

Soltracy3 manual for developpers

Jianfeng Zhang *

April 9, 2014

Contents

1	Change Log	1
2	Introduction	1
3	Non-parallel version Tracy	3
3.1	Compile	3
3.2	Run	3
4	Parallel version Tracy	3
4.1	Compile	3
4.2	Run	5
5	User input script	5
5.1	File path	6
6	File names	6
6.1	Lattice file name	6
6.2	Multipole field error file name (SOLEIL lattice)	7
6.3	Files of multipole field errors of correctors and skew quadrupoles (SOLEIL lattice)	8
6.4	File to define field strength of virtual coupling source elements (SOLEIL lattice)	9
6.5	Cut off value	9
7	Physics: Hamiltonian	9
7.1	Map of drift	12
7.2	Map of mutipoles	14
8	Magnetic field	18
9	FF of dipole	18
9.1	kick map	18

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

10 FF of quadrupole	18
10.1 kick map	18
11 Machine mode	18
11.1 Transfer line	18
11.2 Ring	19
12 General print files	19
12.1 *.sum	19
12.2 Twiss file	21
12.3 Lattice elements property file	22
12.4 File of the ring closed orbit or orbit of the transfer line	27
12.5 6-Dimensional (6D) closed orbit or 6D orbit of the transfer line	29
13 Lattice file	30
13.1 Lattice	30
13.2 Syntax	31
13.3 Variables	32
13.4 Start line	32
13.5 Global variables	32
13.6 DRIFT	32
13.7 DIPOLE	33
13.8 QUADRUPOLE	35
13.9 SKEW QUADRUPOLE	37
13.10SEXTUPOLE	37
13.11MULTIPOLE	38
13.12SOLENOID (to be updated)	41
13.13WIGGLER (To be updated.)	41
13.14FIELD MAP (To be updated..)	41
13.15INSERTION DEVICE (to be updated)	43
13.16RF CAVITY	44
13.17CORRECTOR	44
13.18BEAM POSITION MONITOR (To be updated)	45
13.19GIRDER	46
13.20MARKER	46
13.21ELEMENT BLOCK	46
13.22LINE	47
13.23RING	47
13.24End line	48
14 Commands	48
14.1 QuadFringeOnFlag	49
14.2 QuadFringeOffFlag	49
14.3 RfVoltageFlag	49
14.4 PrintTrackFlag	50
14.5 PrintTrackElemFlag	50

14.6	PrintTwissFlag	51
14.7	PrintCODFlag	51
14.8	ReadChamberFlag	52
14.9	ReadfefileFlag	52
14.10	ReadaefileFlag	52
14.11	CODCorrectFlag	53
14.12	TuneTracFlag	54
14.13	ChromTracFlag	55
14.14	AmplitudeTuneShiftFlag	55
14.15	EnergyTuneShiftFlag	55
14.16	FmapFlag	56
14.17	FmapdpFlag	57
14.18	ErrorCouplingFlag	58
14.19	ErrorCoupling2Flag	59
14.20	CouplingFlag	59
14.21	MomentumAccFlag	59
14.22	ReadMultipoleFlag	60
14.23	ReadVirtualSkewquadFlag	61
14.24	FitTuneFlag	61
14.25	FitTune4Flag	62
14.26	FitChromFlag	63
14.27	TouschekFlag (TO BE UPDATED)?????	63
14.28	IBSFlag (TO BE UPDATED) ????	63
14.29	TousTrackFlag (TO BE UPDATED)	64
14.30	PhaseSpaceFlag	64
14.31	IDCorrFlag (Tested for TaiWan light source; TO BE CONTINUE DEVELOPPED.)?????	64
15	Multipole field error file	65
15.1	Systematic errors	65
15.2	Random error	70
16	Misalignment error file	71
16.1	Systematic errors	71
16.2	Random errors	72
16.3	Vacuum chamber file	73

1 Change 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++ by Johann Bengsston and other scientists (???).

This manual is for SOLEIL version Tracy which is maintained and developed by L. Nadolski and J. Zhang in Synchrotron SOLEIL. The former manuals of Tracy are in the file folder “tracy/tracy/doc” of the Tracy3 code.

Rotation angle The user can define the design rotation angle of the magnetic lattice element in the lattice file.

parameters of the lattice element: rotation, field, misalignment errors? The user can define systematic and random rotation errors, the systematic and random displacement error, and systematic and random multipole field error of the lattice element in an external file (see section. ??), these variables can't be defined in the lattice file as an element parameters.

Simulation method: matrix or tracking? There are two types of simulation in Tracy, Matrix method (Taylor's nonsymplectic map) and symplectic tracking. The default mode is the symplectic tracking. User can active the matrix mode by setting globval.MatMeth in the source code...(Need to update in the input script to active this mode. Be careful, during the lattice reading, the t2init() will active the tracking mode...)

Define RING or transfer line type in the lattice file in Lattice.Read() of t2lat.cc, it seems like we can define a transfer line in Tracy, using the variable V.Symmetry != 0, how to define a transfer line in the lattice? ANSWER: RingType = 1; define the ring, or RingType = 0; define the transfer line. The variable is: globval.RingType = 1/0, 1: ring, 0: transfer line.

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 is acceptable 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 “*\$HOME/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 ??). 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/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 nutil.c nrcheck.c nrlinwww.c
nrframe.c ../tracy/src/tracy_lib.cc →

psoltracy_SOURCES = psoltracy.cc nutil.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/phymach/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/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.

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
```

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 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; the BPM acts as both horizontal BPM and vertical BPM. **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 ??.

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: Hamiltonian

The lattice elements in Tracy can be classified into two types: drift and multipoles (thick or thin).

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 Hamiltonian in the curvilinear coordinate system (Merz...)

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

h_{ref} is the curvature of reference orbit inside the dipole. The $(x, P_x; y, P_y; -t, \delta)$ are the canonical coordinates of the H , and $P_x, P_y, P - z$ are respectively the horizontal, vertical, and longitudinal mechanical momentums. A more convenient way is to expand the electron motion around the motion of a reference particle with total momentum P_0 , so

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

with

$$p_x = \frac{P_x}{P_0} \quad (7)$$

$$p_y = \frac{P_y}{P_0} \quad (8)$$

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

$$\delta = \frac{\Delta P}{P_0} \quad (10)$$

$$B\rho = \frac{P_0}{e} (\text{magnetic rigidity}) \quad (11)$$

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)

$$H(x, p_x; y, p_y; -ct, \delta; s)$$

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

The equation is the exact Hamiltonian used in Tracy. The A_x are due to the longitudinal fringe field, this is a natural result 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

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

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

$$+ \underbrace{\frac{x}{\rho} + \frac{x}{2\rho^2}}_{\text{Geometric components due to the curvilinear coordinate inside the dipole}} \quad (14)$$

$$+ \underbrace{\frac{A_z}{B\rho}}_{\text{magnet contribution}} \quad (15)$$

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

$$(17)$$

From eqn. 12, 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. 12, we can get the Hamiltonian inside a drift is

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

In large ring, in order to increase the tracking speed, we can expand the square root in eqn. 18 to get the approximate Hamiltonian. That is,

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

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

From eqn. 18, it is easy to get the map inside the drift for the **small ring**:

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

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

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

$$\text{or} \quad (24)$$

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

$$p_x = p_{x0} \quad (26)$$

$$p_y = p_{y0} \quad (27)$$

$$\delta = \delta_0 \quad (28)$$

From eqn. 19, it is easy to get the map inside the drift for the **big ring**:

$$x' = \frac{dx}{ds} = \frac{\partial H}{\partial p_{x0}} = \frac{p_{x0}}{1 + \delta_0} \quad (29)$$

$$y' = \frac{dy}{ds} = \frac{\partial H}{\partial p_{y0}} = \frac{p_{y0}}{1 + \delta_0} \quad (30)$$

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

$$\text{or} \quad (32)$$

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

$$p_x = p_{x0} \quad (34)$$

$$p_y = p_{y0} \quad (35)$$

$$\delta = \delta_0 \quad (36)$$

7.2 Map of multipoles

From eqn. 12, we can get the Hamiltonian inside a multipole without fringe field is

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

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

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

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

In eqn. 37, 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 7.1), 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:

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

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

$$+ \frac{x}{\rho} + \frac{x}{2\rho^2} - \frac{A_z}{B\rho} \quad (43)$$

Similarly, by expand the square root in eqn. 37, we can get the approximate Hamiltonian of the multipole as

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

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

$$+ \frac{x}{\rho} + \frac{x}{2\rho^2} - \frac{A_z}{B\rho} \quad (46)$$

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

where h_{ref} is the curvature due to the curvilinear coordinates inside the dipole, $h_{\text{ref}} = 1/\rho_{\text{ref}}$; h_{bend} is due to the bending curvature of the dipole which is

determined by the dipole field, $h_{\text{bend}} = 1/\rho_{\text{bend}}$. $h_{\text{ref}} \neq 0$ only inside dipoles.

In the dipole, $A_z \neq 0$, and $h_{\text{ref}} \neq 0$, so from eqn. 41, we can get the kick map due to the dipole in the **small ring** which use the exact Hamiltonian:

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

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

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

$$p'_y = -\frac{\partial H}{\partial x_0} = \frac{1}{B\rho} \frac{\partial A_z}{\partial y_0} = \frac{B_x}{B\rho} \quad (51)$$

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

or

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

$$\delta = \delta_0 \quad (55)$$

$$B\rho = \frac{p_0}{e} \quad (56)$$

where

$$\frac{\partial A_z}{\partial x} = -B_y \quad (57)$$

$$\frac{\partial A_z}{\partial y} = B_x \quad (58)$$

since

$$A_x = 0 \quad (59)$$

$$A_y = 0 \quad (60)$$

inside the body of the multipoles, and

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

$\hat{x}\hat{y}$	\hat{z}
$\frac{\partial}{\partial x} \frac{\partial}{\partial y}$	$\frac{\partial}{\partial z}$
$A_x A_y$	A_z

In the dipole, $A_z \neq 0$, and $h_{\text{ref}} \neq 0$, so from eqn. 44, we can get the kick map due to the dipole in the **big ring** which use the exact Hamiltonian:

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

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

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

$$p_y' = -\frac{\partial H}{\partial y_0} = \frac{1}{B\rho} \frac{\partial A_z}{\partial y_0} = \frac{B_x}{B\rho} \quad (64)$$

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

$$\text{or} \quad (66)$$

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

$$\delta' = 0 \quad (68)$$

For other multipoles (quadrupoles, sextupole, decapole, octupole, etc), the curvilinear coordinate is the cartesian coordinates, that is $h_{\text{ref}} = 0$, so from

eqn. 48, the kick map for these multipoles in the small ring is:

$$x' = \frac{dx}{ds} = \frac{\partial H}{\partial p_{x0}} = 0 \quad (69)$$

$$y' = \frac{dy}{ds} = \frac{\partial H}{\partial p_{y0}} = 0 \quad (70)$$

$$p'_x = -\frac{\partial H}{\partial y_0} = -(h_{\text{bend}} + \frac{B_y}{B\rho}) \quad (71)$$

$$p'_y = -\frac{\partial H}{\partial y_0} = \frac{1}{B\rho} \frac{\partial A_z}{\partial y_0} = \frac{B_x}{B\rho} \quad (72)$$

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

$$\text{or} \quad (74)$$

$$(ct)' = 0 \quad (75)$$

$$\delta' = 0 \quad (76)$$

With $h_{ref} = 0$, from eqn: 61, the kick map for these multipoles (quadrupoles, sextupole, decapole, octupole, etc) in the big ring is:

$$x' = \frac{dx}{ds} = \frac{\partial H}{\partial p_{x0}} = 0 \quad (77)$$

$$y' = \frac{dy}{ds} = \frac{\partial H}{\partial p_{y0}} = 0 \quad (78)$$

$$p'_x = -\frac{\partial H}{\partial y_0} = -(h_{\text{bend}} + \frac{B_y}{B\rho}) \quad (79)$$

$$p'_y = -\frac{\partial H}{\partial y_0} = \frac{1}{B\rho} \frac{\partial A_z}{\partial y_0} = \frac{B_x}{B\rho} \quad (80)$$

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

$$\text{or} \quad (82)$$

$$(ct)' = 0 \quad (83)$$

$$\delta' = 0 \quad (84)$$

From the eqn 48 to eqn. 77, 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 accelerator.....)

9 FF of dipole

9.1 kick map

E. Forest

10 FF of quadrupole

10.1 kick map

11 Machine mode

11.1 Transfer line

Tracy can simulate the transfer line (TL), to calculate some features of this type of the machine, for example, to transfer the twiss functions and the orbits. User can define the TL type in the lattice file using the command:

RingType = 0

Since the symplecticity is not important to the transfer map of a TL, the twiss functions and orbits of a TL is calculated using a 6-dimensional (6D) linear Taylor transfer matrix.

- The initial phase of the TL is 0.
- The initial parameters of the TL (alpha, beta, mu, and orbit) are defined in the file “soleilcommon.cc”. Maybe these parameters can be defined in the lattice file or “*.prm” in the future.
- The calculated optical functions at i-th lattice element can be called using: Cell[i] → Alpha, Cell[i] → Beta, Cell[i] → Nu, Cell[i] → Eta, Cell[i] → Etab.
- The optical functions (alpha, beta, mu), orbits will be automatically calculated and saved in the file “cod.out”.
- If the BPMs are defined in the lattice file, and the BPMs names are declared in the “*.prm” file with the command:

`bpm_name` `BPMs names`

then the 6D orbits at the BPM positions will be saved in the file “6Dcod.out”.

- See also section 12.

Called functions: `CellTwiss_M()` in `t2ring.cc`.

The functions with “...M()” are for the transfer line.

11.2 Ring

User can define the ring type in the lattice file using the command:

`RingType = 1`

If this machine type is not defined, then the default machine type is ring.

- The momentum compaction factor is calculated and printed to the terminal.
- The twiss functions, and COD files will be automatically printed to a file “cod.out”.
- If the BPMs are defined in the lattice, and the BPMs names are declared in the “*.prm” file using the command:

`bpm_name` `BPM names`

then the 6D closed orbit at the position of the BPMs will be saved in a file “6Dcod.out”.

- See also section 12.

12 General print files

After reading the lattice file, Tracy will print several files with the general informations of the lattice.

12.1 *.sum

This file is with the same name as the lattice file but with the file extention “*.sum”. This file is with the information:

general inforamtion

dPcommon = 1.000e-10	dPparticle = 0.000e+00	Energy [GeV] = 0.050
MaxAmplx [m] = 0.000e+00	MaxAmply [m] = 0.000e+00	RFAccept [%] = +/- 100.00
MatMeth = FALSE	Cavity_On = FALSE	Radiation_On = FALSE
bpm = 0	qt = 0	
hcorr = 0	vcorr = 0	
Chambre_On = FALSE		
alphac = 0.0000e+00		
nux = 1.382583	nuy = 0.859382	
ksix = 0.000000	ksiy = 0.000000	

dPcommon : relative energy spread.

dPparticle :

Energy : beam energy [GeV].

MaxAmplx : maximum horizontal aperture size.

MaxAmply : maximum vertical aperture size.

RFAccept [%] : RF acceptance.

MatMeth : use linear transfer matrix or tracking? True/false.

Cavity_On : with RF cavity or not?

Radiation_On : synchrotron radiation is on or off?

bpm : number of BPMs. If the user doesn't specify the name of the BPM in the "*.prm" file, then this value is 0, it means that Tracy think there is no BPMs defined in the lattice? (need to check...)

qt : number of skew quadrupoles. If the user doesn't specify the name of the qt in the "*.prm" file, then this value is 0, it means that Tracy think there is no skew quadrupoles defined in the lattice? (need to check...)

hcorr : number of horizontal correctors. If the user doesn't specify the name of the hcorr in the "*.prm" file, then this value is 0, it means that Tracy think there is no horizontal correctors defined in the lattice? (need to check...)

vcorr : number of vertical correctors. If the user doesn't specify the name of the vcorr in the "*.prm" file, then this value is 0, it means that Tracy think there is no vertical correctors defined in the lattice? (need to check...)

Chambre_On : read the vacuum chamber definition from an external file? If it is true, then an external file with the definition of the vacuum chamber aperture must be defined in the “*.prm” file.

alphac : first order momentum compaction factor. For a transfer line, this variable is 0.

nux : horizontal phase advance for a transfer line; horizontal tune for the ring. This variable is normalized by 2π .

nuy : vertical phase advance for a transfer line; vertical tune for the ring. This variable is normalized by 2π .

ksix : horizontal chromaticity. This variable is only valid for a ring; it is 0 for a transfer line.

ksiy : vertical chromaticity. This variable is only valid for a ring; it is 0 for a transfer line.

This file also has the 6×6 transfer matrix (one turn map for a ring):

OneTurn matrix:

matrix:

-1.397484e+00	1.028931e+01	0.000000e+00	0.000000e+00	-1.089759e-02	0.000000e+00
1.025593e-01	-1.470689e+00	0.000000e+00	0.000000e+00	1.695979e-01	0.000000e+00
0.000000e+00	0.000000e+00	5.943899e-01	-5.866019e+00	0.000000e+00	0.000000e+00
0.000000e+00	0.000000e+00	5.043746e-01	-3.295264e+00	0.000000e+00	0.000000e+00
0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00	1.000000e+00	0.000000e+00
0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00

Finally, the twiss parameters at the entrance of the first element is saved:

Twiss parameters at entrance:

Betax [m] = 8.100e+00	Alphax = -0.000e+00	Etax [m] = 0.000e+00	Etaxp = 0.000e+00
Betay [m] = 8.100e+00	Alphay = -0.000e+00	Etay [m] = 0.000e+00	Etayp = 0.000e+00

12.2 Twiss file

This file is with the twiss functions at the end of all lattice elements, the file name is the same as the lattice file name, but with the file extension “*.twi”.

name	s	alphax	betax	nux	etax	etapx	alphay	betay	nuy	etay	etapy
	[m]		[m]		[m]			[m]		[m]	

where

name : name of the lattice element.

s [m]: longitudinal position of the element.

alphax α_x : horizontal alpha.

betax β_x [m]: horizontal beta functions.

betay β_y [m]: vertical beta functions.

etax η_x [m]: horizontal linear dispersion function.

etay η_y [m]: vertical linear dispersion function.

etapx $d\eta_x/mathrmds$: horizontal linear dispersion function η_x with respect to longitudinal coordinate s .

etapy $d\eta_y/mathrmds$: vertical linear dispersion function η_y with respect to longitudinal coordinate s .

nux ν_x : horizontal phase advance of the transfer line, or horizontal tune of the ring. This value is normalized by 2π .

nuy ν_y : vertical phase advance of the transfer line, or vertical tune of the ring. This value is normalized by 2π .

12.3 Lattice elements property file

Each time when running Tracy, the design values of lattice elements will be printed to the file “flat_file.dat”. This file is a very important file to check the Tracy is correctly running.

The printed properties of one lattice element are (also see MML) different for the element type: drift and multipoles (thick or thin).

drift:

1st line:

Name; familylist number;

kid number;

index number in the lattice, for example, “1” means the first element;

2nd line: type; method, N

type: type of the lattice element defined in Tracy:

marker -1
drift 0
multipole 1
cavity 2
thin kick 3
wiggler 4
kick_map 6

method: symplectic integration method. For drift, method = 0.

N: number of pieces of the elements to be cut. For drift, N = 0.

3rd line: MaxamplX0, MaxamplX1, MaxamplY0, MaxamplY1.

MaxamplX0: minimum horizontal aperture size;

MaxamplX1: maximum horizontal aperture size;

MaxamplY0: minimum vertical aperture size;

MaxamplY1: maximum vertical aperture size;

4th line: Element length.

For example:

```
sd11      1      1      2
0          0      0
0.000e+00 0.000e+00 0.000e+00 0.000e+00
2.000e-01
```

multipoles:

Depending on the elements length, the printed information of thick and thin multipoles are different. But both types of multipoles have the following general informations as the drift:

1st line:

Name; familylist number; kid number; index number in the lattice, for example, “1” means the first element;

Name: element name.

2nd line: type; method, N

type: type of the lattice element defined in Tracy:

marker	-1
drift	0
thick multipole	1
cavity	2
thin multipole	3
wiggler	4
kick_map	6

method: symplectic integration method defined in the lattice, or the default method for this element.

N: number of pieces of the elements to be cut defined in the lattice or the default value “1”.

3rd line: MaxamplX0, MaxamplX1, MaxamplY0, MaxamplY1.

MaxamplX0: minimum horizontal aperture size;

MaxamplX1: maximum horizontal aperture size;

MaxamplY0: minimum vertical aperture size;

MaxamplY1: maximum vertical aperture size;

For a thick multipoles,

4th line: DsX, DsY, PdTpar, PdTsys, PdTrms*PdTrnd.

DsX: horizontal displacement error of the element.

DsY: vertical displacement error of the element.

PdTpar: total designed rotation angle [degree].

PdTsys: systematic designed rotation angle defined in the lattice.[degree].

PdTrms*PdTrnd: random rotation errors of the element multiplied by a scale factor PdTrnd. The default value of PdTrnd is 0.

For a thin multipole:

4th line: DsX, DsY, PdTsys+ PdTrms*PdTrnd.

DsX: horizontal displacement error of the element.

DsY: vertical displacement error of the element.

PdTsys + pdTrm*PdTrnd: The sum of the systematic field components and random multipole field components, field components of the same order are added.

PdTsys: systematic designed rotation angle defined in the lattice.[degree].

PdTrms*PdTrnd: random rotation errors of the element multiplied by a scale factor PdTrnd. The default value of PdTrnd is 0.

5th line: PL, Pirho, PTx1, PTx2, PH1, PH2, Pgap

PL: length of the element [m].
 Pirho: curvature of the radius of the dipole, $1/\rho$ [1/m]. Pirho denotes the bending angle of the dipole, since $\text{Pirho} = \frac{\theta}{180} * \pi/L$
 PTx1: entrance angle of the dipole, [degree].
 PTx2: exit angle of the dipole, [degree].
 PH1: bending curvature of the entrance pole face of dipole, [degree].
 PH2: bending curvature of the exit pole face of dipole, [degree].
 Pgap: full dipole gap, [m].

The definitions of PTx1, PTx2, PH1, PH2, Pgap can be found on P116 SAC-75.

6th line: nmpole, n_design

nmpole: multipole field order of the element. The value is 0 for dipole.

n_design: design order of the multipole field of the element.

The value of n_design is:

- 0: All
- 1: dipole (n = 1).
- 2: quadrupole (n = 2).
- 3: sextupole (n = 3).
- 4: octupole (n = 4).
- 5: decuple (n = 5).
- 6: dodecapule (n = 6).

7th line and then: i, HOMmax+i, HOMmax-i

i: multipole field order of the element.

HOMmax+i: normal multipole field components, normalized by $B\rho$.

HOMmax-i: skew multipole field components, normalized by $B\rho$.

Notes:

- For a dipole element, there is no multipole field components. This is due to the kick map from the expanded Hamiltonian, the vector component of the dipole is treated separated from other multipoles, and the contribution of the dipoles are included in the curvature of radius of the dipole $1/\rho$.
- The non-zero multipole field components will be printed from 1-th order to the maximum order Porder.

For example:

```

bend1      17          1          15
1          4          4
0.0000e+00 0.0000e+00 0.0000e+00 0.0000e+00
0.0000e+00 0.0000e+00 0.0000e+00 0.0000e+00
2.7645e-01 2.8409e+00 0.0000e+00 0.0000e+00 0.0000e+00 0.0000e+00 0.0000e+00
0          1

sx1        22          1          13
1          4          4
0.0000e+00 0.0000e+00 0.0000e+00 0.0000e+00
0.0000e+00 0.0000e+00 0.0000e+00 0.0000e+00
9.9999e-07 0.0000e+00 0.0000e+00 0.0000e+00 0.0000e+00 0.0000e+00 0.0000e+00
1          3
3          -8.9672e+06 0.0000e+00

```

Cavity

RF cavity.

1st line: Name; familylist number; kid number; index number in the lattice, for example, “1” means the first element;

2nd line: type; method, N

type: type of the lattice element defined in Tracy:
cavity 2

method: symplectic integration method defined in the lattice, or the default method for this element.

N: number of pieces of the elements to be cut defined in the lattice or the default value “1”.

3rd line: MaxamplX0, MaxamplX1, MaxamplY0, MaxamplY1.

MaxamplX0: minimum horizontal aperture size;
MaxamplX1: maximum horizontal aperture size;
MaxamplY0: minimum vertical aperture size;
MaxamplY1: maximum vertical aperture size;

4th line: Pvolt/energy, $2*\pi*Pfreq/co$, phase, energy.

Pvolt/energy: RF voltage normalized by beam energy [eV].
 $2*\pi*Pfreq/co$: RF frequency normalized by the speed of light.
phase: RF phase [degree].
energy: beam energy [eV].

marker

1st line: Name; familylist number; kid number; index number in the lattice, for example, “1” means the first element;

2nd line: type; method, N

type: type of the lattice element defined in Tracy: marker -1

3rd line: MaxamplX0, MaxamplX1, MaxamplY0, MaxamplY1.

MaxamplX0: minimum horizontal aperture size;
MaxamplX1: maximum horizontal aperture size;
MaxamplY0: minimum vertical aperture size;
MaxamplY1: maximum vertical aperture size;

For example:

```
debut      56          1          1
-1         0          0
0.0000e+00 0.0000e+00 0.0000e+00 0.0000e+00
```

wiggler:

insertion device:

12.4 File of the ring closed orbit or orbit of the transfer line

The horizontal and vertical closed orbits of the ring lattice or the orbits of the transfer line at all lattice elements are be printed to the file “cod.out”.

The user can also choose to print the orbit at the position of BPM, but the name of the BPM must be defined at the beginning of the “*.prm” file, since both the horizontal and vertical orbits will be printed at the BPM position, it is only needed to define a general BPM in the lattice file.

i	name	s	code	betax	nux	betay	nuy	xcod	ycod	dSx	dSy	dipx	dipy
i:													
name:	name of the lattice element.												
s:	longitudinal position of the element [m].												
code:	code type of the element.												
	drift: 0.0												
	dipole: 0.5												
	focusing quadrupole: 1												
	defocusing quadrupole: -1												
	focusing sextupole: 1.5												
	defocusing sextupole: -1.5												
	bpm: 2.0;												
	only when the name of the bps are defined in the “*.prm” file, this value is 2.0, otherwise Tracy will not find them.												
others:	0.0												
betax:	horizontal beta function [m].												
nux:	horizontal tune (normalized by 2π).												
betay:	vertical beta function [m].												
nuy:	vertical tune (normalized by 2π).												
xcod:	horizontal closed orbit, [mm].												
ycod:	vertical closed orbit, [mm].												
dSx:	horizontal displacement error, [mm].												
dSy:	vertical displacement error, [mm].												
dipx:	horizontal kick angle, only valid for horizontal correctors, [mrad].												
dipy:	vertical kick angle, only valid for vertical correctors, [mrad].												

For example:

drift:													
2	sd5	0.78	0.0	3.873	0.033	2.137	0.065	-0.002	0.000	0.000	0.000	-0.000	0.000
quadrupole:													
5	qp1	0.93	-1.0	4.403	0.039	2.035	0.076	-0.002	0.000	0.000	0.000	-0.000	0.000
dipole:													
10	dip	2.16	0.5	0.126	0.196	7.446	0.148	0.001	0.000	0.000	0.000	-0.000	0.000
corrector:													
14	ch01	2.29	0.0	0.127	0.420	8.950	0.150	0.001	0.000	0.000	0.000	0.010	0.000
defocusing sextupole:													
19	sx3	2.59	-1.5	2.566	0.523	7.225	0.155	0.005	0.000	0.000	0.000	-0.000	0.000
focusing sextupole:													
27	sx2	3.06	1.5	9.603	0.535	2.375	0.176	0.010	0.000	0.000	0.000	-0.000	0.000

12.5 6-Dimensional (6D) closed orbit or 6D orbit of the transfer line

If the BPMs are defined in the lattice, and the BPM names are defined in the user input file “*.prm” using the command:

bpm_names **BPMx**

then Tracy will print out the 6D closed orbit of the ring or the 6D orbit of the transfer line at the locations of the BPMs to a file “6Dcod.out”. The printed parameters are:

i name s x px y py delta ct

i : index of the lattice element.

name : name of the lattice element.

s : longitudinal position of the element, [m].

x : horizontal orbit at the end of the element, [mm].

px : derivative of the horizontal orbit x with respect to s at the end of the element.

y : vertical orbit at the end of the element, [mm].

py : derivative of the vertical orbit y with respect to s at the end of the element.

delta : energy spread.

`ct` : longitudinal displacement with respect to the reference particle.

For example:

```
3 bpmx 7.75e-01 -2.09e-03 1.53e-03 0.00e+00 0.00e+00 0.00e+00 9.16e-13
```

13 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 obligatory is for the correct calculation of positive/negative momentum compaction factor. To deactivate 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 curvilinear 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 coordinates $h = 1/\rho$ is not zero only inside dipole; in other magnets and drift $h = 0$, and the curvilinear coordinates goes to cartesian coordinates. As a result, the independent variable s inside the dipole hamiltonian is the arc length, while the straight length in other lattice elements.

13.1 Lattice

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

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

$B\rho$ 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

$$b_3 = \frac{1}{B\rho} \frac{1}{2} \frac{\partial^2 B_y}{\partial x^2} \Big|_{x=0, y=0}$$

$$a_3 = \frac{1}{B\rho} \frac{1}{2} \frac{\partial^2 B_x}{\partial x^2} \Big|_{x=0, y=0}$$

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 $n = 0, 1, 2, 3, \dots$, and the n^{th} field strength components of the lattice element b_n and a_n are defined as

$$b_n = \frac{1}{B\rho} \frac{\partial^n B_y}{\partial x^n} \Big|_{x=0, y=0}$$

$$a_n = \frac{1}{B\rho} \frac{\partial^n B_x}{\partial x^n} \Big|_{x=0, y=0}$$

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

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

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

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

13.2 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.

13.3 Variables

User can define 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;
```

13.4 Start line

The lattice file must begin with the sentence:

```
define lattice;
```

This definition is mandatory.

13.5 Global variables

After define the ring, user needs to define the system parameters of the lattice:

RingType = 1/0; define the machine type: ring or transfer line. 1 defines the ring, 0 defines the transfer line. This variable definition is optional, the default machine type is ring.

Energy = ; the beam energy with unit [GeV]. This variable definition is mandatory.

dP = ; the relative momentum offset of the particle. This variable definition is mandatory.

CODEps = ; the convergence for the algorithm to find the closed orbit. This variable definition is mandatory.

Example:

```
Energy = 2.739;  
dP = 1.0d-10;  
CODEps = 1.0d-15;
```

13.6 DRIFT

```
Name: drift, L = (...);
```

The definition of the length of a drift is mandatory.

Example:

Table 1: Parameters of dipole in the lattice file.

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

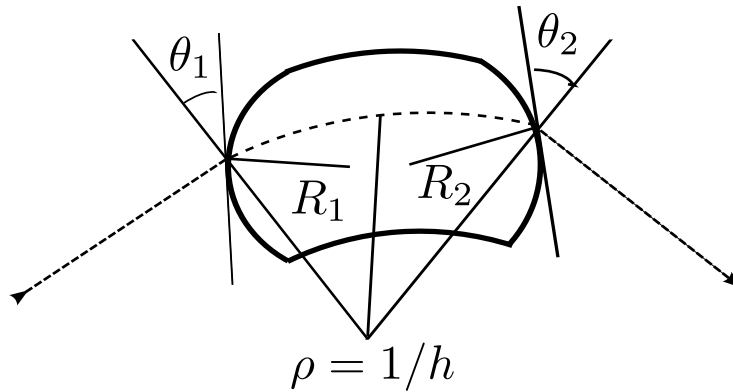


Figure 1: Field boundaries of the dipoles. $T1$ is the entrance angle; $T2$ is the exit angle; $H1$ is the radius of curvature of the entrance pole face; $H2$ is for the exit pole face; ρ is the local radius of the design orbit in the dipole.

SD1a: drift, L= 0.900000;

13.7 DIPOLE

Element_name: bending, L = $\langle \dots \rangle$, T = $\langle \dots \rangle$, T1 = $\langle \dots \rangle$, T2 = $\langle \dots \rangle$, H1 = $[\dots]$, H2 = $[\dots]$, gap = $[\dots]$, edge_effect1 = $[\dots]$, edge_effect2 = $[\dots]$, K = $[\dots]$, method = $\langle \dots \rangle$, N = $\langle \dots \rangle$;

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;

Attention:

- The bending magnet in Tracy is a symplectic element. The independent variable s inside the dipole hamiltonian is the arc length, so the length of the dipole is defined the curved path length of the design particle inside the dipole.

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	$L \neq 0, \text{m}^{-2}.$ $L = 0, \text{m}^{-1}.$	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.

- 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;

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;

13.8 QUADRUPOLE

Definition:

Element_Name: **quadrupole**, L = [...], tilt = [...], K = [...], FF1 = [...], FF2 = [...], FFscaling = [...], method = [...], N= (...);

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.

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 quadrupole.
K	$L \neq 0$, [m ⁻²]. $L = 0$, [m ⁻¹].	0.0	$L \neq 0$, Gradient $\frac{\partial B_y}{B\rho}$; $L = 0$, integrated field strength $\frac{\partial B_y}{\partial x} \frac{L}{B\rho}$.
FF1	-	0	1 or 0. 1: to activate fringe field at the left edge. <code>QuadFringeOnFlag</code> should be set in the Tracy input file. 0: left fringe field off.
FF2	-	0	1 or 0. 1: to activate fringe field at the right edge. <code>QuadFringeOnFlag</code> 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.

13.9 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`

13.10 SEXTUPOLE

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

Table 4: Parameters of sextupoles in the lattice file.

Parameter Name	Units	Default	Description
L [m]		0.0	Length
K	$L \neq 0, [\text{m}^{-3}]$. $L = 0, [\text{m}^{-2}]$.	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.

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 $n = 1, 2, 3, \dots$; 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 $n = 0, 1, 2, 3, \dots$. For sextupole, $n = 2$, and its gradient is defined as

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

13.11 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 = [...], method
= [...];

The multipole field components a_n and b_n are defined as

$$\begin{aligned} b_n &= \frac{1}{B\rho} \frac{1}{(n-1)!} \frac{\partial^{n-1} B_y}{\partial x^{n-1}} \Big|_{x=0, y=0} \\ a_n &= \frac{1}{B\rho} \frac{1}{(n-1)!} \frac{\partial^{n-1} B_x}{\partial x^{n-1}} \Big|_{x=0, y=0} \\ B\rho &= \frac{p_0}{e} \end{aligned} \tag{85}$$

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

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 quadrupole.????
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.

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

$$\begin{aligned} b_n &= \frac{1}{B\rho} \frac{B_y^{tip}}{r_0^{n-1}} \\ a_n &= \frac{1}{B\rho} \frac{B_x^{tip}}{r_0^{n-1}} \\ \vec{B}^{tip} &= B_y^{tip} \hat{y} + B_x^{tip} \hat{x} \end{aligned}$$

The n^{th} order magnetic field is

$$B_y^n + i * B_x^n = (b_n + i * a_n) * (x + i * y)^n \quad (86)$$

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

$$\begin{aligned} B_y + i * B_x &= (b_n + i * a_n) * (x + i * y) * (x + i * y)^{n-1} + O(n-1) \\ &= [(b_n * x - a_n * y) + b_{n-1} + i * (a_n * x + b_n * y + a_{n-1})] * (x + i * y)^{n-1} \\ &+ O(n-2) \\ &+ \dots \end{aligned} \quad (87)$$

The eqn. 87 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 propotional 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 4th 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 15) (???? To be checked, can this routine measured 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 $n = 0, 1, 2, 3, \dots$, and the n^{th} field strength components of the lattice element b_n and a_n are defined as

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

13.12 SOLENOID (to be updated)

13.13 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 >;

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);

13.14 FIELD MAP (To be updated..)

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

;

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.

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";

13.15 INSERTION DEVICE (to be updated)

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;

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.

Example:

WIGSLIC: **insertion**, N = 10, scaling1=1.0, scaling2=1.0, method=2,
file1="/home/sources/phymach/tracy2.7/w150g11pole60_oppose_radia_pour_tracy.txt",
file2= "/home/sources/phymach/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.

13.16 RF CAVITY

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

Table 9: Parameters of RF cavity.

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

Example:

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

To calculate the stable phase for the lattice with positive or negative momentum compaction factor, the definition of the harmonic number of the RF cavity is mandatory. The other parameters of RF cavity are optional, and their default values are 0.

13.17 CORRECTOR

Element_Name: **corrector**, horizontal/vertical, L = , kick = , roll = , N = ,
method = integrated method;

Example 1:

CH: **corrector**, horizontal, method=intmeth;

It defines a horizontal corrector.

Example 2:

Table 10: Parameters of correctors.

Parameter Name	Units	Default	Description
Horizontal / vertical	-	horizontal	horizontal: horizontal corrector; vertical: vertical corrector
L	[m]	0.0	length of the corrector
kick	[rad]	0.0	kick angle
roll	-	0.0	designed rotation angle
N	-	1	cut piece of the element
method	-	1, first order	1, 2, 4. Order of symplectic integration method. Value 1 means 1st order, 2 means 2nd order, and 4 means 4th order

CV: **corrector**, **vertical**, **method=intmeth**;

It defines a vertical corrector.

NOTES:

- The corrector type can be used as a corrector to do both orbit correction or as a kicker to kick the beam.
- The corrector is treated as a multipole in TRACY, that is, if the length of the corrector **L** is non-zero, then the element is treated as a thick multipole, if **L** is zero, then the corrector is treated as a thin multipole.
- 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.

13.18 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_name beaPosMonitor`

Here “beaPosMonitor” is the name of the BPMs defined in the lattice.

13.19 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.

13.20 MARKER

Element_Name: `marker;`

13.21 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.

13.22 LINE

User can define the cell structure using the command:

CELL : `< block name >, SYMMETRY=< symmetry >;`

`<block name >`: the name of a block;

`<symmetry >`: 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.

13.23 RING

To define the ring, use the command:

RING: `elem, block. |`

ts similar to define an element block, but must with the fixed symbol name “RING”.

For example:

RING: `DEBUT,SUP1,SUP2,SUP3,SUP4,CAV,FIN;`

13.24 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.

14 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,

(1)

FitTune4Flag qp7a qp7b qp9a qp9b 18.18 10.28

(2)

ReadMultipoleFlag

(3)

FitTune4Flag qp7a qp7b qp9a qp9b 18.202 10.317

With the command (1), the code will fit tunes to the target tunes (18.18 10.28);

with the command (2), the code then read multipole field errors into the lattice;

with the command (3), the code 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 both the non-parallel and parallel version Tracy; the output files are the same for both versions.

- The key words in the user defined input file “*.prm” is case sensitive, that is, the key word with captical letters is different from the one with small letters.
- The variables and lattice element names in Tracy lattice file are not case sensitive, user can define the variables and lattice elements names with small or captical letters.

14.1 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.

14.2 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.

14.3 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.

14.4 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 [?].

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	-

Table 11: Parameters of command “PrintTrackFlag”.

14.5 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 neleml1 neleml2
```

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 [?].

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	-	-

Table 12: Parameters of command “PrintTrackElemFlag”.

14.6 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”.

14.7 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.

See the details in seciton ...???????????

14.8 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?????.

14.9 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.

14.10 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??????.

14.11 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:

- 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.
- The code corrects the closed orbit distortion.
- 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.
- A file “OrScanFile.out” will be saved with the summaries of the mean and RMS values of the orbits before and after correction.
- 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????.

14.12 TuneTracFlag

Get tunes by tracking

To get tunes by tracking, use command:

TuneTracFlag **T**

the tunes obtained by tracking are printed on the screen.

14.13 ChromTracFlag

Get chromaticities by tracking

To get chromaticity by tracking, use command:

ChromTracFlag

The chromaticities obtained by tracking are printed on the screen.

14.14 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 13: Parameters of the command to calculate tune shift with amplitude.

Parameters	Meaning	Default	units
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	0.025	[m]
ymax	Maximum amplitude of y	0.005	[m]
delta	Energy offset of the particle	0.0	-

14.15 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 14: Parameters of the command to calculate tune shift with energy.

parameter	meaning	default	units
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 -	

14.16 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 $x = 1e-6 + xcod \quad xmax + xcod$, with the step size $xmax/nturn$, $z = 1e-6 + zcod \quad zmax + zcod$, with the step size $ymax/nturn$; $px = 0 + pxcod$; $pz = 0 + pzcod$; $ctau = 0$; $delta = \text{set value}$.

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 15: Parameters of the command to do frequency map analysis for on momentum particle.

Parameters	meaning	Default	units
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	0.025	[m]
ymax	Maximum amplitude in the vertical direction	0.005	[m]
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 particle and external file? If true, the output file is “fmap_file.loss”.	-	-

14.17 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.

parameters	meaning	default	units
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	0.025	[m]
emax	Maximum energy offset of the particle	0.005	-
y	Amplitude in the vertical direction	0.0	[m]
diffusion	Boolean flag to compute tune diffusion	true	-
printloss (optional)	Boolean flag, true/false; print out the last information of the tracked particle to and external file? If true, the output file is “fmapdp_file.loss”.	-	

Table 16: Parameters of the command “FmapdpFlag”.

14.18 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”.

14.19 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.

14.20 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”.

14.21 MomentumAccFlag

Calculate momentum acceptance by tracking

The following command calculate momentum acceptance at a predefined lattice region by tracking:

```
MomentumAccFlag MomAccFile TrackDim irstart istop deltaminp Deltamaxp  
nstepp deltaminn deltamaxn nsteppn 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.

parameter	meaning	Default	units
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]

Table 17: Parameters of the command to calculate momentum acceptance.

14.22 ReadMultipoleFlag

Read multipole field error from a file (SOLEIL lattice)

After defining the file names of multipole field errors on SOLEIL storage ring (see 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.

14.23 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 [].

14.24 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 [?].

Parameters	Meaning	Default	Units
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	-

Table 18: Parameters of the command “FitTuneFlag”.

- 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 quadrupoles.

14.25 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 [?].

Parameters	Meaning	Default	Units
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	-

Table 19: Parameters of the command “FitTune4Flag”.

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.

14.26 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 [?].

Parameters	Meaning	Default	Unit
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	-

Table 20: Parameters of the command “FitChromFlag”.

For example:

FitChromFlag sx9 sx10 2.0 2.6

After fitting the chromaticities, 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.

14.27 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.

14.28 IBSFlag (TO BE UPDATED) ????

Intra Beam Scattering (IBS)

To calculate Intra Beam Scattering, use the command:

IBSFlag

14.29 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.

14.30 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 default values of PhaseSpaceFlag are shown in Table 14.30. If user uses PhaseSpaceFlag without parameters, then the code will use the default values.

14.31 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 Quadrupoles 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 like the following example in Table 14.31:

Parameters	Meaning	Default	Unit
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	-

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

The meanings of the above commands and the default values used to do ID compensation using quadrupoles are shown in Table 14.31.

15 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.

15.1 Systematic errors

To define the systematic multipole field error of the element, the user just need to follow the rules as below.

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_dnuv	0.1
scl_dnuv	0.1
scl_nuv	1e1
scl_nuv	1e1
ID_step	0.7

Table 22: One example to define the parameters to compensate nonlinear dynamics from the insertion device.

This command is commonly used to add the magnets design errors in the lattice.

Input format of multipole error:

keywords/name sys *r0, n, An, Bn*

The parameters of the errors are:

keywords: type of lattice elements or the family name; the keywords of the

dip	dipole
quad	quadrupole
sext	sextupole
hcorr	horizontal corrector
vcorr	vertical corrector
qt	skew quadrupole

type of lattice elements are:

sys: keyword to denote that user are setting the systematic multipole error.

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. For the component of a skew quadrupole or a vertical corrector, $Bn = 0$

Parameters	Meanings	Default	Units
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_dnu _x	Scaling weight factor of the change step of horizontal tune during the ID correction.	0.1	-
scl_dnu _y	Scaling weight factor of the change step of vertical tune during the ID correction.	0.1	-
scl_nu _x	Scaling weight factor of horizontal tune during the ID correction.	100	-
scl_nu _y	Scaling weight factor of vertical tune during the ID correction.	100	-
ID_step	0.7	-	-

Table 23: Parameters of commands to do ID compensation using quadrupoles.

An : n^{th} integrated skew component of the multipole field. For the component of a dipole or upright quadrupole, $An = 0$.

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 coefficient.

For example, to define the decupole field errors inside a horizontal bending dipole with coefficient b_1 , we define the normal scale coefficient as c_5 and skew scale coefficient as d_5 , then the b_5 component of this decupole is c_5

* b_1 and the skew coefficient 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

$$b_1 = \frac{1}{B\rho} B_0. \quad (88)$$

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 coefficients 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

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

- The line start with “#” is a comment line.
- The blank lines in the multipole definition file are neglected by the code.
- 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

```

15.2 Random 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 random multipole error
seed 1000000

#dipole
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			

16 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 systematic misalignment error file works for the lattices with full or half quadrupoles; the random misalignment error file only works for lattice with full quadrupoles.

16.1 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**

keywords: type of lattice elements or the name of the family.

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: 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  keyword  x(m)    y(m)    r (rad)
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
```

16.2 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	keyword	x(m)	y(m)	r (rad)
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

16.3 Vacuum chamber file

User can define the script to set the vacuum chamber limitation around the ring. 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".
- Lines start with # are comment.

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

```

#*****
# Script to set the vacuum chamber
#
#*****
# MK1  MK2   dxmin  dxmax  dymin   dymax
Start  All   -35e-3  35e-3  -12.5e-3 12.5e-3
#sdm1  esdm  -21e-3  21e-3  -5e-3    5e-3
debut  eh600 -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

```