Soltracy3 mannual for developpers

Jianfeng Nadolski *

December 19, 2013

Contents

*LAL, Orsay, France.Email: zhangj@lal.in2p3.fr

1 Log

05/10/2012 Add the control flag to control the entrance and exit edge effects and fringe field of the dipole, with"edge_effect1 = 1" in the dipole definition in the lattice file, the entrance edge effect and fringe field of the dipole are turned on, with "edge_effect1 = 0", this effects are turned off. These two effects at the exit of the dipole are controlled by "edge_effect2".

2 Introduction

Tracy is a code to do long term tracking, and is written in the mixture of c and c++. This code is kept on developing. Soleil version of Tracy 3 is a code with more flexibility and easy to use. User does not need to know the structure of the code, what they need to do is to write an input script, and then run the code. Based on the need, user can write the files to define multipole field errors, misalignment errors of the lattice elements, and vacuum chamber, and then provide the file name in the user input script, in order to set the field errors of the lattice elements and the vacuum limit for the different region of the lattice. Attention: In the user defined file, such as the user input script, the file to define multipole field errors or alignment errors, or the file to define vacuum chambers, the maximum numbers of column is 130 (Not including comment line which starts with symbol #.), and the spaces between each parameters or variables cant contain TAB key, otherwise the code cant work properly. This is because the routine to read the user defined input file is written in C. These limits will be improved in the future development. Although the user can define the file name whatever they want, it is suggested to name the file which is used to define multipole field error with the extension .fe and name the file which is used to define the misalignment error with the extension .ae.

There are two versions of Tracy 3, non-parallel version and parallel version. Non-parallel version Tracy is for the single computer, parallel version of Tracy is for the the cluster. The user defined input script which can be used for both parallel and non-parallel version Tracy. The ascii file with any extension isacceptable for Tracy3, but the extension is suggested to be .prm.

3 Non-parallel version Tracy

3.1 Compile

The make file of Tracy is generated by automake. Based on the compilers used, user needs to update make_for_gcc.sh under path "*$HOME/TracyIII/*" or the "Makefile.am" under path "*$HOME/TracyIII/tracy/tracy/src*" and "*$HOME /TracyIII/tracy/tools*". The compilers used on the server "metis" of SOLEIL Synchrotron are "gfortran", and "gcc".

To compile the code, run the command under the shell terminal:

```
make_for_gcc.sh        opt
```

Then the executive file "soltracy" is automatically generated under path "*$HO ME/TracyIII/tracy/tools*".

3.2 Run

To execute Tracy, the user needs to define an input file with the the lattice file names and the related commands (section 24). The name of the input file can be any valid characters. Although there is no limitation to the extension of the file, the extension ".prm" is preferred. For example, To run the user defined input file "Input_test.prm", user can type the command:

```
soltracy        Input_test.prm
```

4 Parallel version Tracy

In order to reduce the tracking time, the parallel version Tracy can be used on the cluster. Until now, the following three features of Tracy are parallelized:

- Frequency map analysis for on momentum particle, command "FmapFlag".
- Frequency map analysis for off momentum particle, command "FmapdpFlag".
- Track momentum acceptance at lattice elements, command "MomentumAccFlag".

4.1 Compile

The commonly used compilers for parallel computation are MPI 2, and Intel MPI which is based on MPI 2. For the cluster of SOLEIL Synchrotron, Intel MPI is installed. To get the parallel Tracy work, three files of the non-parallel version Tracy need to be modified. The details are shown in the following steps.

1. The path of included files of Intel MPI is added in "Makefile.am" under path "*$HOME/TracyIII/tracy/tracy/src*" (shown in BLUE color):

   INCLUDES=-I../inc-I$(NUM_REC)/inc -I/opt/intel/impi/3.2.2.006/inc lude64

2. The execute file, source file, paths of included files and library of Intel MPI are modified in "Makefile.am" under path "*$HOME/TracyIII/tracy /tools*" (shown in BLUE color):

   bin_PROGRAMS = psoltracy

   soltracy_SOURCES = `soltracy.cc nrutil.c nrcheck.c nrlinwww.c nrframe.c ../tracy/src/tracy_lib.cc` →

   psoltracy_SOURCES = `psoltracy.cc nrutil.c nrcheck.c nrlinwww.c nrframe.c ../tracy/src/tracy_lib.cc`

   LIBS = `-L$(NUM_REC)/lib -L/opt/intel/impi/3.2.2.006/lib64 -L$(LIBPATH) -lrecipes_c_icc -lstdc++ -lgfortran -lmpichcxx`

   INCLUDES = `-I$(TRACY_LIB)/tracy/inc -I$(NUM_REC)/inc -I/opt/intel/impi/3.2.2.006/include64`

3. The compilers used in the parallel computing are defined in the "make_for_psoltracy.sh" located in path "$HOME/TracyIII" as (shown in BLUE color):

   CC = `mpiicc`

   CXX = `mpiicpc`

   F77 = `mpiifort`

   Depending on the compilers used to do parallel computation, user needs to update the compilers in "make_for_psoltracy.sh" and the paths of included files and library which are shown above with blue color.

   After updating compilers, paths of included files and library for parallel computation, user can run

   make_for_psoltracy.sh

   under the shell script to compile the parallel Tracy. After compilation, the execute file "psoltracy" is automatically generated under the path "*$HOME/TracyIII/tracy/tools*".

4.2 Run

As the non-parallel version Tracy, user needs to write the commands in a script which must be with the file extension ".prm". The syntaxes to define the script "*.prm" are the same for both non-parallel and parallel Tracy.

To run the parallel Tracy, user need to contact the administrator of their cluster to know how to run parallel programs on the cluster. For the cluster on Synchrotron SOLEIL, the nodes used to do parallel computation are assigned by PBS (Portable Batch System), so a script is need. For example, user define the input script `test.prm` to tell Tracy what jobs are need to be done on the cluster, define script lance_tracy3_parallel.sh to assign the numbers of CPUs to do parallel computation and lance job to the SOLEIL cluster; and then type the

following command under the bash shell to submit the job and run the parallel Tracy on the cluster:

lance_tracy3_parallel.sh test.prm

5 User input script

There are two types of keywords in the user input script. The first type is to set the file, the file names, and define parameters for the related calculations; the key words for such definitions are ended with the characters "Flag". The second type is to define Boolean commands with or without parameters to do different calculations. The rules of the definition of input scripts are:

- The blank lines and lines starting with "#" (comment line) are ignored by the code.
- Keywords without "Flag" as the final 4 characters are NOT executed according to the defined sequence in user input script. If the same command keywords are defined many times in the user script, only the last defined keyword is executed.
- The commands with the last 4 characters as "Flag" are executed according to the sequence in the code.
- The definition of lattice file at the beginning of the user script is mandatory.
- If the horizontal correctors, or vertical correctors, or girders, Beam Position Monitors, or skew quadrupoles are defined in the lattice file, user must declare the element name at the beginning of Tracy input file.
- One keyword command uses one line. User can not define more than one command at the same line.
- Except explanation, all commands with Boolean flag are the generic commands, that is, the commands are not machine dependent.

5.1 File path

In the user input script, user can specify the name of the files which are used in the calculation, such as the lattice file, multipole field error file, misalignment error file, vacuum chamber file. When specifying the file name without file path in the Tracy input file, Tracy will look for the file in the current work path. Otherwise, user need to provide the absolute path of the file which is not located at the current path, such as: "`/home/physmach/Tracy3/tracy/soleil.lat`". Another method is to provide the filename without absolute path, but put all the called external files at a certain directory, and this directory is defined in the Tracy input file using the command:

in_dir user_defined_path

For example,

in_dir /home/zhang/codes/TracyIII/lattice/

This example tells the code all the files defined with absolute file path in the Tracy input file are located at the directory */home/zhang/codes/TracyIII/lattic e/*. If user declares the files without absolute path and does not set the directory through command in_dir; or the files specified in the script are not found at the current working path, the code will give an error message and stop running.

6 File names

6.1 Lattice file name

In the input script, user must define the lattice name, this is mandatory. The command is:

lat_file lattice_file_name

Here lattice_file_name is the lattice file name without the extension ".lat". For example, the following command sets the lattice file of SOLEIL ring:

lat_file solamor2

n Tracy 3, after reading the lattice, Twiss parameters are automatically printed to the file *linlat.out*.

If one of these elements (Horizontal/vertical correctors or beam position monitors, or girders, or skew quadrupoles.) are defined in the lattice file, for example to read the misalignment errors of the lattice element and then do orbit correction, to read multipole field errors (SOLEIL lattice), etc.; user MUST specify the names of these elements using the corresponding commands in the Tracy input file:

h_corr HCM

v_corr VCM

gs GS

ge GE

bpm_name BPM

qt QT

Here HCM is the name of horizontal correctors for horizontal orbit correction, or the horizontal correctors on which the multipole field errors are added in SOLEIL lattice; VCM is the name of vertical correctors used for vertical orbit correction, or the vertical correctors on which the multipole field errors are added in SOLEIL lattice; GS is the name of the start girder; GE is the name of the end girder; BPM is the name of beam position monitors defined in the lattice file; QT is the name of the skew quadrupoles defined in the lattice. Generally, user needs to define horizontal, vertical correctors and BPMs when reading the misalignment errors or correcting closed orbit distortion.

6.2 Multipole field error file name (SOLEIL lattice)

User can read the multipole field errors on SOLEIL lattice from an external file; the file is specified in the user script using the following command:

multipole_file multipole_file_name

Here multipole_file_name is the user defined multipole field error file, the format of this file is given in section 24.

If the multipole field errors of horizontal, vertical correctors and skew quadrupoles are defined in the multipole_file, user MUST specify the names of these lattice elements in the Tracy input file as the following example:

h_corr CH

v_corr CV

qt QT

Here CH, CV and QT are the names of horizontal, vertical correctors, and skew quadrupoles respectively which are defined in the lattice file.

6.3 Files of multipole field errors of correctors and skew quadrupoles (SOLEIL lattice)

Horizontal, vertical correctors and skew quadrupoles are integrated with the sextupole quadrupoles on SOLEIL storage ring. To define the multipole field errors on these elements, user need to define the orders and relative strengths of multipole field errors in the multipole field error file (section Error: Reference source not found) and specify the file name in the user input script *.prm (section Multipole field error file name), then specify the file names as the following example:

fic_hcorr corh.txt

fic_vcorr corv.txt

fic_skew corqt.txt

Here corh.txt and corv.txt are the files with measured current values corh, corv, corqt (with unit [Ampere]) for the horizontal, vertical correctors and skew quadrupoles respectively. Based on the measured current values, user can get the integrated field strength as:

```
Hcorr_strength [T.m] = corh *_convHcorr /brho; (horizontal correctors)
Vcorr_strength [T.m] = corv *_convVcorr /brho; (vertical correctors)
Qt_strength [T.m] = corqt *_convQt /brho; (skew quadrupoles)
```

Here brho is the momentum rigidity, and the conversion constants between current and field are _convHcorr = 8.14e-4, _convVcorr = 4.642e-4, _convQt = 93.83e-4.

For SOLEIL lattice, the SAME ORDER of multipole field errors on the same elements are added together, so the SAME ORDER of horizontal/vertical,

skew quadrupoles, and sextupoles are added together since these elements are integrated at the same magnets.

6.4 File to define field strength of virtual coupling source elements (SOLEIL lattice)

On SOLEIL storage ring, the coupling is thought to come from the rotation of quadrupoles and vertical displacements of sextupoles. The strengths of these coupling sources are written in a file, then read by the following command:

virtualskewquad_file virtual_skew_quad_currents.txt

Here virtual_skew_quad_currents.txt is the user defined file with strength of vertical coupling source.

In order to use this command to read the virtual sources of coupling, user needs to define the virtual coupling element as a skew quadrupole in the lattice file:

SQ: quadrupole, tilt=45.0, K= 0.0, method=4, N=1;

and this virtual coupling element MUST be with the name SQ. Now there 152 elements defined as the virtual coupling sources in the SOLEIL lattice. The syntaxs to define skew quadrupole are explained in section . The measured current value qtcorr[i] of the ith coupling source is converted to the corresponding integrated skew quadrupole field strength corr_strength as corr_strength [m-1] = qtcorr[i]*conv/brho; here brho is momentum rigidity, and the conversion constant conv is 93.88e-4 [A-1.T].

6.5 Cut off value

Set the cut off value of all random distribution (Gaussian distribution) to n times of the RMS value sigma: normalcut n

7 Physics: Hamiltionian

The dynamic motion of a conserved system can be described by a Hamiltonian. For an accelerator system without radiation damping (energy loss) and RF cavities (energy gain), the motion of a single electron in the magnetic field can be described using the following Hamitonian in the curvilinear coordinate system (Merz...)

$$
\begin{aligned}
H(x, P_x; y, P_y; -t, \delta; s) &= L + V && (1)\\
&= -(1 + h_{\text{ref}} x)[\sqrt{(P^2 - (P_x - eA_x)^2 - (P_y - eA_y)^2)} + eA_z(x, P_x; y, P_y, s)] && (2)\\
P^2 &= P_x^2 + P_y^2 + P_z^2 && (3)
\end{aligned}
$$

h_{ref} is the curvature of reference orbit inside the dipole. The $(x, P_x; y, P_y; -t, \delta)$ are the canonical cooridinates of the H, and $P_x, P_y, P - z$ are respectively the

horizontal, vertical, and longitudinal mechanical momentums. A more convienent way is to expand the electron motion around the motion of a reference particle with total momentum P_0, so

$$\begin{aligned}
\frac{1}{P_0}H(x,P_x;y,P_y;-t,\delta;s) &= \\
H(x,p_x;y,p_y;-t,\delta;s) &= -(1+h_{\rm ref}x)\sqrt{((1+\delta)^2-(p_x-\frac{e}{P_0}A_x)^2-(p_y-\frac{e}{P_0}A_y)^2)}+\frac{e}{P_0}A_z(x,p_x;y,p_y, \\
&= -(1+h_{\rm ref}x)\sqrt{((1+\delta)^2-(p_x-\frac{1}{B\rho}A_x)^2-(p_y-\frac{1}{B\rho}A_y)^2)}+\frac{1}{B\rho}A_z(x,p_x;y,p_
\end{aligned}$$

with

$$p_x = \frac{P_x}{P_0} \tag{7}$$

$$p_y = \frac{P_y}{P_0} \tag{8}$$

$$p = \frac{P}{P_0} = \frac{\Delta P + P_0}{P_0} = (1+\delta) \tag{9}$$

$$\delta = \frac{\Delta P}{P_0} \tag{10}$$

$$B\rho = \frac{P_0}{e}(magneticrigidity) \tag{11}$$

With some operations (J. Bengtsson's linear transverse dynamics for sotrage ring with applications to the low energy antiproton ring (LEAR) at CERN; Tracy 2 manual.), the Hamitonian can be expressed with the canonical coordinates $(x,p_x;y,p_y;-ct,\delta;s)$ (Tracy-2 Manual, J. Bengtsson)

$$
\begin{aligned}
&H(x, p_x; y, p_y; -ct, \delta; s) \\
&\qquad = -(1 + h_{\text{ref}} x)(\sqrt{(1+\delta)^2 - (p_x - \frac{A_x}{B\rho})^2 - (p_y - \frac{A_y}{B\rho})^2} + \frac{A_z}{B\rho}) + \delta \\
&\qquad = -(1 + h_{\text{ref}} x)(\sqrt{(1+\delta)^2 - (p_x - \frac{A_x}{B\rho})^2 - (p_y - \frac{A_y}{B\rho})^2}) + h_{\text{ref}} x \frac{A_z}{B\rho} \\
&\qquad + \frac{A_z}{B\rho} + \delta \\
&\qquad = \underbrace{-(1 + h_{\text{ref}} x)\sqrt{(1+\delta)^2 - (p_x - \frac{A_x}{B\rho})^2 - (p_y - \frac{A_y}{B\rho})^2)}}_{\text{kinetic energy}} \\
&\qquad + \underbrace{\frac{x}{\rho} + \frac{x}{2\rho^2}}_{\substack{\text{Gemometic components due to the curvilinear coordinate inside the dipole}}} \\
&\qquad + \underbrace{\frac{A_z}{B\rho}}_{\text{magnet contribution}} \\
&\qquad + \underbrace{\delta}_{\substack{\text{due to the canonical coordinate } -ct \text{ instead of } -t}} \\
&A_z = A_z(x, p_x; y, p_y, s)
\end{aligned}
$$

The equation is the exact Hamitonian used in Tracy. The A_x are due to the longidutinal fringe field, this is an natural results due to the Maxwell Equation; h_{ref} is not zero only inside the dipoles.

For the system without fringe field of the magnet, or only consider the field

inside the body of the magnet, $A_x = 0$ and $A_y = 0$, so the Hamiltonian is

$$H(x, p_x; y, p_y; -ct, \delta; s) \tag{12}$$

$$= \underbrace{-(1 + h_{\text{ref}} x)\sqrt{(1+\delta)^2 - p_x^2 - p_y^2)}}_{\text{kinetic energy}} \tag{13}$$

$$+ \underbrace{\frac{x}{\rho} + \frac{x}{2\rho^2}}_{\text{Gemometic components due to the curvilinear coordinate inside the dipole}} \tag{14}$$

$$+ \underbrace{\frac{A_z}{B\rho}}_{\text{magnet contribution}} \tag{15}$$

$$+ \underbrace{\delta}_{\text{due to the canonical coordinate } -ct \text{ instead of } -t} \tag{16}$$

$$\tag{17}$$

From eqn. 103, it is easy to get the map of the particle inside different lattice elements.

7.1 Map of drift

In the drift, $A_z = 0$, and $h_{\text{ref}} = 0$, from eqn. 103, we can get the Hamiltonian inside a drift is

$$H(x, p_x; y, p_y; -ct, \delta; s) = -\sqrt{(1+\delta)^2 - p_x^2 - p_y^2)} + \delta \tag{18}$$

In large ring, in order to increase the tracking speed, we can expand the sqare root in eqn. 109 to get the approximate Hamitonian. That is,

$$H(x, p_x; y, p_y; -ct, \delta; s) = -(1+\delta)\left[1 - \frac{p_x^2 + p_y^2}{2(1+\delta)^2}\right] + \delta \tag{19}$$

$$= \frac{p_x^2 + p_y^2}{2(1+\delta)} \tag{20}$$

From eqn. 109, it is easy to get the map inside the drift for the **small ring**:

$$x^{'} = \frac{dx}{ds} = \frac{\partial H}{\partial p_{x0}} = \frac{p_{x0}}{\sqrt{(1+\delta_0)^2 - (p_{x0}^2 + p_{y0}^2)}} \tag{21}$$

$$y^{'} = \frac{dy}{ds} = \frac{\partial H}{\partial p_{y0}} = \frac{p_{y0}}{\sqrt{(1+\delta_0)^2 - (p_{x0}^2 + p_{y0}^2)}} \tag{22}$$

$$(-ct)^{'} = \frac{d(-ct)}{ds} = \frac{\partial H}{\partial \delta_0} = -\frac{1+\delta_0}{\sqrt{(1+\delta_0)^2 - (p_{x0}^2 + p_{y0}^2)}} + 1 \tag{23}$$

$$or \tag{24}$$

$$(ct)^{'} = \frac{1+\delta_0}{\sqrt{(1+\delta_0)^2 - (p_{x0}^2 + p_{y0}^2)}} - 1 \tag{25}$$

$$p_x = p_{x0} \tag{26}$$

$$p_y = p_{y0} \tag{27}$$

$$\delta = \delta_0 \tag{28}$$

From eqn. 110, it is easy to get the map inside the drift for the **big ring**:

$$x^{'} = \frac{dx}{ds} = \frac{\partial H}{\partial p_{x0}} = \frac{p_{x0}}{1+\delta_0} \tag{29}$$

$$y^{'} = \frac{dy}{ds} = \frac{\partial H}{\partial p_{y0}} = \frac{p_{y0}}{1+\delta_0} \tag{30}$$

$$(-ct)^{'} = \frac{d(-ct)}{ds} = \frac{\partial H}{\partial \delta_0} = -\frac{p_{x0^2} + p_{y0}^2}{2(1+\delta_0)} \tag{31}$$

$$or \tag{32}$$

$$(ct)^{'} = \frac{p_{x0^2} + p_{y0}^2}{2(1+\delta_0)} \tag{33}$$

$$p_x = p_{x0} \tag{34}$$

$$p_y = p_{y0} \tag{35}$$

$$\delta = \delta_0 \tag{36}$$

7.2 Map of mutipoles

From eqn. 103, we can get the Hamiltonian inside a multipole without fringe field is

$$H(x, p_x; y, p_y; -ct, \delta; s) \tag{37}$$

$$= \underbrace{-\sqrt{(1+\delta)^2 - p_x^2 - p_y^2} + \delta}_{\text{H drift}} \tag{38}$$

$$\underbrace{-h_{\text{ref}} x \sqrt{(1+\delta)^2 - p_x^2 - p_y^2}}_{\text{H kick}} \tag{39}$$

$$+ \underbrace{\frac{x}{\rho} + \frac{x}{2\rho^2} - \frac{A_z}{B\rho}}_{\text{H kick}} \tag{40}$$

In eqn. 128, the Hamiltonian is decomposed into two part, one is the Hamiltonian of the drift which is due to the kinetic energy of the system (section 15.1), another part is the kick due to the multipoles which will give the kick to the particle. Now we only need to focus on the Hamiltonian of the multipoles which induces the kick map:

$$H(x, p_x; y, p_y; -ct, \delta; s) \tag{41}$$

$$= -h_{\text{ref}} x \sqrt{(1+\delta)^2 - p_x^2 - p_y^2} \tag{42}$$

$$+ \frac{x}{\rho} + \frac{x}{2\rho^2} - \frac{A_z}{B\rho} \tag{43}$$

Similarly, by expand the sqare root in eqn. 128, we can get the approximate Hamiltonian of the multipole as

$$H(x, p_x; y, p_y; -ct, \delta; s) \tag{44}$$

$$= -h_{\text{ref}} x \sqrt{(1+\delta)^2 - p_x^2 - p_y^2} \tag{45}$$

$$+ \frac{x}{\rho} + \frac{x}{2\rho^2} - \frac{A_z}{B\rho} \tag{46}$$

$$= -h_{\text{ref}} x\delta + \frac{1}{2} h_{\text{ref}} h_B x^2 - \frac{A_z}{B\rho} + (h_B - h_{\text{ref}})x \tag{47}$$

where h_{ref} is the curvature due to the curlininear coordinates inside the dipole, $h_{\text{ref}} = 1/\rho_{ref}$; h_{bend} is due to the bending curvature of the dipole which is determined by the dipole field, $h_{\text{bend}} = 1/\rho_{bend}$. $h_{\text{ref}} \neq 0$ only inside dipoles.

In the dipole, $A_z \neq 0$, and $h_{\text{ref}} \neq 0$, so from eqn. 132, we can get the kick

map due to the dipole in the **small ring** which use the exact Hamiltonian:

$$x^{'} = \frac{dx}{ds} = \frac{\partial H}{\partial p_{x0}} = h_{\text{ref}} x_0 \frac{p_{x0}}{\sqrt{(1+\delta_0)^2 - (p_{x0}^2 + p_{y0}^2)}} \tag{48}$$

$$y^{'} = \frac{dy}{ds} = \frac{\partial H}{\partial p_{y0}} = h_{\text{ref}} x_0 \frac{p_{y0}}{\sqrt{(1+\delta_0)^2 - (p_{x0}^2 + p_{y0}^2)}} \tag{49}$$

$$p_x^{'} = -\frac{\partial H}{\partial y_0} = h_{\text{ref}} \sqrt{(1+\delta_0)^2 - p_{x0}^2 - p_{y0}^2} - h_{\text{bend}} - h_{\text{ref}} h_{\text{bend}} x - \frac{B_y}{B\rho} \tag{50}$$

$$p_y^{'} = -\frac{\partial H}{\partial y_0} = \frac{1}{B\rho}\frac{\partial A_z}{\partial y_0} = \frac{B_x}{B\rho} \tag{51}$$

$$(-ct)^{'} = \frac{d(-ct)}{ds} = \frac{\partial H}{\partial \delta_0} = -h_{\text{ref}} x_0 \frac{1+\delta_0}{\sqrt{(1+\delta_0)^2 - (p_{x0}^2 + p_{y0}^2)}} \tag{52}$$

$$or \tag{53}$$

$$(ct)^{'} = h_{\text{ref}} x_0 \frac{1+\delta_0}{\sqrt{(1+\delta_0)^2 - (p_{x0}^2 + p_{y0}^2)}} \tag{54}$$

$$\delta = \delta_0 \tag{55}$$

$$B\rho = \frac{p_0}{e} \tag{56}$$

where

$$\frac{\partial A_z}{\partial x} = -B_y \tag{57}$$

$$\frac{\partial A_z}{\partial y} = B_x \tag{58}$$

since

$$A_x = 0 \tag{59}$$

$$A_y = 0 \tag{60}$$

inside the body of the multipoles, and

$$\vec{B} = \Delta \times \vec{A}$$

$$\begin{array}{cc} \hat{x}\hat{y} & \hat{z} \\ \frac{\partial}{x}\frac{\partial}{y} & \frac{\partial}{z} \\ A_x A_y & A_z \end{array}$$

In the dipole, $A_z \neq 0$, and $h_{\text{ref}} \neq 0$, so from eqn. 135, we can get the kick

map due to the dipole in the **big ring** which use the exact Hamiltonian:

$$x^{'} = \frac{dx}{ds} = \frac{\partial H}{\partial p_{x0}} = h_{\text{ref}} x_0 \frac{p_{x0}}{1+\delta_0} \tag{61}$$

$$y^{'} = \frac{dy}{ds} = \frac{\partial H}{\partial p_{y0}} = h_{\text{ref}} x_0 \frac{p_{y0}}{1+\delta_0} \tag{62}$$

$$p_x^{'} = -\frac{\partial H}{\partial y_0} = h_{\text{ref}}\delta_0 - (h_{\text{bend}} - h_{\text{ref}}) - h_{\text{ref}} h_{\text{bend}} x - \frac{B_y}{B\rho} \tag{63}$$

$$p_y^{'} = -\frac{\partial H}{\partial y_0} = \frac{1}{B\rho}\frac{\partial A_z}{\partial y_0} = \frac{B_x}{B\rho} \tag{64}$$

$$(-ct)^{'} = \frac{d(-ct)}{ds} = \frac{\partial H}{\partial \delta_0} = \frac{-h_{\text{ref}} x_0}{1+\delta_0} \frac{-(p_{x0}^2 + p_{y0}^2)}{2(1+\delta_0)} \tag{65}$$

$$or \tag{66}$$

$$(ct)^{'} = -\frac{h_{\text{ref}} x_0}{1+\delta_0} \frac{(p_{x0}^2 + p_{y0}^2)}{2(1+\delta_0)} \tag{67}$$

$$\delta^{'} = 0 \tag{68}$$

For other multipoles (quadrupoles, sextupole, decapole, octupole, etc), the curvilinear coordinate is the cartesian coordinates, that is $h_{ref} = 0$, so from eqn. 139, the kick map for these multipoles in the small ring is:

$$x^{'} = \frac{dx}{ds} = \frac{\partial H}{\partial p_{x0}} = 0 \tag{69}$$

$$y^{'} = \frac{dy}{ds} = \frac{\partial H}{\partial p_{y0}} = 0 \tag{70}$$

$$p_x^{'} = -\frac{\partial H}{\partial y_0} = -(h_{\text{bend}} + \frac{B_y}{B\rho}) \tag{71}$$

$$p_y^{'} = -\frac{\partial H}{\partial y_0} = \frac{1}{B\rho}\frac{\partial A_z}{\partial y_0} = \frac{B_x}{B\rho} \tag{72}$$

$$(-ct)^{'} = \frac{d(-ct)}{ds} = \frac{\partial H}{\partial \delta_0} = 0 \tag{73}$$

$$or \tag{74}$$

$$(ct)^{'} = 0 \tag{75}$$

$$\delta^{'} = 0 \tag{76}$$

With $h_{ref} = 0$, from eqn: 152, the kick map for these multipoles (quadrupoles,

sextupole, decapole, octupole, etc) in the big ring is:

$$x^{'} = \frac{dx}{ds} = \frac{\partial H}{\partial p_{x0}} = 0 \tag{77}$$

$$y^{'} = \frac{dy}{ds} = \frac{\partial H}{\partial p_{y0}} = 0 \tag{78}$$

$$p_x^{'} = -\frac{\partial H}{\partial y_0} = -(h_{\text{bend}} + \frac{B_y}{B\rho}) \tag{79}$$

$$p_y^{'} = -\frac{\partial H}{\partial y_0} = \frac{1}{B\rho}\frac{\partial A_z}{\partial y_0} = \frac{B_x}{B\rho} \tag{80}$$

$$(-ct)^{'} = \frac{d(-ct)}{ds} = \frac{\partial H}{\partial \delta_0} = 0 \tag{81}$$

$$or \tag{82}$$

$$(ct)^{'} = 0 \tag{83}$$

$$\delta^{'} = 0 \tag{84}$$

From the eqn 139 to eqn. 168, it's clear that the kick map from the main body of dipoles are different in the small ring and big ring; while the kick map from the main body of other type of multipoles are the same in small rings and big rings.

8 Magnetic field

A. Dragt ....(Lie methods for acceleator.....)

9 FF of dipole

9.1 kick map

E. Forest .....

10 FF of quadrupole

10.1 kick map

11 Lattice file

In the lattice, RF cavity with the correct harmonic number must be defined!!! Otherwise Tracy will give the error message: Elem_GetPos: there are no kids in family 0 (). This obigatory is for the correct calculation of positive/negtive momentum compaction factor. To deactive the RF cavity, the cavity voltage can be defined as 0.

The followings are the rules to define a lattice file used in Tracy 3. The curvelinear coordinates are used. **The ideal particle or design particle sees the perfect magnetic field in all magnets, and its orbit is used as the reference orbit and base of the curvilinear coordinate**. Since the reference orbit is a curve inside the dipoles and a straight line in other lattice elements (drifts and the magnets except dipoles), so the length of the dipole is the the path length of the reference particle inside the dipole which is equal to $\rho * \theta$ where ρ is the bending radius of the dipole and *theta* is the bending angle with unit [rad], and all the other magnet lengths are the straight length of the magnets.

Due to the same reason, the curvature of the curvilinear cooridnates h = $1/\rho$ is not zero only inside dipole; in other magnets and drift h = 0, and the curvilinear coordinates goes to cartesian cooridnates. As a result, the independent variable s inside the dipole harmitonian is the arc length, while the staight length in other lattice elements.

11.1 Lattice

The n^{th} field component of the lattice element is defined as in Tracy (n = 1, 2, 3, 4 ...):

$$\begin{aligned} b_n &= \frac{1}{B\rho}\frac{1}{(n-1)!}\frac{\partial^{n-1}B_y}{\partial x^{n-1}}|_{x=0,y=0} \\ a_n &= \frac{1}{B\rho}\frac{1}{(n-1)!}\frac{\partial^{n-1}B_x}{\partial x^{n-1}}|_{x=0,y=0} \\ B\rho &= \frac{p_0}{e} \end{aligned}$$

$B\rho$ is the magnetic rigidity, p_0 is the design beam momentum, e is the electric charge. For example, for sextupole, n = 3, so b_3 and a_3 are defined as

$$\begin{aligned} b_3 &= \frac{1}{B\rho}\frac{1}{2}\frac{\partial^2 B_y}{\partial x^2}|_{x=0,y=0} \\ a_3 &= \frac{1}{B\rho}\frac{1}{2}\frac{\partial^2 B_x}{\partial x^2}|_{x=0,y=0} \end{aligned}$$

.

NOTES: In **AT** (Accelerator Toolbox) and **BETA** code, the definition of are the same as in **Tracy**. While In **MAD8**, **MADX** and **ELEGANT**, the order of the field compoent start from 0, that is n = 0, 1, 2, 3, ..., and the n^{th} field strength components of the lattice element b_n and a_n are defined as

$$\begin{aligned} b_n &= \frac{1}{B\rho}\frac{\partial^n B_y}{\partial x^n}|_{x=0,y=0} \\ a_n &= \frac{1}{B\rho}\frac{\partial^n B_x}{\partial x^n}|_{x=0,y=0} \\ B\rho &= \frac{p_0}{e} \end{aligned}$$

For example, for sextupole, n = 3, its field component b_2 and a_2 are defined as

$$b_2 = \frac{1}{B\rho}\frac{\partial^2 B_y}{\partial x^2}|_{x=0,y=0}$$
$$a_2 = \frac{1}{B\rho}\frac{\partial^2 B_x}{\partial x^2}|_{x=0,y=0}$$

.

11.2 Syntax

Every line embraced by is comment line. For example:
*****drift space *****

Each sentence is ended by ; or no punctuation. Tracy is not sensitive to capital/small letters in the lattice. User can define any lattice element with any valid name (but must start with a character) they want, but the element type is fixed. For the lattice of the ring, the definition of RF cavity is mandatory, and the harmonic number of the RF cavity is also mandatory; for the lattice of the linac, the definition of the RF cavity is optional.

11.3 Variables

User can define the variables in the lattice file. For example:

Intmeth = 4;

Then when the code is running, the variable intmeth in the lattice file will be replaced by 4.

Tracy accept math operation. For example,

DL1: drift, L = 0.1*2 + 0.1-0.01;

11.4 Start line

The lattice file must begin with the sentence:

define lattice;

This definition is mandatory.

11.5 Global variables

After define the ring, user needs to define the system parameters of the lattice: Energy, the beam energy with unit [GeV]. dP, the relative momentum offset of the particle. CODeps, the convergence for the algorism to find the closed orbit. For example: Energy = 2.739; dP = 1.0d-10; CODeps= 1.0d-15; These definitions are mandatory.

11.6 DRIFT

Name: drift, L = ⟨. . .⟩;

Table 1: Parameters of dipole in the lattice file.

Parameter Name	Units	Default	Description
L	[m]	0.0	Length.

The definition of the length of a drift is mandatory.

Example:

SD1a: drift, L= 0.900000;

11.7 DIPOLE

Element_name: bending, L = ⟨...⟩, T = ⟨...⟩, **T1** = ⟨...⟩, **T2**=⟨...⟩, **H1** = [...], **H2**= [...], **gap** = [...], **edge_effect1** = [...], **edge_effect2** = [...], **K** =[...], **method** = ⟨...⟩, **N** = ⟨...⟩;

For example:
beta_gap=37e-3; tracy_gap=beta_gap*2*0.724;

BEND1 : **bending**, **L** = 1.05243, **T** = 11.25, **T1** = 5.5906, **T2** = 5.67658, **K** = 0.00204,gap=tracy_gap, **edge_effect1** = 1, **edge_effect2** = 1, **N** = 4, **method** = intmeth;

The parameters of "bending" are optional, the default values for the missing parameters are 0, and the default value for "method" is also 0.

Attention:

- The bending magnet in Tracy is a symplectic element. The independent variable s inside the dipole harmitonian is the arc length, so the length of the dipole is defined the curved path length of the design particle inside the dipole.
- One can refer to the definition of $T1$,$T2$, $H1$, $H2$ on page 116 of SLAC-75 for more information.
- In the lattice files of ELEGANT, MADX and BETA, the dipole angles are defined with the unit [rad]. But in TRACY, the unit is [degree].
- BETA dipole gap is the Elegant gap; but Tracy dipole gap is different from the definition in both BETA and Elegant. Tracy definition of dipole with fringe field:
  beta_gap=0.04; tracy_gap=beta_gap*2*1/6; Due to the Tanh-like fringe field, K1=0.2, and tracy_gap = beta_gap *2* FINT (K brown para.) tracy_gap = beta_gap*2*0.348;

  DIP: bending, L=0.27646,T= 45, T1=0, T2=0,gap=tracy_gap,edge_effect1=1,edge_effect2=1,method=intmeth,N=4;

Table 2: Parameters of dipole in the lattice file.

Parameter Name	**Units**	**Default**	**Description**
L	[m]	0.0	Curved path length of the design particle inside the dipole. L = $\rho * \theta$, where ρ is the bending radius of the dipole, and θ is the bending angle of the dipole.
T	[degree]	0.0	Total bending angle
T1	[degree]	0.0	Entrance angle
T2	[degree]	0.0	Exit Angle
K	L $\neq$ 0, m^{-2}. L = 0, m^{-1}.	0.0	L $\neq$ 0, quadrupole field strength; L = 0, integrated field strength.
Method	-	4	1,2,4. Symplectic integration method. The value "1" means 1^{st} order, "2" means 2^{nd} order, and "4" means 4^{th} order.
Gap	[m]	0.0	Distance between two poles of the dipole, the gap size determine the fringe field. If the gap size is 0, then the dipole has no fringe field (??????).
edge_effect1	-	0	1/0. "1" to turn on the edge focusing effect and the fringe field at the entrance of the dipole. "0" to turn off the two effects.
edge_effect2	-	0	1/0. "1" to turn on the edge focusing effect and the fringe field at the exit of the dipole. "0" to turn off the two effects.
N	-	1	number of cut slices.

Elegant definition of dipole with dipole fringe field:
DIP: csbend,N_KICKS=100,INTEGRATION_ORDER=4,L=0.27646,angle= 0.785398,
E1=0, E2=0,hgap=0.02,Fint=0.348,nonlinear=1,edge1_effects=2,edge2_effects=2,edge_order=2;

11.8 QUADRUPOLE

Definition:

Element_Name: quadrupole, L = [...], tilt = [...], K =[...], FF1 = [...],
FF2= [...], FFscaling = [...], method = [...], N= ⟨...⟩;

Table 3: Parameters of quadrupoles in the lattice file.

Parameter Name	**Units**	**Default**	**Description**
L	[m]	0.0	Length of the quadrupole
Tilt	[degree]	0.0	tilt angle of the quadrupole. If tilt is non-zero, then the quadrupole is a skew quadruple.
K	L $\neq$ 0, [m^{-2}]. L = 0, [m^{-1}].	0.0	L $\neq$ 0, Gradient $\frac{\frac{\partial B_y}{\partial x}}{B\rho}$; L = 0, integrated field strength $\frac{\frac{\partial B_y}{\partial x}L}{B\rho}$.
FF1	-	0	1 or 0. 1: to avtive fringe field at the left edge. QuadFringeOnFlag should be set in the Tracy input file. 0: left fringe field off.
FF2	-	0	1 or 0. 1: to avtive fringe field at the right edge. QuadFringeOnFlag should be set in the Tracy input file. 0: right fringe field off.
FFscaling	-	1	Scaling factor of the dipole fringe field.
Method	-	4	1,2,4. Order of symplectic integration method. Value "1" means 1st order, "2" means 2nd order, and "4" means 4th order.
N	-	1	Pieces of the quadrupole to be cut when it is treated in the code.

Example:

Nq=8/2; dgsurg=1.00; dgsurgL=1.00; quadfringe=1.0; LQC=0.3602;

QP1a: quadrupole, L=LQC/2, K= -1.073038*dgsurg, FF1=quadfringe, FF2=0, FFscaling =1, method=intmeth, N=Nq;

The parameters of quadrupole are optional, the default value for method is 4, the default value for FFscaling is 1, the default value for the other parameters are 0.

11.9 SKEW QUADRUPOLE

The skew quadrupole is a quadrupole with a 45 [degree] tilt angle. For example:

QT: quadrupole, tilt=45.0, K= 0.0, method=intmeth, N=1;

NOTES: If skew quadrupoles are defined in the lattice file with a name skewquad, User must specify the name of skew quadrupole in the tracy input file using the commands:

qt skewquad

11.10 SEXTUPOLE

Element_Name: sextupole, L = [...], K = [...], FF1 = [...], FF2=[...],method = [...], N=[...];

Example:

NqSx=1; coef=1.0/0.16; method4sextu = 4; sextfringe = 0;

SX1: sextupole, L=0.16, K = 1.719190*coef, method=method4sextu, N = NqSx, FF1=sextfringe, FF2=sextfringe;

The parameters of sextupole are optional, the default value for method is 4, the default value for the other parameters is 0.

NOTES:

In **AT** (Accelerator Toolbox) and **BETA** code, the european notation of the order of magnetic field is used. That is n = 1, 2, 3, ...n; and the definition of K are the same as in **Tracy**. While In **MAD8**, **MADX** and **ELEGANT**, the U.S. notation of the order order of the field is used, that is n = 0, 1, 2, 3, ..n. For sextupole, n = 2, and its gradient is defined as

$$K_2 = \frac{1}{B\rho}\frac{\partial^2 B_y}{\partial x^2}|_{x=0,y=0}$$

11.11 MULTIPOLE

Element_Name: Multipole, L= [...], T =[...], T1=[...], T2=[...], tilt=[...], HOM = (i, $\langle\ b_i\ \rangle$, $\langle\ a_i\ \rangle$, j, $\langle\ b_j\ \rangle$, $\langle\ a_j\ \rangle$,.n, $\langle\ b_n\ \rangle$, $\langle\ a_n\ \rangle$), N =[...]$\langle$, method =[...];

Table 4: Parameters of sextupoles in the lattice file.

Parameter Name	**Units**	**Default**	**Description**
L [m]		0.0	Length
K	L $\neq$ 0, [m^{-3}]. L = 0, [m^{-2}].	0.0	If L $\neq$ 0, $K = \frac{1}{2}\frac{1}{B\rho}\frac{\partial^2 B_y}{\partial x^2}|_{x=0,y=0}$, $B\rho$ is the magnetic rigidity, p_0 is the design beam momentum, e is the electric charge. If L = 0, K is the integrated field strength $L\frac{1}{2}\frac{1}{B\rho}\frac{\partial^2 B_y}{\partial x^2}|_{x=0,y=0}$
FF1	-	0	**1** or **0**. **1**: tringe on the fringe field at the left edge. **0**: tringe off the fringe field at the left edge.
FF2	-	0	**1** or **0**. **1**: tringe on the fringe field at the right edge. **0**: tringe off the fringe field at the right edge.
Method	-	4	**1**,**2**,**4**. Order of symplectic integration method. **1**: 1st order; **2**: 2nd order; **4** 4th order.
N	-	1	Pieces of this element to be cut, used in the symplectic integration.

Table 5: Parameters of the multipoles.

Parameter Name	**Units**	**Default**	**Description**
L	[m]	0.0	Length
T	[degree]	0.0	Total bending angle
T1	[degree]	0.0	Entrance angle
T2	[degree]	0.0	Exit Angle
Tilt	[degree]	0.0	Rotation angle of the quadrupole; if Tilt is non-zero, then the quadrupole is a skew quadruple.????
HOM	-	(0, 0, 0)	Multipole field components of the element. The format is n, b_n, a_n, etc. n is order of the multipole field in U.S. notation (?????), n=1 (dipole field), n=2 (quadrupole field), n=3 (sextupole field), n=4 (octuple field), n=5 (decuple field). b_n is n^{th} component of the upright multipole field with the unit $[\frac{1}{m^{-n}}]$; a_n is n^{th} component of the skew multipole field with the unit $[\frac{1}{m^{-n}}]$.
N	-	1	Pieces of this element to be cut
Method	-	4	1,2,4. Symplectic integration method. 1 is 1^{st} order, 2 is 2^{nd} order, 4 is 4th order.

The multipole field components a_n and b_n are defined as

$$b_n = \frac{1}{B\rho}\frac{1}{(n-1)!}\frac{\partial^{n-1}B_y}{\partial x^{n-1}}|_{x=0,y=0} \quad (85)$$

$$a_n = \frac{1}{B\rho}\frac{1}{(n-1)!}\frac{\partial^{n-1}B_x}{\partial x^{n-1}}|_{x=0,y=0} \quad (86)$$

$$B\rho = \frac{p_0}{e} \quad (87)$$

$B\rho$ is the magnetic rigidity, p_0 is the design beam momentum, e is the electric charge.

If the magnetic field $\vec{B^{tip}}$ is measured at the location with the radius r_0, then b_n and a_n are defined as

$$b_n = \frac{1}{B\rho}\frac{B_y^{tip}}{r_0^{n-1}}$$

$$a_n = \frac{1}{B\rho}\frac{B_x^{tip}}{r_0^{n-1}}$$

$$\vec{B^{tip}} = B_y^{tip}\hat{y} + B_x^{tip}\hat{x}$$

The n^{th} order magnetic field is

$$B_y^n + i * B_x^n = (b_n + i * a_n) * (x + i * y)^n \quad (88)$$

The total magnetic field with 1^{st} to n^{th} order field components can be calculated using

$$B_y + i * B_x = (b_n + i * a_n) * (x + i * y) * (x + i * y)^{n-1} + O(n-1) \quad (89)$$

$$= [(b_n * x - a_n * y) + b_{n-1} + i * (a_n * x + b_n * y + a_{n-1})] * (x + i * y)^{n-1} + O(n-2) \quad (90)$$

$$... \quad (91)$$

The eqn. 89 is used in Tracy3 to calculate the effective fields of the magnet. This equation clearly shows that, all the field errors gives the same effect the beam: horizontal focusing and vertical decusing. But since the field strength of the n^{th} order magnet is proportial to the n^{th} order of x/y and x/y at the entrance of the magnet is small, so the higher n is, the weak of the n^{th} order multipole field is.

Example 1:

B: multipole, L=0.70, T=10.0, T1=5.0, T2=5.0, HOM = (1, -1.0, 0), N=8, Method=2;

In this example, the multipole is a dipole with field component b_1 = -1.0, or the field strength is B_{y0}, and the field direction is down. This field gives a horizontal focusing to the electron.

Example 2:

QF: multipole, L=0.70, HOM = (1, 2.50, 0.0, 4, 1.01e7, 0.0), N=8, Method=4;

In this example, the multipole is a dipole with 4^{th} order upright multipole filed errors (octupole), and $b_4 = 1.01e7$.

To add multipole errors in the defined lattice files, user can also define the multipoles inside an external file (Section 20) (???? To be checked, can this routine mesured with the general one???).

NOTES:

In **AT** (Accelerator Toolbox) and **BETA** code, the definition of are the same as in **Tracy**. While In **MAD8**, **MADX** and **ELEGANT**, the order of the field compoent start from 0, that is n = 0, 1, 2, 3, ..., and the n^{th} field strength components of the lattice element b_n and a_n are defined as

$$\begin{aligned} b_n &= \frac{1}{B\rho}\frac{\partial^n B_y}{\partial x^n}|_{x=0,y=0} \\ a_n &= \frac{1}{B\rho}\frac{\partial^n B_x}{\partial x^n}|_{x=0,y=0} \\ B\rho &= \frac{p_0}{e} \end{aligned}$$

11.12 wiggler (To be updated.)

Element_Name: Wiggler, L = ⟨ length ⟩,BoBrhoV = ⟨ B/Brho ⟩, BoBrhoH = ⟨ B/Brho ⟩, Lambda = ⟨ period ⟩,kxV= ⟨ [m] ⟩,kxH = ⟨ [m] ⟩,phi = ⟨ phase ⟩, harm(n, kxV, BoBrhoV, kxH, BoBrhoH, phi), N = ⟨ no of integration steps ⟩, Method = ⟨ method ⟩;

Table 6: The parameters of wigglers in a lattice file.

Parameter Name	**Units**	**Default**	**Description**
L	[m]	0.0	Length
BoBrhoV	???	???	the normalized vertical field
BoBrhoH	???	???	the normalized horizontal field
Labmda	[m]	???	period length
kxV	[m]	???	???
kxH	[m]	???	???
phi	[degree]	???	???
harm	???	???	???
N	-	-	No of integration steps
Method	-	-	Symplectic integration method. 1 is 1st order, 2 is 2nd order, 4 is 4th order.

Example 1:

U143: wiggler, L=4.80, K=0.5, Lambda=0.15, N=20, Method=0;

Example 2:

EPU: wiggler, L=4.80, Lambda=0.15, N=20, Method=0, harm=(3, kxV_3, BoBrhoV_3, kxH_3, BoBrhoH_3, phi_3);

11.13 FIELD MAP (To be updated..)

Element_Name: Fieldmap, L = ⟨ length [m] ⟩, N = ⟨ no of integration steps ⟩, file1 = ⟨ file name (lower case) ⟩

;

Table 7: Parameters of field map in the lattice file.

Parameter Name	Units	Default	Description
L	[m]	0.0	Length
L	[m]	0.0	Length of the field map
N	-	???	the number of integration steps
file1	-	???	field map file

Example:

FM: Fieldmap, L = 1.0, N = 20, file1 = "U19_Bxyz.dat";

11.14 INSERTION DEVICE

Element_Name: insertion, scaling1 = 1/0, scaling2=1/0,method = interpolation_method, N=Number of slice, file1 = name of the file with 1st order radia map, file2 = name of the file with 2nd order radia map;

Example:

WIGSLIC: insertion, N = 10, scaling1=1.0, scaling2=1.0, method=2, file1="/home/sources/physmach/tracy2.7/w150g11pole60_oppose_radia_pour_tracy.txt", file2= "/home/sources/physmach/brunelle/tracy-2.7/w150g11pole20_fin.dat";

All the parameters for insertion is optional, the default value for scaling1 and scaling2 are 1, the default method is 3 which means spline interpolation, the default N is 1, the default values for all the other parameters are 0.

11.15 RF CAVITY

Element_Name cavity, Frequency = RF frequency, Voltage = RF voltage, Phase = synchrotron Phase, harnum = harmonic number of the RF cavity;

Example:

CAV: Cavity, Frequency = 499.95e6, Voltage=1.22e6, phase = 30, harnum=328;

Table 8: Parameters of the insertion devices.

Parameter Name	Units	Default	Description
scaling1	-	???	scaling factor for the 1st order field map
scaling2	-	???	scaling factor for the 2rd order field map
method	-	???	1, 3. The order of symplectic interpolation method. 1 is linear interpolation, 3 is spline interpolation.
N	-	???	pieces of this element is cut when it is treated in the code
file1	-	???	The 1st order of insertion device field are read from the files generated by RADIA. If user does not specify the file name with the file path, then the code will look for the files in the current working directory. The path of the Radia map file must be in small letters, otherwise the code cant find the file.
file2	-	???	The 2nd order of insertion device field are read from the files generated by RADIA. If user does not specify the file name with the file path, then the code will look for the files in the current working directory. The path of the Radia map file must be in small letters, otherwise the code cant find the file.

Table 9: Parameters of RF cacity.

Parameter Name	Units	Default	Description
frequency	[Hz]	???	RF frequency
voltage	[Volt]	???	RF voltage
phase	[degree]	???	synchrotron phase
harnum	-	???	harmonic number

The harmonic number of the RF cavity is mandatory, and the other parameters of cavity are optional, the default values are 0.

11.16 CORRECTOR

Element_Name: corrector, horizontal/vertical, method = integrated method;

Table 10: Parameters of correctors.

Horizontal / vertical	-	???	horizontal: horizontal corrector; vertical: vertical corrector
method	-	???	1, 2, 4. Order of symplectic integration method. Value 1 means 1st order, 2 means 2nd order, and 4 means 4th order

Example 1:

CH: corrector, horizontal, method=intmeth;

It defines a horizontal corrector.

Example 2:

CV: corrector, vertical, method=intmeth;

It defines a vertical corrector.

The parameters of corrector are optional, the default value for method is 0.

NOTES: For a lattice with correctors, user must specify the name of corrector in the Tracy input file with the commands:

h_corr HCM

or

v_corr VCM

Here HCM is the name of the corrector defined in the lattice for horizontal orbit correction (????); VCM is the name of the corrector defined in the lattice for vertical orbit correction.

11.17 MARKER

Element_Name: marker;

11.18 BEAM POSITION MONITOR (To be updated)

BPM is a special marker in the lattice; the symbol name must be BPM (???). User can define the BPM as:????

BPM: type;

Normally Its type is defined as Marker type, but in order to include the misalignment error of BPM into the lattice, it must be defined as Beam Position Monitor type which is in fact multipole type, since only the element with multipole type is saved with displacement error, field error, etc.

NOTES: For lattice with BPMs, user must specify the name of BPM in the Tracy input file with the command:

bpm beaPosMonitor H

ere beaPosMonitor is the name of the BPMs defined in the lattice.

11.19 GIRDER

Girder is a special element, its the girder used in the real machine to support the magnetic elements and other elements. It is defined as: Symbol: type;

Normally Its type is defined as Marker type, but in order to include the misalignment error of girder into the lattice, it must be defined as multipole type, since only the element with multipole type is saved with displacement error, field error, etc. For convenience, its better to define the beginning of the girder and also the end of the girder, and the elements between the beginning and end of the girders are the elements who are put on the girder in the real machine.

Notice: For lattice with girders, User must specify the name of girder in the input file *.prm with the commands: gs Girder_Start ge Girder_End

Here Girder_Start is the name of the start of girder defined in the lattice; Girder_End is the name of the end of girder defined in the lattice.

11.20 SOLENOID...TO BE UPDATED...

11.21 ELEMENT BLOCK

To construct the element block, use the following format: Symbol: elem1, elem2,., block1,block2; Here Symbol is the name of the element block, and elem1, elem2, block1, block2 are the element or sub element blocks in this element block. If there are N the same element/block subsequently, user can use N*element/block to simply the definition. For example: SINJ: SD1a, ssep, 3*SEP,esep,SD1c,eHU600,SD1d; In this example element block, there are 9 elements/blocks, and 3 elements/blocks SEP subsequently.

11.22 LINE

User can define the cell structure using the command:

CELL : ⟨ block name ⟩, SYMMETRY=⟨ symmetry ⟩;

⟨ block name ⟩ is the name of a block; ⟨ symmetry ⟩ is the number of super symmetry or the number of the block in the ring.

Example:

CELL: BL1, Symmetry=12;

This example defines the cell with block BL1, and the number of super symmetry is 12. The output of the Tracy3 with symmetry large than 1 will give the tunes and chromaticities in one symmetric period.

11.23 RING

To define the ring, use the command: RING: elem, block. Its similar to define an element block, but must with the fixed symbol name RING. For example: RING: DEBUT,SUP1,SUP2,SUP3,SUP4,CAV,FIN; 5.1.23 End line To end the lattice file, user needs to use the following command at the end of the lattice file: End; This command is mandatory.

12 Commands

The following commands turn on the boolean flags in the code to set the machine parameters and carry on different calculations. All these commands are optional; user can choose whichever they need. If user wants to use the flag, they can write the flag in the script, if they do not want to use it, they can delete or comment out (add # at the beginning of command line) the flag. The Boolean flags in the user input script have the following features: If the flags are not active, then the default values for all the boolean commands are false. The code will execute the command according to the sequence defined in the input script *.prm. For example, FitTune4Flag qp7a qp7b qp9a qp9b 18.18 10.28 ReadMultipoleFlag FitTune4Flag qp7a qp7b qp9a qp9b 18.202 10.317 The code will fit tunes to the target tunes (18.18 10.28), and then read multipole field errors into the lattice, then fit the tunes to a new set of values (18.202 10.317). User can define the same Boolean commands as often as they want in the same input script; but user can only define maximum 500 commands in one input script. The user defined script can be used both for the non-parallel and parallel version Tracy; the output files are the same for both versions.

12.1 QuadFringeOnFlag

To activate quadrupole fringe field, use the command:
QuadFringeOnFlag
in the *.prm script. With this command, user can define the fringe field at the entrance or exit of the quadrupole together with the command FF1 = 1 or FF2 = 1 of the quadrupoles which are defined in the lattice file; if FF1 or FF2 not equals to 1, then there is no fringe field at the entrance or exit of the quadrupoles even if QuadFringeOnFlag is active in the *.prm file. This flag is a global flag, if user set this flag in the input script, it will always have effects until this flag is reset.

12.2 QuadFringeOffFlag

To deactivate quadrupole fringe field, use the command: QuadFringeOffFlag With this command, user can deactivate the fringe field at the entrance and exit of the quadrupole, even if FF1 = 1 or FF2 = 1 for the quadrupole in the lattice file. This flag is a global flag, if user set this flag in the input script, it will always have effects until this flag is reset.

12.3 RFvoltageFlag

User can reset the RF voltage by setting RFvoltageFlag to replace the value of RF voltage which is defined in the lattice. For example: RFvoltageFlag 3000000 Here RFvoltageFlag is the name of the keyword command, 3000000 is the value of RF voltage with the unit [volt]. If the ring has more than one RF cavities, the related parameters are defined as the total values for one RF cavity.

12.4 PrintTrackFlag

To print the coordinates tracked around COD at each element to a file, use the command: PrintTrackFlag track_file x px y py delta ctau nturn

For example: PrintTrackFlag track.out 0.001 0.0 0.0 0.0 0.0 0.0 50

The parameters and the default values of PrintTrackFlag are shown in Table .

Table 11:

Table Parameters of command "PrintTrackFlag". Name Description Default value Unit track_file File to save the tracked coordinates around each lattice element track.out - x Start tracking horizontal coordinate 0.001 [m] px Start tracking horizontal canonical momentum px normalized by reference momentum p0, that is: px = px/p0. Normally px is approximate as horizontal deviation with unit [rad]. 0.0

- y Start tracking vertical coordinate 0.0 [m] py Start tracking vertical canonical momentum py normalized by reference momentum p0, that is: py = py/p0. Normally px is approximate as vertical deviation with unit [rad]. 0.0

- delta Start tracking relative energy offset 0.0 - ctau Start tracking longitudinal coordinate 0.0 [m] nturn Number of turn for tracking 50 -

12.5 PrintTrackElemFlag

To print the coordinates tracked (NOT around COD) at a certain element to a file, use the command: PrintTrackFlag track_file x px y py delta ctau nelem1 nelem2

For example: PrintTrackFlag track.out 0.001 0.0 0.0 0.0 0.0 0.0 50 1 2

The parameters and the default values of PrintTrackElemFlag are shown in Table .

Table 12:

Table Parameters of command "PrintTrackElemFlag". Name Description Default value Unit track_file File to save the tracked coordinates around each lattice element track.out - x Start tracking horizontal coordinate 0.001 [m] px Start tracking horizontal canonical momentum px normalized by reference momentum p0, that is: px = px/p0. Normally px is approximate as horizontal deviation with unit [rad]. 0.0

- y Start tracking vertical coordinate 0.0 [m] py Start tracking vertical canonical momentum py normalized by reference momentum p0, that is: py = py/p0. Normally px is approximate as vertical deviation with unit [rad]. 0.0

- delta Start tracking relative energy offset 0.0 - ctau Start tracking longitudinal coordinate 0.0 [m] nelem1 index of start lattice element for tracking

nelem2 index of end lattice element for tracking

12.6 PrintTwissFlag

Print Twiss parameters to a user defined file

With the command PrintTwissFlag, Tracy 3 will print the Twiss parameters to a user defined file. The format is: PrintTwissFlag user_defined_file

If user use the command PrintTwissFlag but without define the file name, then the code will print the Twiss parameters to a default file twiss.out.

12.7 PrintCODFlag

Print COD (Close Orbit Distortion) to a user defined file

With the command PrintCODFlag, the code will print the close orbit distortion to a user defined file. The format is: PrintCODFlag user_defined_file If user use the command PrintCODFlag but without defining the file name, then the closed orbit will be printed to the default file printcod.out. In Tracy 3, close orbit file cod.out is automatically generated after reading the lattice. In the outuput file of the command PrintCODFlag, the data is saved in the following format: # i name s code betax nux betay nuy xcod ycod dSx dSy dipx dipy # [m] [m] [m] [mm] [mm] [mm] [mm] [mrad] [mrad] # Where # denotes the command line, the meanings of the above parameters are shown in Table .

Table 13:

Table Parameters in the output file of close orbit. Name Description Unit i Number of the element - name Element name defined in lattice - s Longitudinal length [m] code Symbol for the element: 0.5 dipole -1.0 defocusing quadrupole 1.0 focusing quadrupole -1.5 defocusing sextupole 1.5 focusing sextupole 0 other element

- betax Horizontal beta function [m] nux Horizontal tune - betay Vertical beta function [m] nuy Vertical tune - xcod Horizontal closed orbit distortion [mm] ycod Vertical closed orbit distortion [mm] dSx Horizontal displacement of the element [mm] dSy vertical displacement of the element [mm[dipx Horizontal dipole strength of the element [mrad] dipy vertical dipole strength of the element [mrad]

12.8 ReadChamberFlag

Read vacuum chamber setting from an external file

To read the vacuum chamber from the user defined chamber file, use the command: ReadChamberFlag Chamber_example.dat In the file Chamber_example.dat, user can specify the vacuum limit at the different region of the lattice; the format of the chamber file is given in section .

12.9 ReadfefileFlag

Read lattice element multipole field errors from an external file

To read the multipole field errors of the lattice elements from the user defined file, use the command: ReadfefileFlag dip.fe Tracy will read the systematic and random multipole field errors of the lattice elements defined in the file dip.fe, and then replace the corresponding field components of the elements with the new field errors. The formats to specify the systematic and random multipole field errors in a file are given in section Error: Reference source not found.

12.10 ReadaefileFlag

Read lattice element misalignment errors from a file

To read the misalignment error of the lattice elements from the user defined file, use the command: ReadaefileFlag dip.ae Tracy will read the systematic and random misalignment errors of the lattice elements from the file dip.fe, and replace the misalignment errors of the corresponding components of the elements. The formats to define the systematic and random misalignment errors of the lattice elements in a file are given in section .

12.11 CODCorrectFlag

Closed orbit (COD) correction

The orbit distortion is corrected using SVD (Singular Value Decomposition) method in Tracy 3. In order to do orbit correction, user needs to call the command CODCorrectFlag and then specify the following parameters in the user defined *.prm file. Specify the element names of horizontal, vertical correctors, and beam position monitors used in the orbit correction as the following examples: h_corr HC v_corr VC BPM bpm User also need to specify the states of the correctors to trigger on/off the correction using the following parameters: hcorr_file hcorr_56nom.state vcorr_file vcorr_56nom.state In the file hcorr_56nom.state, a list of numbers (1 or 0) are given to the horizontal correctors, according to the sequence in the lattice; 1 means the corresponding corrector is used for horizontal orbit correction, 0 means this corrector is not used in the horizontal orbit correction. The definition rules of vertical corrector states in vcorr_56nom.state are the same as vcorr_56nom.state.

This parameter defines number of iterations to correct the orbit distortion, this value should be an integer number not smaller than 1. n_orbit 3 This parameter defines number of singular values in H-plane, must be not larger than the number of correctors used for orbit correction nwh 60 This defines number of singular values in V-plane, must be not larger than the number of correctors used for orbit correction nwv 60

In Tracy 3, during the closed orbit correction: 1) Beam response matrices between beam position monitors and horizontal/vertical correctors are calculated and written to the files svdh.out/ svdv.out, respectively. The maximum number of horizontal/vertical correctors used for orbit correction is 250. 2) The code corrects the closed orbit distortion. 3) Horizontal and vertical orbits at the locations of all beam position monitors during the correction are saved to the files horbit.out and vorbit.out, respectively. 4) A file OrScanFile.out will be saved with the summaries of the mean and RMS values of the orbits before and after correction. 5) Finally, Twiss parameters, closed orbit distortion at the lattice elements are saved to a summary file summary_miserr_codcorr.out, the format of this file is explained in Table .

12.12 TuneTracFlag

Get tunes by tracking

To get tunes by tracking, use command: TuneTracFlag The tunes obtained by tracking are printed on the screen.

12.13 ChromTracFlag

Get chromaticities by tracking

To get chromaticity by tracking, use command: ChromTracFlag The chromaticities obtained by tracking are printed on the screen.

12.14 AmplitudeTuneShiftFlag

Tune shift with amplitude

To calculate tune shift with amplitude, one needs to use the following command: AmplitudeTuneShiftFlag nudx_file nudz_file nxpoint nypoint nturn xmax ymax delta For example: AmplitudeTuneShiftFlag nudx.out nudz.out 50 30 516 0.035 0.02 0.0

The meanings of parameters and default values of command AmplitudeTuneShiftFlag are shown in Table . If user uses the command AmplitudeTuneShiftFlag without parameters, then the code will use all the default values.

Table 14:

Table Parameters of the command to calculate tune shift with amplitude Parameters Meaning Default values nudx_file File to save the calculated tune shift with horizontal amplitude nudx.out nudz_file File to save the calculated tune shift with vertical amplitude nudz.out nxpoint Number of points in horizontal direction 31 nypoint Number of points in vertical direction 21 nturn Number of turns to track tune 516 xmax Maximum amplitude of x with the unit [m] 0.025 ymax Maximum amplitude of y with the unit [m] 0.005 delta Energy offset of the particle 0.0

12.15 EnergyTuneShiftFlag

Tune shift with energy

To calculate tune shift with energy, one needs to use the command: EnergyTuneShiftFlag nudp_file npoint nturn deltamax For example: EnergyTuneShiftFlag nudptest.out 31 1026 0.06 The meaning of parameters and default values of this command are shown in Table . If user uses command EnergyTuneShiftFlag without parameters, then the code will use all the default values.

Table 15:

Table Parameters of the command to calculate tune shift with energy parameter meaning default value nudp_file File to save the calculated tune shift with energy nudp.out npoint Number of points 31 nturn Number of turns for tracking 516 deltamax Maximum energy offset of the particle 0.06

12.16 FmapFlag

Frequency map analysis for on momentum particle. Track the coordinates (x, px, y, py, delta, ctau) of the particle, the coordinates (x, px, y, py) are tracked around the closed orbit; while (delta, ctau) is tracked around the zero orbit.

The grid of the particle is x= 1e-6 + xcod xmax + xcod, with the step size xmax/nturn, z= 1e-6 + zcod zmax + zcod, with the step size ymax/nturn; px = 0 + pxcod; pz = 0 + pzcod; ctau = 0; delta = set value.

To do frequency map analysis for the on momentum particle, use the command: FmapFlag fmap_file nxpoint nypoint nturn xmax ymax delta diffusion or FmapFlag fmap_file nxpoint nypoint nturn xmax ymax delta diffusion printloss

For example: FmapFlag fmap.out 31 21 516 0.025 0.005 0.0 true or FmapFlag fmap.out 31 21 516 0.025 0.005 0.0 true true

The meaning of parameters and default values of this command are shown in Table . If user uses command FmapFlag without parameters, then the code will use all the default values.

Table 16:

Table Parameters of the command to do frequency map analysis for on momentum particle. Parameters meaning Default values fmap_file File to save the calculated frequency map analysis fmap.out nxpoint Number of points in the horizontal direction 31 nypoint Number of points in the vertical direction 21 nturn Number of turns for tracking; if "diffusion" is true, then the tunes will calculated in the first Nturn and then the second Nturn, and the tune difference between these two tunes is the tune diffusion. 516 xmax Maximum amplitude in the horizontal direction with the unit [m] 0.025 ymax Maximum amplitude in the vertical direction with the unit [m] 0.005 delta Energy offset of the particle 0.0 diffusion Boolean flag, true/false; to compute tune diffusion at the first N turns and second N turns. true

printloss (optional) Boolean flag, true/flase; print out the last information of the tracked particleto and external file? If true, the output file is "fmap_file.loss".

12.17 FmapdpFlag

Frequency map analysis for off momentum particle

To do frequency map analysis for the off momentum particle, use the command: FmapdpFlag fmapdp_file nxpoint nepoint nturn xmax emax y diffusion or FmapdpFlag fmapdp_file nxpoint nepoint nturn xmax emax y diffusion printloss

The meaning of parameters and default values are shown in Table . If user only uses the command FmapdpFlag but without defining all the parameters,

then the code uses the default values.

Table 17:

Table Parameters of the command FmapdpFlag. parameters meaning default value fmapdp_file File to save the calculated frequency map analysis fmapdp.out nxpoint Number of points in the horizontal direction 31 nepoint Number of points for the energy 21 nturn Number of turns for tracking 516 xmax Maximum amplitude in the horizontal direction with the unit [m] 0.025 emax Maximum energy offset of the particle 0.005 y Amplitude in the vertical direction with the unit [m] 0.0 diffusion Boolean flag to compute tune diffusion true

printloss (optional) Boolean flag, true/flase; print out the last information of the tracked particleto and external file? If true, the output file is "fmapdp_file.loss".

12.18 ErrorCouplingFlag

Add coupling by the random rotation of the full quadrupoles

To simulate coupling in the lattice, use can add the random rotation error to all the full quadrupole, using the command as the following example: ErrorCouplingFlag 0 0.0007 In this example, 0 is the random seed number; 0.0007 is the RMS value of the rotation angles of all the quadrupoles with the unit [rad]. After setting the rotation error in the lattice, the code will generate a file with the file name flat_file_errcoupling_full.dat at the current working directory, user can check the error setting of quadrupoles in this file; then the coupling will be calculated and Twiss parameters after adding the random rotation errors will be saved to the file linlat_errcoupling.out.

12.19 ErrorCoupling2Flag

Add coupling by random rotation of the half quadrupoles

In order to get the beam parameters in the middle of the quadrupoles, each quadrupole in the lattice can be cut into two parts. In such case, the coupling of the lattice can be generated by random rotation of all the half quadrupoles in the lattice, using the command as the following example: ErrorCoupling2Flag 0 0.0007 In this example, 0 is the random seed number; 0.0007 is the RMS value of the rotation angle of the quadrupole with the unit [rad]. After setting the errors in the lattice, the code will generate a file at the current working directory with the file name flat_file_errcoupling_half.dat, user can check the error setting of quadrupoles in this file. After adding the random rotation errors, the coupling will be calculated and Twiss parameters will be saved to the file linlat_errcoupling2.out. This command is dedicated for Soleil lattice in which each quadrupole is cut into two half quadrupoles.

12.20 CouplingFlag

Calculate coupling and emittance

To calculate coupling and emittance, use command: CouplingFlag After calculation, the coupling and the emittance will be printed on the screen, and the Twiss parameters will be automatically saved to the file linlat_coupling.out.

12.21 MomentumAccFlag

Calculate momentum acceptance by tracking

The following command calculate momentum acceptance at a predefined lattice region by tracking: MomentumAccFlag MomAccFile TrackDim istart istop deltaminp Deltamaxp nstepp deltaminn deltamaxn nstepn turns zinitial For example: MomentumAccFlag momacc.out 4D 1 209 0.01 0.05 100 -0.01 -0.05 100 1026 0.0001 The meaning of parameters and default values are shown in Table . If user uses MomentumAccFlag without parameters, then the code will use the default values.

Table 18:

Table Parameters of the command to calculate momentum acceptance parameter meaning Default value MomAccFile File to save the tracked momentum acceptance at each element; saved in the current directory. momentumacceptance.out TrackDim 4D/6D tracking to get the momentum acceptance 6D istart Start element in the lattice for the tracking 1 istop End element in the lattice for tracking 108 nstepp Number of steps to do tracking in the positive energy range 100 nstepn Number of steps to do tracking in the negative energy range 100 Deltaminp Positive start energy of the tracking 0.01 Deltamaxp Positive end energy of the tracking 0.05 Deltaminn Negative start energy of the tracking -0.01 Deltamaxn Negative end energy of the tracking -0.05 nturns Number of turn 1026 zinitial The initial vertical coordinate which is used to search for 6D closed orbit. This value should be a small value. 0.0003 [m]

12.22 ReadMultipoleFlag

Read multipole field error from a file (SOLEIL lattice)

After defining the file names of multipole field errors on SOLEIL storage ring (section and), use the command: ReadMultipoleFlag to read multipole field errors and set the corresponding values to SOLEIL lattice. The multipole field errors of correctors and skew quadrupoles are added on the thick sextupoles which are integrated at the same magnets. The format of multipole errors file is given in section .

After setting the multipole field errors in the lattice, the code will generate a file at the current working directory, and the file name is flat_file_errmultipole.dat, user can check the field components of the lattice elements in this file to verify the multipole field errors.

12.23 ReadVirtualSkewquadFlag

Read the sources of coupling from a file (SOLEIL lattice)

The sources of coupling on SOLEIL storage ring can be read from an external file. Use the command: ReadVirtualSkewquadFlag to read and set the field strength to the virtual skew quadrupoles. Currently this command only works for Soleil lattice. The coupling sources MUST be defined as the skew quadrupoles with the name SQ, the rules and related information are explained in section.

12.24 FitTuneFlag

Fit tunes for the lattice with full quadrupole

Betatron tunes can be fit using two families of quadrupoles. The command is: FitTuneFlag Quad1 Quad2 nux nuz For example: FitTuneFlag q7 q9 18.202 10.317 The parameters of this command are shown in Table . Table Parameters of the command FitTuneFlag. Parameters Meaning Default values Quad1 Quadrupole family used to fit the tunes - Quad2 Quadrupole family used to fit the tunes - nux Target horizontal tune 0.0 nuz Target vertical tune 0.0

After fitting the tunes, field strengths of the fitted quadrupoles before and after the fitting are printed to the screen; user can copy the new quadrupole field strengths to the lattice file for further analysis. FitTuneFlag is a generic command; it works for the lattices with full qudrupoles.

12.25 FitTune4Flag

Fit tunes for the lattice with half quadrupoles

For the lattice with each quadrupole cut into two pieces, betatron tunes can be fit using two families of quadrupoles. The command is: FitTune4Flag Q1a Q1b Q2a Q2b nux nuz The parameters of this command are shown in Table .

Table 19:

Table Parameters of the command FitTune4Flag. Parameters Meaning Default values Q1a First half of the quadrupole family used to fit the tunes - Q1b Second half of the quadrupole family used to fit the tunes - Q2a First half of the quadrupole family used to fit the tunes - Q2b Second half of the quadrupole

family used to fit the tunes - nux Target horizontal tune 0.0 nuz Target vertical tune 0.0

For example: FitTune4Flag qp7a qp7b qp9a qp9b 18.202 10.317 In this example, all the variables have the same meaning as the ones in the command FitTuneFlag, except qp7a and qp7b are the two half pieces of the full quadrupole qp7, and qp9a and qp9b are the two half pieces of the full quadrupole 'qp9'. After fitting the tunes, the field strengths of fitted quadrupole before and after the fitting are printed to the screen; user can copy the new field strengths of quadrupoles to the lattice file for further analysis. FitTune4Flag is a command that works for the lattices in which each quadrupole are cut into two halves.

12.26 FitChromFlag

Fit chromaticity

Chromaticities can be fit using two families of sextupoles, the command is: FitChromFlag SX1 SX2 epsilon_x epsilon_z The parameters of this command are shown in Table .

Table 20:

Table Parameters of the command FitChromFlag. Parameters Meaning Default values SX1 First sextupole family used to fit the chromaticities - SX2 Second sextupole family used to fit the chromaticities - epsilon_x Target horizontal chromaticity 0.0 epsilon_z Target vertical chromaticity 0.0

For example: FitChromFlag sx9 sx10 2.0 2.6 After fitting the chromaticites, the field strengths of fitted sextupoles before and after the fitting are printed to the screen; user can copy the new field strengths of sextupoles to the lattice file for further analysis.

12.27 TouschekFlag (TO BE UPDATED)

Touschek lifetime determined by RF acceptance

To calculate Touschek lifetime, use the following command: TouschekFlag Here the momentum acceptance is limited by the RF acceptance.

12.28 IBSFlag (TO BE UPDATED)

Intra Beam Scattering (IBS)

To calculate Intra Beam Scattering, use the command: IBSFlag

12.29 TousTrackFlag (TO BE UPDATED)

Touschek lifetime determined by the minimum value of RF acceptance and momentum acceptance

Touschek lifetime can be calculated by TousTrackFlag In this case, the energy acceptance at each lattice element is limited by the minimum value of RF acceptance and momentum acceptance obtained by tracking, and the chamber file MUST be defined in the user script.

12.30 PhaseSpaceFlag

Obtain phase space by tracking

To calculate phase space, use the command:
PhaseSpaceFlag Phase_phase_file Phase_Dim Phase_X Phase_Px Phase_Y Phase_Py Phase_delta Phase_ctau Phase_nturn damping_flag

For example: **PhaseSpaceFlag** phasespace.out 6D 1e-6 0.0 1e-6 0.0 0.012 0.0 1000 false

The meanings of parameters and defaults values of PhaseSpaceFlag are shown in Table 21. If user uses PhaseSpaceFlag without parameters, then the code will use the default values.

Table 21: Parameters of the command **PhaseSpaceFlag** to calculate phase space.

Symbol	Units	Parameter
Phase_phase_file	File to save tracked phase space; saved in the current directory.	phase.out
Phase_Dim	4D / 6D tracking	4D
Phase_X	Horizontal coordinate at the start point of tracking.	0.0
Phase_Px	Horizontal canonical momentum / derivative at the start point of tracking.	0.0
Phase_Y	Vertical coordinate at the start point of tracking.	0.0
Phase_Py	Vertical canonical momentum / derivative at the start point of tracking.	0.0
Phase_delta	Energy at the start point of tracking	0.0
Phase_ctau	Longitudinal position at the start point of tracking.	0.0
Phase_nturn	Number of turns for tracking	512
Damping_flag	Boolean flag to turn on / off the radiation damping during the tracking.	false

12.31 IDCorrFlag (Tested for TaiWan light source; TO BE CONTINUE DEVELOPPED.)

Insertion device (ID) compensation

To compensate the beta beat introduced by the insertion devices, several families of Quadruoples can be used. Defining the following command in the *.prm can active this action: IDCorrFlag User also needs to define the following parameters used for the compensation of insertion device: N_calls 1 N_steps 1 N_Fam 11 IDCquads qs1 qs2 qs3 qs4 qs5 ql1 ql2 ql3 q1 q2 q3 scl_dbetax 5e-1 scl_dbetay 5e-1 scl_dnux 0.1 scl_dnuy 0.1 scl_nux 1e1 scl_nuy 1e1 ID_step 0.7

The meanings of the above commands and the default values used to do ID compensation using quadrupoles are shown in Table .

Table 22:

Table Parameters of commands to do ID compensation using quadrupoles. Parameters Meanings Default values N_calls Number of calls to do ID compensation 1 N_steps Number of steps. 1 N_Fam Number of quadrupole families used to do ID correction. 15 IDCquads Name of quadrupole families used to do ID correction. - scl_dbetax Scaling weight factor of the change step of horizontal beta function during the ID correction. 1 scl_dbetay Scaling weight factor of the change step of vertical beta function during the ID correction. 1 scl_dnux

Scaling weight factor of the change step of horizontal tune during the ID correction. 0.1 scl_dnuy Scaling weight factor of the change step of vertical tune during the ID correction. 0.1 scl_nux

Scaling weight factor of horizontal tune during the ID correction. 100 scl_nuy
Scaling weight factor of vertical tune during the ID correction. 100 ID_step 0.7
Frequency map analysis for on momentum particle, command “FmapFlag”.

- Frequency map analysis for off momentum particle, command “FmapdpFlag”.
- Track momentum acceptance at lattice elements, command “MomentumAccFlag”.

12.32 Compile

The commonly used compilers for parallel computation are MPI 2, and Intel MPI which is based on MPI 2. For the cluster of SOLEIL Synchrotron, Intel MPI is installed. To get the parallel Tracy work, three files of the non-parallel version Tracy need to be modified. The details are shown in the following steps.

1. The path of included files of Intel MPI is added in “Makefile.am” under path “$HOME/TracyIII/tracy/tracy/src” (shown in BLUE color):

   INCLUDES = -I../inc -I$(NUM_REC)/inc -I/opt/intel/impi/3.2.2.006/include64

2. The execute file, source file, paths of included files and library of Intel MPI are modified in "Makefile.am" under path "$HOME/TracyIII/tracy/tools" (shown in BLUE color):

   bin_PROGRAMS = psoltracy

   soltracy_SOURCES = soltracy.cc nrutil.c nrcheck.c nrlinwww.c nrframe.c ../tracy/src/tracy_lib.cc -¿ psoltracy_SOURCES = psoltracy.cc nrutil.c nrcheck.c nrlinwww.c nrframe.c ../tracy/src/tracy_lib.cc

   LIBS = -L$(NUM_REC)/lib -L/opt/intel/impi/3.2.2.006/lib64 -L$(LIBPATH) -lrecipes_c_icc -lstdc++ -lgfortran -lmpichcxx

   INCLUDES = -I$(TRACY_LIB)/tracy/inc -I$(NUM_REC)/inc -I/opt/intel/impi/3.2.2.006/include64

3. The compilers used in the parallel computing are defined in the "make_for_psoltracy.sh" located in path "$HOME/TracyIII" as (shown in BLUE color):

   CC = mpiicc

   CXX = mpiicpc

   F77 = mpiifort

   Depending on the compilers used to do parallel computation, user needs to update the compilers in "make_for_psoltracy.sh" and the paths of included files and library which are shown above with blue color.

   After updating compilers, paths of included files and library for parallel computation, user can run

   make_for_psoltracy.sh

   under the shell script to compile the parallel Tracy. After compilation, the execute file "psoltracy" is automatically generated under the path "$HOME/TracyIII/tracy/tools".

12.33 Run

As the non-parallel version Tracy, user needs to write the commands in a script which must be with the file extension .prm. The syntaxes to define the script *.prm are the same for both non-parallel and parallel Tracy. To run the parallel Tracy, user need to contact the administrator of their cluster to know how to run parallel programs on the cluster. For the cluster on Synchrotron SOLEIL, the nodes used to do parallel computation are assigned by PBS (Portable Batch System), so a script is need. For example, user define the input script test.prm to tell Tracy what jobs are need to be done on the cluster, define script lance_tracy3_parallel.sh to assign the numbers of CPUs to do parallel computation and lance job to the SOLEIL cluster. and then type the following command under the bash shell to submit the job and run the parallel Tracy on the cluster:

lance_tracy3_parallel.sh test.prm

13 User input script

There are two types of keywords in the user input script. The first type is to set the file, the file names, and define parameters for the related calculations; the key words for such definitions are ended with the characters Flag. The second type is to define Boolean commands with or without parameters to do different calculations. The rules of the definition of input scripts are: The blank lines and lines starting with "#" (comment line) are ignored by the code. Keywords without Flag as the final 4 characters are NOT executed according to the defined sequence in user input script. If the same command keywords are defined many times in the user script, only the last defined keyword is executed. The commands with the last 4 characters as Flag are executed according to the sequence in the code. The definition of lattice file at the beginning of the user script is mandatory. If the lattice file is with any of the following elements: horizontal correctors, vertical correctors, girders, BPM, skew quadrupoles, user must declare the element name at the beginning of input script. One keyword command uses one line. User can not define more than one command at the same line. Except explanation, all commands with Boolean flag are the generic commands, and can be used on all the lattice of the ring.

13.1 File path

In the user input script, user can specify the name of the files which are used in the calculation, such as the lattice file, multipole field error file, misalignment error file, vacuum chamber file. When specifying the file name, user can provide the absolute path for the file which is not located at the current path, such as: /home/physmach/Tracy3/tracy/soleil.lat; or for convenience, user provides the filename without absolute path, but put the files at a certain directory and then specific this directory using the command: in_dir user_defined_path For example: in_dir /home/zhang/codes/TracyIII/lattice/

This command tells the code all the files specified in the script are located at the directory /home/zhang/codes/TracyIII/lattice/. If user declares the files without absolute path and does not set the directory through command in_dir; or the files specified in the script are not found at the current working path, the code will give an error message and stop running.

14 File names

14.1 Lattice file name

In the input script, user must define the lattice name, this is mandatory. The command is: lat_file lattice_file_name Here lattice_file_name is the lattice file name without .lat extension. For example, the following command sets the lattice file of SOLEIL ring: lat_file solamor2_reglage_focalisation_chcvqt_thicksextu_LQPintermediaire_QFF In Tracy 3, after reading the lattice, Twiss parameters are automatically printed to the file linlat.out. If one of these elements (Horizontal/vertical correctors or

beam position monitors, or girders, or skew quadrupoles.) are defined in the lattice file, for example to read the misalignment errors of the lattice element and then do orbit correction, to read multipole field errors (SOLEIL lattice), etc.; user MUST specify the names of these elements using the corresponding commands in the *.prm script: h_corr HCM v_corr VCM gs GS ge GE bpm_name BPM qt QT Here HCM is the name of horizontal correctors used for horizontal orbit correction, or the horizontal correctors on which the multipole field errors are added in SOLEIL lattice; VCM is the name of vertical correctors used for vertical orbit correction, or the vertical correctors on which the multipole field errors are added in SOLEIL lattice; GS is the name of the start girder; GE is the name of the end girder; BPM is the name of beam position monitors defined in the lattice file; QT is the name of the skew quadrupoles defined in the lattice. Generally, user needs to define horizontal, vertical correctors and BPMs when reading the misalignment errors or correcting closed orbit distortion

14.2 Multipole field error file name (SOLEIL lattice)

User can read the multipole field errors on SOLEIL lattice from an external file; the file is specified in the user script using the following command: multipole_file multipole_file_name Here multipole_file_name is the user defined multipole field error file, the format of this file is given in section Error: Reference source not found.

If the multipole field errors of horizontal, vertical correctors and skew quadrupoles are defined in the multipole_file, user MUST specify the names of these lattice elements in the *.prm file as the following example: h_corr CH v_corr CV qt QT Here CH, CV and QT are the names of horizontal, vertical correctors, and skew quadrupoles respectively which are defined in the lattice file.

14.3 Files of multipole field errors of correctors and skew quadrupoles (SOLEIL lattice)

Horizontal, vertical correctors and skew quadrupoles are integrated with the sextupole quadrupoles on SOLEIL storage ring. To define the multipole field errors on these elements, user need to define the orders and relative strengths of multipole field errors in the multipole field error file (section Error: Reference source not found) and specify the file name in the user input script *.prm (section Multipole field error file name), then specify the file names as the following example: fic_hcorr corh.txt fic_vcorr corv.txt fic_skew corqt.txt Here corh.txt and corv.txt are the files with measured current values corh, corv, corqt (with unit [Ampere]) for the horizontal, vertical correctors and skew quadrupoles respectively. Based on the measured current values, user can get the integrated field strength as: Hcorr_strength [T.m] = corh *_convHcorr /brho; (horizontal correctors) Vcorr_strength [T.m] = corv *_convVcorr /brho; (vertical correctors) Qt_strength [T.m] = corqt *_convQt /brho; (skew quadrupoles) here brho is the momentum rigidity, and the conversion constants between current and field are _convHcorr = 8.14e-4, _convVcorr = 4.642e-4, _convQt = 93.83e-4. For

SOLEIL lattice, the SAME ORDER of multipole field errors on the same elements are added together, so the SAME ORDER of horizontal/vertical, skew quadrupoles, and sextupoles are added together since these elements are integrated at the same magnets.

14.4 File to define field strength of virtual coupling source elements (SOLEIL lattice)

On SOLEIL storage ring, the coupling is thought to come from the rotation of quadrupoles and vertical displacements of sextupoles. The strengths of these coupling sources are written in a file, then read by the following command: virtualskewquad_file virtual_skew_quad_currents.txt

Here virtual_skew_quad_currents.txt is the user defined file with strength of vertical coupling source. In order to use this command to read the virtual sources of coupling, user needs to define the virtual coupling element as a skew quadrupole in the lattice file: SQ: quadrupole, tilt=45.0, K= 0.0, method=4, N=1; and this virtual coupling element MUST be with the name SQ. Now there 152 elements defined as the virtual coupling sources in the SOLEIL lattice. The syntaxs to define skew quadrupole are explained in section . The measured current value qtcorr[i] of the ith coupling source is converted to the corresponding integrated skew quadrupole field strength corr_strength as corr_strength [m-1] = qtcorr[i]*conv/brho; here brho is momentum rigidity, and the conversion constant conv is 93.88e-4 [A-1.T].

14.5 Cut off value

Set the cut off value of all random distribution (Gaussian distribution) to n times of the RMS value sigma: normalcut n

15 Physics: Hamiltionian

The dynamic motion of a conserved system can be described by a Hamiltonian. For an accelerator system without radiation damping (energy loss) and RF cavities (energy gain), the motion of a single electron in the magnetic field can be described using the following Hamitonian in the curvilinear coordinate system (Merz...)

$$H(x, P_x; y, P_y; -t, \delta; s) = L + V \tag{92}$$

$$= -(1 + h_{\text{ref}}x)[\sqrt{(P^2 - (P_x - eA_x)^2 - (P_y - eA_y)^2)} + eA_z(x, P_x; y, P_y; s)] \tag{93}$$

$$P^2 = P_x^2 + P_y^2 + P_z^2 \tag{94}$$

h_{ref} is the curvature of reference orbit inside the dipole. The $(x, P_x; y, P_y; -t, \delta)$ are the canonical cooridinates of the H, and $P_x, P_y, P - z$ are respectively the horizontal, vertical, and longitudinal mechanical momentums. A more convienent way is to expand the electron motion around the motion of a reference

particle with total momentum P_0, so

$$\begin{aligned}
\frac{1}{P_0}H(x,P_x;y,P_y;-t,\delta;s) &= \\
H(x,p_x;y,p_y;-t,\delta;s) &= -(1+h_{\rm ref}x)\sqrt{((1+\delta)^2-(p_x-\frac{e}{P_0}A_x)^2-(p_y-\frac{e}{P_0}A_y)^2)}+\frac{e}{P_0}A_z(x,p_x;y,p_y, \\
&= -(1+h_{\rm ref}x)\sqrt{((1+\delta)^2-(p_x-\frac{1}{B\rho}A_x)^2-(p_y-\frac{1}{B\rho}A_y)^2)}+\frac{1}{B\rho}A_z(x,p_x;y,p
\end{aligned}$$

with

$$p_x = \frac{P_x}{P_0} \tag{98}$$

$$p_y = \frac{P_y}{P_0} \tag{99}$$

$$p = \frac{P}{P_0} = \frac{\Delta P + P_0}{P_0} = (1+\delta) \tag{100}$$

$$\delta = \frac{\Delta P}{P_0} \tag{101}$$

$$B\rho = \frac{P_0}{e}(magneticrigidity) \tag{102}$$

With some operations (J. Bengtsson's linear transverse dynamics for sotrage ring with applications to the low energy antiproton ring (LEAR) at CERN; Tracy 2 manual.), the Hamitonian can be expressed with the canonical coordinates $(x,p_x;y,p_y;-ct,\delta;s)$ (Tracy-2 Manual, J. Bengtsson)

$$
\begin{aligned}
&H(x, p_x; y, p_y; -ct, \delta; s) \\
&\qquad = -(1 + h_{\text{ref}} x)(\sqrt{(1+\delta)^2 - (p_x - \frac{A_x}{B\rho})^2 - (p_y - \frac{A_y}{B\rho})^2} + \frac{A_z}{B\rho}) + \delta \\
&\qquad = -(1 + h_{\text{ref}} x)(\sqrt{(1+\delta)^2 - (p_x - \frac{A_x}{B\rho})^2 - (p_y - \frac{A_y}{B\rho})^2}) + h_{\text{ref}} x \frac{A_z}{B\rho} \\
&\qquad + \frac{A_z}{B\rho} + \delta \\
&\qquad = \underbrace{-(1 + h_{\text{ref}} x)\sqrt{(1+\delta)^2 - (p_x - \frac{A_x}{B\rho})^2 - (p_y - \frac{A_y}{B\rho})^2)}}_{\text{kinetic energy}} \\
&\qquad + \underbrace{\frac{x}{\rho} + \frac{x}{2\rho^2}}_{\substack{\text{Gemometic components due to the curvilinear coordinate inside the dipole}}} \\
&\qquad + \underbrace{\frac{A_z}{B\rho}}_{\text{magnet contribution}} \\
&\qquad + \underbrace{\qquad \delta}_{\text{due to the canonical coordinate } -ct \text{ instead of } -t} \\
&A_z = A_z(x, p_x; y, p_y, s)
\end{aligned}
$$

The equation is the exact Hamitonian used in Tracy. The A_x are due to the longidutinal fringe field, this is an natural results due to the Maxwell Equation; h_{ref} is not zero only inside the dipoles.

For the system without fringe field of the magnet, or only consider the field

inside the body of the magnet, $A_x = 0$ and $A_y = 0$, so the Hamiltonian is

$$H(x, p_x; y, p_y; -ct, \delta; s) \tag{103}$$

$$= \underbrace{-(1 + h_{\text{ref}} x)\sqrt{(1+\delta)^2 - p_x^2 - p_y^2)}}_{\text{kinetic energy}} \tag{104}$$

$$+ \underbrace{\frac{x}{\rho} + \frac{x}{2\rho^2}}_{\text{Gemometic components due to the curvilinear coordinate inside the dipole}} \tag{105}$$

$$+ \underbrace{\frac{A_z}{B\rho}}_{\text{magnet contribution}} \tag{106}$$

$$+ \underbrace{\delta}_{\text{due to the canonical coordinate } -ct \text{ instead of } -t} \tag{107}$$

$$\tag{108}$$

From eqn. 103, it is easy to get the map of the particle inside different lattice elements.

15.1 Map of drift

In the drift, $A_z = 0$, and $h_{\text{ref}} = 0$, from eqn. 103, we can get the Hamiltonian inside a drift is

$$H(x, p_x; y, p_y; -ct, \delta; s) = -\sqrt{(1+\delta)^2 - p_x^2 - p_y^2)} + \delta \tag{109}$$

In large ring, in order to increase the tracking speed, we can expand the sqare root in eqn. 109 to get the approximate Hamitonian. That is,

$$H(x, p_x; y, p_y; -ct, \delta; s) = -(1+\delta)\left[1 - \frac{p_x^2 + p_y^2}{2(1+\delta)^2}\right] + \delta \tag{110}$$

$$= \frac{p_x^2 + p_y^2}{2(1+\delta)} \tag{111}$$

From eqn. 109, it is easy to get the map inside the drift for the **small ring**:

$$x^{'} = \frac{dx}{ds} = \frac{\partial H}{\partial p_{x0}} = \frac{p_{x0}}{\sqrt{(1+\delta_0)^2 - (p_{x0}^2 + p_{y0}^2)}} \tag{112}$$

$$y^{'} = \frac{dy}{ds} = \frac{\partial H}{\partial p_{y0}} = \frac{p_{y0}}{\sqrt{(1+\delta_0)^2 - (p_{x0}^2 + p_{y0}^2)}} \tag{113}$$

$$(-ct)^{'} = \frac{d(-ct)}{ds} = \frac{\partial H}{\partial \delta_0} = -\frac{1+\delta_0}{\sqrt{(1+\delta_0)^2 - (p_{x0}^2 + p_{y0}^2)}} + 1 \tag{114}$$

$$or \tag{115}$$

$$(ct)^{'} = \frac{1+\delta_0}{\sqrt{(1+\delta_0)^2 - (p_{x0}^2 + p_{y0}^2)}} - 1 \tag{116}$$

$$p_x = p_{x0} \tag{117}$$

$$p_y = p_{y0} \tag{118}$$

$$\delta = \delta_0 \tag{119}$$

From eqn. 110, it is easy to get the map inside the drift for the **big ring**:

$$x^{'} = \frac{dx}{ds} = \frac{\partial H}{\partial p_{x0}} = \frac{p_{x0}}{1+\delta_0} \tag{120}$$

$$y^{'} = \frac{dy}{ds} = \frac{\partial H}{\partial p_{y0}} = \frac{p_{y0}}{1+\delta_0} \tag{121}$$

$$(-ct)^{'} = \frac{d(-ct)}{ds} = \frac{\partial H}{\partial \delta_0} = -\frac{p_{x0^2} + p_{y0}^2}{2(1+\delta_0)} \tag{122}$$

$$or \tag{123}$$

$$(ct)^{'} = \frac{p_{x0^2} + p_{y0}^2}{2(1+\delta_0)} \tag{124}$$

$$p_x = p_{x0} \tag{125}$$

$$p_y = p_{y0} \tag{126}$$

$$\delta = \delta_0 \tag{127}$$

15.2 Map of mutipoles

From eqn. 103, we can get the Hamiltonian inside a multipole without fringe field is

$$H(x, p_x; y, p_y; -ct, \delta; s) \tag{128}$$

$$= \underbrace{-\sqrt{(1+\delta)^2 - p_x^2 - p_y^2} + \delta}_{\text{H drift}} \tag{129}$$

$$\underbrace{-h_{\text{ref}} x \sqrt{(1+\delta)^2 - p_x^2 - p_y^2}}_{\text{H kick}} \tag{130}$$

$$+ \underbrace{\frac{x}{\rho} + \frac{x}{2\rho^2} - \frac{A_z}{B\rho}}_{\text{H kick}} \tag{131}$$

In eqn. 128, the Hamiltonian is decomposed into two part, one is the Hamiltonian of the drift which is due to the kinetic energy of the system (section 15.1), another part is the kick due to the multipoles which will give the kick to the particle. Now we only need to focus on the Hamiltonian of the multipoles which induces the kick map:

$$H(x, p_x; y, p_y; -ct, \delta; s) \tag{132}$$

$$= -h_{\text{ref}} x \sqrt{(1+\delta)^2 - p_x^2 - p_y^2} \tag{133}$$

$$+ \frac{x}{\rho} + \frac{x}{2\rho^2} - \frac{A_z}{B\rho} \tag{134}$$

Similarly, by expand the sqare root in eqn. 128, we can get the approximate Hamiltonian of the multipole as

$$H(x, p_x; y, p_y; -ct, \delta; s) \tag{135}$$

$$= -h_{\text{ref}} x \sqrt{(1+\delta)^2 - p_x^2 - p_y^2} \tag{136}$$

$$+ \frac{x}{\rho} + \frac{x}{2\rho^2} - \frac{A_z}{B\rho} \tag{137}$$

$$= -h_{\text{ref}} x\delta + \frac{1}{2} h_{\text{ref}} h_B x^2 - \frac{A_z}{B\rho} + (h_B - h_{\text{ref}}) x \tag{138}$$

where h_{ref} is the curvature due to the curlininear coordinates inside the dipole, $h_{\text{ref}} = 1/\rho_{ref}$; h_{bend} is due to the bending curvature of the dipole which is determined by the dipole field, $h_{\text{bend}} = 1/\rho_{bend}$. $h_{\text{ref}} \neq 0$ only inside dipoles.

In the dipole, $A_z \neq 0$, and $h_{\text{ref}} \neq 0$, so from eqn. 132, we can get the kick

map due to the dipole in the **small ring** which use the exact Hamiltonian:

$$x^{'} = \frac{dx}{ds} = \frac{\partial H}{\partial p_{x0}} = h_{\text{ref}} x_0 \frac{p_{x0}}{\sqrt{(1+\delta_0)^2 - (p_{x0}^2 + p_{y0}^2)}} \tag{139}$$

$$y^{'} = \frac{dy}{ds} = \frac{\partial H}{\partial p_{y0}} = h_{\text{ref}} x_0 \frac{p_{y0}}{\sqrt{(1+\delta_0)^2 - (p_{x0}^2 + p_{y0}^2)}} \tag{140}$$

$$p_x^{'} = -\frac{\partial H}{\partial y_0} = h_{\text{ref}} \sqrt{(1+\delta_0)^2 - p_{x0}^2 - p_{y0}^2} - h_{\text{bend}} - h_{\text{ref}} h_{\text{bend}} x - \frac{B_y}{B\rho} \tag{141}$$

$$p_y^{'} = -\frac{\partial H}{\partial y_0} = \frac{1}{B\rho}\frac{\partial A_z}{\partial y_0} = \frac{B_x}{B\rho} \tag{142}$$

$$(-ct)^{'} = \frac{d(-ct)}{ds} = \frac{\partial H}{\partial \delta_0} = -h_{\text{ref}} x_0 \frac{1+\delta_0}{\sqrt{(1+\delta_0)^2 - (p_{x0}^2 + p_{y0}^2)}} \tag{143}$$

$$or \tag{144}$$

$$(ct)^{'} = h_{\text{ref}} x_0 \frac{1+\delta_0}{\sqrt{(1+\delta_0)^2 - (p_{x0}^2 + p_{y0}^2)}} \tag{145}$$

$$\delta = \delta_0 \tag{146}$$

$$B\rho = \frac{p_0}{e} \tag{147}$$

where

$$\frac{\partial A_z}{\partial x} = -B_y \tag{148}$$

$$\frac{\partial A_z}{\partial y} = B_x \tag{149}$$

since

$$A_x = 0 \tag{150}$$

$$A_y = 0 \tag{151}$$

inside the body of the multipoles, and

$$\vec{B} = \Delta \times \vec{A}$$

$$\begin{array}{cc} \hat{x}\hat{y} & \hat{z} \\ \frac{\partial}{x}\frac{\partial}{y} & \frac{\partial}{z} \\ A_x A_y & A_z \end{array}$$

In the dipole, $A_z \neq 0$, and $h_{\text{ref}} \neq 0$, so from eqn. 135, we can get the kick

map due to the dipole in the **big ring** which use the exact Hamiltonian:

$$x^{'} = \frac{dx}{ds} = \frac{\partial H}{\partial p_{x0}} = h_{\text{ref}} x_0 \frac{p_{x0}}{1+\delta_0} \tag{152}$$

$$y^{'} = \frac{dy}{ds} = \frac{\partial H}{\partial p_{y0}} = h_{\text{ref}} x_0 \frac{p_{y0}}{1+\delta_0} \tag{153}$$

$$p_x^{'} = -\frac{\partial H}{\partial y_0} = h_{\text{ref}}\delta_0 - (h_{\text{bend}} - h_{\text{ref}}) - h_{\text{ref}} h_{\text{bend}} x - \frac{B_y}{B\rho} \tag{154}$$

$$p_y^{'} = -\frac{\partial H}{\partial y_0} = \frac{1}{B\rho}\frac{\partial A_z}{\partial y_0} = \frac{B_x}{B\rho} \tag{155}$$

$$(-ct)^{'} = \frac{d(-ct)}{ds} = \frac{\partial H}{\partial \delta_0} = \frac{-h_{\text{ref}} x_0}{1+\delta_0} \frac{-(p_{x0}^2 + p_{y0}^2)}{2(1+\delta_0)} \tag{156}$$

$$or \tag{157}$$

$$(ct)^{'} = -\frac{h_{\text{ref}} x_0}{1+\delta_0} \frac{(p_{x0}^2 + p_{y0}^2)}{2(1+\delta_0)} \tag{158}$$

$$\delta^{'} = 0 \tag{159}$$

For other multipoles (quadrupoles, sextupole, decapole, octupole, etc), the curvilinear coordinate is the cartesian coordinates, that is $h_{ref} = 0$, so from eqn. 139, the kick map for these multipoles in the small ring is:

$$x^{'} = \frac{dx}{ds} = \frac{\partial H}{\partial p_{x0}} = 0 \tag{160}$$

$$y^{'} = \frac{dy}{ds} = \frac{\partial H}{\partial p_{y0}} = 0 \tag{161}$$

$$p_x^{'} = -\frac{\partial H}{\partial y_0} = -(h_{\text{bend}} + \frac{B_y}{B\rho}) \tag{162}$$

$$p_y^{'} = -\frac{\partial H}{\partial y_0} = \frac{1}{B\rho}\frac{\partial A_z}{\partial y_0} = \frac{B_x}{B\rho} \tag{163}$$

$$(-ct)^{'} = \frac{d(-ct)}{ds} = \frac{\partial H}{\partial \delta_0} = 0 \tag{164}$$

$$or \tag{165}$$

$$(ct)^{'} = 0 \tag{166}$$

$$\delta^{'} = 0 \tag{167}$$

With $h_{ref} = 0$, from eqn: 152, the kick map for these multipoles (quadrupoles,

sextupole, decapole, octupole, etc) in the big ring is:

$$x^{'} = \frac{dx}{ds} = \frac{\partial H}{\partial p_{x0}} = 0 \tag{168}$$

$$y^{'} = \frac{dy}{ds} = \frac{\partial H}{\partial p_{y0}} = 0 \tag{169}$$

$$p_x^{'} = -\frac{\partial H}{\partial y_0} = -(h_{\text{bend}} + \frac{B_y}{B\rho}) \tag{170}$$

$$p_y^{'} = -\frac{\partial H}{\partial y_0} = \frac{1}{B\rho}\frac{\partial A_z}{\partial y_0} = \frac{B_x}{B\rho} \tag{171}$$

$$(-ct)^{'} = \frac{d(-ct)}{ds} = \frac{\partial H}{\partial \delta_0} = 0 \tag{172}$$

$$or \tag{173}$$

$$(ct)^{'} = 0 \tag{174}$$

$$\delta^{'} = 0 \tag{175}$$

From the eqn 139 to eqn. 168, it's clear that the kick map from the main body of dipoles are different in the small ring and big ring; while the kick map from the main body of other type of multipoles are the same in small rings and big rings.

16 Magnetic field

A. Dragt ....(Lie methods for acceleator.....)

17 FF of dipole

17.1 kick map

E. Forest .....

18 FF of quadrupole

18.1 kick map

19 Lattice file

In the lattice, RF cavity must be defined!!! Otherwise Tracy will give the error message: Elem_GetPos: there are no kids in family 0 (). This obigatory is for the correct calculation of positive/negtive momentum compaction factor.

The followings are the rules to define a lattice file used in Tracy 3. The curvelinear coordinates are used. **The ideal particle or design particle sees the perfect magnetic field in all magnets, and its orbit is used as**

the reference orbit and base of the curvilinear coordinate. Since the reference orbit is a curve inside the dipoles and a straight line in other lattice elements (drifts and the magnets except dipoles), so the length of the dipole is the the path length of the reference particle inside the dipole which is equal to $\rho * \theta$ where ρ is the bending radius of the dipole and *theta* is the bending angle with unit [rad], and all the other magnet lengths are the straight length of the magnets.

Due to the same reason, the curvature of the curvilinear cooridnates h = $1/\rho$ is not zero only inside dipole; in other magnets and drift h = 0, and the curvilinear coordinates goes to cartesian cooridnates. As a result, the independent variable s inside the dipole harmitonian is the arc length, while the staight length in other lattice elements.

19.1 Lattice

The n^{th} field component of the lattice element is defined as in Tracy (n = 1, 2, 3, 4 ...):

$$\begin{aligned} b_n &= \frac{1}{B\rho}\frac{1}{(n-1)!}\frac{\partial^{n-1}B_y}{\partial x^{n-1}}|_{x=0,y=0} \\ a_n &= \frac{1}{B\rho}\frac{1}{(n-1)!}\frac{\partial^{n-1}B_x}{\partial x^{n-1}}|_{x=0,y=0} \\ B\rho &= \frac{p_0}{e} \end{aligned}$$

$B\rho$ is the magnetic rigidity, p_0 is the design beam momentum, e is the electric charge. For example, for sextupole, n = 3, so b_3 and a_3 are defined as

$$\begin{aligned} b_3 &= \frac{1}{B\rho}\frac{1}{2}\frac{\partial^2 B_y}{\partial x^2}|_{x=0,y=0} \\ a_3 &= \frac{1}{B\rho}\frac{1}{2}\frac{\partial^2 B_x}{\partial x^2}|_{x=0,y=0} \end{aligned}$$

.

Notes: In **AT** (Accelerator Toolbox) and **BETA** code, the definition of are the same as in **Tracy**. While In **MAD8**, **MADX** and **ELEGANT**, the order of the field compoent start from 0, that is n = 0, 1, 2, 3, ..., and the n^{th} field strength components of the lattice element b_n and a_n are defined as

$$\begin{aligned} b_n &= \frac{1}{B\rho}\frac{\partial^n B_y}{\partial x^n}|_{x=0,y=0} \\ a_n &= \frac{1}{B\rho}\frac{\partial^n B_x}{\partial x^n}|_{x=0,y=0} \\ B\rho &= \frac{p_0}{e} \end{aligned}$$

For example, for sextupole, n = 3, its field component b_2 and a_2 are defined as

$$\begin{aligned} b_2 &= \frac{1}{B\rho}\frac{\partial^2 B_y}{\partial x^2}|_{x=0,y=0} \\ a_2 &= \frac{1}{B\rho}\frac{\partial^2 B_x}{\partial x^2}|_{x=0,y=0} \end{aligned}$$

.

19.2 Syntax

Every line embraced by is comment line. For example:

*****drift space *****

Each sentence is ended by ; or no punctuation. Tracy is not sensitive to capital/small letters in the lattice. User can define any lattice element with any valid name (but must start with a character) they want, but the element type is fixed. For the lattice of the ring, the definition of RF cavity is mandatory, and the harmonic number of the RF cavity is also mandatory; for the lattice of the linac, the definition of the RF cavity is optional.

19.3 Variables

User can define the variables in the lattice file. For example: Intmeth = 4; so when the code is running, each intmeth in the lattice file will be replaced by 4.

19.4 Start line

The lattice file must begin with the sentence: define lattice; This definition is mandatory.

19.5 Global variables

After define the ring, user needs to define the system parameters of the lattice: Energy, the beam energy with unit [GeV]. dP, the relative momentum offset of the particle. CODeps, the convergence for the algorism to find the closed orbit. For example: Energy = 2.739; dP = 1.0d-10; CODeps= 1.0d-15; These definitions are mandatory.

20 Multipole field error file

The multipole field errors of the lattice elements can be defined in a file, and then the file is read into the lattice. User can define the systematic or random multipole field error of the lattice elements. There are two ways to define the multipole field errors, one way is to define the errors for all the families with the same type, for example, the error for all the quadrupoles; another way is to define the error for each family, for example, the Q1 family of the quadrupoles.

20.1 Systematic errors

To define the systematic multipole field error of the element, the user just need to follow the rules as below. This command is commonly used to add the mangets design errors in the lattice. Input format of multipole error: keywords/name sys $r0$(radius where the multipole field error is measured. If $r0 = 0$, then the An and Bn are the integrated multipole field strength at the position $x = z = 0$). n(order of multipole field error, in US. notation. Dip errors is 1, quadrupole error is 2, sextupole field error is 3, decaper field error is 4, octoper field error is 5, etc.), Bn(n^{th} integrated upright component of the multipole field) An(n^{th} integrated skew component of the multipole field) m, Bm, Am,..... The "keywords" means one type of lattice elements or the family name; the keywords of the type of lattice elements are: dip dipole
quad quadrupole
sext sextupole
hcorr horizontal corrector
vcorr vorizontal corrector
qt skew quadrupole

sys is a keyword to denote that user are setting the systematic multipole error. Bn defines the upright component of the magnetic field, then for the component of a skew quadrupole or a vertical corrector, Bn = 0 An defines the skew component of the magnetic field, then for the component of a dipole or upright quadrupole, An = 0. The line start with # is a comment line. The blank lines in the multipole definition file are neglected by the code.

There are two ways to define the multipole field errors strength, one is to define the measured field errors at the pole tip location r_0, then b_n and a_n are respectively the scale coefficient of the main magnetic field coeffeicient. For example, to define the decupole field errors inside a horizontal bending dipole with coefficient b_1, we define the normal scale coefficience as c_5 and skew scale coefficience as d_5, then the b_5 component of this decupole is c_5 * b_1 and the skew coefficience a_5 is d_5 * b_1; that is

$$\begin{aligned} b_5 &= \frac{1}{B\rho}\frac{c_5 * B_0}{r_0^5} \\ a_5 &= \frac{1}{B\rho}\frac{d_5 * B_0}{r_0^5} \end{aligned}$$

where

$$b_1 = \frac{1}{B\rho}B_0. \tag{176}$$

This method is the simplest way to define the multipole errors inside a lattice element. The default scaling factors c_n and d_n are 1.

Another way is to direct define the multipole field coefficiences b_n and a_n, that is, set $r_0 = 0$. If the multipole field $\vec{B^{tip}}$ are measured at the tips with

radius r_0, then b_n and a_n can be calculated using

$$\begin{aligned} b_n &= \frac{1}{B\rho}\frac{1}{(n-1)!}\frac{\partial^{n-1}B_y}{\partial x^{n-1}}|_{x=0,y=0} = \frac{1}{B\rho}\frac{B_y^{tip}}{r_0^{n-1}} \\ a_n &= \frac{1}{B\rho}\frac{1}{(n-1)!}\frac{\partial^{n-1}B_x}{\partial x^{n-1}}|_{x=0,y=0} = \frac{1}{B\rho}\frac{B_x^{tip}}{r_0^{n-1}} \\ \vec{B^{tip}} &= B_x^{tip}\hat{x} + B_y^{tip}\hat{y} \end{aligned}$$

The assigned multipoles errors will be added to the corresponding n^{th} order multipole errors of the lattice element.

For the soleil lattice, the use can define the multipole errors for the type or each family. But to define the multipole error for all the quadrupoles, user can NOT define the multipole errors by the type. There are two choice: one is to define the multipole errors for each quadrupole family; second is to define the field errors by quadrupole type, and then define the multipole errors on Q2 and Q7 families (the lattice with full quadrupoles) or QP2a, QP2b, QP7a and QP7b families (lattice with quadrupoles which are cut into two halves). This is due to that Q2/QP2a/QP2b and Q7/QP7a/QP7b are the quadrupoles which have lengths larger than other quadrupoles in the lattice, and the multipole errors on them are different from the ones on the other short quadrupoles.

The following is an example file to define systematic multipole errors on Soleil lattice:

#dipole

dip sys 20e-3 2 2.2e-40 0.0 3 -3.0e-4 0.0 4 2.0e-5 0.0 5 -1.0e-4 0.0 6 -6.0e-5 0.0 7 -1.0e-4 0.0

#quadrupole

#for all short quadrupoles

quad sys 30e-3 6 2.4e-4 0.0 10 0.7e-4 0.0 14 0.9e-4 0.0

#for all long quadrupoles qp2 and qp7

qp2a sys 30e-3 6 0.7e-4 0.0 10 1.9e-4 0.0 14 1.0e-4 0.0 qp2b sys 30e-3 6 0.7e-4 0.0 10 1.9e-4 0.0 14 1.0e-4 0.0 qp7a sys 30e-3 6 0.7e-4 0.0 10 1.9e-4 0.0 14 1.0e-4 0.0 qp7b sys 30e-3 6 0.7e-4 0.0 10 1.9e-4 0.0 14 1.0e-4 0.0

#for all short quadrupoles,sextupole mesure quadrupoles longs

quad sys 30e-3 3 -1.6e-4 0.0 4 -3.4e-4 0.0

#for long quadrupoles qp2 and qp7

qp2a sys 30e-3 3 2.9e-4 0.0 4 -8.6e-4 0.0

qp2b sys 30e-3 3 2.9e-4 0.0 4 -8.6e-4 0.0

qp7a sys 30e-3 3 2.9e-4 0.0 4 -8.6e-4 0.0

qp7b sys 30e-3 3 2.9e-4 0.0 4 -8.6e-4 0.0

for sextupoles

sext sys 32e-3 5 5.4e-4 0.0 7 3.3e-4 0.0 9 -4.7e-4 0.0 15 -9.0e-4 0.0 21 -20.9e-4 0.0 27 0.8e-4 0.0

for horizontal correctors, all An=0 hcorr sys 35e-3 5 0.430 0.0 7 0.063 0.0 11 -0.037 0.0

for vertical correctors, all Bn=0

vcorr sys 35e-3 5 0.0 -0.430 7 0.0 0.063 11 0.0 0.037
for sextupole associated skew quadrupole, all Bn=0
qt sys 35e-3 4 0.0 -0.0
qt sys 35e-3 4 0.0 -0.680

20.2 Ramdom error

To define random multipole field errors on the lattice elements, user needs to define the seed of the random errors, and then follow the same rule as the ones to define systematic multipole error except replacing sys by rms. For example: seed seed_number quad rms 30e-3 6 2.4e-4 0.0 10 0.7e-4 0.0 14 0.9e-4 0.0

The random multipole error is multiplied by the random scale factor; the new value is added to the corresponding components of the magnetic field. The random scale factor is generated by a random function which follows the normal distribution (mean value is 0 and standard deviation is 1). The cut off value for the normal distribution function is 2 times of the RMS value. If user does not define seed for the random function before the setting of errors, then the code will stop and give an error message.

Here is example file to define random multipole error in the lattice:

#define seed for the ramdom multipole error seed 1000000

#dipole # dip 20e-3 2 2.2e-4 0.0 3 -3.0e-4 0.0 4 2.0e-5 0.0 5 -1.0e-4 0.0 6 -6.0e-5 0.0 7 -1.0e-4 0.0 dip rms 20e-3 2 2.2e-40 0.0 3 -3.0e-4 0.0 4 2.0e-5 0.0 5 -1.0e-4 0.0 6 -6.0e-5 0.0 7 -1.0e-4 0.0

#quadrupole quad rms 30e-3 3 -1.6e-4 0.0 4 -3.4e-4 0.0 6 2.4e-4 0.0 10 0.7e-4 0.0 14 0.9e-4 0.0

Q2 rms 30e-3 3 2.9e-4 0.0 4 -8.6e-4 0.0 6 0.7e-4 0.0 10 1.9e-4 0.0 14 1.0e-4 0.0 Q7 rms 30e-3 3 2.9e-4 0.0 4 -8.6e-4 0.0 6 0.7e-4 0.0 10 1.9e-4 0.0 14 1.0e-4 0.0

21 Misalignment error file

The misalignment error of the lattice elements can be defined in a file, and then the file is read into the lattice. User can define the systematic or random misalignment error of the lattice elements. There are two ways to define the misalignment error, one way is to define the error for all the families in one type, for example, the error for all the quadrupoles; another way is to define the error for each family, for example, the Q1 family of the quadrupoles. The systermatic misalignment error file works for the lattices with full or half quadrupoles; the random misalignment error file only works for lattice with full quadrupoles.

21.1 Systematic errors

To define the systematic misalignment error of the element, user just needs to follow the rules as below. input format of misalignment error: type/family name sys dx dy dr

The "keywords" means one type of lattice elements or the name of the family, and keywords of the type of lattice elements are: All all the elements in the lattice girder girder dipole dipole quad quadrupole sext sextupole bpm beam position monitor family name family name of the elements sys is a keyword to denote that user are setting the systematic displacement error. dx defines the displacement in x direction with unit [m]. dy defines the displacement in y direction with unit [m]. dr defines the rotation angle with unit [rad]). The line start with # is comment line. The blank line in the misalignment error file is neglected by the code.

The following is an example file to define systematic multipole error on Soleil lattice:

#———————————————————— # systematic alignment error for SOLEIL # name x(m) y(m) r (rad) #————————————————————

Table 23:

girder sys 100.0e-6 100.0e-6 0.5e-03 quad sys 30.0e-6 30.0e-6 80.0e-06 sext sys 30.0e-6 30.0e-6 100.0e-06 dipole sys 500.0e-6 500.0e-6 0.2e-03

21.2 Random errors

To define random misalignment errors on the lattice elements, user need to follow the same rule as the ones to define systematic misalignment error except replacing sys by rms. That is: input format of misalignment error: seed seed_number type/family name rms dx dy dr The random misalignment error is multiplied by the random scale factor; the new value is added to the corresponding components of the misalignment components. The random scale factor is generated by a random function which follows the normal distribution (mean value is 0 and standard deviation is 1), the cut off value for the normal distribution function is 2 times of the RMS value. If user does not define seed for the random function before the setting of errors, then the code will stop and give an error message. Here is example file to define random misalignment error in the lattice: #———————————————————— # random alignment error for SOLEIL # name x(m) y(m) r (rad) #————————————————————

Table 24:

girder rms 100.0e-6 100.0e-6 0.5e-03 quad rms 30.0e-6 30.0e-6 80.0e-06 sext rms 30.0e-6 30.0e-6 100.0e-06 dipole rms 500.0e-6 500.0e-6 0.2e-03 5.4 Vacuum chamber file User can define the script to set the vacuum chamber limitation around the ring. The characteristic for the vacuum chamber script are: Lines

start with # are comment. The format of the vacuum chamber definition is MK1, MK2, minimum x, maximum x, minimum y, maximum y. To set the vacuum chamber, it is needed to add two markers in the lattice, such as MK1 and MK2, MK1 is before the first element and MK2 is after the end element of the vacuum chamber region. The numbers of MK1 and MK2 are the same in the lattice. The units are [meter] for minimum x, maximum x, minimum y, maximum y. The first line is to define the global vacuum chamber limit around the ring, and the key words should be "Start","All".

The following is one example of the user vacuum chamber script:

#** # Script to set the vacuum chamber # #** # MK1 MK2 dxmin dxmax dymin dymax (Apertures in meter) Start All -35e-3 35e-3 -12.5e-3 12.5e-3 #sdm1 esdm -21e-3 21e-3 -5e-3 5e-3 debut ehu600 -35e-3 35e-3 -7e-3 7e-3 ssep esep -20e-3 35e-3 -7e-3 7e-3 ssdm esdm -21e-3 21e-3 -5e-3 5e-3 ssdac esdac -35e-3 25e-3 -2.5e-3 2.5e-3