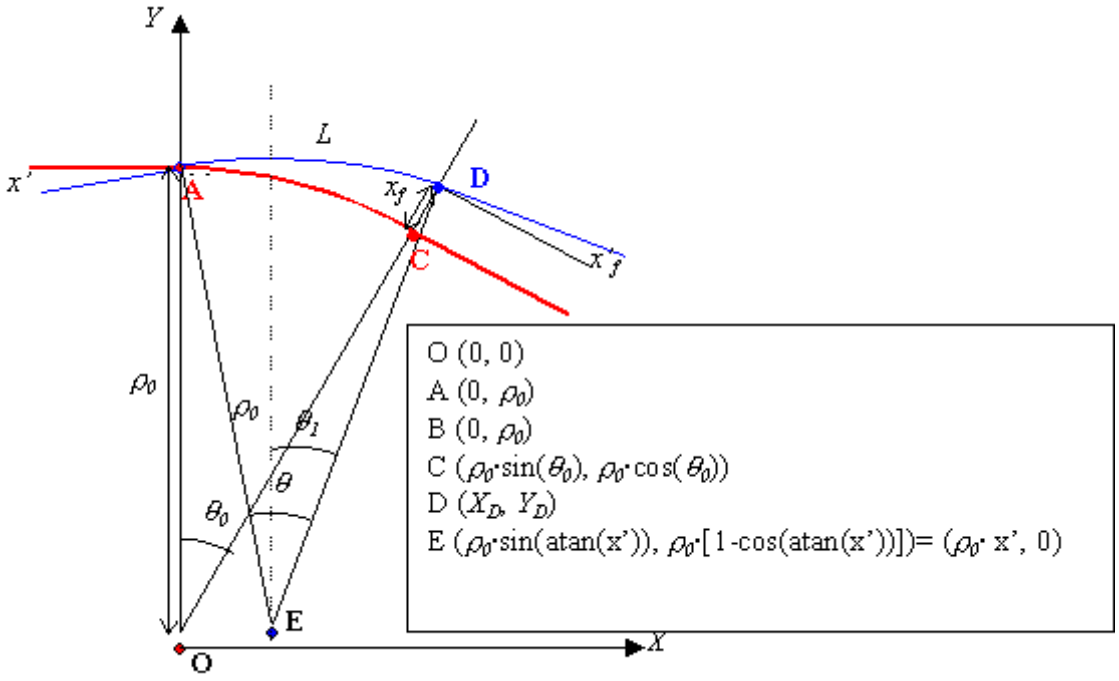
with: $d\theta = \cos^2\theta_0 \cdot d(\tan\theta) = \cos^2\theta_0 \cdot (\tan\theta - \tan\theta_0) = \sin\theta_0 \cdot \left(\frac{x - d\rho}{\rho_0}\right)$.

One obtains finally:

$$\varphi = \frac{2\pi \cdot f_{RF}}{\beta_0 c} \cdot \left(\sin\theta_0 + (\theta_0 - \sin\theta_0) \cdot \frac{\rho_0 \sin\theta_s}{X_s} \right) \cdot x = T_{5,1} \cdot x .$$

The other terms ($T_{3,1}$, $T_{4,1}$,$T_{6,1}$) are equal to zero.

Matrix second column: variation with x'



Calculus of $T_{1,2}$

D is at the intersection of (O, C) and the circle with centre E and radius ρ. Its coordinates satisfy:

$$\begin{cases} X_D = \tan\theta_0 \cdot Y_D \\ (X_D - \rho_0 \cdot x')^2 + Y_D^2 = \rho_0^2 \end{cases}.$$

Giving:

$$(1 + \tan^2\theta_0) \cdot Y_D^2 - 2 \cdot \rho_0 \cdot x' \cdot \tan\theta_0 \cdot Y_D - \rho_0^2 \cdot (1 - x'^2) = 0 .$$

The solution of (X) is obtained giving a reduced discriminator:

$$\Delta' = \rho_0^2 \cdot \tan^2\theta_0 \cdot x'^2 + \frac{\rho_0^2 \cdot (1 - x'^2)}{\cos^2\theta_0} .$$

A first order development in $x' << 1$ gives:

$$\Delta' = \frac{\rho_0^2}{\cos^2\theta_0}$$

This gives the coordinates of D:

$$\begin{cases} X_D = \rho_0 \cdot \sin\theta_0 \cdot (1 + \sin\theta_0 \cdot x') \\ Y_D = \rho_0 \cdot \cos\theta_0 \cdot (1 + \sin\theta_0 \cdot x') \end{cases}$$

The final position of the particle in the moving frame is then:

$$\boxed{x_f = \frac{Y_D - Y_C}{\cos\theta_0} = \rho_0 \cdot \sin\theta_s \cdot x' = T_{1,2} \cdot x'}$$

Calculus of $T_{2,2}$

$$x'_f = \tan(\theta_0 - \theta_1) = \frac{\tan\theta_0 - \tan\theta_1}{1 + \tan\theta_0 \cdot \tan\theta_1}$$

with: $\tan\theta_1 = \frac{X_D - X_E}{Y_D - Y_E}$.

At first order, one has:

$$\tan\theta = \frac{X_D - X_E}{Y_D - Y_E},$$

giving :

$$\boxed{x'_f = \cos\theta_0 \cdot x' = T_{2,2} \cdot x'}$$

Calculus of $T_{5,2}$

$$\varphi = \frac{2\pi \cdot f_{RF}}{\beta_0 c}(L - \rho_0\theta_0) = \frac{2\pi \cdot f_{RF}}{\beta_0 c}\rho_0 \cdot d\theta$$

.

with : $d\theta = \cos^2\theta_0 \cdot d(\tan\theta) = \cos^2\theta_0 \cdot (\tan\theta - \tan\theta_0)$,

as: $\tan\theta = \tan(\theta_1 + \theta_0) = \tan\theta_0 \cdot \left(1 + \frac{1 - \cos\theta_0}{\sin\theta_0 \cdot \cos\theta_0} \cdot x'\right)$,

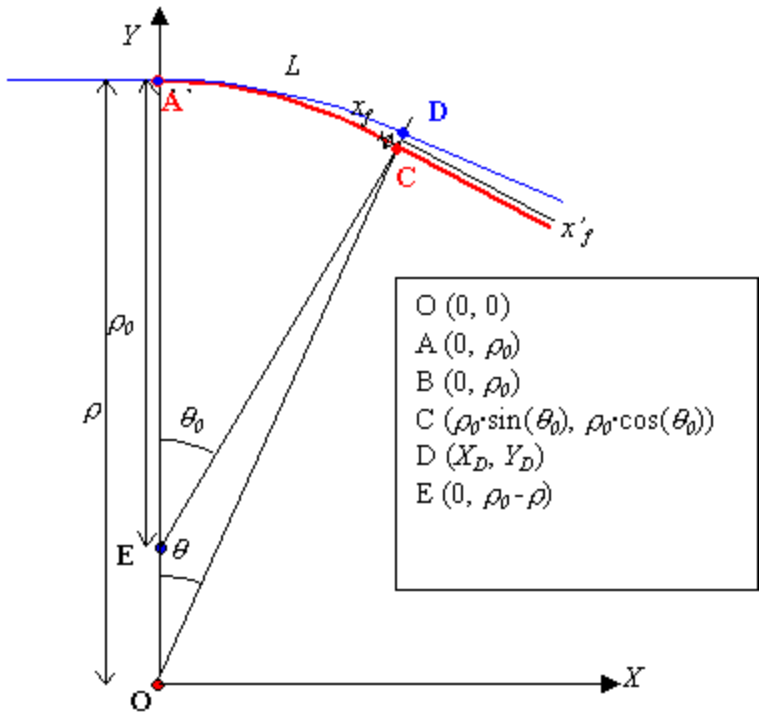
one has: $d\theta = (1 - \cos\theta_0) \cdot x'$.

One obtains finally:

$$\boxed{\varphi = \frac{2\pi \cdot f_{RF}}{\beta_0 c} \cdot \rho_0 \cdot (1 - \cos\theta_0) \cdot x' = T_{5,2} \cdot x'}.$$

The other terms ($T_{3,2}$, $T_{4,2}$,$T_{6,2}$) are equal to zero.

Matrix sixth column: variation with □



Calculus of $T_{1,6}$

D is at the intersection of (O, C) and the circle with centre E and radius ρ. Its coordinates satisfy:

$$\begin{cases} X_D = \tan(\theta_0)\cdot Y_D \\ X_D^2 + (Y_D - (\rho_0 - \rho))^2 = \rho^2 \end{cases}.$$

Giving:

$$(1+\tan^2\theta_0)\cdot Y_D^2 + 2\cdot d\rho\cdot Y_D - \rho_0\cdot(\rho_0 + 2\cdot d\rho) = 0$$

with: $\rho = \rho_0 + d\rho = \rho_0\cdot\left(1+\frac{d\rho}{\rho_0}\right)$.

As: $\rho = \frac{p}{q\cdot B}$, one has: $\frac{d\rho}{\rho_0} = \delta$.

The solution of (X) is obtained giving a reduced discriminator:

$$\Delta' = d\rho^2 + \frac{\rho_0\cdot(\rho_0 + 2\cdot d\rho)}{\cos^2\theta_0}.$$

A first order development in $\frac{d\rho}{\rho_0} << 1$ gives:

$$\Delta' = \frac{\rho_0^2}{\cos^2\theta_0}\cdot\left[1+2\cdot\frac{d\rho}{\rho_0}\right]$$

This gives the coordinates of D:

$$\begin{cases} X_D = \rho_0 \cdot \sin\theta_0 \cdot \left(1 + (1-\cos\theta_0)\cdot\dfrac{d\rho}{\rho_0}\right) \\ Y_D = \rho_0 \cdot \cos\theta_0 \cdot \left(1 + (1-\cos\theta_0)\cdot\dfrac{d\rho}{\rho_0}\right) \end{cases}$$

The final position of the particle in the moving frame is then:

$$\boxed{x_f = \frac{Y_D - Y_C}{\cos\theta_0} = \rho_0 \cdot (1-\cos\theta_0)\cdot\delta = T_{1,6}\cdot\delta}$$

Calcul of $T_{2,6}$

$$x'_f = \tan(\theta_0 - \theta) = \frac{\tan\theta_0 - \tan\theta}{1 + \tan\theta_0 \cdot \tan\theta}$$

wih: $\tan\theta = \dfrac{X_D - X_E}{Y_D - Y_E}$.

At first order, one has:

$$\tan\theta = \tan\theta_0 \cdot \left(1 - \frac{d\rho}{\rho_0 \cos\theta_0}\right),$$

giving :

$$\boxed{x'_f = \sin\theta_0 \cdot \delta = T_{5,5}\cdot\delta}$$

Calcul of $T_{5,6}$

$$\varphi = \frac{2\pi \cdot f_{RF}}{\beta_0 c}(L - \rho_0\theta_0) = \frac{2\pi \cdot f_{RF}}{\beta_0 c}\rho_0\theta_0 \cdot \left(\frac{d\rho}{\rho_0} + \frac{d\theta}{\theta_0}\right).$$

with: $d\theta = \cos^2\theta_0 \cdot d(\tan\theta) = \cos^2\theta_0 \cdot (\tan\theta - \tan\theta_0) = -\sin\theta_0 \cdot \dfrac{d\rho}{\rho_0}$,

One obtains finally:

$$\boxed{\varphi = \frac{2\pi \cdot f_{RF}}{\beta_0 c} \cdot \rho_0\theta_0 \cdot \left(1 - \frac{\sin\theta_0}{\theta_0}\right)\cdot\delta = T_{5,6}\cdot\delta}.$$

The terms ($T_{3,6}$, $T_{4,6}$) are equal to zero, the term $T_{6,6}$ is equal to 1.

Matrix fifth column: variation with φ

The output position, slope, energy do not depend on the input phase φ:
The terms ($T_{1,5}$, $T_{2,5}$, $T_{3,5}$, $T_{4,5}$, $T_{6,5}$) are equal to zero, the term $T_{5,5}$ is equal to 1.

Matrix third and fourth columns: motion in y (or Z)

The equation of motion along Z direction is:

$$\frac{dp_Z}{dt} = -q \cdot \frac{p_y}{\gamma m} \cdot B_X = -q \cdot \frac{p_y}{\gamma m} \cdot k \cdot Z \,.$$

For the matrix calculation, one uses a first order development of the force, giving:

$$\frac{dp_Z}{dt} = -q \cdot \frac{p_0 \cdot \sin\theta_0}{\gamma m} \cdot k \cdot Z \,.$$

This equation is the classical one in a quadrupole with gradient: $k \cdot \sin\theta_0$.
The associated matrix coefficients are:

$$T_{3,3} = T_{4,4} = \cos\left(\sqrt{K}\rho_0\theta_0\right),$$
$$T_{3,4} = \frac{\sin\left(\sqrt{K}\rho_0\theta_0\right)}{\sqrt{K}},$$
$$T_{4,3} = -\sqrt{K} \cdot \sin\left(\sqrt{K}\rho_0\theta_0\right),$$

with: $K = \frac{k \cdot \sin\theta_s}{p_0}$.
The other coefficients are equal to 0.

Alpha magnet matrix

The final matrix of a fraction of a alpha magnet (on which, X_s and θ_s are kept almost constant) :

$$\begin{pmatrix}
\cos\theta_0 + (1-\cos\theta_0)\cdot\frac{\rho_0\cdot\sin\theta_s}{X_s} & \rho_0\cdot\sin\theta_s & 0 & 0 & 0 & \rho_0\cdot(1-\cos\theta_0) \\
-\frac{\sin\theta_0}{\rho_0}\cdot\left(1-\frac{\rho_0\cdot\sin\theta_s}{X_s}\right) & \cos\theta_0 & 0 & 0 & 0 & \sin\theta_0 \\
0 & 0 & \cos\left(\sqrt{K}\rho_0\theta_0\right) & \frac{\sin\left(\sqrt{K}\rho_0\theta_0\right)}{\sqrt{K}} & 0 & 0 \\
0 & 0 & -\sqrt{K}\cdot\sin\left(\sqrt{K}\rho_0\theta_0\right) & \cos\left(\sqrt{K}\rho_0\theta_0\right) & 0 & 0 \\
K_\varphi\cdot\left(\sin\theta_0 + (\theta_0-\sin\theta_0)\cdot\frac{\rho_0\sin\theta_s}{X_s}\right) & K_\varphi\cdot\rho_0\cdot(1-\cos\theta_0) & 0 & 0 & 1 & K_\varphi\cdot\rho_0\cdot(\theta_0-\sin\theta_0) \\
0 & 0 & 0 & 0 & 0 & 1
\end{pmatrix}$$

with: $K = \frac{k \cdot \sin\theta_s}{X_s}$,

and: $K_\varphi = \frac{2\pi \cdot f_{RF}}{\beta_0 c}$.

The matrix of the full element is a product of all matrixes for varying X_s and θ_s.

Explanation about the way to obtain this matrix

Dynamics calculations

General description

Usual mathematics formulas

$$\gamma = \frac{W}{mc^2} + 1 \qquad \gamma^{-2} = 1 - \beta^2 \qquad \lambda = \frac{c}{f} \quad \frac{dp}{p} = \frac{\gamma}{\gamma + 1} \frac{dW}{W}$$

γ and β are the usual relativistic parameters, λ and f are respectively the free-space wavelength and the frequency of the RF. W is the kinetic energy, mc^2 the mass and p the momentum of a particle.

Description

If all the forces are considered linear, the phase-space coordinates of a particle at location s_2 can be deduced from those at the location s_1 along a transport system, by a single matrix multiplication:

$$\vec{x}(s_2) = R \cdot \vec{x}(s_1),$$

Where $\vec{x}(s)$ is a vector representing the particle position in the phase-space: $\vec{x} = \begin{pmatrix} x \\ x' = dx/ds \\ y \\ y' = dy/ds \\ z \\ \delta = \Delta p / p_s \end{pmatrix}$,

x, y and z being respectively the horizontal, vertical and longitudinal position of the particle in the bunch (relative to a synchronous particle). p_s is the synchronous particle momentum, and $\delta=(p-p_s)/p_s$, with p being the particle momentum.
R is the 6×6 transfer matrix between s_1 and s_2. In TraceWin this matrix is partitioned into 2x2 matrices to simplify and accelerate the calculations.

$$R = \left[\begin{array}{cc:cc:cc} a_{00} & a_{01} & a_{02} & a_{03} & a_{04} & a_{05} \\ a_{10} & a_{11} & a_{12} & a_{13} & a_{14} & a_{15} \\ \hdashline a_{20} & a_{21} & a_{22} & a_{23} & a_{24} & a_{25} \\ a_{30} & a_{31} & a_{32} & a_{33} & a_{34} & a_{35} \\ \hdashline a_{40} & a_{41} & a_{42} & a_{43} & a_{44} & a_{45} \\ a_{50} & a_{51} & a_{52} & a_{53} & a_{54} & a_{55} \end{array}\right] = \begin{bmatrix} R_{xx} & R_{xy} & R_{xz} \\ R_{yx} & R_{yy} & R_{yz} \\ R_{zx} & R_{zy} & R_{zz} \end{bmatrix}.$$

In order to be able to use this formalism, the space charge force is considered as linear. To calculate the space-charge effect, the real beam is replaced by an equivalent uniform beam having identical rms properties (sizes and emittances). The total emittance of the equivalent uniform bunched beam in each phase plane is then 5 times the rms emittance, and its envelope size is $\sqrt{5}$ times its rms size. For a continuous beam these factors are 4 for the emtittance and $2(=\sqrt{4})$ for envelope.

Twiss parameters

Lets define $\langle w \rangle$ as the mean value of the w particle property over the beam at location *s*.

The beam phase-space position is defined by: $\begin{pmatrix} \bar{x} = \langle x \rangle \\ \overline{x'} = \langle x' \rangle \\ \bar{y} = \langle y \rangle \\ \overline{y'} = \langle y' \rangle \\ \bar{z} = \langle z \rangle \\ 0 \end{pmatrix}$

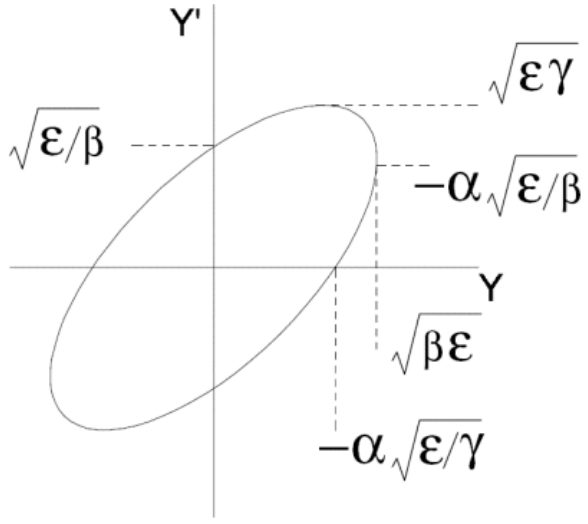
The beam rms sizes are defined by: $\tilde{w} = \sqrt{\langle (w - \bar{w})^2 \rangle}$, with w used for x, x', y, y', z or δ .

The beam rms correlation: $\overline{wv} = \langle (w - \bar{w}) \cdot (v - \bar{v}) \rangle$, with w and v used for x, x', y, y', z or δ.

The beam rms unnormalized emittances: $\tilde{\varepsilon}_w = \sqrt{\tilde{w}^2 \, \tilde{w}'^2 - \langle w w' \rangle^2}$ with w used for x, y and z.

Note: In the preceding definition, *z'* is defined as $z' = (v_z - v_{zs})/v_{zs}$, v_z and $v_{z\delta}$ being the longitudinal velocities of respectively the particle and the synchronous particle. Most of the time, δ is used rather than *z'*. In that last case, the emittance is defined by the conversion from ε_z to $\varepsilon_{z\delta}$.
In case of linear forces, the beam can be represented in sub phase planes by ellipses whose equation can be written: $\gamma_w w^2 + 2\alpha_w w w' + \beta_w w'^2 = \varepsilon_w$, where :

ε_w is the unnormalized beam effective emittance (which is the full emittance of a homogenous beam) define as 5 times the rms-emittance for bunched beam and 4 times that of a continuous beam.
$\alpha_w = -\overline{ww'}/\tilde{\varepsilon}_w$, $\beta_w = \tilde{w}^2/\tilde{\varepsilon}_w$, and $\gamma_w = \tilde{w}'^2/\tilde{\varepsilon}_w$ are the beam Twiss parameters satisfying the relationship: $\beta_w \gamma_w - \alpha_w^2 = 1$.



The beam can be represented by a matrix, called the σ-matrix defined as:

$$[\sigma] = 5 \cdot \begin{bmatrix} \widetilde{x}^2 & \overline{xx'} & \overline{xy} & \overline{xy'} & \overline{xz} & \overline{x\delta} \\ \overline{xx'} & \widetilde{x'}^2 & \overline{x'y} & \overline{x'y'} & \overline{x'z} & \overline{x'\delta} \\ \overline{xy} & \overline{x'y} & \widetilde{y}^2 & \overline{yy'} & \overline{yz} & \overline{y\delta} \\ \overline{xy'} & \overline{x'y'} & \overline{yy'} & \widetilde{y'}^2 & \overline{y'z} & \overline{y'\delta} \\ \overline{xz} & \overline{x'z} & \overline{yz} & \overline{y'z} & \widetilde{z}^2 & \overline{z\delta} \\ \overline{x\delta} & \overline{x'\delta} & \overline{y\delta} & \overline{y'\delta} & \overline{z\delta} & \widetilde{\delta}^2 \end{bmatrix}.$$

The evolution of the *σ -matrix* along the line from s_1 to s_2 can be calculated with the transfer matrix *R*:

$$[\sigma](s_2) = R \cdot [\sigma](s_2) \cdot R^T,$$

Where R^T is the transpose of *R* and *[σ]* is the beam σ-matrix. Like with the transfer matrixes the *σ-matrixes* can be partitioned into 2x2 matrices:

$$[\sigma] = \begin{bmatrix} \sigma_{xx} & \sigma_{xy} & \sigma_{xz} \\ \sigma_{yx} & \sigma_{yy} & \sigma_{yz} \\ \sigma_{zx} & \sigma_{zy} & \sigma_{zz} \end{bmatrix}.$$

The elements are divided into small steps, whose transfer matrixes are used to transport the beam *σ-matrix*. The space-charge effect is applied at each step.

Definition of the matched beam

The 2x2 extracted *σ-matrix* can be written in terms of Twiss parameters.

$$\sigma_{ww} = \begin{bmatrix} \beta_w \varepsilon_w & -\alpha_w \varepsilon_w \\ -\alpha_w \varepsilon_w & \gamma_w \varepsilon_w \end{bmatrix}.$$

We observe that $\det(\sigma_{ww}) = (\beta_w \gamma_w - \alpha_w^2) \cdot \varepsilon_w^2 = \varepsilon_w^2$

Let's *R* be the transfer matrix of a lattice of a periodic structure: $R = \begin{bmatrix} R_{xx} & R_{xy} & R_{xz} \\ R_{yx} & R_{yy} & R_{yz} \\ R_{zx} & R_{zy} & R_{zz} \end{bmatrix}$.

The 2x2 extracted diagonal matrix can be written:

$$R_{ww} = \begin{bmatrix} \cos(\sigma_{w0}) + \alpha_{w0} \sin(\sigma_{w0}) & \beta_{w0} \sin(\sigma_{w0}) \\ -\gamma_{w0} \sin(\sigma_{w0}) & \cos(\sigma_{w0}) - \alpha_{w0} \sin(\sigma_{w0}) \end{bmatrix},$$

Where α_{wo}, β_{wo}, and γ_{wo} are the Twiss parameters of the beam matched to the lattice, and σ_{wo} is the zero-current phase advance per lattice in the *[w-w']* phase plane.

Twiss parameters and acceleration

In case of acceleration, the determinant of the transfer matrix is not equal to 1, and the matrix cannot be written with the Twiss parameters, as defined before. To extract the Twiss parameters of the matched beam from the transfer matrix when there is acceleration, we use the matrix defined as below:

$$R' = \left[\begin{array}{cc:cc:cc} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & \beta_o\gamma_o & 0 & 0 & 0 & 0 \\ \hdashline 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \beta_o\gamma_o & 0 & 0 \\ \hdashline 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & \beta_o\gamma_o \end{array}\right] \cdot R \cdot \left[\begin{array}{cc:cc:cc} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1/\beta_i\gamma_i & 0 & 0 & 0 & 0 \\ \hdashline 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/\beta_i\gamma_i & 0 & 0 \\ \hdashline 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1/\beta_i\gamma_i \end{array}\right],$$

Where β_i and γ_i are the relativistic parameters at the input and β_o and γ_o are the relativistic parameters at the output of the lattice. Now, the R' matrix determinant equals 1.
The Twiss parameters of the matched beam under acceleration conditions can then be deduced from the R' matrix:

$$\sigma_{x0} = \cos^{-1}\left(\frac{r'_{00}+r'_{11}}{2}\right), \quad \sigma_{y0} = \cos^{-1}\left(\frac{r'_{22}+r'_{33}}{2}\right), \quad \sigma_{z0} = \cos^{-1}\left(\frac{r'_{44}+r'_{55}}{2}\right),$$

$$\alpha_{x0} = \frac{(r'_{00}-r'_{11})}{2\sin(\sigma_{x0})}, \quad \alpha_{y0} = \frac{(r'_{22}-r'_{33})}{2\sin(\sigma_{y0})}, \quad \alpha_{z0} = \frac{(r'_{44}-r'_{55})}{2\sin(\sigma_{z0})},$$

$$\beta_{x0} = \beta_s\gamma_s\frac{r'_{01}}{\sin(\sigma_{x0})}, \quad \beta_{y0} = \beta_s\gamma_s\frac{r'_{23}}{\sin(\sigma_{y0})}, \quad \beta_{z0} = \beta_s\gamma_s^3\frac{r'_{45}}{\sin(\sigma_{z0})},$$

$$\gamma_{x0} = \frac{-r'_{10}}{\beta_s\gamma_s\sin(\sigma_{x0})}, \quad \gamma_{y0} = \frac{-r'_{32}}{\beta_s\gamma_s\sin(\sigma_{y0})}, \quad \gamma_{z0} = \frac{-r'_{54}}{\beta_s\gamma_s^3\sin(\sigma_{z0})},$$

Where, r'_{ij} is a R' matrix coefficient (i^{th} row, j^{th} column) and α_{wo}, β_{wo}, and γ_{wo} are output Twiss parameters of the matched beam in the *[w-w']* phase plane.

Conversions between *[z-z']*, *[z-δ]* and *[Δφ-ΔW]* phase planes

β and γ being the beam reduced velocity and energy, λ the RF wavelength in vacuum, mc^2 the particle rest energy, we have in the paraxial approximation conditions the following relationship between parameters:

$$\Delta\varphi = -\frac{360°}{\beta\lambda}\cdot z,$$

But if the beam shows a divergence this relation becomes:

$$\Delta\varphi = -\frac{360°}{\beta\lambda}\cdot\sqrt{1+\frac{(\Delta x')^2}{4}+\frac{(\Delta y')^2}{4}}z$$

$\Delta\varphi$ and z being the RF phase and the position of a beam particle relative to the synchronous one.

$$\Delta W = \beta^2\gamma^3 mc^2 \cdot z' = \beta^2\gamma\, mc^2 \cdot \delta ,$$

In the general case with a synchronous particle and a generator particle these relation become:

$$\delta = \frac{\Delta W\gamma_G + (\gamma_G - 1)(m_S c^2 - m_G c^2)}{m_G c^2(\gamma_G^2 - 1)} \qquad \delta = z'\gamma_G^2 + \frac{(m_S c^2 - m_G c^2)}{m_G c^2}$$

ΔW, z' and δ being the energy, velocity and momentum of a beam particle relative to the synchronous one.

$$\varepsilon_w = \frac{360° \cdot mc^2}{\lambda} \cdot \varepsilon_{zn} = \frac{360° \cdot mc^2}{\lambda} \cdot \beta\gamma^3 \cdot \varepsilon_z = \frac{360° \cdot mc^2}{\lambda} \cdot \beta\gamma \cdot \varepsilon_{z\delta} ,$$

εw and $\varepsilon z\delta$ being the normalized longitudinal emittances, εz and $\varepsilon z\delta$ being the unnormalized longitudinal emittances of the beam in respectively the *[z-z']* and the *[z- δ]* phase planes.

$$\beta_w = \frac{360°}{mc^2\gamma^3\beta^3\lambda} \cdot \beta_z = \frac{360°}{mc^2\gamma\,\beta^3\lambda} \cdot \beta_{z\delta} ,$$

βw, βz and $\beta z\delta$ are the β-Twiss parameters of the beam in respectively the *[Δφ-ΔW], [z-z']* and *[z-δ]* phase planes.

$$\alpha_w = \alpha_z = \alpha_{z\delta} ,$$

αw, αz and $\alpha z\delta$ are the α-Twiss parameters of the beam in respectively *the [Δφ-ΔW], [z-z']* and *[z-δ]* phase planes.

$$\gamma_w = \frac{mc^2\gamma^3\beta^3\lambda}{360°} \cdot \gamma_z = \frac{mc^2\gamma\,\beta^3\lambda}{360°} \cdot \gamma_{z\delta} ,$$

γw, γz and $\gamma z\delta$ are the γ-Twiss parameters of the beam in respectively the *[Δφ-ΔW], [z-z']* and *[z-δ]* phase planes.

Normalization of the emittance

$$\varepsilon_{tn} = \beta\gamma \cdot \varepsilon_t ,$$

$$\varepsilon_{zn} = \beta\gamma^3 \cdot \varepsilon_z = \beta\gamma \cdot \varepsilon_{z\delta} .$$

Beta X&Y function

The betaX and betaY function plotted in envelope charts are defined as following:

$$\beta_{xx'} = \frac{\sigma_{11} - \sigma_{16} \cdot T_{16}}{\varepsilon_{xx'}} \text{ and } \beta_{yy'} = \frac{\sigma_{11} - \sigma_{16} \cdot T_{16}}{\varepsilon_{yy'}}$$

σ, the beam matrix and T, the transfer matrix.

Energy gain and synchronous phase

For a rf cavity with a length *L*, the energy gain for a charged particle which traverses the cavity on the axis with an amplitude profile of the longitudinal component $E_z(s)$
can be calculated with the formula

$$\Delta W = \int_{S_0}^{S_0+L} qE_z(s)\cdot\cos[\phi(s)]\cdot ds \qquad \text{(A1)}$$

with *q* the charge of the particle, and *s* the beam axis coordinate. The function $\phi(s)$ is the rf phase when the particle is at the coordinate *s*. It is defined by

$$\phi(s) = \phi_0 + \frac{\omega_{rf}}{c}\int_{s_0}^{s_0+s}\frac{ds'}{\beta_z(s')}. \qquad \text{(A2)}$$

With *c* the Einstein constant, w_{rf} the rf pulsation, ϕ_0 is the rf phase when the particle is at the cavity entrance, and $\beta_z(s')$ is the longitudinal component of the particle reduced speed at the *s'* location. Writing $\phi(s)+\phi_s-\phi_s$, with ϕ_s being an arbitrary phase and using trigonometric relations, we found that the energy gain can be written

$$\Delta W = \cos(\phi_s)\int_{s_0}^{s_0+L} qE_z(s)\cdot\cos[\phi(s)-\phi_s]\cdot ds - \sin(\phi_s)\int_{S_0}^{s_0+L} qE_z(s)\cdot\sin[\phi(s)-\phi_s]\cdot ds$$

(A3)

We can define ϕ_s such as

$$\int_{S_0}^{S_0+L} qE_z(s)\cdot\sin[\phi(s)-\phi_s]\cdot ds = 0 \qquad \text{(A4)}$$

It gives

$$\phi_s = \arctan\left[\frac{\int_{S_0}^{S_0+L} qE_z(s)\cdot\sin[\phi(s)]\cdot ds}{\int_{S_0}^{S_0+L} qE_z(s)\cdot\cos[\phi(s)]\cdot ds}\right] \qquad \text{(A5)}$$

Then the energy gain can be rewritten:

$$\Delta W = \left[q\int_{S_0}^{S_0+L}|E_z(s)|\cdot ds\right]\cdot T\cdot\cos\phi_s = qV_0\cdot T\cdot\cos\phi_s \qquad \text{(A6)}$$

With

$$T = \frac{1}{V_0}\int_{s_0}^{s_0+L} qE_z(s)\cdot\cos[\phi(s)-\phi_s]\cdot ds \qquad \text{(A7)}$$

T is known as the transit time factor. It depends on the speed of the particle and the field amplitude. *A crucial point is that no hypothesis about the field profile in the cavity has been made to develop these formulas*. This general approach can then be used for a standing wave resonator with a fixed or not fixed geometrical β. This definition of the synchronous phase is different compared to the one which

can be found in the literature which is the rf phase when the particle reaches the middle of the cavity. This definition and our definition correspond exactly when the cavity is symmetric and the speed variation can be neglected. It has to be noticed that, in fixed _ cavities, there is no synchronism between cells, but this redefinition of the phase allows us to use the classical formalism [1]. This phase can then be called "effective" phase or "reference" phase. With this last hypothesis, the parameter *T* depends only on the average speed of the particle in the cavity. The calculation of *T* with Eq. (A7) may be difficult because it is required to know the value of β_s. But when the speed gain is weak enough, this dependence can be neglected and it is possible to use another equation for the transit time factor: *T*.

$$T = \frac{1}{V_0}\left|\int_{s_0}^{s_0+L} qE_z(s)\cdot e^{j\phi(s)}\cdot ds\right| \qquad \text{(A8)}$$

[1] P. Lapostolle and M. Weiss, CERN-PS-2000-001-DR,2000.

Transit time factor definition

These following definitions are used in all accelerating element.
T: is the usual time factor transit given by SUPERFISH.
T': is the *T'* from SUPERFISH time *-2π*
T'': is the *T''* from SUPERFISH time *-4 π²*

$$T(\beta) = T_s + kT_s'\cdot(1-\kappa) + k^2T_s''\cdot\frac{(1-\kappa)^2}{2},$$

$$kT'(\beta) = kT_s' + k^2T_s''\cdot(1-\kappa) \ldots \qquad \text{With } \kappa = \frac{k_s}{k} = \frac{\beta}{\beta_s}.$$

The electric field is corrected according to *T(β)*.
Coordinate transformations are given in Wangler's book page 202:

Phase advance definition

TraceWin calculates the particle phase advance in two ways. The first one is the extraction of the phase advance from the transfer matrix of the lattice (μ or σ_0). The second one is done by the beta function integration along a lattice giving the phase advance with (σ) or without σ_0 space charge:

$\sigma_x = \int_0^L \frac{dl}{\beta(l)}$ with L is the lattice length and $\beta = \frac{< x^2 >}{\varepsilon_x}$.

x is the beam RMS or effective size and ε_x the unnormalized RMS or effective emittance.
The first phase advance type can be plotted from the "***Phase advance"->"Structure"***.of "***Chart***" page
And the second one can be plotted from the "***Phase advance"->"Beam"*** of "***Chart***" page

Residual orbit

Use in the error studies in order to know the beam gravity evolution, It's defined like below:

$$\bar{x} = \frac{1}{N}\sum_{1}^{N} x \qquad x_{rms} = \sqrt{\frac{1}{N}\sum_{1}^{N} x^2 - \bar{x}^2}$$

$$\bar{y} = \frac{1}{N}\sum_{1}^{N} y \qquad y_{rms} = \sqrt{\frac{1}{N}\sum_{1}^{N} y^2 - \bar{y}^2}$$

$$\bar{r} = \sqrt{\bar{x}^2 + \bar{y}^2} \qquad r_{rms} = \sqrt{\frac{1}{N}\sum_{1}^{N} r^2 - \bar{r}^2}$$

x & y are the beam gravity position and *N* is the number of run or linac.

Halo definition

If the motion is uncoupled between phase planes, the following quantities are kinematic invariants of motion:

$$I_2^i \equiv \left\langle q_i^2 \right\rangle \left\langle p_i^2 \right\rangle - \left\langle q_i\, p_i \right\rangle^2$$

$$I_4^i \equiv \left\langle q_i^4 \right\rangle \left\langle p_i^4 \right\rangle + 3\left\langle q_i^2\, p_i^2 \right\rangle^2 - 4\left\langle q_i p_i^3 \right\rangle \left\langle q_i^3 p_i \right\rangle$$

Then we define the halo intensity parameter, *Hi*, in the *i*th phase plane, as

$$H_i \equiv \frac{\sqrt{3I_4^i}}{2I_2^i} - 2 = \frac{\sqrt{3\left\langle q_i^4 \right\rangle \left\langle p_i^4 \right\rangle + 9\left\langle q_i^2\, p_i^2 \right\rangle^2 - 12\left\langle q_i\, p_i^3 \right\rangle \left\langle q_i^3\, p_i \right\rangle}}{2\left\langle q_i^2 \right\rangle \left\langle p_i^2 \right\rangle - 2\left\langle q_i\, p_i \right\rangle^2}$$

From ref: "PHYSICAL REVIEW SPECIAL TOPICS - ACCELERATORS AND BEAMS,VOLUME 5, 124202 (2002)", "Beam halo definitions based upon moments of the particle distribution (C. K. Allen and T. P.Wangler) "

Space charge

In the dynamics the space charge is calculated at each step. The number of calculation and obviously the calculation time depend on the parameter "***Step of calculation per βλ***". The space-charge calculation as it is done in TraceWin is presented in this paragraph.
In a free space, the motion equation of a particle, feeling only space-charge force, can be written along one direction in the **beam frame *R****:

$$\frac{d^2 w^*}{dt^{*2}} = F_w^*,$$

F_w^* being the *w* component of the space-charge force (divided by the particle rest mass).

All the quantities with a star (*) are expressed in R^*, all quantities without star are expressed in the **laboratory frame *R***. We have, from the Lorentz transform:

$dt^* = dt/\gamma$, $x^* = x$, $y^* = y$, $z^* = \gamma \cdot z$,

And:

$ds = \beta c \cdot dt$.

The derivation with *s*, the beam longitudinal position used as independent variable gives:

$$\frac{d^2 w^*}{ds^2} = F_w = \frac{F_w^*}{(\gamma\beta c)^2}.$$

In the longitudinal direction, we have:

$$\frac{d\delta}{ds} = \gamma^2 \frac{d^2 z}{ds^2} = \gamma \frac{d^2 z^*}{ds^2} = \gamma \frac{F_w^*}{(\gamma\beta c)^2} = \gamma F_w.$$

DC beam

When the beam is continuous, there is no space-charge force acting along the longitudinal direction. We have:

$$F_x = \frac{2K}{a_x(a_x + a_y)} \cdot x, \qquad F_y = \frac{2K}{a_y(a_x + a_y)} \cdot y,$$

With $K = \frac{|q| \cdot I}{\pi\varepsilon_0 mc^3\beta^3\gamma^3}$, the beam generalized perveance.

a_x, a_y, are the semiaxes of the homogeneous ellipse (2 times the rms beam sizes), I is the average beam current, ε_o is the vacuum permittivity.

Bunched beam

The *w*-component of the space-charge force acting on the particle in the bunch frame is:

$$F_w^* = \frac{K_{w^*}}{a_x^* a_y^* a_z^*} \cdot w^*,$$

Giving:

$$F_w = \frac{K_w}{a_x^* a_y^* a_z^*} \cdot \frac{1}{(\gamma\beta c)^2} \cdot w^*,$$

With: *w* for *x*, *y* or *z*,

$$K_w = \frac{3 \cdot |q| \cdot I\, \lambda\, h}{4\pi\varepsilon_0 mc} \cdot f_w(a_x^*, a_y^*, a_z^*),$$

And $f_w(a_x, a_y, a_z) = \int_0^\infty \frac{ds}{(a_w^2 + s)\sqrt{(a_x^2 + s)(a_y^2 + s)(a_z^2 + s)}}$ is the form factor such as:

$\forall$ a_x, a_y and a_z, $f_x + f_y + f_z = 1$.

a_x, a_y, a_z, are the semiaxes in the laboratory frame of the homogeneous ellipsoid ($\sqrt{5}$ times the rms beam sizes), $a_x^* = a_x$, $a_y^* = a_y$ and $a_z^* = \gamma a_z$ are the beam semiaxes along longitudinal direction in the beam frame, I is the average beam current, h is the harmonic number (ratio between the RF frequency and the bunch frequency), ε_0 is the vacuum permittivity. Note that the longitudinal beam size in the beam frame is equal to γ times the one in the laboratory frame.

Form factor calculation

The form factor integral calculation depends on the ratios a_z^* / a_x^* and a_z^* / a_y^*. If they are lower than 12 the integral is calculated by the Gauss method with a very good precision. If they are greater than 12 (which happens obviously when the beam is ultra-relativistic due to the Lorentz transformation) an expended development is used and slightly reducing the result precision.

Space-charge application

Frame change

The space charge impulse should be applied in the beam frame. Before any application, the beam *σ-matrix* should be written from the laboratory frame to the beam frame by making the transformation:

$[\sigma] = R_\gamma \cdot [\sigma] \cdot R_\gamma^T$,

With: $R_\gamma = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & \gamma & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$.

After the application of space-charge impulse (taking into account the beam coupling), the *σ-matrix* should be written back to the laboratory frame:

$[\sigma] = R_{\gamma^{-1}} \cdot [\sigma] \cdot R_{\gamma^{-1}}^T$,

With: $R_{\gamma^{-1}} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & \gamma^{-1} & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$.

Beam coupling or tilted.

Due to the elements (magnetic coil or dipole) or initial conditions, the beam ellipsoid in *[x-y-z]* space can be tilted. In this case, the beam ellipsoid (in the beam frame) must first be transformed to a coordinate system in which it is upright before applying the space-charge impulses.
If the ellipsoid is tilted in the *[x-y]* plane, the angle between the x-axis and the axis of the elliptical projection on the *[x-y]* plane is:

$\theta_{xy} = \frac{1}{2}\tan^{-1}\left(\frac{2\sigma_{02}}{\sigma_{22}-\sigma_{00}}\right)$ With σ_{ij} , σ -matrix elements.

If the ellipsoid is tilted in the *[x-z]* plane: $\theta_{xz} = \frac{1}{2}\tan^{-1}\left(\frac{2\sigma_{04}}{\sigma_{44}-\sigma_{00}}\right)$.

If the ellipsoid is tilted in the *[y-z]* plane: $\theta_{yz} = \frac{1}{2}\tan^{-1}\left(\frac{2\sigma_{24}}{\sigma_{44}-\sigma_{22}}\right)$.

The ellipsoid can be brought upright by rotations of angles $-\theta_{xy}$, $-\theta_{xz}$ and $-\theta_{yz}$ accomplished by applying the transfer matrixes:

$$R_{xy} = \left[\begin{array}{cc|cc|cc} \cos(\theta_{xy}) & 0 & \sin(\theta_{xy}) & 0 & 0 & 0 \\ 0 & \cos(\theta_{xy}) & 0 & \sin(\theta_{xy}) & 0 & 0 \\ \hline -\sin(\theta_{xy}) & 0 & \cos(\theta_{xy}) & 0 & 0 & 0 \\ 0 & -\sin(\theta_{xy}) & 0 & \cos(\theta_{xy}) & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{array}\right],$$

R_{xz} and R_{yz} can be obtained the same way.
The rotation is then applied:

$$[\sigma] = R_{yz}\cdot R_{xz}\cdot R_{xy}\cdot[\sigma]\cdot {R_{xy}}^{T}\cdot {R_{xz}}^{T}\cdot {R_{yz}}^{T},$$

$$[\sigma] = Rot\cdot[\sigma]\cdot Rot^{T}.$$

When the ellipse is upright, the space-charge impulses can be applied. The three reverse rotations can then be applied:

$$[\sigma] = {R_{xy}}^{T}\cdot {R_{xz}}^{T}\cdot {R_{yz}}^{T}\cdot[\sigma]\cdot R_{yz}\cdot R_{xz}\cdot R_{xy},$$

$$[\sigma] = Rot^{T}\cdot[\sigma]\cdot Rot.$$

Space-charge impulse

The space-charge transfer matrix R_{ce} applies on a distance Δs (the calculation step):

$$R_{ce} = \left[\begin{array}{cc|cc|cc} 1 & 0 & 0 & 0 & 0 & 0 \\ F_x\Delta s & 1 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & F_y\Delta s & 1 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & \gamma F_z\Delta s & 1 \end{array}\right].$$

The space-charge impulse is applied in the bunch frame, where the beam ellipse is upright:
$[\sigma] = R_{ce} \cdot [\sigma] \cdot R_{ce}{}^{T}$.

Finally, the total space charge effect is given by:

$$\boxed{[\sigma] = R_{\gamma^{-1}} \cdot Rot^{T} \cdot R_{ce} \cdot Rot \cdot R_{\gamma} \cdot [\sigma] \cdot R_{\gamma}{}^{T} \cdot Rot^{T} \cdot R_{ce}{}^{T} \cdot Rot \cdot R_{\gamma^{-1}}{}^{T}}.$$

Particle motion in electromagnetic field

The equations of particle motions in electromagnetic field (RF or static) defined below are mainly uses in FIED_MAP elements for each particle (Partran) and for beam centroid (Envelope) with linearization.

Equations générales de la dynamique

La variation de la quantité de mouvement $\vec{p}$ d'une particule de charge *q* et de masse *m* dans un champ électromagnétique est :

$$\frac{d\vec{p}}{dt} = q \cdot \left(\vec{E} + \frac{\vec{p}}{\gamma m} \times \vec{B} \right)$$

En coordonnées cartésiennes, l'équation se projette :

$$\begin{cases} \dot{p}_x = \dfrac{dp_x}{dt} = q \cdot \left(E_x + \dfrac{p_z}{\gamma m} \cdot \left(y' \cdot B_z - B_y \right) \right) \\ \dot{p}_y = \dfrac{dp_y}{dt} = q \cdot \left(E_y + \dfrac{p_z}{\gamma m} \cdot \left(B_x - x' \cdot B_z \right) \right) \\ \dot{p}_z = \dfrac{dp_z}{dt} = q \cdot \left(E_z + \dfrac{p_z}{\gamma m} \cdot \left(x' \cdot B_y - y' \cdot B_x \right) \right) \end{cases}$$

avec :

$$\frac{dx}{ds} = x' = \frac{p_x}{p_z} \text{ et } \frac{dy}{ds} = y' = \frac{p_y}{p_z}$$

Nous avons alors :

$$\frac{dx'}{ds} = \frac{d(p_x / p_z)}{dt} \cdot \frac{dt}{ds} = \frac{1}{\beta_z c} \cdot \frac{\dot{p}_x - \dot{p}_z \cdot x'}{p_z}$$

Ce qui donne :

$$\frac{dx'}{ds} = \frac{1}{\gamma \beta_z^2 \left(mc^2 / q \right)} \cdot \left(E_x - x' \cdot E_z + \beta_z c \cdot \left(x'y' \cdot B_x - \left(1 + x'^2 \right) \cdot B_y + y' \cdot B_z \right) \right)$$

De même nous avons :

$$\frac{dy'}{ds} = \frac{1}{\gamma \beta_z^2 \left(mc^2 / q \right)} \cdot \left(E_y - y' \cdot E_z + \beta_z c \cdot \left(\left(1 + y'^2 \right) \cdot B_x - x'y' \cdot B_y - x' \cdot B_z \right) \right)$$

En ce qui concerne la dynamique longitudinale, les variables utilisées sont généralement soit l'énergie cinétique *T* soit la quantité de mouvement *p* de la particule. La variation de ces grandeurs est déduite de la relation :

$$p^2 = p_x^2 + p_y^2 + p_z^2 = p_z^2 \cdot \left(1 + x'^2 + y'^2\right) = \frac{\left(E_0 + T\right)^2 - E_0^2}{c^2} .$$

Donnant :

$$\frac{dp}{ds} = \frac{\frac{dp_z}{ds} \cdot \left(1 + x'^2 + y'^2\right) + p_z \cdot \left(x' \cdot \frac{dx'}{ds} + y' \cdot \frac{dy'}{ds} \right)}{\sqrt{1 + x'^2 + y'^2}}$$

$$p^2 = p_x^2 + p_y^2 + p_z^2 = p_z^2 \cdot \left(1 + x'^2 + y'^2\right) = \frac{\left(E_0 + T\right)^2 - E_0^2}{c^2}$$

soit :

$$\frac{dp}{ds} = \frac{q}{\beta \cdot c} \cdot \left(E_z + x' \cdot E_x + y' \cdot E_y\right)$$

et :

$$\frac{dT}{ds} = q \cdot \left(E_z + x' \cdot E_x + y' \cdot E_y\right)$$

Dans le code PIC, ces équations doivent être intégrées pas à pas.

Linéarisation

Dans le cadre de l'hypothèse de paraxialité, la linéarisation des équations nous donne :

$$\frac{dx'}{ds} = \frac{1}{\gamma\beta^2\left(mc^2/q\right)} \cdot \left(E_x - x' \cdot E_z + \beta c \cdot \left(-B_y + y' \cdot B_z\right)\right)$$

$$\frac{dy'}{ds} = \frac{1}{\gamma\beta^2\left(mc^2/q\right)} \cdot \left(E_y - y' \cdot E_z + \beta c \cdot \left(B_x - x' \cdot B_z\right)\right)$$

$$\frac{dp}{ds} = \frac{1}{\beta \cdot c} \cdot \left(E_z + x' \cdot E_x + y' \cdot E_y\right)$$

Dans TraceWin, les variables longitudinales sont :

$$z = -\frac{\varphi}{2\pi} \cdot \beta\lambda \quad \text{et} \qquad \delta = \frac{p - p_s}{p_s}$$

p_s est la quantité de mouvement de la particule synchrone.
L'évolution de δ est alors :

$$\frac{d\delta}{ds} = \frac{1}{p_s} \cdot \left(\frac{dp}{ds} - \left(1 + \delta\right) \cdot \frac{dp_s}{ds} \right) \frac{dp}{ds} = \frac{1}{\beta \cdot c} \cdot \left(E_z + x' \cdot E_x + y' \cdot E_y\right)$$

Les champs peuvent être modélisés par :

$$E_x = E_{x0} + \frac{\partial E_x}{\partial x} \cdot x + \frac{\partial E_x}{\partial y} \cdot y + \frac{\partial E_x}{\partial z} \cdot z \ \ldots$$

$$B_x = B_{x0} + \frac{\partial B_x}{\partial x} \cdot x + \frac{\partial B_x}{\partial y} \cdot y + \frac{\partial B_x}{\partial z} \cdot z \ \ldots$$

Les équations du mouvement linéarisées deviennent alors :

$$\frac{dx'}{ds}=\frac{q}{\gamma_s\beta_s^2mc^2}\cdot\begin{pmatrix}\left(E_{x0}-\beta_sc\cdot B_{y0}\right)-E_{z0}\cdot x'+\beta_sc\cdot B_{z0}\cdot y'-\left[\left(2-\beta_s^2\right)\cdot E_{x0}+\beta_sc\cdot B_{y0}\right]\cdot\delta\\ \left(\frac{\partial E_x}{\partial x}-\beta_sc\cdot\frac{\partial B_y}{\partial x}\right)\cdot x+\left(\frac{\partial E_x}{\partial y}-\beta_sc\cdot\frac{\partial B_y}{\partial y}\right)\cdot y+\left(\frac{\partial E_x}{\partial z}-\beta_sc\cdot\frac{\partial B_y}{\partial z}\right)\cdot z\end{pmatrix}$$

$$\frac{dy'}{ds}=\frac{q}{\gamma_s\beta_s^2mc^2}\cdot\begin{pmatrix}\left(E_{y0}+\beta_sc\cdot B_{x0}\right)-\beta_sc\cdot B_{z0}\cdot x'-E_{z0}\cdot y'-\left[\left(2-\beta_s^2\right)\cdot E_{y0}-\beta_sc\cdot B_{x0}\right]\cdot\delta\\ \left(\frac{\partial E_y}{\partial x}+\beta_sc\cdot\frac{\partial B_x}{\partial x}\right)\cdot x+\left(\frac{\partial E_y}{\partial y}+\beta_sc\cdot\frac{\partial B_x}{\partial y}\right)\cdot y+\left(\frac{\partial E_y}{\partial z}+\beta_sc\cdot\frac{\partial B_x}{\partial z}\right)\cdot z\end{pmatrix}$$

$$\frac{d\delta}{ds}=\frac{q}{\gamma_s\beta_s^2mc^2}\cdot\left(\frac{\partial E_z}{\partial x}\cdot x+\frac{\partial E_z}{\partial y}\cdot y+\frac{\partial E_z}{\partial z}\cdot z+E_{x0}\cdot x'+E_{y0}\cdot y'-\left(2-\beta_s^2\right)\cdot E_{z0}\cdot\delta\right)$$

Nous avons utilisé ici :

$$d\left(\left(\gamma\beta^2\right)^{-1}\right)=-\frac{2-\beta_s^2}{\gamma_s\beta_s^2}\cdot\delta,$$

$$d\beta=\frac{\beta_s}{\gamma_s^2}\cdot\delta,$$

$$d\left(\beta^{-1}\right)=-\frac{\delta}{\beta_s\cdot\gamma_s^2}.$$

Quelques éléments de transport

Eléments à symétrie de révolution

De nombreux éléments (solénoïdes, lentilles de Einzel, cavités RF (DTL, CCL, supra-elliptique,…)) présentent une symétrie de révolution autour de l'axe du faisceau. Dans ce cas, les champs sont représentés dans le référentiel cylindrique : (r, θ, z).

On a alors : $\begin{pmatrix}V_x\left(r\cdot\cos\theta,r\cdot\sin\theta\right)\\ V_y\left(r\cdot\cos\theta,r\cdot\sin\theta\right)\end{pmatrix}=\begin{pmatrix}\cos\theta & -\sin\theta\\ \sin\theta & \cos\theta\end{pmatrix}\cdot\begin{pmatrix}V_r\left(r,\theta\right)\\ V_\theta\left(r,\theta\right)\end{pmatrix}$

Solénoïde magnétique

Dans un solénoïde, on a, au premier ordre :

$\vec{E}=\vec{0}$,

$B_\theta=0$,

$B_r=\frac{\partial B_r}{\partial r}\cdot r$,

$B_z=B_{z0}$.

Les équations linéarisées du mouvement deviennent alors :

$$\frac{dx'}{ds}=\frac{q}{\gamma_s\beta_s^2mc^2}\cdot\left(\beta_sc\cdot B_{z0}\cdot y'-\beta_sc\cdot\frac{\partial B_r}{\partial r}\cdot y\right)$$

$$\frac{dy'}{ds}=\frac{q}{\gamma_s\beta_s^2mc^2}\cdot\left(-\beta_sc\cdot B_{z0}\cdot x'+\beta_sc\cdot\frac{\partial B_r}{\partial r}\cdot x\right)$$

Lentille électrostatique de Einzel

Dans une lentille électrostatique d'Einzel, on a, au premier ordre :

$$\vec{B} = \vec{0},$$

$$E_\theta = 0,$$

$$E_r = \frac{\partial E_r}{\partial r} \cdot r,$$

$$E_z = E_{z0}.$$

Les équations linéarisées du mouvement deviennent alors :

$$\frac{dx'}{ds} = \frac{q}{\gamma_s \beta_s^2 mc^2} \cdot \left(\frac{\partial E_r}{\partial r} \cdot x - E_{z0} \cdot x' \right)$$

$$\frac{dy'}{ds} = \frac{q}{\gamma_s \beta_s^2 mc^2} \cdot \left(\frac{\partial E_r}{\partial r} \cdot y - E_{z0} \cdot y' \right)$$

$$\frac{d\delta}{ds} = -\frac{q}{\gamma_s \beta_s^2 mc^2} \cdot \left(2 - \beta_s^2\right) \cdot E_{z0} \cdot \delta$$

A ces équations, il faut ajouter la variation de l'énergie de la particule synchrone :

$$\frac{dT_s}{ds} = q \cdot E_{z0}.$$

Cavités accélératrice RF

Dans une cavité accélératrice RF, on a :

$$E_z(r,s,t) = E_{z0}(s) \cdot \left(1 - \frac{1}{4} \cdot \left(\frac{1}{E_{z0}(s)} \cdot \frac{d^2 E_{z0}(s)}{ds^2} + \frac{\omega^2}{c^2} \right) \cdot r^2 \right) \cdot \cos(\omega \cdot t + \varphi),$$

$$E_r(r,s,t) = -\frac{1}{2} \cdot \frac{dE_{z0}(s)}{ds} \cdot \left(1 - \frac{1}{8} \cdot \left(\frac{1}{E_{z0}(s)} \cdot \frac{d^2 E_{z0}(s)}{ds^2} + \frac{\omega^2}{c^2} \right) \cdot r^2 \right) \cdot r \cdot \cos(\omega \cdot t + \varphi),$$

$$B_\theta(r,s,t) = -\frac{\omega}{2 \cdot c^2} \cdot E_{z0}(s) \cdot \left(1 - \frac{1}{8} \cdot \left(\frac{1}{E_{z0}(s)} \cdot \frac{d^2 E_{z0}(s)}{ds^2} + \frac{\omega^2}{c^2} \right) \cdot r^2 \right) \cdot r \cdot \sin(\omega \cdot t + \varphi).$$

Soit au premier ordre :

$$E_z(r,s,t) = E_{z0}(s) \cdot \cos(\omega \cdot t + \varphi),$$

$$E_r(r,s,t) = -\frac{1}{2} \cdot \frac{dE_{z0}(s)}{ds} \cdot r \cdot \cos(\omega \cdot t + \varphi),$$

$$B_\theta(r,s,t) = -\frac{\omega}{2 \cdot c^2} \cdot E_{z0}(s) \cdot r \cdot \sin(\omega \cdot t + \varphi).$$

Soient, en coordonnées cartésiennes :

$$E_x(r,s,t) = -\frac{1}{2} \cdot \frac{dE_{z0}(s)}{ds} \cdot x \cdot \cos(\omega \cdot t + \varphi) = \frac{\partial E_x}{\partial x} \cdot x,$$

$$E_y(r,s,t) = -\frac{1}{2} \cdot \frac{dE_{z0}(s)}{ds} \cdot y \cdot \cos(\omega \cdot t + \varphi) = \frac{\partial E_y}{\partial y} \cdot y,$$

$$B_x(r,s,t) = \frac{\omega}{2 \cdot c^2} \cdot E_{z0}(s) \cdot y \cdot \sin(\omega \cdot t + \varphi) = \frac{\partial B_x}{\partial y} \cdot y,$$

$$B_y(r,s,t) = -\frac{\omega}{2\cdot c^2}\cdot E_{z0}(s)\cdot x\cdot\sin(\omega\cdot t+\varphi) = \frac{\partial B_y}{\partial x}\cdot x\,.$$

Les équations linéarisées du mouvement deviennent alors :

$$\frac{dx'}{ds} = \frac{q}{\gamma_s\beta_s^2 mc^2}\cdot\left(-E_{z0}\cdot\cos\varphi\cdot x' + \left(\frac{\partial E_x}{\partial x} - \beta_s c\cdot\frac{\partial B_y}{\partial x}\right)\cdot x\right),$$

$$\frac{dy'}{ds} = \frac{q}{\gamma_s\beta_s^2 mc^2}\cdot\left(-E_{z0}\cdot\cos\varphi\cdot y' + \left(\frac{\partial E_y}{\partial y} + \beta_s c\cdot\frac{\partial B_x}{\partial y}\right)\cdot y\right),$$

$$\frac{d\delta}{ds} = \frac{q}{\gamma_s\beta_s^2 mc^2}\cdot\left(\frac{\partial E_z}{\partial z}\cdot z - \left(2-\beta_s^2\right)\cdot E_{z0}\cdot\cos\varphi\cdot\delta\right).$$

Eléments sans symétrie particulière

Quadripole magnétique

Dans un quadripôle magnétique :

$\vec{E} = \vec{0}$, $B_x = G\cdot y$ et $B_y = G\cdot x$

On obtient alors :

$$\frac{dx'}{ds} = -\frac{q}{\gamma_s\beta_s^2 mc^2}\cdot\beta_s c\cdot G\cdot x\,,$$

$$\frac{dy'}{ds} = \frac{q}{\gamma_s\beta_s^2 mc^2}\cdot\beta_s c\cdot G\cdot y\,.$$

Quadripole électrique

Dans un quadripôle electrique :

$\vec{B} = \vec{0}$, $E_x = -G\cdot x$ et $E_y = G\cdot y$

On obtient alors :

$$\frac{dx'}{ds} = \frac{q}{\gamma_s\beta_s^2 mc^2}\cdot G\cdot x$$

$$\frac{dy'}{ds} = \frac{q}{\gamma_s\beta_s^2 mc^2}\cdot G\cdot y$$

3D field development

(*Note write by Ciprian Plostinar*)

The 3D magnetic field components and their derivates for a multipole magnet in the region close to the axis can be expressed as given by the gradient of a scalar magnetic potential function, V.

The proposed solution in the literature for the scalar potential is of the form:

$$V_n(x,y,z) = (n!)^2 \left(\sum_{q=0}^{\infty} (-1)^q \frac{G^{(2q)}(z)\left(x^2+y^2\right)^q}{4^q q!(n+q)!} \right) \left(\sum_{m=0}^{n} \frac{\sin\left(m\frac{\pi}{2}\right) x^{n-m}\ \mathrm{y}^{\mathrm{m}}}{m!(\mathrm{n}-\mathrm{m})!} \right)$$

Where G(z) is the magnetic gradient along the longitudinal axis.

More explicitly, for a quadrupole (n=2), the scalar potential is:

$$V(x,y,z) = \left(G(z) - \frac{G''(z)\left(\mathrm{x}^2+y^2\right)}{12} + \frac{G''''(z)\left(\mathrm{x}^2+y^2\right)^2}{384} - \frac{G''''''(z)\left(\mathrm{x}^2+y^2\right)^3}{23040} \right) xy$$

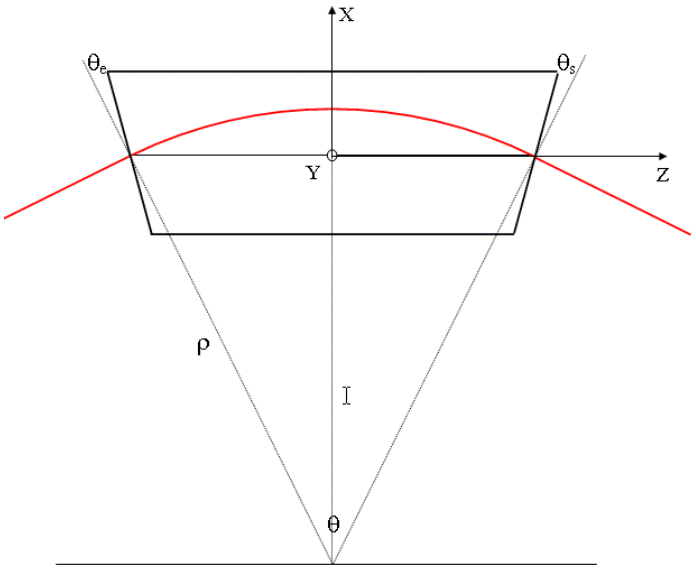
And the three magnetic field components in Cartesian coordinates are given by:

$$B_x = \frac{\partial V}{\partial x}$$
$$By = \frac{\partial V}{\partial y}$$
$$Bz = \frac{\partial V}{\partial z}$$

Bend error treatment

DY :

y_e -= *DY*
y_s += *DY*

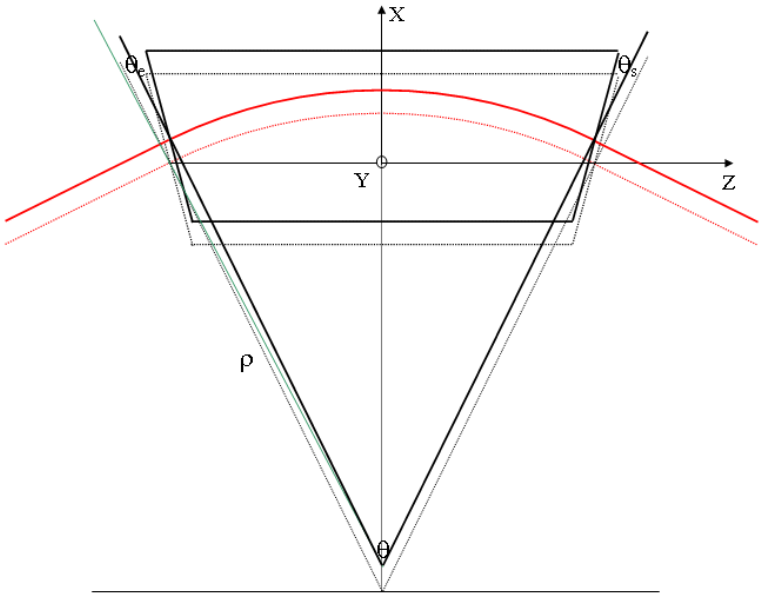


DX :

Input drift (*DX*·sin(*θ*/2)

$$\begin{cases} x_e -= DX \cdot \cos(\theta/2) \\ x_s += DX \cdot \cos(\theta/2) \end{cases}$$

Output drift (*DX*·sin(*θ* /2)

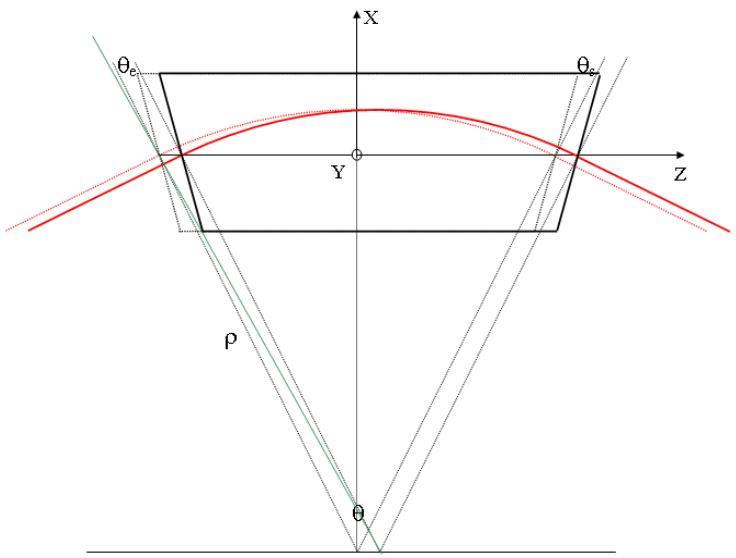


DZ :

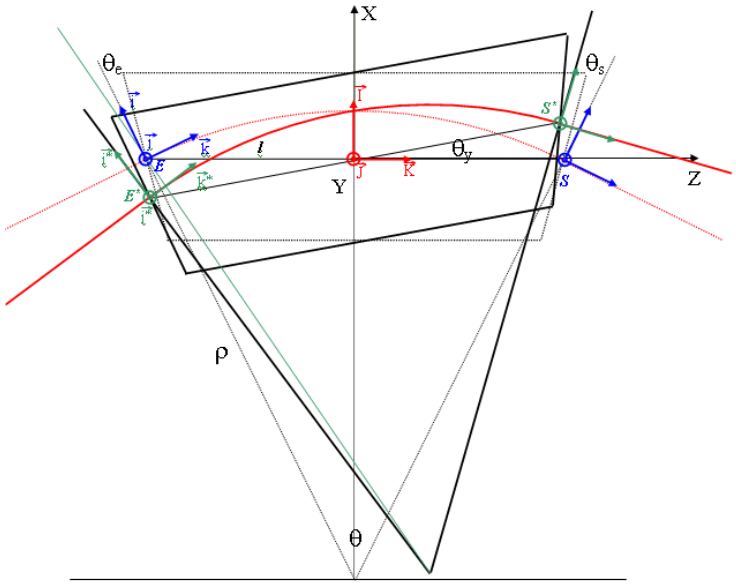
Input drift (*DZ*·cos(*θ* /2)

$$\begin{cases} x_e += DZ \cdot \sin(\theta/2) \\ x_s += DZ \cdot \sin(\theta/2) \end{cases}$$

Output drfit (-*DZ*·cos(*θ*/2)



Input of the dipole



At the input :

$$\begin{pmatrix} \vec{I} \\ \vec{J} \\ \vec{K} \end{pmatrix} = \begin{pmatrix} \cos\theta/2 & 0 & \sin\theta/2 \\ 0 & 1 & 0 \\ -\sin\theta/2 & 0 & \cos\theta/2 \end{pmatrix} \cdot \begin{pmatrix} \vec{i}_e \\ \vec{j}_e \\ \vec{k}_e \end{pmatrix} = T_{r_e \to R} \cdot \begin{pmatrix} \vec{i}_e \\ \vec{j}_e \\ \vec{k}_e \end{pmatrix}$$

Soit M_θ, la matrice de rotation du dipôle (autour d'un axe, sens donné par la loi du tir-bouchon) :

$$M_{\theta_x} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_x & \sin\theta_x \\ 0 & -\sin\theta_x & \cos\theta_x \end{pmatrix} \approx \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1-\frac{1}{2}\theta_x^2 & \theta_x \\ 0 & -\theta_x & 1-\frac{1}{2}\theta_x^2 \end{pmatrix}$$

$$M_{\theta_y} = \begin{pmatrix} \cos\theta_y & 0 & -\sin\theta_y \\ 0 & 1 & 0 \\ \sin\theta_y & 0 & \cos\theta_y \end{pmatrix} \approx \begin{pmatrix} 1-\frac{1}{2}\theta_y^2 & 0 & -\theta_y \\ 0 & 1 & 0 \\ \theta_y & 0 & 1-\frac{1}{2}\theta_y^2 \end{pmatrix}$$

$$M_{\theta_z} = \begin{pmatrix} \cos\theta_z & \sin\theta_z & 0 \\ -\sin\theta_z & \cos\theta_z & 0 \\ 0 & 0 & 1 \end{pmatrix} \approx \begin{pmatrix} 1-\frac{1}{2}\theta_z^2 & \theta_z & 0 \\ -\theta_z & 1-\frac{1}{2}\theta_y^2 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Nous avons donné aussi le développement au deuxième ordre de la matrice (attention, dans ce cas, le déterminant est non nul).
La matrice exprimant la rotation totale du dipôle est le produit des trois matrices de rotation. Cependant, ce produit n'est pas commutatif.
Par contre, en supposant que les angles de rotation sont petits et en restant au deuxième ordre, on peut alors trouver une matrice simplifiée pour la rotation des éléments :

$$M_\theta \approx \begin{pmatrix} 1-\frac{1}{3}\cdot\left(\theta_x^2+\theta_y^2+\theta_z^2\right) & \theta_z & -\theta_y \\ -\theta_z & 1-\frac{1}{3}\cdot\left(\theta_x^2+\theta_y^2+\theta_z^2\right) & \theta_x \\ \theta_y & -\theta_x & 1-\frac{1}{3}\cdot\left(\theta_x^2+\theta_y^2+\theta_z^2\right) \end{pmatrix}$$

C'est cette matrice que l'on utilisera dans PARTRAN. Elle est évidemment approchée, mais très certainement moins que la connaissance de l'amplitude des erreurs. Remarquons tout de même que son déterminant est non nul (à l'ordre 4). Le choix de la diagonale n'est pas unique. C'est cependant celui qui minimise le déterminant pour des angles de rotation égaux dans toutes les directions.

La quantité de mouvement $\vec{p}$ dans le repère $\left(\vec{i}_e, \vec{j}_e, \vec{k}_e\right)$ s'exprime alors $\vec{p}^*$ dans le repère $\left(\vec{i}_e^*, \vec{j}_e^*, \vec{k}_e^*\right)$:

$$\vec{p}^* = M_{-\theta} \cdot \vec{p} \cdot \quad (1)$$

Une particule de coordonnées $\left(x_e, y_e, 0\right)$ dans le repère $\left(\vec{i}_e, \vec{j}_e, \vec{k}_e\right)$ centré sur *E* possède alors les coordonnées $\left(x_e^*, y_e^*, z_e^*\right)$ dans le repère $\left(\vec{i}_e^*, \vec{j}_e^*, \vec{k}_e^*\right)$ centré sur E^* telles que :

$$\begin{pmatrix} x_e^* \\ y_e^* \\ z_e^* \end{pmatrix} = T_{R\to r_e} \cdot \left(\left(Id - M_{-\theta}\right) \cdot \begin{pmatrix} 0 \\ 0 \\ -l \end{pmatrix} + M_{-\theta} \cdot T_{r_e \to R} \cdot \begin{pmatrix} x_e \\ y_e \\ 0 \end{pmatrix} \right)$$

l est la demi-longueur de la corde à la trajectoire principale de l'aimant.
On a : $l = \rho \cdot \sin\varphi/2 \cdot$

soit :

$$\begin{pmatrix} x_e^* \\ y_e^* \\ z_e^* \end{pmatrix} = \begin{pmatrix} \left(1-\varepsilon\right)\cdot x_e - \left(\theta_z \cos\varphi/2 + \theta_x \sin\varphi/2\right)\cdot y_e - \left(\theta_y \cos\varphi/2 - \varepsilon\cos\varphi/2\right)\cdot l \\ \left(1-\varepsilon\right)\cdot y_e + \left(\theta_z \cos\varphi/2 + \theta_x \sin\varphi/2\right)\cdot x_e + \theta_x \cdot l \\ -\theta_y \cdot x_e - \left(\theta_z \sin\varphi/2 - \theta_x \cos\varphi/2\right)\cdot y_e - \left(\theta_y \sin\varphi/2 + \varepsilon\cos\varphi/2\right)\cdot l \end{pmatrix} \quad (2)$$

avec : $\varepsilon = \frac{1}{3} \cdot \left(\theta_x^2 + \theta_y^2 + \theta_z^2\right)$.

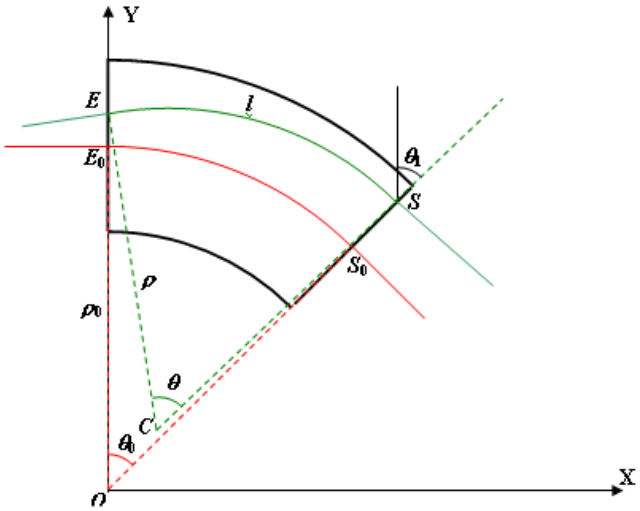
La procédure en entrée est la suivante :
Une particule entre dans le dipôle avec les coordonnées $\left(x_e, x'_e, y_e, y'_e, phi_e, E_e\right)$ dans le référentiel $\left(\vec{i}_e, \vec{j}_e, \vec{k}_e\right)$.
Sa quantité de mouvement réduite $\vec{p}_e = \left(x'_e \cdot p_{ze}, y'_e \cdot p_{ze}, p_{ze}\right)$ dans $\left(\vec{i}_e, \vec{j}_e, \vec{k}_e\right)$ est calculée avec $p_{ze} = \sqrt{\frac{\gamma^2 - 1}{1 + x'^2_e + y'^2_e}}$.

Elle est ensuite transformée en $\vec{p}_e^{\,*}$ avec (1), puis on prend : $x'^{*}_e = \frac{p^*_{xe}}{p^*_{ze}}$, …
Sa position dans le repère $\left(\vec{i}_e^{\,*}, \vec{j}_e^{\,*}, \vec{k}_e^{\,*}\right)$ est obtenue avec (2).
Un drift de longueur $-z_e^*$ doit être appliqué avant la matrice du dipôle.

Transport dans le dipôle

Le coin d'entré est traité en lentille mince "à la PARMILA" en prenant en compte l'influence du champ de fuite magnétique.

La matrice du dipôle telle qu'elle était utilisée avant est remplacée par un transport non linéaire dans le référentiel $\left(\vec{i}_e^{\,*}, \vec{j}_e^{\,*}, \vec{k}_e^{\,*}\right)$. Le calcul est détaillé ci-dessous :



Les coordonnées des points sont données dans le repère (X,Y) centré sur O.
$E_0\left(0, \rho_0\right)$, $E\left(0, \rho_0 + x_e\right)$, $S_0\left(\rho_0 \cos\theta_0, \rho_0 \sin\theta_0\right)$

La trajectoire de la particule est un cercle de rayon de courbure ρ :
$$\rho = \frac{p}{qB} = \rho_0 \cdot \frac{1 + \delta p / p_0}{1 + \delta B / B_0}$$

Recherchons les coordonnées du centre de ce cercle C :

$$\begin{cases} x'_e = \dfrac{X_C}{Y_E - Y_C} \\ (X_E - X_C)^2 + (Y_E - Y_C)^2 = \rho^2 \end{cases}$$

$$\Rightarrow \quad \begin{cases} X_C = x'_e \cdot \dfrac{\rho}{\sqrt{1 + x'^2_e}} \\ Y_C = Y_E - \dfrac{\rho}{\sqrt{1 + x'^2_e}} \end{cases}$$

Recherchons les coordonnées de la position de sortie *S* de la particule :

$$\begin{cases} X_S = Y_S \cdot \tan\theta_0 \\ (X_S - X_C)^2 + (Y_S - Y_C)^2 = \rho^2 \end{cases}$$

$$\Rightarrow \quad \begin{cases} X_S = \sin\theta_0 \cdot \left[X_C \sin\theta_0 + Y_C \cos\theta_0 + \sqrt{\rho^2 - (X_C \cos\theta_0 - Y_C \sin\theta_0)^2} \right] \\ Y_S = \cos\theta_0 \cdot \left[X_C \sin\theta_0 + Y_C \cos\theta_0 + \sqrt{\rho^2 - (X_C \cos\theta_0 - Y_C \sin\theta_0)^2} \right] \end{cases}$$

On en déduit sa position de sortie $x_s = SS_0$ de la particule dans le référentiel lié à la trajectoire de référence :

$$x_s = X_C \sin\theta_0 + Y_C \cos\theta_0 + \sqrt{\rho^2 - (X_C \cos\theta_0 - Y_C \sin\theta_0)^2} - \rho_0$$

Calculons l'angle de la particule en sortie dans le référentiel lié à la trajectoire de référence $x'_s = \arctan(\theta_0 - \theta_1)$:

$$\tan\theta_1 = \frac{X_S - X_C}{Y_S - Y_C},$$

$$\Rightarrow \quad x'_s = \frac{(Y_S - Y_C) \cdot \tan\theta_0 - (X_S - X_C)}{(X_S - X_C) \cdot \tan\theta_0 + (Y_S - Y_C)}$$

Calculons la longueur de la trajectoire : $l = \rho \cdot \theta$

$$\tan\theta = \frac{(X_S - X_C) + (Y_S - Y_C) \cdot x'_e}{(Y_S - Y_C) - (X_S - X_C) \cdot x'_e}$$

$$\Rightarrow \quad l = \rho \cdot \arctan\left(\frac{(X_S - X_C) + (Y_S - Y_C) \cdot x'_e}{(Y_S - Y_C) - (X_S - X_C) \cdot x'_e} \right).$$

Dans le plan vertical, l'aimant peut-être considéré comme un espace de glissement :

$$\begin{cases} y'_s = y'_e \\ y_s = y_e + y'_e \cdot l \end{cases}.$$

Sortie du dipôle

En sortie du dipôle, la particule est à la position $\left(x_s^*, x_s'^*, y_s^*, y_s'^*, phi_s^*, E_s^*\right)$.

En sortie, on a :

$$\begin{pmatrix} \vec{I}^* \\ \vec{J}^* \\ \vec{K}^* \end{pmatrix} = \begin{pmatrix} \cos\theta/2 & 0 & -\sin\theta/2 \\ 0 & 1 & 0 \\ \sin\theta/2 & 0 & \cos\theta/2 \end{pmatrix} \cdot \begin{pmatrix} \vec{i}_s^* \\ \vec{j}_s^* \\ \vec{k}_s^* \end{pmatrix} = T_{r_s^* \to R^*} \cdot \begin{pmatrix} \vec{i}_s^* \\ \vec{j}_s^* \\ \vec{k}_s^* \end{pmatrix}$$

Le vecteur $\vec{p}^*$ dans le repère $\left(\vec{i}_s^*, \vec{j}_s^*, \vec{k}_s^*\right)$ s'exprime alors $\vec{p}$ dans le repère $\left(\vec{i}_s, \vec{j}_s, \vec{k}_s\right)$:

$$\vec{p} = M_\theta \cdot \vec{p}^* . \quad (3)$$

Une particule de coordonnées $\left(x_s^*, y_s^*, 0\right)$ dans le repère $\left(\vec{i}_s^*, \vec{j}_s^*, \vec{k}_s^*\right)$ centré sur S^* possède alors les coordonnées $\left(x_s, y_s, z_s\right)$ dans le repère $\left(\vec{i}_s, \vec{j}_s, \vec{k}_s\right)$ centré sur S telles que :

$$\begin{pmatrix} x_s \\ y_s \\ z_s \end{pmatrix} = T_{R \to r_s^*} \cdot \left(\left(Id - M_\theta\right) \cdot \begin{pmatrix} 0 \\ 0 \\ l \end{pmatrix} + M_\theta \cdot T_{r_s^* \to R} \cdot \begin{pmatrix} x_s^* \\ y_s^* \\ 0 \end{pmatrix} \right) \quad (4)$$

soit :

$$\begin{pmatrix} x_s \\ y_s \\ z_s \end{pmatrix} = \begin{pmatrix} \left(1-\varepsilon\right) \cdot x_s^* + \left(\theta_z \cos\varphi/2 - \theta_x \sin\varphi/2\right) \cdot y_s^* - \left(\theta_y \cos\varphi/2 - \varepsilon\cos\varphi/2\right) \cdot l \\ \left(1-\varepsilon\right) \cdot y_s^* - \left(\theta_z \cos\varphi/2 - \theta_x \sin\varphi/2\right) \cdot x_s^* + \theta_x \cdot l \\ \theta_y \cdot x_s^* - \left(\theta_z \sin\varphi/2 + \theta_x \cos\varphi/2\right) \cdot y_s^* + \left(\theta_y \sin\varphi/2 + \varepsilon\cos\varphi/2\right) \cdot l \end{pmatrix} \quad (2)$$

avec : $\varepsilon = \frac{1}{3} \cdot \left(\theta_x^2 + \theta_y^2 + \theta_z^2\right)$.

La procédure en sortie est la suivante :

Une particule sort du dipôle avec les coordonnées $\left(x_s^*, x_s'^*, y_s^*, y_s'^*, phi_s^*, E_s^*\right)$ dans le référentiel $\left(\vec{i}_s^*, \vec{j}_s^*, \vec{k}_s^*\right)$.

Sa quantité de mouvement réduite $\vec{p}_s^* = \left(x_s'^* \cdot p_{zs}^*, y_s'^* \cdot p_{zs}^*, p_{zs}^*\right)$ dans $\left(\vec{i}_s^*, \vec{j}_s^*, \vec{k}_s^*\right)$ est calculée avec $p_{zs}^* = \sqrt{\frac{\gamma^2 - 1}{1 + x_s'^{*2} + y_s'^{*2}}}$.

Elle est ensuite transformée en $\vec{p}_s$ avec (3), puis on prend : $x_s' = \frac{p_{xs}}{p_{zs}}$, …

Sa position dans le repère $\left(\vec{i}_s, \vec{j}_s, \vec{k}_s\right)$ est obtenue avec (2).

Un drift de longueur $-z_s$ doit être appliqué.

Errors study management

Remote & Local computers

For an error study (only statistical study) with PARTRAN or Toutatis or even in envelope, which spend a very long time, several computers can be used via client/server architecture (multiparameters scheme). These remote machines have to work under Window, Linux and MacOS operating system. You can decide to use these computers only during the night or weekend. Some computer can be add or remove during the error study process. (You have to install and launch into each computer the code "***twserver***". For each computer you have to select the number of core you want to use.
If your run is very short (a few seconds), it's probably much better to use only your local computer (IP=0.0.0.0) and set a number of core bigger than the real available number of core. You should have to check that point in your own computer.

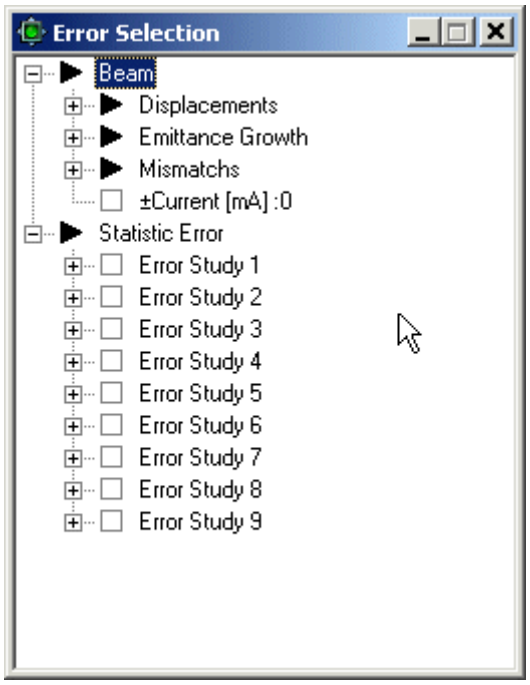
TraceWin server installation

For Windows you have to install "twserver.exe" and for Linux or MacOS install "twserver" code on your remote computers. You have just to copy the executable file somewhere and launch it. Setup is completely automatic and the final installation will be "*C:\TWSever*" for Windows and "*Home/.TWServer*" for Linux or MacOS. If you need and automatic start, print "*./twserver auto&*" for Linux or "*twserve.exer auto*" for Windows. All new installation will destroy the running old version and will install the new one in the good directory. Don't try to start "twserver" directly in its final directory.
Sometime, some remote machines don't reply to TraceWin requests. A few reasons can explain that:
Check you firewall: The TraceWin server need access rights to TCP port between 1024 and 10000.
Have a look on "*Remote computer*" box management in order to check if you select or not the option "*Work only out of working hours*". Check also the minimum "*Needed memory*" requirement.
Under Windows, remote server can be disabled locally (right click on right bottom tray icon). That means TraceWin will use this computer only if its CPU charge is very low.
Before any error study, test all your computers with "*Test computers*" button from "*Remote computers*" Box.

For CEA Saclay users only:
You can install "twserver" on the Irfu clusters called "DAPHPC" or "DAPINT", for that launch twserver by the command *"./twserver daphpc*" or *"./twserver dapint*" according to cluster where you are. Set the number of cores for the corresponding cluster

Generality

TraceWin allows to study many kinds of errors. They are separated in two types: The input "*beam errors*" and the "*statistical errors*". The first type concerns the input beam errors, like beam displacement, emittance growths, beam mismatches and so on. The second is based on a Monte Calo approach, it concerns quadrupole, cavity, RFQ and input beam, this method needs a statistical study with several runs.



The input beam errors are study one after the other. You have to select one or several errors and put the error amplitude values. There are 9 different statistical error studies, which are also study one after the other, but the errors defined inside are combined during a study including *N* run, ("***Number of run"***) and the error amplitude have to be insert in the data file by including the error commands.

Different tools are available in order to analyze the statistical results. All the results are located in

Input beam errors

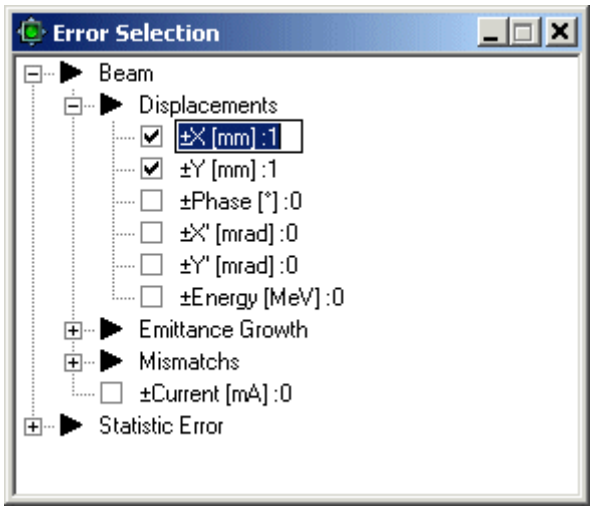
This kind of study, you have to use only one computer with one core.

Four kind of error:

- **Beam displacement**: The beam input position is not centered.
- **Emittance growth**: The input beam emittance is increased by a percentage.
- **Beam mismatch**: The input beam is mismatched by a percentage. A 20 % mismatch in x plane means α_x and β_x are multiplied by $(1.2)^2$.
- **Beam Current error**: Allows to study the effect of the input beam current variation.

The best way to understand the way of using is to look at an example:

Example: Know the linac behavior and losses when the input beam positions *X* and *Y* move from -1 mm to 1 mm in ten steps [-1.0, -0.9, -0.8,….,0.8, 0.9, 1.0]. You have to select these two kind of errors and put the error amplitude (Click two time one the item). Then, put "***Nbr of step***" to and to finally select "***Envelope error study***" or "***Particles error study***"

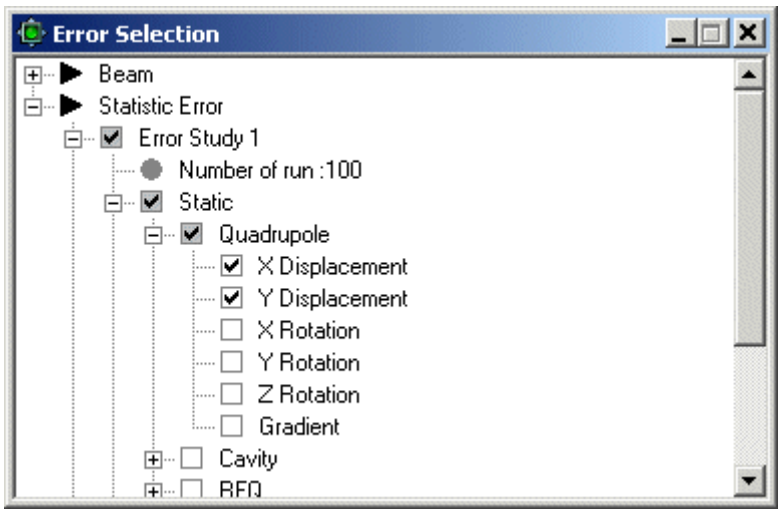


For the beam errors, in the box Remote and Local computer you have to select only your local computer (IP:0.0.0.0).

Now, you just have to run TraceWin, the two errors study will perform one after the other. At the end, you can look at the results like the output beam behavior according to the input beam errors by using the two file "*X_Displacement_ENV.txt*" and "*Y_Displacement_ENV.txt*" if "***Envelope study***" is selected or "*X_Displacement_PAR.txt*" and "*Y_Displacement_PAR.txt*" if "***Particles study***" is selected. You find them in the "***Calculation directory***" defined in the "Main" TraceWin page. One of these files has to be select in "***Studies results***" of the "Errors" page.

Statistical errors

The first stage is to include in the data file the different error commands with all the amplitude errors, the second one is to select the different kinds of error in the box "*Error selection*" of "*Error*" page, For example, if you want to include in the "*Error study 1*", some static quadrupole displacement and rotation errors you have to do the following selection.



Third stage: Selects le number of linacs, which will be generated, here 100, "***Number of run***". Finally, select the number of step to reach the maximum amplitude of the errors, "***Nbr of step***" of "Error"

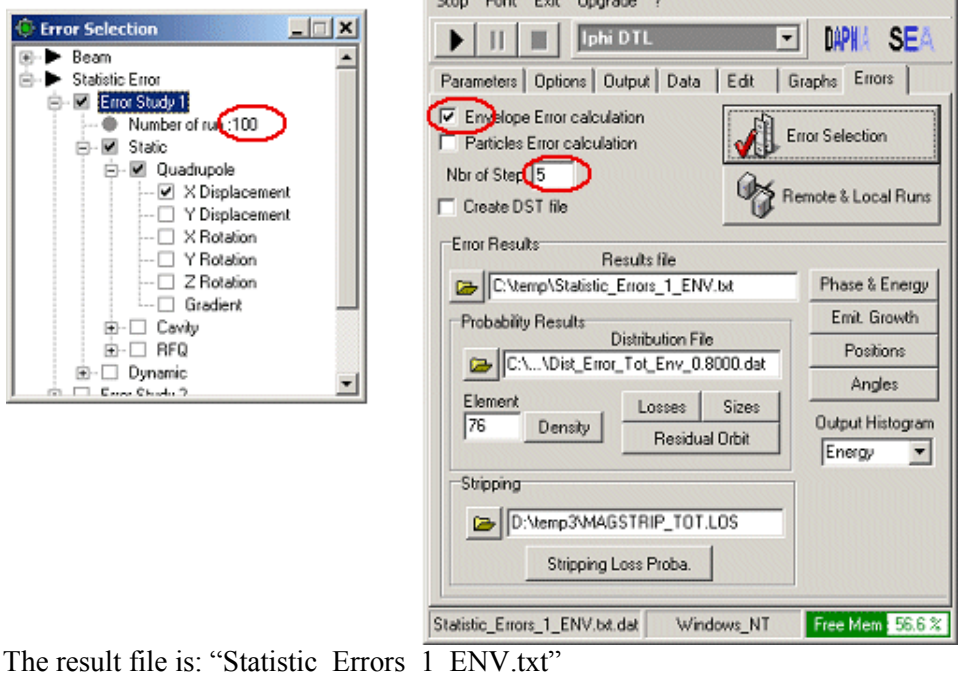
page. For example 5 mean [20% of the max amplitudes, 40%, 60%…]. In this case 500 runs will be performed. All errors are randomly set in a uniform probability law *[-σ, σ]*.

Correction scheme

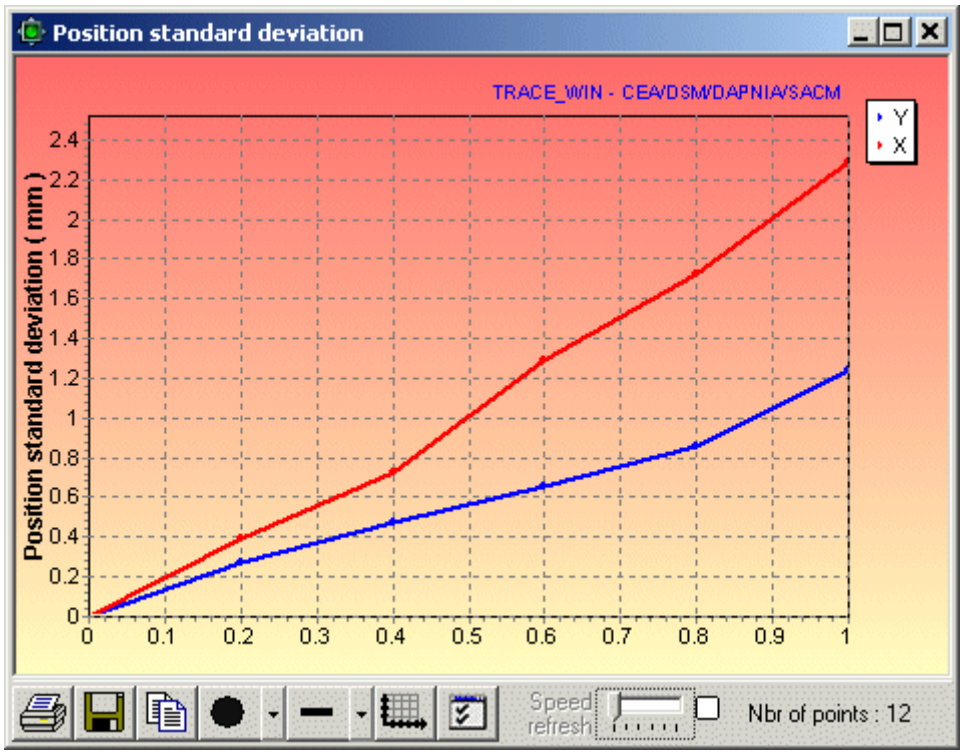
If you have defined a correction scheme in you data file, this scheme will be applied to each machine like the following procedure:
Use your reference machine tuned with your correction scheme if you defined one (diagnostic associated with ADJUST command) and if you check "*Match using diagnostic*" in "*Matching*" page.
Static random errors are applied to your linac according to your static error commands.
Your correction scheme is performed.
Dynamic random errors are applied to your linac according to your dynamic error commands.
The final run is performed.

Error study example

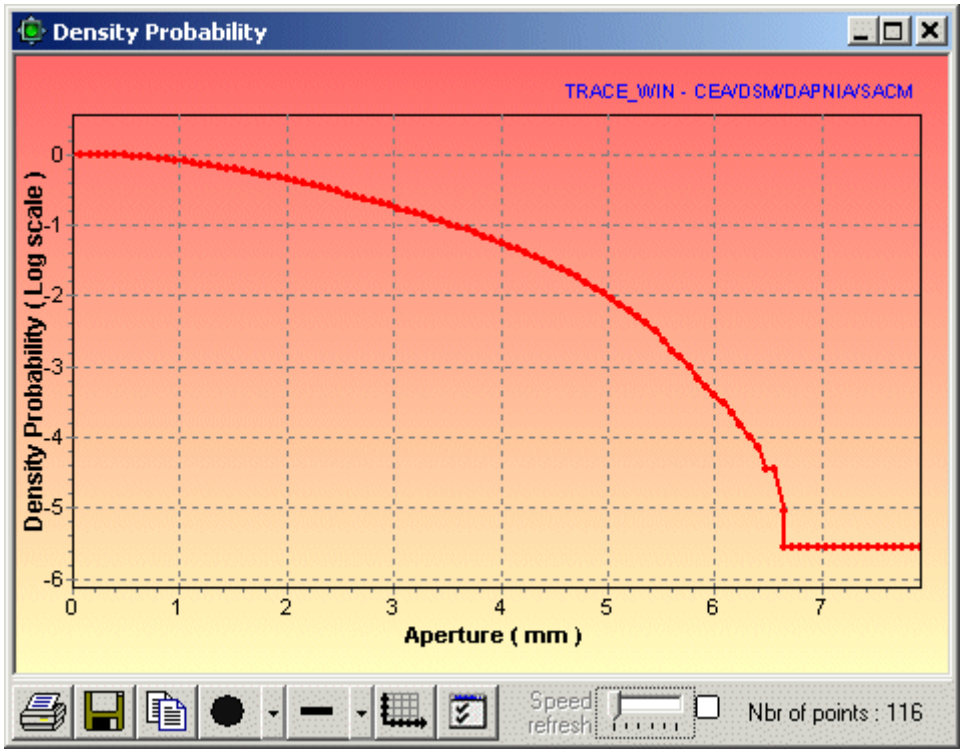
A classical error study is concerning the quadrupole misalignments in several DTL tanks, for example: Quadrupoles misalignment study from 0 mm to 0.2 mm in 5 steps. 100 linacs with 1000 macro-particles for each of them.



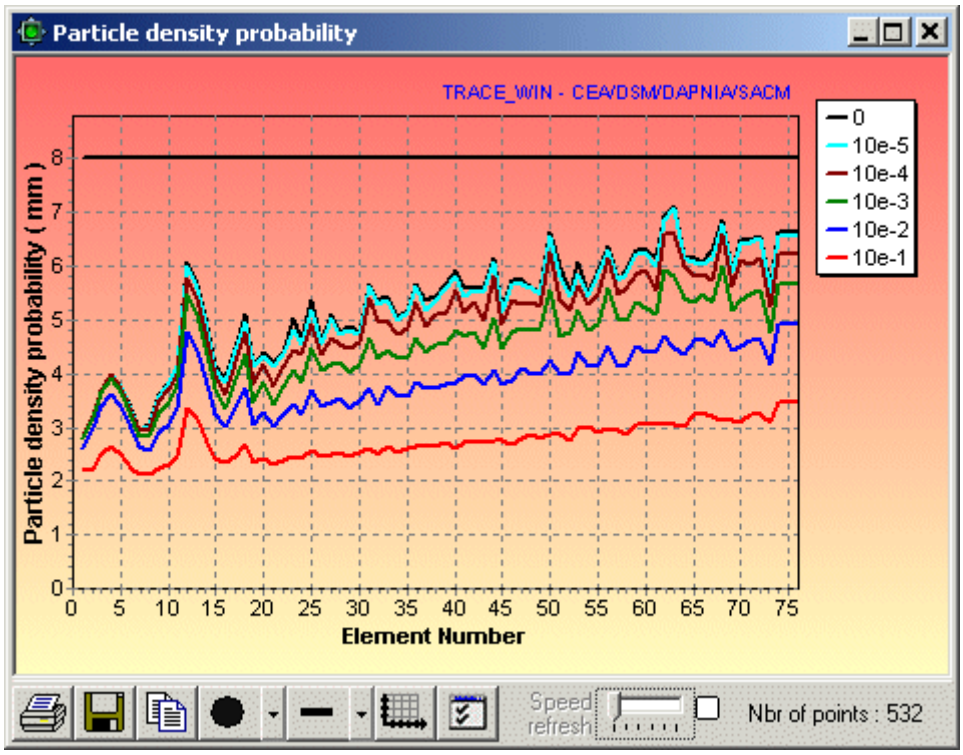
The result file is: "Statistic_Errors_1_ENV.txt"
The distribution file results are:"*Dist_Error_Tot_Env_0.2000.dat*" for 20%
"Dist_Error_Tot_Env_0.4000.dat" for 40%,
…
The rms output beam position for 100% of the error (0.2 mm) is about 2.2 mm for X.



The density probability of the beam located at the end of the element 76 shows a 6.6 mm maximum beam size.



The particle density probability repartition all along the structure show for example than the maximum size of 99% of the beam is lower than 5 mm when the errors are randomly distributed at 80% of the maximum amplitude. (Red curve: 90% of the 100.000 particles, 100 linacs of 1000 particles, Blue curve: 99%, Green curve 99.9% …)



If you want to combine other quadrupole errors, you just have to select them and restart the study. Obviously, cavity errors can be combined with quadrupole error. You could also insert in your data file a scheme of correction; steerers associated with beam monitor positions. Start a new study in order to see if your scheme is efficient.

Copy following examples:

examples\Supra_4.ini
examples\Supra_4.dat
examples\Supra_4.cal

examples\Supra_5.ini
examples\Supra_5.dat
examples\Supra_5.cal