SIMULINAC User's Guide
v7.0.1

W.D. Klotz, wdklotz@alceli.com

October 7, 2018

1 Getting Started

SIMULINAC is a set of pure Python3 modules to simulate proton dynamics in a LINAC lattice. It has two main modules *simu.py* and *tracker.py*. *simu.py* is an envelope code and *traker.py* is a tracking code.

All files should be installed in a root directory like `$HOME/SIMULINAC`. The Python3 executable should be on your PATH environment variable. The Python installation must have `matplotlib`, `numpy` and `pyaml` installed.

To start *simu.py* type `python simu.py`.

To start *tracker.py* type `python tracker.py`.

Both will print on your terminal and before finishing display results in several figures.

2 Preparing Input

Input to the two programs has to be provided by a text file written in `YAML` syntax. The *input*-file is generated from a *template*-file also written in `YAML`. The *template*-file contains immutable information and mutable information coded as `m4` macros. Before each program reads its *input*-file it invokes the `m4` macro processor which reads a *macro-definition*-file and replaces the macros in the *template*-file.

standard file names		
	simu.py	tracker.py
macro-definitions	yml/macros.sh	yml/macos.sh
template	ymlworktmpl.yml	yml/worktmpl.yml
input	yml/simuIN.yml	yml/trackerIN.yml

The standard files in the distribution are good examples to see how this works.

2.1 Special Input Parameters

In the current version some input parameters have to set directly in the Python code. The corresponding code lines are behind the `if __name__ == '__main__':` line. For *simu.py* these are:

```
if __name__ == '__main__':

    # launch m4 to fill macros in template file
    template_file = 'yml/worktmpl.yml'      # template file
    input_file    = 'yml/simuIN.yml'        # input file
    macros_file   = 'yml/macros.sh'         # macro definitions
```

they allow to customize the standard file names.

In *tracker.py* the code is:

```
if __name__ == '__main__':
    :
    :
    # launch m4 to fill macros in template file
    template_file = 'yml/worktmpl.yml'       # template file
    input_file    = 'yml/trackIN.yml'        # input file
    macros_file   = 'yml/macros.sh'          # macro definitions
    :
    :
    options = dict( input_file = input_file,
                    particles_per_bunch = 10000,
                    show    = True,
                    save    = False,
                    skip    = 1
```

The standard file names, the `particles_per_bunch` and the two flags `show` and `save` can be customized here. The `skip`-flag is not used in this version. The `show`-flag is used to switch fugures on/off. The `save`-flag is supposed to save figures but is not working in this version.

3 Input File Structure

In this section the structure of the standard *input*-file is discussed.

3.1 Flags

The *flags*-block is used to select different features the programs provide. Both programs have access to the flag-settings but make different use of them or ignore them. The current set of flags is given below. The `{value}`s are the default values. To change from default values the corresponding line in the input-file has to be uncommented.

```
flags:
    # - accON:        False           # {True} acceleration on/off flag
    # - egf:          True            # {False} emittance growth flag
    # - sigma:        False           # {True} beam sizes by sigma-matrix
    # - KVout:        True            # {False} print a dictionary of Key-Value pairs
    # - periodic:     True            # {False} treat lattice as ring
    # - express:      True            # {False} use express version of thin quads
    # - useaper:      True            # {False} use aperture check for lattice elements
    # - csTrak:       False           # {True} plot CS trajectories
    # - bucket:       True            # {False} plot the bucket
    # - pspace:       True            # {False} plot the twiss ellipses at entrance
    # - verbose:      2               # {0} print flag (0 = minimal print), try 0,1,2,3
```

Their use in both programs is tabeled below.

flag	simu.py	tracker.py
accON	acceleration on/off	acceleration on/off
egf	emittance growth flag	emittance growth flag
sigma	beam sizes by sigma-matrix	not used
KVout	no display, Key/Value dictionary only	not used
periodic	treat lattice as ring	meaningless
express	express version of thin quads	express version of thin quads
useaper	aperture check for lattice elements	not used
csTrack	plot CS trajectories	not used
bucket	plot the RF bucket	plot the RF bucket
pspace	plot entrance twiss ellipses	not used
verbose	verbose level 0=minimal	not used

Notes:

- egf stands for emittance growth formula. For details see the TRACE 3-D Documentation[1].
- sigma selects the way to calculate beam envelope sigmas. If True the sigma-matrix method is used to calulate beam sizes[2] else the standard formulas from twiss[3] functions are used.

[1] TRACE 3-D Documentation by K.R.Crandal, D.P.Rusthoi, 1997, Appendix F
[2] TRACE 3-D Documentation by K.R.Crandal, D.P.Rusthoi, 1997, Appendix A
[3] see Snyder & Courant therory

- KVout = True prints a long dictionary of internal parameters and supresses garaphics output.
- periodic is used for testing purposes to see if the linear matrices produce correct results.
- express replaces thick quadrupoles by thin quadrupoles with up- and downstream drift space.
- useapaer: in envelope calculation the transverse 1-sigma beam envelope is checked against aperture limitations.
- pspace plots the two transverse sphase space ellipses to check for transverse beam matching.

3.2 Sections

The *sections*-block can be used to divide the lattice into different sections. Sections are not mandatory. If commented the whole lattice is one single unnamed section. If sections are defined, each element[4] must be assigned a section.

```
sections:
- [&LE 5/30,  &HE 30/200]
```

Note:

- [&LE 5/30, &HE 30/200] defines a list of section tags. &LE defines a link that can be referenced elsewhere in the YAML-file as ∗LE. The entry behind '5/30' can be any name and defines the name of the section. In this example a shortcut for the section from 5 to 30 Mev.

3.3 parameters

The *parameters*-block defines global parameters as key-value pairs. Some prameters are defined with fixed values others by macro-variables.

```
parameters:
    - Tkin:                 _TKIN       # [MeV] energy @ entrance (injection)
    - emitx_i:      &emx    _EMITX      # [m*rad] {x,x'} emittance @ entrance
    - emity_i:      &emy    _EMITY      # [m*rad] {y,y'} emittance @ entrance
    - emitw_i:      &emw    _EMITW      # [rad] {Dphi,w} emittance @ entrance
    - betax_i:      &btx    _BETAX      # [m] twiss beta @ entrance x
    - betay_i:      &bty    _BETAY      # [m] twiss beta @ entrance y
    - phi_sync:     &phs    _PHISY      # [deg] synchronous phase
    - alfax_i:              0.          # [1] twiss alpha x @ entrance
    - alfay_i:              0.          # [1] twiss alpha y @ entrance
```

[4] see Elements below

```
- frequency:    &p01    816.e+6     # [Hz] frequency
- ql0:          &p02    0.10        # [m] quad-length
- ql:           &p03    0.05        # [m] 1/2 quad-length
- quad_bore:    &p04    0.011       # [m] quad bore radius
- windings:             30          # [1] quad-coil windings
- gap:          &p15    0.048       # [m] RF gap
- n_sigma:              10          # [m] sigma aperture
- aperture:             15.e-3      # [m] global aperture setting (default = None)
```

Notes:

- Tkin: is the key for the kinetic energy at injection in [Mev]. Its value is the macro-name _TKIN.
- emitx_i: is the transvese emmittance in x-plane in $[m * rad]$. Its value is the macro-name _EMITX and &emx its link-id.
- aperture has a fixed value of $15. * 10^{-3}$

3.4 elements

The *elements*-block defines the elements (a.k.a nodes) in the lattice.

```
elements:
# HE
    - D3:        &D3             # ID:&link
        - type:     D            # type:class
        - length:   0.08         # [m]
        - sec:      *HE          # section
    - D5:    &D5
        - type:     D
        - length:   0.022
        - sec:      *HE
    - QFH:   &QFH                # ID:&link
        - type:     QF           # type:class
        - length:   *p03         # [m]
        - aperture: *p04         # [m] quad bore
        - B':       &Bgrad   30. # [T/m] gradient
        - slices:   0            # slices
        - sec:      *HE          # section
    - QDH:   &QDH
        - type:     QD
        - length:   *p02
        - aperture: *p04
        - B':       *Bgrad
        - slices:   0
        - sec:      *HE
```

```
- RFGH:  &RFGH               # ID:&link
    - type:     RFG          # type:class
    - EzAvg:    1.00         # [MV/m] average E-field
    - EzPeak:   1.40         # [MV/m] peak E-field
    - PhiSync:  *phs         # [deg] synchronous phase
    - fRF:      *p01         # [Hz] frequency
    - gap:      *p15         # [m] length
    - aperture: *p04         # [m] quad bore
    - aperture: 10.e-3       # [m] quad bore
    - SFdata:   SF_WDK2g44.TBL # superfish tbl-data file
    - mapping:   t3d         # Trace 3D linear map model
    - mapping:   simple      # Shishlo/Holmes linear map model
    - mapping:   base        # Shishlo/Holmes base map model
    # - mapping:   ttf       # Shishlo/Holmes three point TTF RF gap-model
    # - mapping:   dyn       # Tanke/Valero RF gap-model
    :
    :
```

Notes:

- ```
  - D3:        &D3            # the node-ID is D3, link-id is &D3
      - type:     D           # dipole node
      - length:   0.08        # [m]
      - sec:      *HE         # section
  ```

  type:D defines the node as an object of class 'D', which is a dipole. length:0.08 defines its length in [m] and sec:∗HE says that this dipole belongs to section 'HE' by reference.

- ```
  - QFH:   &QFH               # ID:&link
      - type:     QF          # type:class
      - length:   *p03        # [m]
      - aperture: *p04        # [m] quad bore
      - B':       &Bgrad   30. # [T/m] gradient
      - slices:   0           # slices
      - sec:      *HE         # section
  ```

  The type:QF defines an x-focussing quadrupole. It has attributes length, aperture, B', slices and belongs to a section. ∗p03 and ∗p04 are references to links. B' has the value of 30. $[T/m]$ and defines a link-id &Bgrad. The slices attribute defines the number of slices a thick node is cut into. slices = 0 or 1 means don't slice the thick node. slices = n means means cut it into n slices. The sec attribute references the HE section.

- ```
  - RFGH:  &RFGH              # ID:&link
      - type:     RFG         # type:class
      - EzAvg:    1.00        # [MV/m] average E-field on axis
      - EzPeak:   1.40        # [MV/m] (EzAvg = 1.00[MV/m])
      - PhiSync:  *phs        # [deg] synchronous phase
  ```
  ```

```
    - fRF:      *p01         # [Hz] frequency
    - gap:      *p15         # [m] length
    - aperture: 10.e-3       # [m]
    - SFdata:   SF_WDK2g44.TBL # superfish tbl-data file
    - mapping:   t3d         # Trace 3D linear map model
    - mapping:   simple      # Shishlo/Holmes linear map model
    - mapping:   base        # Shishlo/Holmes base map model
    # - mapping:   ttf       # Shishlo/Holmes three point TTF RF gap-model
    # - mapping:   dyn       # Tanke/Valero RF gap-model
    - sec:      *HE          # section
```

The type:RFC defines a radio frequency cavity. It is modeled as a kick of zero length. EzAvg is the average E-field on the axis E(r=0,z). PhiSync is the synchronous phase in [deg]. It takes its value from the link reference ∗phs. fRF is the rf frequency in [Hz] given here from the link reference ∗p15. aperture is the cavity bore radius. SFdata is the file name of a table for field profile data from 'SuperFish'. The mapping attribute specifies which cavity model to use. The cavity node is part of section HE - sec attribute.

3.5 segments

```
segments:
# LE            # empty section
# HE            # high energy section
    - SEG1H:
        - *QFH
        - *D3
    - SEG2H:
        - *D3
        - *QDH
        - *D3
    - SEG3H:
        - *D3
        - *QFH
    - RFGH:
        - *D5
        - *RFGH
        - *D5
        #
        - *D5
        - *RFGH
        - *D5
        #
        - *D5
```

```
- *RFGH
- *D5
#
- *D5
- *RFGH
- *D5
#
- *D5
- *RFGH
- *D5
#
- *D5
- *RFGH
- *D5
#
- *D5
- *RFGH
- *D5
#
- *D5
- *RFGH
- *D5
#
- *D5
- *RFGH
- *D5
#
# - *D5     # 10th cavity makes it unstable!!
# - *RFGH
# - *D5
#
```