OCELOT Documentation

Release 16.7

IA

July 20, 2016

CHAPTER

ONE

OVERVIEW

OCELOT is a framework

cpbd module

adaptors

Installation notes

CHAPTER

TWO

CONTENTS:

2.1 Charged Particle Beam Dynamics (CPBD) module

2.1.1 Overview

Charged Particle Beam Dynamics module provides features for charged particle (electron) beam optics, including calculating and matching Twiss parameters, single-particle tracking as well as tracking with collective effects (CSR, space charge and wakefields)

2.1.2 Getting started

Import OCELOT

```
from ocelot import *
```

Define a magnetic lattice

```
q1 = Quadrupole(l = 0.3, k1 = 5)
q2 = Quadrupole(l = 0.3, k1 = -5)
d = Drift(l = 0.5)
lat = MagneticLattice( (d, q1, d, q2, d, q1, d,q2,d) )
```

Use `twiss()` to find linear optics (Twiss) functions for given initial values

```
tw0 = Twiss()
tw0.beta_x = 5.
tw0.alpha_x = -0.87
tw0.beta_y = 2.1
tw0.alpha_y = 0.96
tws = twiss(lat, tw0)
```

Find periodic Twiss solution

```
tws = twiss(lat)
```

Find periodic Twiss solution with given longitudinal resolution (500 points)

```
tws = twiss(lat, nPoints=500)
```

Plot Twiss parameters

```
from pylab import *
plot([t.s for t in tws], [t.beta_x for t in tws])
plot([t.s for t in tws], [t.beta_y for t in tws])
```

Plot Twiss parameters in the lattice display

```
from ocelot.gui.accelerator import *
plot_opt_func(lat, tws)
show()
```

2.1.3 Linear optics functions

twiss (*lat*[, *nPoints=None*])

2.1.4 Matching

match (*lattice*, *constarints*, *variables*[, *start=0*])
lattice a MagneticLattice object

2.1.5 Tracking

2.1.6 Elements

class MagneticLattice

class Drift

class Quadrupole

class Bend
same as SBend

class SBend

class RBend

2.1.7 Transfer maps

Transfer maps define how the element map acts in tracking. The default transfer map attachment scheme is as follows:

- Drifts, Quadrupoles, and bends have first order transfer maps
- Sextupoles have a drift-kick-drift map

2.1.8 API documentation

ocelot.cpbd.optics.**twiss** (*lattice*, *tws0=None*, *nPoints=None*)

ocelot.cpbd.match.**match** (*lat*, *constr*, *vars*, *tw*, *verbose=True*, *max_iter=1000*, *method='simplex'*, *weights=<function weights_default at 0x2aedb4f16668>*, *vary_bend_angle=False*, *min_i5=False*)
matching stuff

2.2 Synchrotron radiation (rad) module

2.2.1 Overview

Synchrotron radiation from undulators and bending magnets

2.3 Photon optics (optics) module

2.3.1 Overview

photon optics

2.4 Machine interface (mint) module

2.4.1 Overview

Machine interface module

2.5 Adaptors (adaptors) module

2.5.1 Overview

CHAPTER

THREE

INDICES AND TABLES

- *genindex*
- *modindex*
- *search*

B

D

M

Q

R

S

T