P12 Manual

Contents

1

Introduction

P12 is a fast calculating, analytical program for lattice design, it is a non-tracking program. It assumes, that the particles, i.e. electrons, are relativistic ($E \gg E_0$). Since not only properties of magnet elements are important for lattice design but also the properties of particle beams, special features, e.g. subroutines for space charge effects, are implementated in P12. The original idea of P12 was a fast calculating algorithm for precalculations of FEL (Genesis1.3) lattices including transversal space charge defocusing. This requires undulator focusing as well. By this way several different types of undulator modells are implementated in P12.
At present the effects of undulator focusing, space charge effects etc. are calculated in linear optics. Although there are many programs for lattice design, most of them don't have any optical userinterface, whereby the first steps as well as further applications become more complex. The user interface design of P12 should make the use of P12 more comfortable. It is explained in detail in the following pages.

2

First Steps

In this chapter the basics of the application of P12 are described.
The topics are

1. **Bunch properties**
   Which parameter can be taken into account in P12?
   How to defined them in P12?
   How to convert from other programs?
   Layout of I/O Files(bunch)

2. **Lattice**
   Which elements are implementated (at present)?
   How to create and test a lattice?
   How to convert lattice files from other programs?
   Layout of I/O Files(lattice)

3. **Space Charge Calculation**
   Short introduction of the calculation scheme
   Limits produced by the approximation and the assumptions.

2.1 MainTitle

The program P12 usually starts with the main title, shown in figure 2.1.

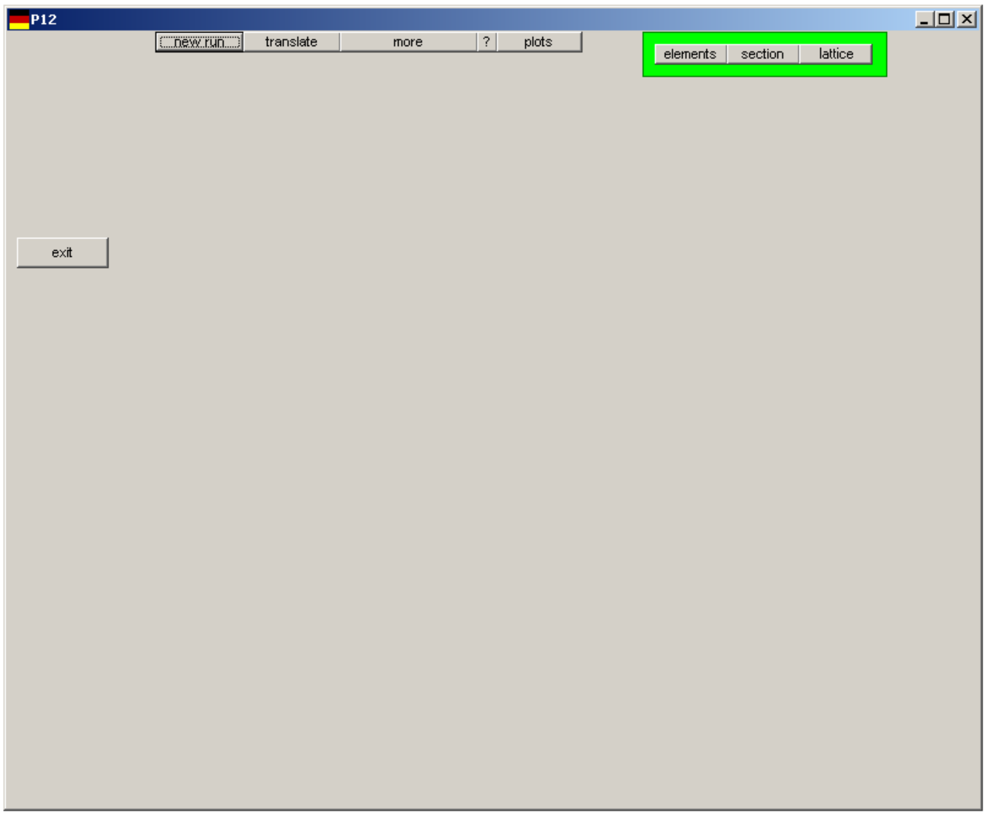


Figure 2.1: MainTitle of P12
This screen shoot shows the main title of the program P12.

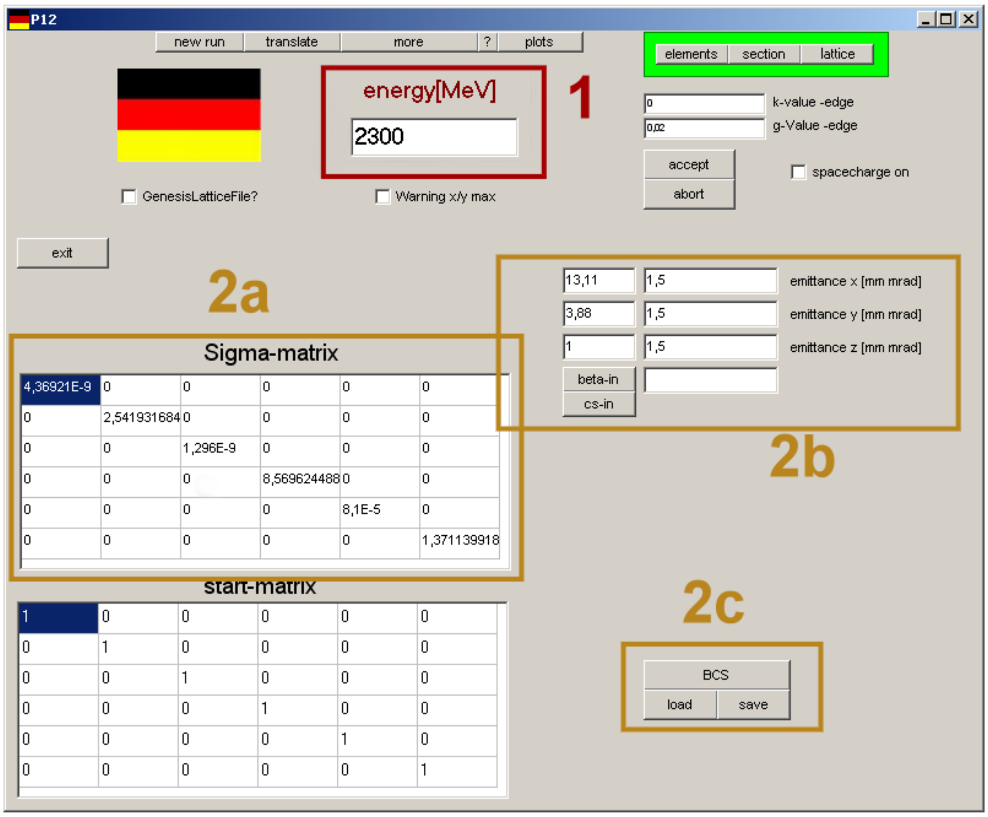
There are nine buttons shown. Starting left-hand-side:

- ***exit:*** This button closes the program.
- ***new run:*** The program P12 reboots.
- ***translate:*** This button opens an additional user interface, which is equiped with a conversion algorhythm for bunch and lattice properties.
- ***more:*** With the button "more" the setup of bunch properties and space charge calculation opens.
- ***?:*** Some informations of P12 will be shown. (Koordinate system, properties of beam optic elements, etc.)

- ***plots:*** A plot user interface opens, where the user can observe the development properties of the bunch and the lattice.
- ***elements/section/lattice:*** These three buttons opens three different lattice-creation user-interfaces.(chapter 2.3)

2.2 Bunch Properties

In P12, two sets of bunch properties are required, one for the reference particle and one for the particle ensemble.



energy[MeV] 2300

□ GenesisLatticeFile?

□ Warning x/y max

0 k-value -edge

0,02 g-Value -edge

□ spacecharge on

13,11	1,5	emittance x [mm mrad]
3,88	1,5	emittance y [mm mrad]
1	1,5	emittance z [mm mrad]

Sigma-matrix

4,36921E-9	0	0	0	0	0
0	2,541931684	0	0	0	0
0	0	1,296E-9	0	0	0
0	0	0	8,569624488	0	0
0	0	0	0	8,1E-5	0
0	0	0	0	0	1,371139918

start-matrix

1	0	0	0	0	0
0	1	0	0	0	0
0	0	1	0	0	0
0	0	0	1	0	0
0	0	0	0	1	0
0	0	0	0	0	1

Figure 2.2: BunchTitle of P12 - "more"
Here is the setup for bunch properties, space charge calculations etc. .

1 - First part is set by the energy(E) of the reference particle,

$$E = \sqrt{p^2c^2 + m_0^2c^4}.$$

The energy of the reference particle is important, because at least one of the transfermatrix elements depends on $\gamma = \frac{E}{m_0c^2}$.

***2* -** The second part describes properties of the particle ensemble refering to the reference particle. The ensemble is characterised by the "Sigma[σ]-matrix", figure 2.2. The values of the Sigma-matrix describe the RMS-values of the particle distribution in SI-unit system, e.g. $\sigma_{11} = \sigma_x^2 = 4.37 \cdot 10^{-9} m^2 \ \rightarrow \sigma_x = 66.1 \mu m$. For creating the Sigma-matrix, 2.2 -2b can be used. A hide algorithm creates a Sigma-matrix without correlation elements with the input-data of emittance and either the RMS-values (button 'cs-in') of the distribution or the beta-function value (button'beta-in'). Additional conversion algorithm are given in 'translate', chapter 3.
For check, save and load of the entries for the particle distribution 2.2 -2c is implementated. "BCS", bunch cross section, calculates the bunch cross section from the Sigma-matrix, whereby any correlation between the planes are ignored. The values of the projected bunch cross section as well as the square of emittance will be presented.

! emittance$^2 > 0$, otherwise non-physical character & program - error!

With "save" and "load" the current values can be saved into a file or replaced by values from a bunch file. In the following subsection the layout of the bunch files will be explained.

2.2.1 Bunch - File

The bunch file is coded in (MS-)ASCII. From the bunch file only values marked by keywords, "energy" and "Sigma-Matrix", are loaded, so it offers the possibility to add specific information about file background. Please, avoid using these words for these informations. (If necessary they can be localed at the very first position of the line. At this position they will be ignored.) Every line of input data starts and ends with a single space bar. The value of the reference particle energy is separated from the keyword 'energy' by a single space bar as well. For the Sigma-matrix the keyword 'Sigma-matrix' implies, that the following 6 lines are reservated for the $\sigma - matrix$. These 6 lines start and end with a single space bar. The values of the Sigma-matrix are also seperated by a single space bar. Here is a short example of a bunch file, where the symbol '_' marks a single space bar.

```
_energy:_2300_MeV_
_Sigma-Matrix_
_4,37E-9_0_0_0_0_0_
_0_2,54E-11_0_0_0_0_
_0_0_1,29E-9_0_0_0_
_0_0_0_8,57E-11_0_0_
_0_0_0_0_8,1E-5_0_
_0_0_0_0_0_1,37E-15_
```

2.3 Lattice

This second part focuses on the beam optic elements and the creation of lattices.

2.3.1 Beam Optic Elements in P12

In P12 the following elements are implemenated (at present):

- ***drift***,
- ***bending magnets***,
- ***kick / edge***,
- ***quadrupole***,
- ***undulator***,

which will be discussed in the following.

Drift

The drift is characterised by two parameter, drift length($lsdr$) and the relativistic gamma(γ). The corresponding transfermatrix is

$$M = \begin{pmatrix} 1 & lsdr & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & lsdr & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & \frac{lsdr}{\gamma^2} \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

Kick - Edge

Kicks and Edges are similar to thin quadrupoles. They are used to simulate the focusing effects of edge focusing etc.. Since these elements have an infinitesimal longitudinal dimension, their effect can be described by:

$$Edge = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ \frac{1}{f} & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{f} & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix},$$

where f corresponds to the focus of the thin quadrupole. In case of edges of bending magnets $\frac{1}{f} = \frac{tan(\phi)}{R}$, where R is the curvature radius an ϕ stands for the entrance angle(α_{in}) or the angle of emergence(α_{out}).

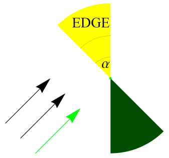


Figure 2.3: Edge

Bending Magnets

Bending magnets, alias bends or dipoles, are defined by five parameter, path in the bend (s), curvature radius (R), entrance angle (α_{in}), angle of emergence (α_{out}) as well as the relativistic gamma (γ). The path in the bend and the curvature radius result from the physical properties of the bending magnet; magnetic field strength and length of the magnet. If the entrance and the exit angle are sero, i.e. $\alpha_{in} = 0$ and $\alpha_{out} = 0$, the bending magnet is called sector bend, SBend. The corresponding transfermatrix is

$$SB = \begin{pmatrix} cos(\frac{s}{R}) & Rsin(\frac{s}{R}) & 0 & 0 & 0 & R(1-cos(\frac{s}{R})) \\ \frac{-1}{R}sin(\frac{s}{R}) & cos(\frac{s}{R}) & 0 & 0 & 0 & sin(\frac{s}{R}) \\ 0 & 0 & 1 & s & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ -sin(\frac{s}{R}) & -R(1-cos(\frac{s}{R})) & 0 & 0 & 1 & -\frac{s}{\gamma^2}+(Rsin(\frac{s}{R})-s) \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

An additional focusing effect appears, if at least one of the angle does not vanish. The effects of edge focusing is already described, see chapter 2.3.1. A special kind of Rbend is also implementated in P12. The angles α_{in} and α_{out} are equivalent to the half of the bending angle.

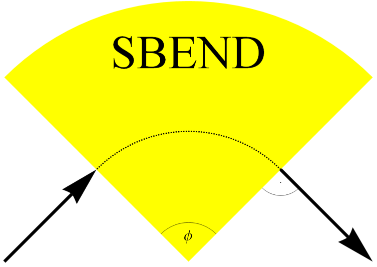


Figure 2.4: Sbend

P12 Standard Sbend configuration

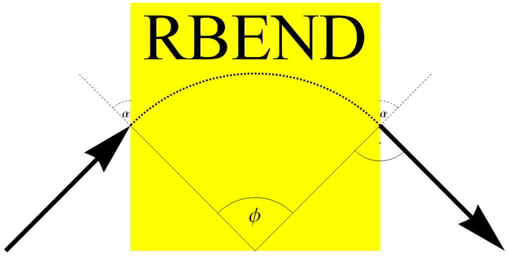


Figure 2.5: Rbend

P12 Standard Rbend configuration

With the combined application of Sbend and Edge any customary Rbend can be simulated. Therefore the bending direction has to be variable and the Edges have to be adopted to the corresponding Sbends. This progress has to be made by a lattice file. An example is given at the end of this chapter.

Quadrupole

Quadrupoles are characterised by their field gradients (k) and their lengths (lqF). In P12 the quadrupoles focus in x-plane, QF, or in y-plane, QD. In the following the transfermatrix of a QF is presented. The effect of a QD is describe in a similar way, by changing the planes of focusing and defocusing.

$$QF = \begin{pmatrix} cos(\phi_x) & \frac{1}{\sqrt{|k_x|}} sin(\phi_x) & 0 & 0 & 0 & 0 \\ -\sqrt{|k_x|} sin(\phi_x) & cos(\phi_x) & 0 & 0 & 0 & 0 \\ 0 & 0 & cosh(\phi_y) & \frac{1}{\sqrt{|k_y|}} sinh(\phi_y) & 0 & 0 \\ 0 & 0 & \sqrt{|k_y|} sinh(\phi_y) & cosh(\phi_y) & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & \frac{lqF}{\gamma^2} \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix},$$

$\phi_i = \sqrt{|k_i|} lqF$ stands for the focusing phase.

Undulator

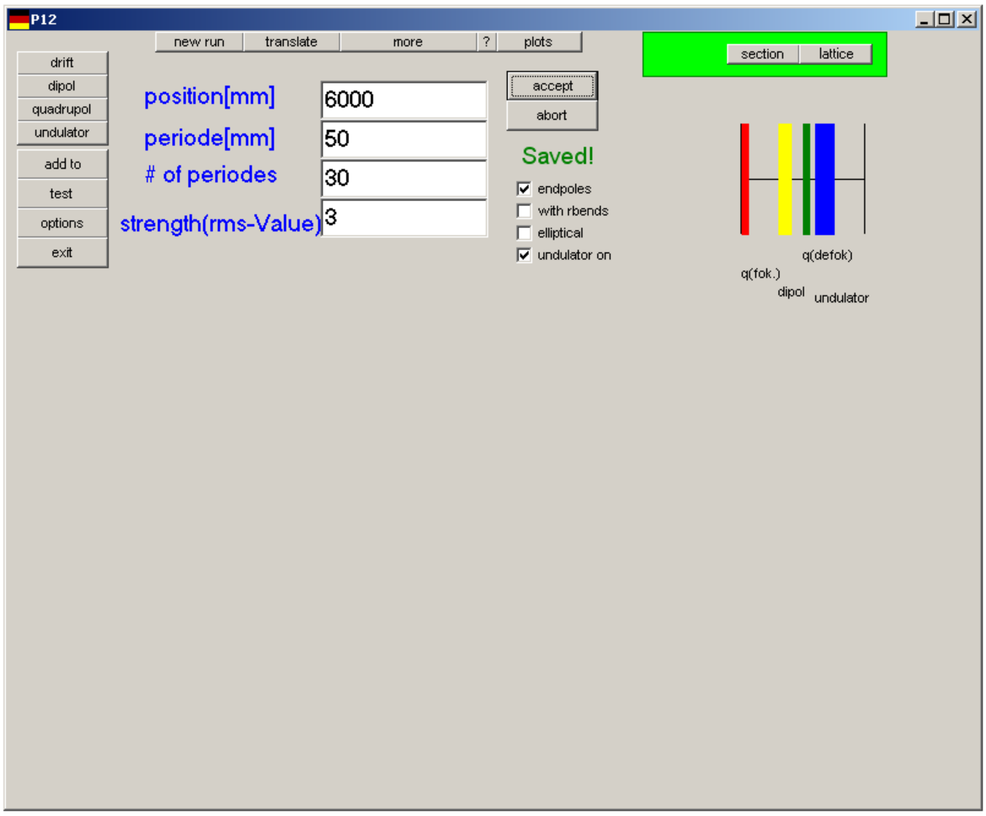
In P12 there are many different approaches for undulators eligible. Undulators are sequences of bending magnets, whereby focusing effects as well as dispersive effects occurs. A model in P12 is the Genesis undulator, where only focusing effects are taken into account. Another model is the simulation of an planar undulator as a sequence of alternating Sbends and Edges. For elliptical undulator a similar Sbend-approxamiation is eligible in P12.

2.3.2 Lattice Designing

In P12 a lattice can be designed by adding single elements, precast sections, lattices files and the combination of these subroutines. The corresponding user interfaces can be activated by the buttons 'elements', 'section' and 'lattice'.

elements

In the subroutine 'elements' single beam optic elements can be combinated to sections. The current section is ploted, while input data will be added or corrected.



position[mm] 6000

periode[mm] 50

# of periodes 30

strength(rms-Value) 3

Saved!

☑ endpoles
□ with rbends
□ elliptical
☑ undulator on

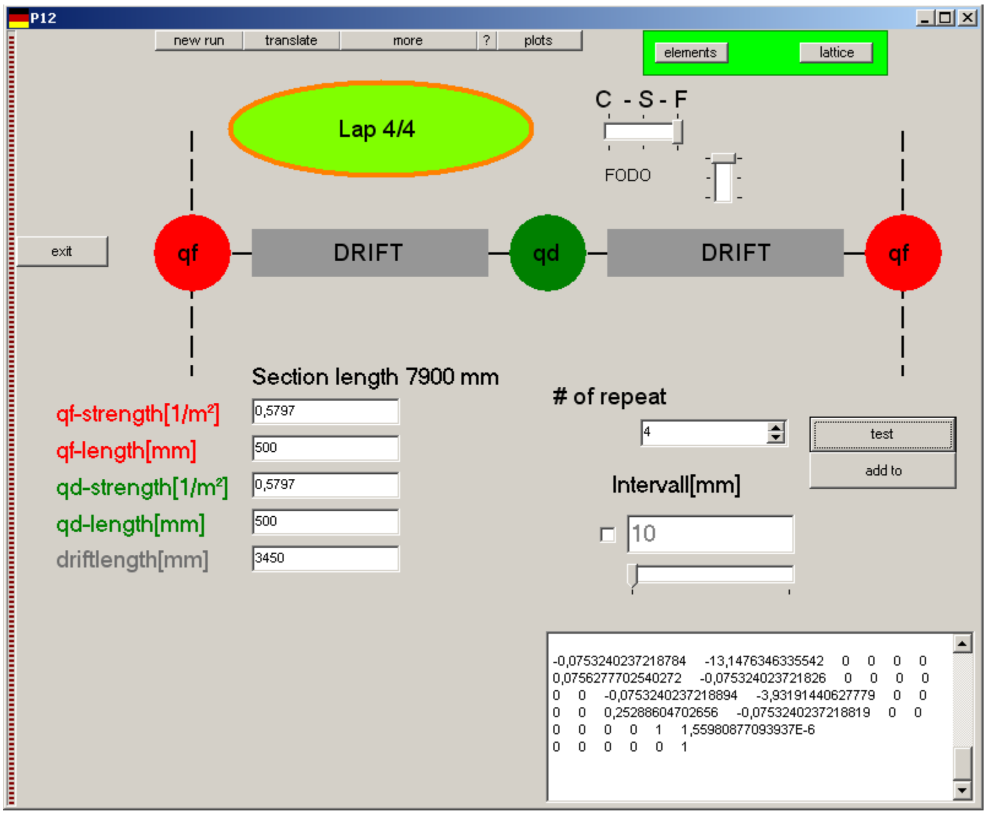
Figure 2.6: User Interface of 'Elements'

Layout of the user interface of the subroutine 'Elements'

In this subroutine as well as in the other lattice design subroutines the lastest section can be tested before plugging in. This can be done with the button 'test'. The 'old' sections as well as the 'new' section is observable by using the implimentated plot-function. If the section is correct, it can be added by using button 'add to'.

section

In this subroutine precasted section, like fodo-cells etc., can be used. The current section layout is plotted.



Section length 7900 mm

qf-strength[1/m²] 0,5797
qf-length[mm] 500
qd-strength[1/m²] 0,5797
qd-length[mm] 500
driftlength[mm] 3450

# of repeat 4

Intervall[mm] □ 10

```
-0,0753240237218784   -13,1476346335542   0   0   0   0
0,0756277702540272   -0,075324023721826   0   0   0   0
0   0   -0,0753240237218894   -3,93191440627779   0   0
0   0   0,25288604702656   -0,0753240237218819   0   0
0   0   0   0   1   1,55980877093937E-6
0   0   0   0   0   1
```

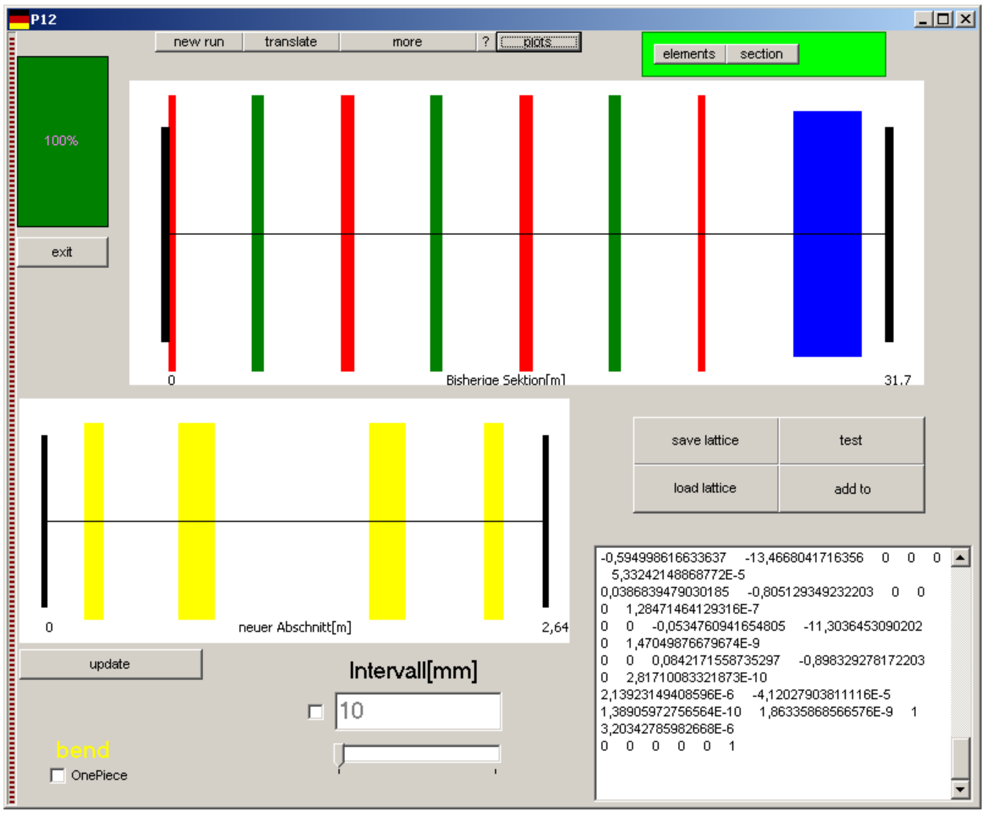
Figure 2.7: User Interface of 'Section'

Layout of the user interface of the subroutine 'Section'

For calculations with the impact of space charge effects the variation of the calculation intervalls can be useful. In the memo of the user interface the transfer matrix as well as additional informations about the choosen section and the calculation progress is given.

lattice

The subroutine 'Lattice' requires a lattice file, whose exact layout will be discussed in chapter 2.3.3. In this user interface the current, fixed lattice, picture above, as well as the *new* section, defined by the lattice file, is plotted. The four button on the right hand side of the interface enable the user to save the current, fixed lattice as a lattice file, to load another lattice file, to test the *new* section and the add the *new* section to the *old* one. In the memo of the user interface the transfer matrix, additional informations about the choosen section and the calculation progress is given as well.



```
-0,594998616633637   -13,4668041716356   0   0   0
  5,33242148868772E-5
0,0386839479030185   -0,805129349232203   0   0
0   1,28471464129316E-7
0   0   -0,0534760941654805   -11,3036453090202
0   1,47049876679674E-9
0   0   0,0842171558735297   -0,898329278172203
0   2,81710083321873E-10
2,13923149408596E-6   -4,12027903811116E-5
1,38905972756564E-10   1,86335868566576E-9   1
3,20342785982668E-6
0   0   0   0   0   1
```

Figure 2.8: User Interface of 'Lattice'

Layout of the user interface of the subroutine 'Lattice'

2.3.3 Lattice - files

The lattice files are codes in MS-ASCII. In the following the codes of the elements will be explained by examples. For marking space bars the symbol ’_’ is used.

drift

With the help of the code of a drift the end of a section can be defined. It is
”_MD_EndPosition_” ,
whereby the endposition has to be defined in meter.

Examples:	
1. Drift to 2.78m	_MD_2,78_
2. Drift to 8.7m	_MD_8,7_

bends

The bends in P12 are defined by physical properties of the bending magnet, e.g. magnetic field strength and physical length. This is also reflected in the codes. **Sbend** is
”_SB_StartPosition[m]_FieldStrength[T]_Length[m]_T1_T2_”. The starting position and the physical length are defined in meter and the field strength in Tesla. The bending direction is characterised by the variables $T1$ and $T2$. Their values are f or t, which stands for false and true. Due to $T1$ the bending direction in one plane can be defined, i.e. whether the bend bends left or right hand side and $T2$ sets whether the bending plane is horizontal or vertical.

Examples:	
1. SBend starting at 0,2 m with 34,97mT and a length of 0,3m to 2.78m and bends in vertical direction.	_SB_0,2_0,03497_0,3_f_t_
2. This Sbend acts in horizontal direction.	_SB_0,2_0,03497_0,3_f_f_

In P12 there is a special kind of **Rbend** defined as well. Its properties are already discussed above, chapter 2.3.1. The corresponding code is
”_RB_StartPosition[m]_FieldStrength[T]_Length[m]_T1_T2_”. It is very similar to the code of Sbends.

quadrupoles

Quadrupoles are defined by their focusing strength and their physical length. In P12 there are two different kind of quadrupoles implimentated. One of them focuses in x-direction and defocuses in y-direction and the other one vice versa. The codes are almost identical. For the first quadrupole(focus in x-direction) the

code is
”_QF_StartPosition[m]_FocusingStrength[$\frac{1}{m^2}$]_Length[m]_”.
The code for the other quadrupole is
”_QD_StartPosition[m]_FocusingStrength[$\frac{1}{m^2}$]_Length[m]_”.

Examples:
1. Quadrupole, focus in x-direction, starts at 0.4m,focusing strength 0.58$\frac{1}{m^2}$, length 0.5m .
2. Quadrupole, defocus in x-direction, starts at 0m,focusing strength 0.7$\frac{1}{m^2}$, length 0.3m .

```
_QF_0,4_0,58_0,5_

_QD_0_0,7_0,3_
```

undualtor

In HGHG-FELs, like FLASH II, different types of undulators can be used for seeding and radiation. E.g. one of the schemes of the FLASH II layout contains a combination of three planar undulators and one helical undulator, whereby the helical undulator is used as the final radiator.

a) planar undulator
Two different approaches for planar undulators are implemented in the code P12. In the first model the planar undulators are approximated by a sequence of dipoles, i.e. cobination of Sbends and Edges. Due to the closed dispersion in linear optics the undulators characteristics are restricted to the focusing properties. This leads to a quadrupole similar approach. The lattice code for **planar undulators** is:
”_UP_StartPosition[m]_rmsUndulatorStrength[-]_ UndulatorPeriodeLength[m]_NumbersOfPeriod[-]_Tapproach_Tendpole_”.
”Tapproach” stands for type of approach. For dipole approach *Tapproach* = *d* , **d**ipole approach and for the other approach *Tapproach* = *q* , **q**uadrupole approach. Real undulators are equiped with endpoles, whose effect is to center the oscillating motion of the electrons in the undulator. In the dipole approach this can be estimated as well. If *Tendpole* = *t*, then the endpole are also taken into account, whereby at two periods of the undulator-field strength-configuration becomes changed, ($\frac{1}{4}, \frac{3}{4}$) configuration. The total length of the undulator maintains constant, but the core section decreses by 2 periodes, which are used for the end poles.

Examples:
1. Undulator starts at 0m with K[rms]=3 and $\lambda_u = 5cm$ over 20 periods in dipole approach with endoles.
2. Undulator starts at 4m with K[rms]=2.47 and $\lambda_u = 6.2cm$ over 30 periods in quadrupole approach. No Endpoles.

```
_UP_0_3_0,05_20_d_t_

_UP_4_2,47_0,062_30_q_f_
```

b) planar undulator

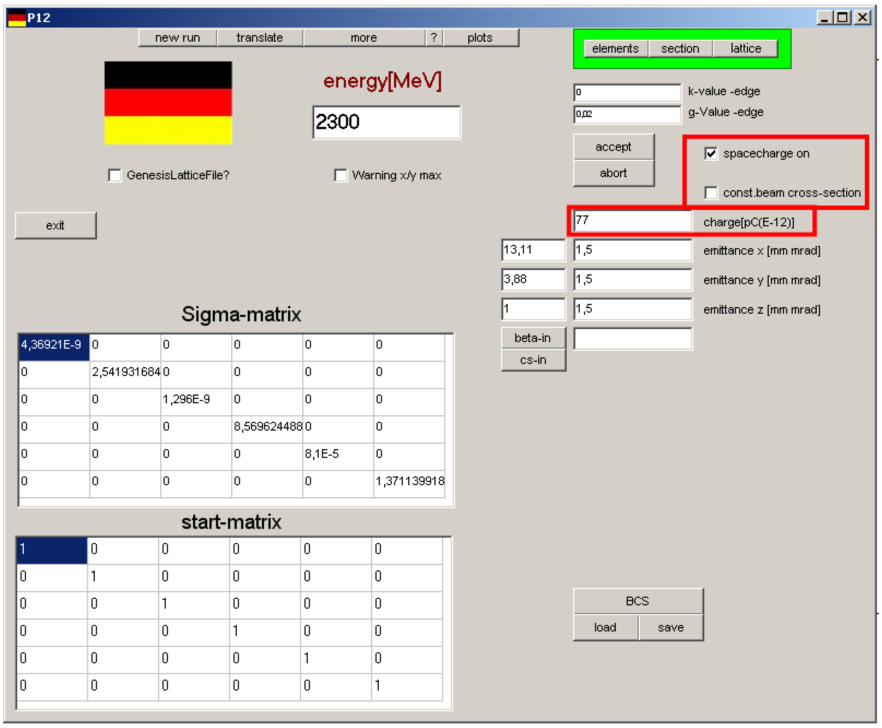
The effects of elliptical undulators are hard to simulate without tracking. In P12 the inner part of the undulator is simulated with a very good agreement with established tracking programs. It is simulated by a superposition of two sbends bending in separated planes.
The lattice code for the **elliptical undulator model** is:
”_UE_StartPosition[m]_rmsUndulatorField$_1$[t]_rmsUndulatorField$_2$[T]_UndulatorPeriodeLength[m]_NumbersOfPeriod[-]_Tendpole_”.

Examples:
Undulator(ell.) starts at 0m with $B_1 = 1.01T$, $B_2 = 0.71T$ and $\lambda_u = 29mm$ over 70 periods without endoles.

_UE_0_1,01_0,71_0,029_70_f_

2.4 Space Charge Calculation

In the user interface *”more”* the space charge options are selectable. For the space charge calculation the bunch dimension as well as the bunch charge is



energy[MeV] 2300

0 k-value -edge
0,02 g-Value -edge

□ GenesisLatticeFile?
□ Warning x/y max

☑ spacecharge on
□ const.beam cross-section

77 charge[pC(E-12)]
13,11 1,5 emittance x [mm mrad]
3,88 1,5 emittance y [mm mrad]
1 1,5 emittance z [mm mrad]

Sigma-matrix

4,36921E-9	0	0	0	0	0
0	2,541931684	0	0	0	0
0	0	1,296E-9	0	0	0
0	0	0	8,569624488	0	0
0	0	0	0	8,1E-5	0
0	0	0	0	0	1,371139918

start-matrix

1	0	0	0	0	0
0	1	0	0	0	0
0	0	1	0	0	0
0	0	0	1	0	0
0	0	0	0	1	0
0	0	0	0	0	1

Figure 2.9: User Interface of ’more’

Layout of the user interface of the subroutine ’more’ with space charge characteristics.

required. The dimensions of the bunch are determinated by the σ-matrix. The bunch charge has to be defined by the user. The bunch can also be estimated as a continuous beam, whereby the input data consists of an average current and the bunch cross section. In this case the longitudinal dimension is ignored in the calculation routine.

2.5 Output/Input files

The file format of all input and output file is ASCII.

Output - Bunch Properties

The bunch properties are defined by the σ-matrix. While the bunch propagates through lattices the σ-matrix changes. The progress is written down into two files, *p.001* and *BETAEnd.dat*. In these files the σ-matrix at positions along the lattice is noted. The file *p.001* is used for the *test*-routine and *BETAEnd.dat* for the *add to*-routine.
Path: \applicationPath \out\[..]

Output - Transfer Matrix

The transfermatrix as well as its corresponding position is written down into *t.001* and *TransEnd.dat*, where the file *t.001* is used for the *test*-routine and *TransEnd.dat* for the *add to*-routine.
Path: \applicationPath \out\[..]

Output - Lattice Design

For the defination of the lattice design the lattice conversion, chapter 2.3.3, is used. The lattice design of the *add to*-routine is save in *design.txt* at
Path: \applicationPath \out\[..].
The lattice design of the *test*-routine is save in *designt.txt* at
Path: \applicationPath \Temp\[..].

Starting Picture

A starting background picture of P12 can set.
The picture file \applicationPath \bmp\Start.bmp.

Input Files

Input file of P12 are the bunch-input file, chapter 2.2, and the lattice file, chapter 2.3.3, which are already discussed above.

3
Additional Features

3.1 new elements

Maybe sometimes it is necessary to use elements or combined elements, which is **not** implimentated in P12. Their specific behaviour can be taken into account for lattice design by using the 'START-Matrix'-Subroutine. If the transfermatrix of the unimplementated element is known, this matrix can be entered to P12. At each run of the subroutine 'elements', chapter 2.3.2, the corresponding transfermatrix is taken into account as the very first element of the section.

3.2 Plot Function

For fast investigations the program P12 is equiped with a plot function, where the development of major bunch properties as well as properties of the transfermatrix can be observed. The created plot is saved as *application path* \out \Plot.bmp.

3.3 Warning Function

The task of the *Warning Function*-Subroutine is the detection of a passing-over of a limit of the bunch cross section. The user can define limits of a maximum bunch extension for both transversal dimension. A warning is given is at least one of the limits is passed over.

3.4 Genesis1.3 Lattice

The program P12 is also developed for Genesis1.3 lattice design, whereby a subroutine for the creating lattice files is required. Since the lattice design of Genesis1.3 is scaled on the unit length corresponding on the undulator period, the subroutine will adopte the lattice file. The lattice is save as
\application path \out \Genesis13.lat.
The file *Genesis.lat* contains the identical information, but it is coded in MS-ASCII instead of Unix-ASCII.