

# LCODE user manual

K.V. Lotov, A.P. Sosedkin

November 1, 2013

branches/stable r386

## Contents

<b>1</b>	<b>Overview</b>	<b>2</b>
<b>2</b>	<b>Notation and units of measure</b>	<b>2</b>
<b>3</b>	<b>Underlying physics</b>	<b>5</b>
3.1	Kinetic plasma model . . . . .	5
3.2	Fluid plasma model . . . . .	5
3.3	Beam model . . . . .	5
3.3.1	Low-noise beam . . . . .	5
3.4	Energy fluxes and energy densities . . . . .	5
3.5	Laser pulse . . . . .	5
<b>4</b>	<b>Running the code</b>	<b>5</b>
<b>5</b>	<b>Configuration file</b>	<b>6</b>
5.1	Simulation area . . . . .	7
5.2	Particle beams . . . . .	7
5.2.1	Newly generated beams . . . . .	8
5.3	Plasma . . . . .	8
5.3.1	Options specific to particle plasma models . . . . .	9
5.3.2	Option specific to the fluid plasma model . . . . .	10
5.4	Every-time-step diagnostics . . . . .	10
5.5	Periodical diagnostics . . . . .	11
5.5.1	Colored maps . . . . .	11
5.5.2	Functions of $\xi$ . . . . .	13
5.5.3	Beam particle information as pictures . . . . .	14
5.5.4	Beam information as histograms . . . . .	15
5.5.5	Trajectories of plasma particles . . . . .	15
5.5.6	Detailed (substepped) plasma response . . . . .	16
5.6	Saving run state . . . . .	16
5.7	Logging preferences . . . . .	17
5.8	UNTESTED parameters . . . . .	17
<b>6</b>	<b>Initial beam shape</b>	<b>17</b>
6.1	Segment parameters . . . . .	18
6.2	Example of beam profile . . . . .	20
<b>7</b>	<b>Format of other input and output files</b>	<b>20</b>
7.1	beamfile.bin . . . . .	20
7.2	beamfile.bit . . . . .	20
7.3	plasma.bin . . . . .	20
7.4	fields.bin . . . . .	20
7.5	elocf.dat, glocf.dat, emaxf.dat, gmaxf.dat . . . . .	21
7.6	hystog.dat . . . . .	21
7.7	n1.dat . . . . .	21
7.8	tt*.dat . . . . .	21
7.9	r*****.det . . . . .	21
7.10	u*****.det, v*****.det, u*****.swp, v*****.swp . . . . .	22
7.11	xi*****.det . . . . .	22

7.12	d?*****.swp . . . . .	22
7.13	??*****.swp, ??*****w.swp, ??*****.det . . . . .	22
7.14	captured.pls . . . . .	22
7.15	*****.pls, s*****.pls . . . . .	22
7.16	fl*****.pls, pl*****.swp, tb*****.swp . . . . .	22
7.17	partic.swp . . . . .	22
7.18	Temporary files . . . . .	22

## 1 Overview

LCODE is a freely-distributed code for simulations of particle beam-driven plasma wakefield acceleration. The code is 2-dimensional (2d3v), with both plane and axisymmetric geometries possible. In the code, the simulation window moves with the light velocity, and the quasi-static approximation is used for calculating plasma response. The beams are modeled by fully relativistic macro-particles. The plasma is modeled either by macro-particles (kinetic solver), or as the electron fluid (fluid solver). Transversely inhomogeneous plasmas, non-neutral plasmas, and mobile ions are possible with the kinetic solver. The code is furnished with extensive diagnosing tools which include the possibility of graphical presentation of the results.

The essence of the quasi-static approximation is illustrated by Figure 1. When we calculate the plasma response, the beam is considered as a rigid (not evolving in time) distribution of charges and currents which propagates with the speed of light  $c$ . The fields generated by this beam depend on the longitudinal coordinate  $z$  and time  $t$  only in combination  $\xi = z - ct$  and can be found layer-by-layer starting from the beam head. Since the beam is not changing, all particles started from some transverse position  $r_0$  copy the motion of each other, and their parameters (transverse coordinate and momenta) can be found as functions of  $\xi$ . Thus, a plasma macroparticle in the quasi-static model is not a “big” particle, but a particle tube, i.e., a group of real particles started from a given radius with a given initial momentum. This greatly reduces the memory required for storing plasma particles.

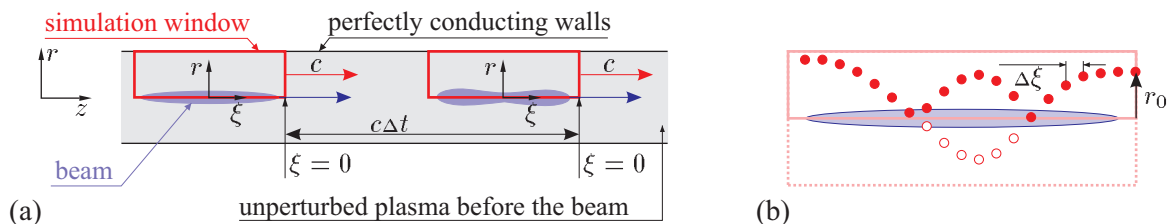


Figure 1: Geometry of the problem (a), and trajectory of a plasma particle in the simulation window (b).

The calculated fields are then used to modify the beam. For highly relativistic beams, the time step  $\Delta t$  for beam particles can be made large, which speeds up simulations several orders of magnitude. The quasi-static approximation is thus useful if and only if the time scale of beam evolution is much longer than the period of plasma wave.

Various details of LCODE and underlying physics are described in the following papers:

- K.V.Lotov, *Simulation of ultrarelativistic beam dynamics in plasma wake-field accelerator*. Phys. Plasmas **5** (1998), p.785-791. — The fluid plasma model.
- K.V.Lotov, *Fine wakefield structure in the blowout regime of plasma wakefield accelerators*. Phys. Rev. ST - Accel. Beams **6** (2003), p.061301. — The beam model and the kinetic plasma model.
- K.V.Lotov, *Blowout regimes of plasma wakefield acceleration*. Phys. Rev. E **69** (2004), p.046405. — Energy fluxes in the co-propagating window.
- K.V.Lotov, V.I.Maslov, I.N.Onishchenko, and E.N.Svistun, *Resonant excitation of plasma wakefields by a non-resonant train of short electron bunches*. Plasma Phys. Control. Fusion **52** (2010), p.065009. — Discussion on applicability of quasi-static codes to simulations of long beams.

## 2 Notation and units of measure

We use cylindrical coordinates  $(r, \varphi, \xi)$  for the axisymmetric geometry and Cartesian coordinates  $(x, y, \xi)$  for the plane geometry. The beam propagates in positive  $\xi$ -direction.

The code works with dimensionless quantities. Units of measure depend on some basic plasma density  $n_0$ . It is recommended to use the initial unperturbed plasma density as  $n_0$ . All times are in units of  $\omega_p^{-1}$ , where  $\omega_p = \sqrt{4\pi n_0 e^2/m}$  is the electron plasma frequency,  $e$  is the elementary charge, and  $m$  is the electron mass. All distances are in units of  $c/\omega_p$ . The unit velocity is  $c$ . The notation used and units of measure for various quantities are given in Table 1.

Table 1: Notation, units of measure, and places of definition for various quantities.

<b>Times:</b>		$\omega_p^{-1}$
$t$	Propagation time (for the beam)	
$\Delta t$	Main time step for the beam, Sec. 5.1, <b>time-step</b>	
$\Delta t'$	Reduced time step for the beam, Sec. 5.1, <b>beam-substepping-energy</b>	
$t_{\max}$	Time limit for the run, Sec. 5.1, <b>time-limit</b>	
$t_F$	Period of the external beam focusing, Sec. 5.2, <b>foc-period</b>	
$t_B$	Oscillation period for the external magnetic field, Sec. 5.3, <b>magnetic-field-period</b>	
$\Delta t_{\text{out}}$	Periodicity of the detailed output, Sec. 5.5, <b>output-period</b>	
<b>Lengths:</b>		$c/\omega_p$
$\xi$	The co-moving coordinate, Sec. 1	
$\vec{r}_b, r_b, x_b, \xi_b$	Coordinates of a beam macro-particle, Sec. 3.3	
$\Delta\xi$	Longitudinal grid step, Sec. 5.1, <b>xi-step</b>	
$\xi_{\max}$	Length of the simulation window, Sec. 5.1, <b>window-length</b>	
$\Delta r$	Transverse grid step, Sec. 5.1, <b>r-step</b>	
$r_{\max}$	Transverse size of the simulation window, Sec. 5.1, <b>window-width</b>	
$r_p, r_{p2}$	Width parameters for some plasma density profiles, Sec. 5.3.1, <b>plasma-profile</b>	
$L_{\text{trap}}$	Path limit for trapped plasma particles, Sec. 5.3.1, <b>trapped-path-limit</b>	
$\xi_{si}$	Check points for diagnosing beam survival, Sec. 5.4, <b>output-survival-xi</b>	
$X_{b0}$	Transverse displacement of the beam slice, Sec. 5.4, <b>output-beam-slices</b>	
$R_b, X_b$	Radius or half-width of the beam slice, Sec. 5.4, <b>output-beam-slices</b>	
$\epsilon$	Emittance of a beam slice, Sec. 5.4, <b>output-beam-slices</b>	
$\xi_{\text{from}}, \xi_{\text{to}}$	Left and right boundaries of the subwindow, Sec. 5.5.1, <b>subwindow-xi-from</b>	
$r_{\text{from}}, r_{\text{to}}$	Bottom and top boundaries of the subwindow, Sec. 5.5.1, <b>subwindow-xi-from</b>	
$r_{\text{ax}}, r_{\text{aux}}$	Two transverse coordinates to output functions of $\xi$ at, Sec. 5.5.2, <b>axis-radius</b>	
$l_s$	Length of the beam segment, Sec. 6.1, <b>length</b>	
$\delta\xi$	Distance to beginning of the beam segment, Sec. 6.1, <b>xishape</b>	
$\xi_s$	$\xi$ -coordinate the segment begins at ( $\xi_s < 0$ ), Sec. 6.1, <b>xishape</b>	
$\sigma_r$	Transverse size of the beam segment, Sec. 6.1, <b>xishape</b>	
$\sigma_z$	Length of the Gaussian beam segment, Sec. 6.1, <b>xishape</b>	
$x_0$	Transverse displacement of the segment, Sec. 6.1, <b>vshift</b>	
<b>Velocities:</b>		$c$
$\vec{v}$	Velocity of a plasma macro-particle or of an electron fluid element, Sec. 3.1, Sec. 3.2	
$\vec{v}_i$	Velocity of $i$ -th plasma macro-particle, Sec. 3.1	
$\vec{v}_b$	Velocity of a beam macro-particle, Sec. 3.3	
<b>Momenta:</b>		$mc$
$\vec{p}$	Momentum of a plasma macro-particle, Sec. 3.1	
$\vec{p}_b$	Momentum of a beam particle (not heavy macro-particle), Sec. 3.3	
$\Delta p_{\text{foc}}$	Extra momentum gained by a beam particle due to external focusing, Sec. 5.2, <b>focusing</b>	
$\vec{p}_e$	Momentum of a plasma electron, Sec. 5.5.1, <b>colormaps-full</b>	
$\vec{p}_i$	Momentum of a plasma ion, Sec. 5.5.1, <b>colormaps-full</b>	
$p_{b,\text{ref}}$	Reference value for displaying beam momentum, Sec. 5.5.4, <b>output-reference-energy</b>	
$p_{b0}$	Basic longitudinal momentum of beam particles in the segment, Sec. 6.1, <b>energy</b>	
$p_a$	Auxiliary value of the longitudinal momentum for the beam segment, Sec. 6.1, <b>espread</b>	
<b>Angular momentum:</b>		$mc^2/\omega_p$
$M_b$	Angular momentum of a beam particle, Sec. 7.1	
<b>Masses:</b>		$m$
$M$	Mass of a plasma macro-particle, Sec. 3.1	
$m_b$	Mass of a beam particle, Sec. 3.3	
<b>Number densities:</b>		$n_0$
$n(r)$	Initial transverse profile of the plasma density, Sec. 5.3.1, <b>plasma-profile</b>	
$n_{p2}$	A parameter for some plasma density profiles, Sec. 5.3.1, <b>plasma-profile</b>	
$n_e$	Density of plasma electrons, Sec. 3.2, Sec. 5.5.1, <b>colormaps-full</b>	

$n_i$	Density of plasma ions, Sec. 5.5.1, <b>colormaps-full</b>
<b>Charge densities:</b> $en_0$	
$\rho$	Charge density of the plasma, Sec. 3.1
$\rho_b$	Charge density of the beam, Sec. 5.5.1, <b>colormaps-full</b>
<b>Current densities:</b> $ecn_0$	
$\vec{j}$	Total current density of plasma particles, Sec. 3.1
$\vec{j}_b$	Current density of the beam, Sec. 3.2
<b>Charges:</b> $e$	
$q$	Charge of a plasma macro-particle, Sec. 3.1
$q_b$	Charge of a beam particle, Sec. 3.3
<b>Currents:</b> $mc^3/e$	
$I_{b0}$	Base beam current, Sec. 5.2, <b>beam-current</b>
$I_b$	Current of the beam slice, Sec. 6.1, <b>xishape</b>
<b>Fields:</b> $E_0 \equiv mc\omega_p/e$	
$\vec{E}$	Electric field in the plasma, Sec. 3.1
$\vec{B}$	Magnetic field in the plasma, Sec. 3.1
$\vec{E}_r, \vec{B}_r$	Auxiliary fields used in the kinetic plasma solver, Sec. 3.1
$B_0$	Amplitude of the external longitudinal magnetic field, Sec. 5.3, <b>magnetic-field</b>
$B_{z0}$	External longitudinal magnetic field, Sec. 5.3, <b>magnetic-field-type</b>
$E_{az}$	Average longitudinal electric field acting on the beam slice, Sec. 5.5.2, <b>f(xi)</b>
<b>Potential:</b> $mc^2/e$	
$\Phi$	Wakefield potential, Sec. 3.2
<b>Energies:</b> $mc^2$	
$W_{\text{kin}}$	Kinetic energy of a plasma electron, Sec. 3.1
$W_{\text{ss}}$	Substepping energy for the beam, Sec. 5.2, <b>beam-substepping-energy</b>
<b>Energy flux densities:</b> $n_0mc^3$	
$\vec{S}_f$	Total energy flux density, Sec. 3.4, Sec. 5.5.1, <b>colormaps-full</b>
$\vec{S}_h$	Collective energy flux density, Sec. 3.4, Sec. 5.5.1, <b>colormaps-full</b>
$\vec{S}_e$	Electromagnetic energy flux density, Sec. 3.4, Sec. 5.5.1, <b>colormaps-full</b>
<b>Energy fluxes:</b> $n_0mc^5/\omega_p^2$	
$\Psi_f$	Total energy flux along the simulation window, Sec. 3.4, Sec. 5.5.2, <b>f(xi)</b>
$\Psi_h$	Collective energy flux along the simulation window, Sec. 3.4, Sec. 5.5.2, <b>f(xi)</b>
$\Psi_e$	Electromagnetic energy flux along the simulation window, Sec. 3.4, Sec. 5.5.2, <b>f(xi)</b>
<b>Energy densities:</b> $n_0mc^2$	
$W_f$	Total energy density, Sec. 3.4, Sec. 5.5.1, <b>colormaps-full</b>
$W_h$	Collective energy density, Sec. 3.4, Sec. 5.5.1, <b>colormaps-full</b>
<b>Linear energy densities:</b> $n_0mc^4/\omega_p^2$	
$W_{\text{int}}$	Linear density of the total energy, Sec. 3.4, Sec. 5.5.2, <b>f(xi)</b>
$dW_{\text{int}}$	Difference between total and collective linear energy densities, Sec. 3.4, Sec. 5.5.2, <b>f(xi)</b>
<b>Distribution functions:</b>	
$f_{\perp}$	Sec. 6.1, <b>rshape</b> $\omega_p/(m^2c^3)$
$f_{4d}$	Sec. 6.1, <b>rshape</b> $\omega_p^2/(m^2c^4)$
$f_{\parallel}$	Sec. 6.1, <b>eshape</b> $1/(mc)$
<b>Focusing strength:</b> $m\omega_p^2$	
$F_s$	Strength of the external beam focusing, Sec. 5.2, <b>foc-strength</b>
<b>Dimensionless:</b>	
$q_i$	Charge of $i$ -th plasma macro-particle, Sec. 3.1
$A$	Normalization factor for calculation of plasma currents and charge densities, Sec. 3.1
$\vec{e}_z$	Unit vector in $z$ -direction, Sec. 3.2
$\tilde{N}$	Quantity used by the fluid solver instead of the electron density, Sec. 3.2
$\gamma$	Relativistic factor of a plasma particle or of a fluid element, Sec. 3.2
$N_r$	Number of grid steps in the transverse direction, Sec. 5.1, <b>r-step</b>
$l$	Number of twofold reductions of the time step for a low-energy beam particle, Sec. 5.2, <b>beam-substepping-energy</b>
$N_b$	Number of beam macro-particles in the slice of current $I_{b0}$ , Sec. 5.2.1, <b>beam-particles-in-layer</b>
$D_{\text{ss}}$	Maximum sub-stepping depth allowed for the kinetic plasma model (0...4), Sec. 5.3.1, <b>substepping-depth</b>
$A_{\text{sub}}$	Sensitivity of the substepping trigger, Sec. 5.3.1, <b>substepping-sensitivity</b>
$d_b$	Fraction of beam particles to output, Sec. 5.4, <b>write-beam-particles</b>

$\eta_{\text{draw}}$	Fraction of the simulation window to be drawn as the full window, Sec. 5.5.1,
$N_{mr}$	Number of grid points to be merged transversely into a single pixel on colored maps, Sec. 5.5.1, <b>colormaps-merging-r</b>
$N_{m\xi}$	Number of grid points to be merged longitudinally into a single pixel on colored maps, Sec. 5.5.1, <b>colormaps-merging-z</b>
$N_{\text{av},r}$	Rate of data smoothing in transverse direction, Sec. 5.5.1, <b>colormaps-smoothing-r</b>
$N_{\text{av},\xi}$	Rate of data smoothing in longitudinal direction, Sec. 5.5.1, <b>colormaps-smoothing-r</b>
$X_m, Y_m$	Substitutes for axis dimensions, Sec. 5.5.2, Sec. 5.5.3
$D_b$	Fraction of beam macro-particles to be drawn, Sec. 5.5.3, <b>draw-each</b>
$h_{\text{fig}}$	Height of beam portraits and histograms in pixels, Sec. 5.5.3, <b>beam-picture-height</b>
$\gamma_{\text{min}}$	Minimum relativistic factor for the plasma particle to be drawn, Sec. 5.5.5, <b>trajectories-min-energy</b>
$C_{\text{step}}$	Color step for visualization of particle energies, Sec. 5.5.5, <b>trajectories-energy-step</b>
$N_{\text{col}}$	Ordinal number of a color in the palette, Sec. 5.5.5, <b>trajectories-energy-step</b>
$\alpha_b$	Angular spread of the beam slice, Sec. 6, <b>angshape</b>
$\alpha_0$	Maximum angular spread in the segment, Sec. 6, <b>angspread</b>
$I_a$	Maximum current in the beam segment, Sec. 6.1, <b>ampl</b>

### 3 Underlying physics

To appear soon.

#### 3.1 Kinetic plasma model

#### 3.2 Fluid plasma model

#### 3.3 Beam model

##### 3.3.1 Low-noise beam

#### 3.4 Energy fluxes and energy densities

#### 3.5 Laser pulse

### 4 Running the code

To run LCODE, execute the file (`lcode.exe`) (Windows) or (`lcode`) (UNIX, execute with `./lcode`) in the folder where the necessary input files are located.

While the program is running its execution could be controlled by pressing some keys in the terminal. Note that this feature won't be available if the program is not run in terminal and that the UNIX realization is still highly experimental. The following keys could be used:

- comma, ', ':** pause the execution after next step, could be used repeatedly to pause after each step
- dot, '.', ':** gracefully stop the execution after next step
- space:** pause the execution immediately
- escape:** terminate the execution immediately
- any key:** resume the paused execution

If the execution ends successfully, the program returns 0. If the execution fails the program will print an error description and terminate with an exit code 1, but if the program was gracefully stopped with pressing a dot or manually interrupted with pressing **Escape** it will terminate with exit codes 2 or 3 respectively.

Possible input files are:

- (lcode.cfg):** the config file, optional (see section 5)
- (beamfile.bin):** the beam state file (see section 7.1)
- (beamfile.bit):** continuation run indicator (see section 7.2)
- (plasma.bin):** plasma state file (see section 7.3)
- (fields.bin):** fields state file (see section 7.4)
- arbitrarily named:** other config files (see section 6), beam constructor file (see **beam-profile**)

Additional command-line options may be provided after the executable name.

Examples:

```
./lcode executes LCODE with the values read from lcode.cfg, if present. (UNIX)
lcode.exe filename.cfg --beam-tune-charge=y same as the above, also applies the values from
(filename.cfg) and explicitly enables the beam-tune-charge option. (Windows)
lcode.exe filename1.cfg filename2.cfg --dump first applies the values from (lcode.cfg), if present;
then applies the values from the files (filename1.cfg) and (filename2.cfg); then does not start the
calculations, but instead prints the resulting values after reading these files. (Windows)
```

A more detailed explanations of all possible options:

- Options `--option=value` are those usable in configuration file **lcode.cfg**.
- Directives `filename` may be used to include other files containing configuration options.
- Special options interrupt the normal process of parsing command-line parameters and terminate execution of the program with the following actions:
  - `--help` or `--usage` outputs a brief description of all possible configuration options.
  - `--dump` or `--dump-config` outputs the already read values of configuration options to console. To save the output to a file (`config.cfg`) execute this `lcode.exe --dump-config > config.cfg` (Windows) or `./lcode --dump-config > config.cfg` (UNIX)
  - `--dump-defconfig` outputs the the configuration file with default values of options to console. To save it to a file (`default.cfg`) execute this `lcode.exe --dump-defconfig > default.cfg` (Windows) or `./lcode --dump-defconfig > default.cfg` (UNIX)
  - `--dump-docconfig` outputs the the configuration file with default values of options, but in a more detailed format. To save it to a file (`docconfig.cfg`) execute this `lcode.exe --dump-docconfig > docconfig.cfg` (Windows) or `./lcode --dump-docconfig > docconfig.cfg` (UNIX)

If an option occurs several times in configuration files or command line, then the latter value is used. The order of reading the values is the following:

1. Initially, all options have the default hard-coded values (the ones specified as default in this manual, section 5).
2. The main configuration file 'lcode.cfg', if present, with all the files included. The file is read from the beginning to the end, with the included files, if any.
3. The command-line options processed left-to-right, if provided, with the included files if any.

After reading the values get validated. If some values are invalid, instead of exiting the program may attempt an automatic 'config recovery', which consists of resetting some of the invalid values to the hardcoded defaults. Warnings will be printed and logged during the recovery. An example config recovery session looks like this:

```
invalid histogram-bins=0 (Must be positive and <= 300) was reset to default 300
config recovered
```

```
Error filling config:  invalid config, some defaults used
```

```
The config is still valid though
```

```
Trying to use a recovered config
```

If the recovery has been completed successfully, the program will reattempt the execution with the recovered config. The exact (possibly recovered) config values at the start of the computation could be written to (`lcode.runas.cfg`) to ease the diagnostics (see **save-config**, **save-config-filename**).

## 5 Configuration file

The configuration file is a text file containing the list of options in the form

```
option=value   or   option = value.
```

The order of options is of no importance, though in case of specifying the option multiple times the latter value overrides the previous ones.

Several options may coexist on the same line of the file, provided they are separated by semicolons. Semicolons may be omitted in several cases, but it is strongly recommended not to rely on that.

The configuration file can contain comments or empty lines which are ignored by the program, but improve visual readability of the file. A comment begins with **#** and extends to the end of line.

In this section, configuration options are described like this:

**option**, *type* (default value): brief description  
 Detailed description, if available.

Possible types of options are:

*choice*: a value from a predefined set.

*float*: a floating-point number, may be in scientific notation.

*string*: a string (multiple characters). Strings with spaces, semicolons or hash signs must be enclosed in double quotes, multiline strings must be enclosed in triple double quotes.

*multichoice*: multiple values from the predefined set, separated with commas.

*y/n*: ‘yes’ or ‘no’. Brief forms ‘y’ and ‘n’ are also possible.

The following is the complete list of options grouped by purpose.

## 5.1 Simulation area

**geometry**, *choice* (c): The geometry of the problem:

‘c’ or ‘axisymmetric’: Axisymmetric (cylindrical) geometry

‘p’ or ‘plane’: Plane (2d Cartesian) geometry

**window-width**, *float* (5): Transverse size of the simulation window,  $r_{\max}$

Transverse size of the simulation window: radius in the cylindrical geometry or full width in the plane geometry.

**window-length**, *float* (15): Length of the simulation window,  $\xi_{\max}$

**r-step**, *float* (0.05): Transverse grid step,  $\Delta r$

The number of grid steps in transverse direction  $N_r$  must not be too large:  $N_r = r_{\max}/\Delta r \leq 10000$ . Otherwise  $\Delta r$  is automatically increased.

**xi-step**, *float* (0.05): Longitudinal grid step,  $\Delta \xi$

**time-limit**, *float* (200.5): Time limit for the run,  $t_{\max}$

To insure against round off accumulation, put here a somewhat greater value than the last time moment you wish to calculate.

**time-step**, *float* (25): Main time step for the beam,  $\Delta t$

**continuation**, *choice* (n): Mode of plasma continuation:

‘n’ or ‘no’: Evolution of the beam. Every time step the beam enters an unperturbed plasma of a prescribed profile.

‘y’ or ‘beam’: Evolution of the beam sequence. Every time step (except the first) the beam enters the perturbed plasma; the plasma state is taken from the end of the previous simulation window. Does not work for the fluid plasma model.

‘Y’ or ‘longplasma’: Evolution of the plasma. A long beam creates the wake, and long-term behavior of this wake is followed by the sequence of simulation windows. In this mode, only a rigid beam can extend to several simulation windows. Does not work for the fluid plasma model.

Latter two regimes need  $\Delta t = \xi_{\max}$ , otherwise  $\Delta t$  is automatically corrected.

## 5.2 Particle beams

**beam-current**, *float* (0.1): Base beam current (in 17 kA),  $I_{b0}$

The common multiplier (in units of  $mc^3/e \approx 17\text{kA}$ ) for dimensionless beam currents specified in **beam-profile**. In plane geometry, corresponds to the current through  $c/\omega_p$  in the third dimension. Also the unit current for beam macro-particles.

**rigid-beam**, *y/n* (n): Switch for evolution of the beam

‘y’ or ‘yes’: A rigid (not evolving in time) distribution of the beam current.

‘n’ or ‘no’: The beam is modeled by macro-particles.

**beam-substepping-energy**, *float* (2): Substepping energy for the beam,  $W_{ss}$

Threshold of reducing the time step for beam particles. For each beam particle, the minimal integer  $l$  is found that meets the condition  $2^{2l}(1 + p_{bz}) > W_{ss}$ , and the reduced time step  $\Delta t' = \Delta t/2^l$  is then calculated. Plasma fields are calculated with periodicity  $\Delta t$ , and each beam particle is propagating in these fields with its own time step  $\Delta t'$ . This feature is particularly useful if low energy beam particles are present in the system, which otherwise would require undesirable reduction of the main time step  $\Delta t$ .

**focusing**, *choice* (n): External focusing for the beam:

‘n’ or ‘no’: No focusing

‘c’ or ‘cosine’: Cosine-varying focusing force. Each beam particle (located at radius  $r_b$ ) at each time step gets the extra radial momentum  $\Delta p_{\text{foc}} = F_s r_b \Delta t \cos(2\pi t/t_F)$ .

‘r’ or ‘rectangular’: Piecewise-constant focusing force. Each beam particle (located at radius  $r_b$ ) at each time step gets the extra radial momentum  $\Delta p_{\text{foc}} = \pm F_s r_b \Delta t$ , where “+” is chosen if the fractional part of  $(t/t_F + 0.25)$  is less than 0.5, and “-” is chosen otherwise.

In the plane geometry,  $\Delta p_{\text{foc}}$  is the addition to  $x$ -momentum, and the distance between the particle and the midplane of the simulation window is used instead of  $r_b$ .

**loc-period**, *float* (100): Period of the external focusing,  $t_F$

**loc-strength**, *float* (0.01): Strength of the external focusing,  $F_s$

### 5.2.1 Newly generated beams

The following beam options determine parameters of newly generated particle beams or rigid beams. If the beam is allowed to evolve by **rigid-beam** and the files (**beamfile.bit**) and (**beamfile.bin**) are present in the working folder, then the beam state is imported from (**beamfile.bin**), the time is read from (**beamfile.bit**), and these options are ignored.

**beam-particles-in-layer**, *integer* (200): Number of beam particles in the layer ( $\leq 100000$ ),  $N_b$

A beam slice which has the current  $I_{b0}$  and length  $\Delta \xi$  contains  $N_b$  macro-particles. Beam slices with lower currents have correspondingly fewer macro-particles.

**beam-profile**, *string* (“xishape=cos, length=1”): Initial distribution of beam particles

The name of the file that specifies the initial distribution of beam particles in 6d phase space, or the initial beam distribution itself (as a multiline description enclosed in triple quotes). If the value consists of a single line without spaces or equal signs, it is interpreted as a filename to read the beam distribution from; otherwise the value itself is considered the beam distribution description. The format of beam distribution is described in section 6.

**beam-tune-charge**, *y/n* (n): Tuning the charge of beam particles to match the beam profile better

If the option is off, all beam particles are equally charged. Only the absolute value of the charge can be different. Consequently, the beam current can have only discrete values and therefore is slightly different from the specified value.

If this option is on, the charge of particles in each beam slice is slightly modified to better match the specified beam profile. This option lowers the shot noise produced by the beam.

**rng-seed**, *integer* (1): Random generator seed

An integer number initiating the generator of random numbers. Different seeds generate different statistical ensembles for the beam.

## 5.3 Plasma

**plasma-model**, *choice* (P): Plasma model:

‘f’ or ‘fluid’: The fluid model. It is the fastest one, but works only for the initially uniform plasma with immobile ions. Neither wave-breaking, nor near-wall plasma perturbations are allowed.

‘p’ or ‘particles’: Old kinetic model.

‘P’ or ‘newparticles’: New kinetic model.



**magnetic-field**, *float* (0): External longitudinal magnetic field,  $B_0$

For zero  $B_0$ , the code runs faster since some equations (which are identically zeros) are not solved.

**magnetic-field-type**, *choice* (c): Time dependence of the external magnetic field seen by the beam:

‘c’ or ‘constant’: Always  $B_0$ .

‘r’ or ‘random’: Random value from 0 to  $B_0$  for each time step.

‘p’ or ‘periodic’:  $B_{z0} = B_0 \cos(2\pi t/t_B)$ .

**magnetic-field-period**, *float* (200): Period of magnetic field oscillations,  $t_B$

Used only with the periodic external magnetic field.

### 5.3.1 Options specific to particle plasma models

These options have effect only if **plasma-model** is a kinetic one.

**plasma-particles-number**, *int* (1000): Number of plasma macro-particles ( $\leq 40000$ )

All plasma electrons are modeled by this number of macro-particles. Mobile ions, if chosen, are modeled by the same number of heavier macro-particles; in this case this number must be  $\leq 20000$ .

**plasma-profile**, *choice* (1): The initial transverse profile of the plasma density:

‘1’ or ‘uniform’: Uniform up to the walls.

‘2’ or ‘stepwise’: Uniform up to  $r_p$ , zero at  $r > r_p$ .

‘3’ or ‘gaussian’: Gaussian,  $n(r) = n_0 \exp(-r^2/2r_p^2)$ , obtained by changing weights of macro-particles.

‘4’ or ‘arbitrary’: Arbitrary, with particle parameters imported from (**plasma.bin**) and initial fields from (**fields.bin**).

‘5’ or ‘channel’: Zero up to  $r_p$ , uniform at  $r > r_p$ .

‘6’ or ‘sub-channel’: Constant ( $n_{p2}$ ) up to  $r_{p2}$ , then linear growth from  $n_{p2}$  to  $n_0$  between  $r_{p2}$  and  $r_p$ , then  $n_0$  at  $r > r_p$ ; obtained by changing weights of macro-particles.

**plasma-width**, *float* (2): Main parameter of initial plasma density distributions,  $r_p$

Used if **plasma-profile** is ‘2’, ‘3’, or ‘6’.

**plasma-width-2**, *float* (1): Auxiliary parameter of initial plasma density distributions,  $r_{p2}$

**plasma-density-2**, *float* (0.5): Auxiliary parameter of initial plasma density distributions,  $n_{p2}$

Used if **plasma-profile** is ‘6’.

**plasma-temperature**, *float* (0): Initial temperature of mobile plasma particles in units of  $mc^2$

**ion-model**, *choice* (y): Model of plasma ions:

‘Y’ or ‘mobile’: Half of plasma macro-particles are single-charged mobile ions initially located at the same positions as plasma electrons.

‘y’ or ‘background’: Ions are immobile background charge.

‘n’ or ‘absent’: No ions, plasma electrons are initially at rest.

**ion-mass**, *float* (1836): Ion mass in units of the electron mass (for mobile ions)

**substepping-depth**, *integer* (3): Maximum sub-stepping depth allowed (0...4),  $D_{ss}$

Substepping is usually needed for strongly nonlinear wakefields when some plasma particles are close to trapping. If necessary, the longitudinal grid step can be automatically reduced up to  $10^{D_{ss}}$  times with respect to the basic **xi-step**.

**substepping-sensitivity**, *float* (0.2): Sensitivity of substepping trigger,  $A_{sub}$ .

If the longitudinal current density ( $j_z$ ) is so high that the product of  $|j_z|$  and  $\xi$ -step exceeds  $A_{sub}$  at some point, then the  $\xi$ -step is automatically divided by 10 (if allowed by **substepping-depth**). Reverse action (increasing the  $\xi$ -step) is taken when  $j_z$  gets small again.

**trapped-path-limit**, *float* (0): Path limit for trapped plasma particles,  $L_{\text{trap}}$

With this option, it is possible to treat trapping of plasma particles by the wakefield (to the extent allowed by the quasi-static approximation). The charge of a macro-particle is put to zero when the time spent by this macro-particle in the simulation window exceeds  $L_{\text{trap}}$ . With this trick we obtain the correct plasma state and fields at the distance  $L_{\text{trap}}$  from the beam entrance to the plasma even if some plasma particles get trapped by the wakefield. It is necessary to put  $L_{\text{trap}} \gg \xi_{\text{max}}$ , otherwise the result will have no physical meaning. Zero value of  $L_{\text{trap}}$  switches this option off.

For every “trapped” macro-particle, 10 values are appended to file (`captured.pls`) (one particle per line):

1. Time
2. Number of the macro-particle (which contains information on its initial position)
3. Final  $\xi$ -coordinate
4. Final transverse coordinate
- 5-7. Final  $r$ -,  $\varphi$ -, and  $z$ -components of the particle momentum
8. Mass of the macro-particle
9. Charge-to-mass ratio
10. Relativistic factor

### 5.3.2 Option specific to the fluid plasma model

**viscosity**, *float* (0): Artificial viscosity

Artificial viscosity is used for suppressing high-frequency numerical noises and forcing the fluid model to work beyond the applicability area. Reasonable values are between 0 and 0.01.

## 5.4 Every-time-step diagnostics

If activated, these diagnostics work at each main time step  $\Delta t$ .

**indication-line-format**, *choice* (1): Format of the on-screen progress indication:

- ‘1’ or ‘eachdt’: One line each time step: time, total number of survived beam macro-particles, maximum and minimum electric field  $E_z$  on the axis.
- ‘2’ or ‘eachdxi’: One line each  $\xi$ -step: time,  $|\xi|$ ,  $E_z$  on axis, total energy flux  $\Psi_f$ , number of beam macro-particles in this layer, number of sub-steps in  $\xi$  within the last  $\xi$ -step.

Location of the axis is determined by **axis-radius**. After finishing a time step (just before drawing pictures), the word “finished” is printed.

**output-Ez-minmax**, *y/n* (n): Write absolute extrema of the on-axis  $E_z$  into `(emaxf.dat)`

**output-Phi-minmax**, *y/n* (n): Write absolute extrema of the on-axis  $\Phi$  into `(gmaxf.dat)`

**output-Ez-local**, *y/n* (n): Write local extrema of the on-axis  $E_z$  into `(elocf.dat)`

**output-Phi-local**, *y/n* (n): Write local extrema of the on-axis  $\Phi$  into `(glocf.dat)`

If enabled, a line of 5 values is appended to the corresponding file immediately after an extremum is found:

1. Current time  $t$
2. The maximum value of the quantity
3.  $\xi$ -coordinate of this maximum
4. The minimum value of the quantity (earlier found in case of local extrema)
5.  $\xi$ -coordinate of this minimum

**output-beam-survival**, *y/n* (n): Output of the number of survived beam particles

**output-survival-r**, *float* (2): Radius of the area in which beam particles are considered survived

**output-survival-xi**, *string* (-2, -4, -6, -8, -10, -12): Check points for diagnosing beam survival

6 negative values separated by commas, in the descending order,  $\xi_{si}$  ( $i = 1 \dots 6$ ).

If enabled, then a line of 7 values is appended to file (`n1.dat`) at each time step:

1. Time  $t$ ,
- 2-7. Number of survived beam macro-particles with  $\xi_b > \xi_{si}$

The information is written by one value immediately after passing  $\xi_{si}$ .

**output-beam-slices**, *y/n* (n): Output of beam characteristics at selected cross-sections

**output-slices-xi**, *string* (-2, -4, -6, -8, -10, -12): Cross-sections for diagnosing beam properties  
6 negative values separated by commas, in the descending order.

If the diagnostics is enabled, characteristics of 6 beam slices are appended to files (**tt\*.dat**), where (\*) is a cross-section number (1...6). When the slice is passed, 6 values are appended to the corresponding file:

1. Time  $t$ ,
2. On-axis electric field  $E_z$  at this  $\xi$ ,
3. Emittance  $\epsilon$  of the slice,
4. Beam transverse displacement  $X_{b0}$  (is nonzero only in plane geometry),
5. Beam radius  $R_b$  or half-width  $X_b$ ,
6. Number of beam macro-particles in this layer.

**write-beam-particles**, *y/n* (n): Output of individual characteristics of beam particles:

**write-beam-particles-each**, *integer* (1000): Fraction of beam particles to output,  $d_b$

A beam particle is selected for output if its ordinal number is a multiple of  $d_b$ .

**write-beam-particles-from**, *float* (0): Right limit.

**write-beam-particles-to**, *float* (-10): Left limit

Only beam particles located between these two non-positive values are selected for output.

If the diagnostics is enabled, then the following information about selected beam particles is appended to file (**partic.swp**) at every time step, one line (9 values) per particle:

1. Time  $t$ ,
2. Longitudinal coordinate  $\xi_b$ ,
3. Transverse coordinate  $r_b$  or  $x_b$ ,
4. Longitudinal momentum  $p_{bz}$ ,
5. Transverse momentum  $p_{br}$  or  $p_{bx}$ ,
6. Angular momentum  $M_b$  or third component of the momentum  $p_{by}$ ,
7. Charge-to-mass ratio (absolute value),
8. Current carried by the particle,
9. Ordinal number.

The same particle characteristics can be also extracted from (**beamfile.bin**).

**output-lost-particles**, *y/n* (n): Output of lost particles to (**beamlost.dat**)

If enabled, then beam particles which exit the simulation window are kept in file (**beamlost.dat**). One line of 9 values is written for each particle. These values correspond to the last time particle was in the simulation window. The file format is the same as for (**partic.swp**).

## 5.5 Periodical diagnostics

These diagnostics are periodically triggered with the time interval given by

**output-period**, *float* (100): Time periodicity of detailed output,  $\Delta t_{\text{out}}$

If  $\Delta t_{\text{out}} < \Delta t$ , then each time step is diagnosed. The first time step is always diagnosed.

### 5.5.1 Colored maps

These keys control output of various quantities as functions of  $r$  and  $\xi$ , either in the form of a colored map, or in the form of a data array. The colored maps are produced as separate files named (??\*\*\*\*\*[w].png) where (??) stands for two-character quantity abbreviation, (\*\*\*\*\*) is the time of output, and the optional suffix 'w' denotes subwindow output.

**colormaps-full**, *multichoice* (""): A list of quantities to output in the full window

**colormaps-subwindow**, *multichoice* (""): A list of quantities to output in the subwindow

Full window means the **drawn-portion** of the whole simulation window, the area of size  $\xi_{\max} \times r_{\max} \eta_{\text{draw}}$ .  
Subwindow size and location are controlled manually. Output of the following quantities is possible:

'Er': first transverse component of the electric field, $E_r$ or $E_x$	(er*****[w].png .swp)
'Ef': second transverse component of the electric field, $E_\varphi$ or $E_y$	(ef*****[w].png .swp)
'Ez': $z$ -component of the electric field, $E_z$	(ez*****[w].png .swp)
'Phi': wakefield potential, $\Phi$	(fi*****[w].png .swp)
'Bf': $\varphi$ or $y$ component of the magnetic field, $B_\varphi$ or $B_y$	(bf*****[w].png .swp)
'Bz': $z$ -component of the magnetic field, $B_z$	(bz*****[w].png .swp)
'pr': first transverse component of the electron momentum, $p_{er}$ or $p_{ex}$	(pr*****[w].png .swp)
'pf': second transverse component of the electron momentum, $p_{e\varphi}$ or $p_{ey}$	(pf*****[w].png .swp)
'pz': $z$ -component of the electron momentum, $p_{ez}$	(pz*****[w].png .swp)
'pri': first transverse component of the ion momentum, $p_{ir}$ or $p_{ix}$	(ir*****[w].png .swp)
'pfi': second transverse component of the ion momentum, $p_{i\varphi}$ or $p_{iy}$	(if*****[w].png .swp)
'pzi': $z$ -component of the ion momentum, $p_{iz}$	(iz*****[w].png .swp)
'nb': charge density of particle beams, $\rho_b$	(nb*****[w].png .swp)
'ne': density of plasma electrons, $n_e$	(ne*****[w].png .swp)
'ni': density of plasma ions, $n_i$	(ni*****[w].png .swp)
'Sf': $z$ -component of the total energy flux density (e.f.d.), $S_{fz}$	(sf*****[w].png .swp)
'Sr': $r$ -component of the total e.f.d., $S_{fr}$	(sr*****[w].png .swp)
'dS': $z$ -component of the thermal e.f.d., $S_{fz} - S_{hz}$	(sd*****[w].png .swp)
'Sf2': $r$ -weighted $z$ -component of the total e.f.d., $2\pi r S_{fz}$	(2f*****[w].png .swp)
'Sr2': $r$ -weighted $r$ -component of the total e.f.d., $2\pi r S_{fr}$	(2r*****[w].png .swp)
'dS2': $r$ -weighted $z$ -component of the thermal e.f.d., $2\pi r(S_{fz} - S_{hz})$	(2d*****[w].png .swp)
'Wf': full energy density, $W_f$	(wf*****[w].png .swp)
'dW': thermal energy density, $W_f - W_h$	(wd*****[w].png .swp)
'SEB': $z$ -component of the electromagnetic e.f.d., $S_{ez}$	(se*****[w].png .swp)

**colormaps-type**, *choice* (y): Controls whether functions of  $(r, \xi)$  are output as data files or as pictures:

- 'n' or 'numbers': Only data files.
- 'y' or 'pictures': Only pictures.
- 'F' or 'both': Both pictures and data files.

**drawn-portion**, *float* (1): Fraction of the simulation window to be referred as the full window,  $\eta_{\text{draw}}$   
A number between 0 and 1. Used to exclude uninteresting near-wall regions from the output.

**subwindow-xi-from**, *float* (-5): Right boundary of the subwindow,  $\xi_{\text{from}}$

**subwindow-xi-to**, *float* (-10): Left boundary of the subwindow,  $\xi_{\text{to}}$

**subwindow-r-from**, *float* (0): Bottom boundary of the subwindow,  $r_{\text{from}}$

**subwindow-r-to**, *float* (3): Top boundary of the subwindow,  $r_{\text{to}}$

Four parameters controlling the size and position of the subwindow.

**colormaps-merging-r**, *integer* (1): Merging in  $r$ , grid points,  $N_{mr}$

**colormaps-merging-z**, *integer* (1): Merging in  $\xi$ , grid points,  $N_{m\xi}$

These numbers control output of one average value from several grid points. Values from the specified number of grid points ( $N_{mr}$  in  $r$  and  $N_{m\xi}$  in  $\xi$ ) are added together and divided by the number of points ( $N_{mr}N_{m\xi}$ ). This procedure is made after the passage of the whole simulation window, when all the values are written to an auxiliary file.

**colormaps-smoothing-r**, *integer* (0): Averaging in  $r$ , times,  $N_{\text{av},r}$

**colormaps-smoothing-z**, *integer* (0): Averaging in  $z$ , times,  $N_{\text{av},\xi}$

Smoothing out data arrays prepared for drawing as color maps. The elementary smoothing procedure consists in delivering 1/4 of the value per adjacent cell (Fig.??). The procedure is repeated  $N_{\text{av},r}$  times in  $r$ -direction and  $N_{\text{av},\xi}$  times in  $\xi$ -direction. Smoothing is made after merging of grid points, if any.

**palette**, *choice* (d): Colormap palette: default/greyscale/hue/bluewhitered

Coloring method used to display values of functions on  $(r, \xi)$  plane (Fig.??):

- 'd' or 'default': Distinctly-colored palette
- 'g' or 'greyscale': Greyscale palette
- 'h' or 'hue': Hue-equidistant palette
- 'b' or 'bluewhitered': Blue-white-red palette

The following parameters (color steps) set up a correspondence between an increment of a quantity and color

change on the picture:

- E-step**, *float* (0.1): Components of the electric field: ‘Er’, ‘Ef’, ‘Ez’.
- Phi-step**, *float* (0.1): Wakefield potential: ‘Phi’.
- Bf-step**, *float* (0.1): Transverse magnetic field: ‘Bf’.
- Bz-step**, *float* (0.1): Longitudinal magnetic field: ‘Bz’.
- electron-momenta-step**, *float* (0.1): Electron momenta: ‘pr’, ‘pf’, ‘pz’.
- ion-momenta-step**, *float* (0.1): Ion momenta: ‘pri’, ‘pfi’, ‘pzi’.
- nb-step**, *float* (0.1): Charge density of particle beams: ‘nb’.
- ne-step**, *float* (0.1): Density of plasma electrons: ‘ne’.
- ni-step**, *float* (0.01): Density of plasma ions: ‘ni’.
- flux-step**, *float* (0.01): Components of energy flux densities: ‘Sf’, ‘Sr’, ‘dS’, ‘SEB’.
- r-corrected-flux-step**, *float* (0.01): *r*-corrected energy flux densities: ‘Sf2’, ‘Sr2’, ‘dS2’.
- energy-step**, *float* (0.01): Energy densities: ‘Wf’, ‘dW’.

### 5.5.2 Functions of $\xi$

These keys control output of various quantities as functions of  $\xi$ , either in the form of a graph, or in the form of a data array. The graphs are plotted in files named (**u\*\*\*\*\*.png**) or (**v\*\*\*\*\*.png**) by different colors, where (**\*\*\*\*\***) is the time of output. Data arrays are named (**u\*\*\*\*\*.swp**) and (**v\*\*\*\*\*.swp**). Each line in the data array correspond to certain  $\xi$  and contains 13 (**u\*\*\*\*\*.swp**) or 17 (**v\*\*\*\*\*.swp**) columns. First columns in the files are values of  $\xi$ , others are output quantities. If a quantity is not chosen for output, zero value is written in its place.

**f(xi)**, *multichoice* (Ez,nb2,Ez2,dS,Sf,SEB): A list of quantities to output as functions of  $\xi$ .

The quantities can be listed in an arbitrary order. There are two groups of quantities, which differ by files of output. Indicated in braces are: the column number in the data array (**?\*\*\*\*\*.swp**), the color of the graph in (**?\*\*\*\*\*.png**), the style of the graph.

First group:

			( <b>u*****.png</b> ) ( <b>swp</b> )
‘ne’	{2, grey line}	$n_e(r_{ax})$	on-axis density of plasma electrons
‘nb’	{3, blue line}	$\rho_b(r_{ax})$	on-axis charge density of the beam
‘Ez’	{4, green line}	$E_z(r_{ax})$	on-axis <i>z</i> -component of the electric field
‘⟨Ez⟩’	{5, white line}	$E_{az}$	average longitudinal electric field acting on the beam
‘Bz’	{6, dark red line}	$B_z(r_{ax})$	on-axis <i>z</i> -component of the magnetic field
‘Phi’	{7, magenta line}	$\Phi(r_{ax})$	on-axis wakefield potential
‘pz’	{8, dark blue line}	$p_{ez}(r_{ax})$	on-axis <i>z</i> -momentum of plasma electrons
‘emitt’	{9, green points}	$\epsilon$	emittance of the beam slice
‘dW’	{10, dark grey line}	$dW_{int}$	thermal energy per unit length
‘Wf’	{11, dark grey line}	$W_{int}$	total energy per unit length
‘ni’	{12, white line}	$n_i(r_{ax})$	on-axis density of plasma ions
‘pzi’	{13, dark blue line}	$p_{iz}(r_{ax})$	on-axis <i>z</i> -momentum of plasma ions

Second group:

			( <b>v*****.png</b> ) ( <b>swp</b> )
‘nb2’	{2, blue line}	$\rho_b(r_{aux})$	off-axis charge density of the beam
‘Er’	{3, cyan line}	$E_r$ or $E_x(r_{aux})$	off-axis transverse electric field
‘Ez2’	{4, dark grey line}	$E_z(r_{aux})$	off-axis longitudinal electric field
‘Bf’	{5, brown line}	$B_\varphi$ or $B_y(r_{aux})$	off-axis $\varphi$ - or <i>y</i> component of the magnetic field
‘Bz2’	{6, dark red line}	$B_z(r_{aux})$	off-axis <i>z</i> -component of the magnetic field
‘Fr’	{7, yellow line}		average focusing force, $(\Phi_{aux} - \Phi_{ax})(r_{aux} - r_{ax})$
‘pr’	{8, dark green line}	$p_{er}$ or $p_{ex}(r_{aux})$	off-axis <i>r</i> - or <i>x</i> -momentum of plasma electrons
‘pf’	{9, dark red line}	$p_{e\varphi}$ or $p_{ey}(r_{aux})$	off-axis $\varphi$ - or <i>y</i> -momentum of plasma electrons
‘⟨rb⟩’	{10, red points}	$R_b$ or $X_b$	radius/width of the beam
	{11, dark red points}	$0$ or $X_{b0}$	transverse displacement of the beam
‘dS’	{12, dark grey line}	$\Psi_f - \Psi_h$	integral thermal energy flux
‘Sf’	{13, dark grey line}	$\Psi_f$	integral total energy flux
‘SEB’	{14, dark grey line}	$\Psi_e$	integral electromagnetic energy flux
‘pri’	{15, dark green line}	$p_{ir}$ or $p_{ix}(r_{aux})$	off-axis <i>r</i> - or <i>x</i> -momentum of plasma ions
‘pfi’	{16, dark red line}	$p_{i\varphi}$ or $p_{iy}(r_{aux})$	off-axis $\varphi$ - or <i>y</i> -momentum of plasma ions
‘Ef’	{17, magenta line}	$E_\varphi$ or $E_y(r_{aux})$	off-axis $\varphi$ - or <i>y</i> component of the electric field

**f(xi)-type, choice (n):** Output mode for the functions of  $\xi$ :

- 'y' or 'pictures': Only pictures (**u\*\*\*\*\*.png**) and (**v\*\*\*\*\*.png**) are created.
- 'Y' or 'numbers': Only data files (**u\*\*\*\*\*.swp**) and (**v\*\*\*\*\*.swp**) are created.
- 'F' or 'both': Both pictures and data files are created.
- 'n' or 'n': No output of this kind.

**axis-radius, float (0):** Position of the 'probe' line 1,  $r_{ax}$

**auxiliary-radius, float (1):** Position of the 'probe' line 2,  $r_{aux}$

Two transverse positions at which functions of  $\xi$  can be output. Both are measured from the geometrical axis (in cylindrical geometry) or middle of the simulation window (in plane geometry). It makes sense to put  $r_{ax} \neq 0$  only if the beam goes off-axis (in plane geometry) or if the plasma density is too noisy at the geometrical axis (in cylindrical geometry).

The following parameters ( $X_m$ ) determine the size of the vertical axis when drawing functions of  $\xi$ . The axis is:

- (0,  $X_m$ ) for fixed sign quantities,
- ( $-X_m$ ,  $X_m$ ) for sign-changing quantities.

For ion and electron densities, the axis type depends on  $X_m$ :

- if  $X_m \geq 1$ , then (0,  $X_m$ ), the absolute value of the density is drawn,
- if  $X_m < 1$ , then ( $-X_m$ ,  $X_m$ ), the density perturbation is drawn.

**E-scale, float (1):** Most of the fields: 'Er', 'Ef', 'Ez', 'Ez2', '<Ez>', 'Bf', 'Fr'

**Phi-scale, float (1):** Wakefield potential: 'Phi'

**Bz-scale, float (1):** Longitudinal magnetic field: 'Bz', 'Bz2'

**electron-momenta-scale, float (1):** Electron momenta: 'pr', 'pf', 'pz'

**ion-momenta-scale, float (1):** Ion momenta: 'pri', 'pfi', 'pzi'

**beam-radius-scale, float (5):** Beam width and displacement: '<rb>'

**nb-scale, float (1):** Charge density of particle beams: 'nb', 'nb2'

**ne-scale, float (2):** Density of plasma electrons: 'ne'

**ni-scale, float (1):** Density of plasma ions: 'ni'

**flux-scale, float (1):** Integral energy fluxes: 'dS', 'Sf', 'SEB'

**energy-scale, float (1):** Energies per unit length: 'dW', 'Wf'

**emittance-scale, float (5):** Beam emittance: 'emitt'

### 5.5.3 Beam particle information as pictures

These keys control output of beam portraits in various parameter spaces. The selected beam macro-particles are drawn as cyan dots in pictures (**?\*\*\*\*\*.png**), where (?) shows the content, and (\*\*\*\*\*) is the time.

**output-beam-particles, multichoice (""):** A list of beam projections to be drawn

Beam macro-particles can be output on the following planes in the following files:

- 'r': ( $r, \xi$ ) plane (real space) **(p\*\*\*\*\*.png)**
- 'pr': ( $p_{br}, \xi$ ) plane (transverse momentum) **(r\*\*\*\*\*.png)**
- 'pz': ( $p_{bz}, \xi$ ) plane (longitudinal phase space) **(z\*\*\*\*\*.png)**
- 'M': ( $M_b, \xi$ ) plane (angular momentum) **(m\*\*\*\*\*.png)**

**draw-each, integer (20):** Fraction of beam macro-particles to be drawn,  $D_b$

A beam macro-particle is drawn if its ordinal number is a multiple of  $D_b$ .

**beam-picture-height, integer (300):** Height of beam portraits in pixels,  $h_{fig}$

This number determines the visual height of all histograms and beam portraits except the real space portrait. The latter has the same height as colored maps (equal to  $\eta_{draw} r_{max} / \Delta r / N_{mr}$ ).

**output-reference-energy, float (1000):** Reference value for displaying beam momentum,  $p_{b,ref}$ .

The following parameters ( $Y_m$ ) determine the size of the vertical axis for corresponding beam portraits:

**beam-pr-scale, float (100):** Axis is ( $-Y_m, Y_m$ ) for  $r$ -momentum of beam particles ('pr')

**beam-a-m-scale, float (100):** Axis is (0,  $Y_m$ ) for the angular momentum of beam particles ('M')

**beam-pz-scale, float (2000):** Axis for  $z$ -momentum of beam particles ('pz')

The latter axis depends on the ratio between  $Y_m$  and **output-reference-energy**  $p_{b,ref}$ . If  $Y_m > p_{b,ref}$ , the axis is (0,  $Y_m$ ), otherwise it is ( $p_{b,ref} - Y_m, p_{b,ref} + Y_m$ ).

### 5.5.4 Beam information as histograms

These keys control output of beam characteristics in the histogram form, either as a picture (`d?*****.png`), or as a data file (`d?*****.swp`), where (?) shows the content, and (\*\*\*\*\*) is the time. If a macro-particle falls outside the chosen histogram interval, the particle is ignored.

Vertical size of the pictures in pixels is determined by **beam-picture-height**. The picture height in physical units equals the highest column. To retain the normalization, the line of 3 values is appended to file (`hystog.dat`) each time a histogram picture is produced. These values are:

1. Time
2. Histogram type (second character of the filename)
3. Height of the highest column.

The data files for histograms have 2 columns:

1. Number of the cell
2. Number of macro-particles in this cell.

**histogram-output**, *multichoice* (""): Output of various beam properties in the histogram form

Available options, corresponding output quantities, keys determining histogram intervals, and corresponding file names are:

'r':	transverse momentum, $p_{br}$ or $p_{bx}$ , <b>beam-pr-scale</b> ,	( <code>dr*****.png</code> ) ( <code>dr*****.swp</code> )
'z':	longitudinal momentum, $p_{bz}$ , <b>beam-pz-scale</b> ,	( <code>dz*****.png</code> ) ( <code>dz*****.swp</code> )
'M':	angular or $y$ -momentum, $M_b$ or $p_{by}$ , <b>beam-a-m-scale</b> ,	( <code>dm*****.png</code> ) ( <code>dm*****.swp</code> )
'a':	angle between the particle momentum and $z$ -axis, <b>beam-angle-scale</b> ,	( <code>da*****.png</code> ) ( <code>da*****.swp</code> )

**histogram-output-accel**, *multichoice* (""): Output of histograms for accelerated particles only

The same as **histogram output**, but only beam macro-particles with  $z$ -momentum greater than **output-reference-energy** are taken into account. Available options, corresponding output quantities, keys determining histogram intervals, and corresponding file names are:

'r':	transverse momentum, $p_{br}$ or $p_{bx}$ , <b>beam-pr-scale</b> ,	( <code>dR*****.png</code> ) ( <code>dR*****.swp</code> )
'z':	longitudinal momentum, $p_{bz}$ , <b>beam-pz-scale</b> ,	( <code>dZ*****.png</code> ) ( <code>dZ*****.swp</code> )
'M':	angular or $y$ -momentum, $M_b$ or $p_{by}$ , <b>beam-a-m-scale</b> ,	( <code>dM*****.png</code> ) ( <code>dM*****.swp</code> )
'a':	angle between the particle momentum and $z$ -axis, <b>beam-angle-scale</b> ,	( <code>dA*****.png</code> ) ( <code>dA*****.swp</code> )

**histogram-type**, *choice* (y): Mode of the histogram output:

- 'y' or 'pictures': Pictures only,
- 'n' or 'data': Data files only,
- 'F' or 'both': Both pictures and data files.

**histogram-bins**, *integer* (300): Number of histogram bins ( $\leq 3000$ )

**beam-angle-scale**, *float* (0.1): Axis size for the histogram of beam angular distribution

The axis is from zero to this value.

### 5.5.5 Trajectories of plasma particles

These diagnostics visualizes trajectories of plasma macro-particles in the simulation window, either as the picture (`pp*****.png`), or as the data file (`*****.pls`), where (\*\*\*\*\*) is the time. The diagnostics works for kinetic plasma models only. Each line of the data file contains 9 values corresponding to one position of one macro-particle:

1. Current  $\xi$ -coordinate
2. Number of the macro-particle
3. Transverse coordinate ( $r$  or  $x$ )
- 4-6.  $r$ -,  $\varphi$ -, and  $z$ -components of the particle momentum
7. Mass of the macro-particle
8. Charge-to-mass ratio
9. Relativistic factor

**trajectories-draw**, *choice* (n): Output mode for trajectories of plasma particles:  
 ‘n’ or ‘n’: No output.  
 ‘y’ or ‘y’: Picture mode. Trajectories of chosen plasma macro-particles on  $(r, \xi)$  plane are shown by dotted lines with a specified interval in  $\xi$  between dots. The dot color represents the gamma-factor  $\gamma$  of the particle.  
 ‘Y’ or ‘Y’: Data mode. Information about chosen macro-particles is written to the data file with a specified interval in  $\xi$  between successive outputs.  
 ‘F’ or ‘F’: Both picture and data file.

**trajectories-each**, *integer* (10): Fraction of plasma macro-particles to output  
 A trajectory of a plasma macro-particle can be drawn or written if the ordinal number of the macro-particle is a multiple of this number.

**trajectories-spacing**, *integer* (10): Spacing in  $\xi$  for consecutive output of macro-particle parameters  
 This interval is measured in units of the grid step  $\Delta\xi$ .

**trajectories-min-energy**, *float* (1): Minimum relativistic factor  $\gamma$  for the particle to be drawn,  $\gamma_{\min}$   
 Particles with smaller relativistic factors are ignored. This is useful if you want to exclude weakly perturbed plasma regions from the output.

**trajectories-energy-step**, *float* (0.5): Color step for visualization of particle energies,  $C_{\text{step}}$   
 The dot color in pictures is determined by  $N_{\text{col}} = (\gamma - \gamma_{\min})/C_{\text{step}}$ . The order of colors is the same as for the default palette of colored maps.

### 5.5.6 Detailed (substepped) plasma response

These keys control output of various plasma characteristics at a reduced  $\xi$ -step.

**substepping-output-depth**, *integer* (4): Substepping depth for the output (0..4)  
 How deep sub-stepping in  $\xi$  is included into the following output.

**substepping-output-map**, *y/n* (n): Substepped output of functions of  $(r, \xi)$   
 If this option is enabled, then quantities chosen in **colormaps-subwindow** are also output with a reduced  $\xi$ -step into data files named (??\*\*\*\*\*.det), where (??) is the two-character quantity abbreviation, and (\*\*\*\*\*) is the time. The subwindow for the output is the same as for subwindow colored maps. No merging or smoothing of values is made. If this diagnostics is on, the file (xi\*\*\*\*\*.det) is also created, which has the same number of lines as data files and contains corresponding values of  $\xi$ , one value per line.

**substepping-output-f(xi)**, *y/n* (n): Substepped output of functions of  $\xi$   
 If enabled, then quantities chosen in **f(xi)** are also output with a reduced  $\xi$ -step into data files named (u\*\*\*\*\*.det) and (v\*\*\*\*\*.det), where (\*\*\*\*\*) is the time. The file format is the same as for (u\*\*\*\*\*.swp) or (v\*\*\*\*\*.swp).

**substepping-output-particles**, *y/n* (n): Substepped output of plasma particles  
 If enabled, information about plasma macro-particles is written to data file at each  $\xi$ -step or sub-step of the depth specified by **substepping-output-depth**. Particle filtering criteria imposed by **trajectories-each** and **trajectories-min-energy** are applicable.

**substepping-output-particles-area**, *choice* (f): Area for substepped output of plasma particles:  
 ‘n’ or ‘no’: The output window is the same as for colored maps. Particle parameters are written to file (\*\*\*\*\*.pls), where (\*\*\*\*\*) is the time.  
 ‘y’ or ‘yes’: The output window is the same as subwindow for colored maps. Particle parameters are written to file (s\*\*\*\*\*.pls), where (\*\*\*\*\*) is the time. The file format is the same as for (\*\*\*\*\*.pls).

## 5.6 Saving run state

The following options control periodical savings of the run state.

**saving-period**, *float* (1000): Time interval between run saves  
 The run is saved if the time  $t$  differs from a multiple of the saving period by less than  $\Delta t/2$ . Saving is made after completion of the  $\xi$ -cycle and drawing all pictures.



**save-beam**, *y/n* (n): Saving the beam state

The state of the beam is saved in binary file (**tb\*\*\*\*\*.swp**). The format of the file is the same as that of (**beamfile.bin**). Option works for non-rigid beams only. Independently on this key, the beam state is also saved to (**beamfile.bin**) on reaching the time limit  $t_{\max}$  or after manual interruption of the run (by the dot), so that the run can be continued without data loss.

**save-plasma**, *y/n* (n): Saving the plasma state

The plasma state at the end of the simulation window is saved to files (**pl\*\*\*\*\*.swp**) (plasma particles) and (**fl\*\*\*\*\*.swp**) (electric and magnetic fields). Formats of the files are the same as those of (**plasma.bin**) and (**fields.bin**). Option works for kinetic plasma models only. Independently on this key, in regimes with plasma continuation the final plasma state is saved to (**plasma.bin**) and (**fields.bin**) on reaching the time limit  $t_{\max}$  or after manual interruption of the run (by the dot), so that the run can be continued without data loss.

The run can be resumed from any saved time \*\*\*\*\* by changing the content of (**beamfile.bit**) to \*\*\*\*\* , renaming (**tb\*\*\*\*\*.swp**) to (**beamfile.bin**), and, if necessary, renaming (**pl\*\*\*\*\*.swp**) to (**plasma.bin**) and (**fl\*\*\*\*\*.swp**) to (**fields.bin**). Any changes in code options (e.g., changes of diagnostics) will take effect on the resumed run.

## 5.7 Logging preferences

**log-stdout-level**, *choice* (d): Verbosity level for printing messages

**log-file-level**, *choice* (w): Verbosity level for logging to a file:

“e” or “error”: Output error messages only

“w” or “warning”: Output error and warning messages

“i” or “info”: Output error, warning and info messages

“d” or “debug”: Output error, warning, info and debug messages

**log-filename**, *string* (“lcode.log”): Name of the file to log to

**log-file-clean**, *y/n* (y): Remove empty log file

If enabled, then empty log files are automatically removed at the end of the run, if any.

**save-config**, *y/n* (y): Enable saving config to a file

**save-config-filename**, *string* (“lcode.runas.cfg”): Name of the file to save the current config to

This config includes the final version of all parameters, including those set by command-line switches. This may become useful in later debugging, for exact replication of the execution process, and especially after config recovery, as it reflects the automatic changes introduced by the recovery process.

## 5.8 UNTESTED parameters

**write-beam-particles-q-m-from**, *float* (0): Filter particles by their q/m ratio

Sets the lower limit for the q/m parameter of a particle to get it written to the (**partic.swp**). Setting those two parameters to the same value disables this filtering.

**write-beam-particles-q-m-to**, *float* (0): Filter particles by their q/m ratio

Sets the higher limit for the q/m parameter of a particle to get it written to the (**partic.swp**). Setting those two parameters to the same value disables this filtering.

## 6 Initial beam shape

There are two ways to define the initial distribution of beam particles in the six-dimensional phase space.

The first way is to specify all necessary parameters for each beam macro-particle. For this, information about the macro-particles must be written to the binary file (**beamfile.bin**), e.g., by an external program. The initial time must be written to the text file (**beamfile.bit**). The code will read this information as the initial state of the beam.

The second way is to specify macroscopic parameters for individual beam segments. A segment is a beam piece of variable length  $l_s$ . The segments follow one by one starting from the beginning of simulation window (at  $\xi = 0$ ). A description of a segment has the format

parameter1=value1, parameter2=value2, ..., parameterN=valueN

If a parameter definition not followed by comma, then it is considered as the last one in the description of this segment. A description must contain at least one parameter defined. Segment descriptions follow one by one in **beam-profile** or in a separate file which **beam-profile** refers to.

Numerical parameters can be specified in scientific notation (like **1e6**) and postfixed with **PI** or **Pi** which acts like multiplication by 3.1415926.

If a parameter is not explicitly defined in the segment description, then it is assigned the default value. The initial default values are given in the following description. To modify default values for all subsequent segments, a special line prefixed with '**default:**' can be used. This line has the same syntax as segment descriptions, but does not define any segment and only changes the default values of some parameters. Re-definition of default values can be made several times.

## 6.1 Segment parameters

**length**, *float* (2PI): Length of the segment,  $l_s$

**ampl**, *float* (0.5): Maximum current in the segment,  $I_a$   
 Maximum current for this segment in units of **beam-current**  $I_{b0}$ .

**xishape**, *choice* (c): Dependence of the beam current on the longitudinal coordinate,  $I_b(\xi)$   
 In the following variants, the formulae are applicable in the interval  $l_s > \delta\xi > 0$ , where  $\delta\xi = \xi_s - \xi$ , and the segment starts at  $\xi_s < 0$ .

'c' or 'cos': Shifted cosine shape,	$I_b(\xi) = 0.5 I_a I_{b0} (1 - \cos(2\pi\delta\xi/l_s)).$
't' or 't': Linear rise of the current,	$I_b(\xi) = I_a I_{b0} \delta\xi/l_s.$
'T' or 'T': Linear decrease of the current,	$I_b(\xi) = I_a I_{b0} (1 - \delta\xi/l_s).$
'l' or 'l': Constant current,	$I_b(\xi) = I_a I_{b0}.$
'h' or 'half-cos': Rising half-period of the shifted cosine,	$I_b(\xi) = 0.5 I_a I_{b0} (1 - \cos(\pi\delta\xi/l_s)).$
'b' or 'b': Decreasing half-period of the shifted cosine,	$I_b(\xi) = 0.5 I_a I_{b0} (1 + \cos(\pi\delta\xi/l_s)).$
'g' or 'g': Decreasing part of Gaussian function with $\sigma_z = l_s/6$ ,	$I_b(\xi) = 0.5 I_a I_{b0} \exp(-18 \delta\xi^2/l_s^2).$

**radius**, *float* (1): Radius of the beam segment,  $\sigma_r$

**energy**, *float* (1000): Basic longitudinal momentum of beam particles,  $p_{b0}$

**vshift**, *float* (0): Transverse displacement of the segment (used in plane geometry only),  $x_0$

**rshape**, *choice* (g): Initial distribution of beam particles in the transverse phase space

Available options:

‘g’ or ‘gaussian’: Random distribution with the probability density

$$f_{\perp}(r_b, p_{br}, p_{b\varphi}) = \frac{r_b}{2\pi\sigma_r^2\alpha_b^2 p_{b0}^2} \exp\left(-\frac{r_b^2}{2\sigma_r^2} - \frac{p_{br}^2 + p_{b\varphi}^2}{2\alpha_b^2 p_{b0}^2}\right), \quad (\text{axisymmetric beam}),$$

$$f_{\perp}(x_b, p_{bx}, p_{by}) = \frac{1}{(2\pi)^3/2\sigma_r\alpha_b^2 p_{b0}^2} \exp\left(-\frac{(x_b - x_0)^2}{2\sigma_r^2} - \frac{p_{bx}^2 + p_{by}^2}{2\alpha_b^2 p_{b0}^2}\right), \quad (\text{plane geometry}).$$

Here  $\alpha_b$  is the angular spread of the beam slice, and  $x_b$  is measured from the central plane.

‘r’ or ‘regular’: The distribution with the same probability density as for ‘gaussian’, but with a regular distribution of beam particles.

‘f’ or ‘flattop’: Random distribution of particles with the “flattop” probability density in the 4-dimensional phase space,

$$f_{4d}(x_b, y_b, p_{bx}, p_{by}) = \frac{1}{\pi^2\sigma_r^2\alpha_b^2 p_{b0}^2}, \quad \text{if } \left(\frac{(x_b - x_0)^2}{\sigma_r^2} + \frac{p_{bx}^2}{\alpha_b^2 p_{b0}^2}\right) < 1 \quad \text{and} \quad \left(\frac{y_b^2}{\sigma_r^2} + \frac{p_{by}^2}{\alpha_b^2 p_{b0}^2}\right) < 1;$$

$$f_{4d}(x_b, y_b, p_{bx}, p_{by}) = 0, \quad \text{otherwise,}$$

in which one spatial coordinate is ignored for each particle:

$$r_b = \sqrt{x_b^2 + y_b^2}, \quad p_{br} = \frac{x_b p_{bx} + y_b p_{by}}{r_b}, \quad p_{b\varphi} = \frac{y_b p_{bx} - x_b p_{by}}{r_b}, \quad \varphi_b \text{ ignored, } (\text{axisymmetric}),$$

$$y_b \text{ ignored, } (\text{plane}).$$

**angspread**, *float* (1e-5): Maximum angular spread in the segment,  $\alpha_0$

**angshape**, *choice* (l): Dependence of the angular spread on the longitudinal coordinate,  $\alpha_b(\xi)$

In the following variants, the formulae are applicable in the interval  $l_s > \delta\xi > 0$ , where  $\delta\xi = \xi_s - \xi$ , and the segment starts at  $\xi_s < 0$ .

‘c’ or ‘cos’: Shifted cosine,	$\alpha_b(\xi) = 0.5 \alpha_0 (1 - \cos(2\pi\delta\xi/l_s)).$
‘t’ or ‘t’: Linear rise,	$\alpha_b(\xi) = \alpha_0 \delta\xi/l_s.$
‘T’ or ‘T’: Linear decrease,	$\alpha_b(\xi) = \alpha_0 (1 - \delta\xi/l_s).$
‘l’ or ‘l’: Constant,	$\alpha_b(\xi) = \alpha_0.$
‘h’ or ‘half-cos’: Rising half-period of the shifted cosine,	$\alpha_b(\xi) = 0.5 \alpha_0 (1 - \cos(\pi\delta\xi/l_s)).$
‘b’ or ‘b’: Decreasing half-period of the shifted cosine,	$\alpha_b(\xi) = 0.5 \alpha_0 (1 + \cos(\pi\delta\xi/l_s)).$
‘g’ or ‘g’: Decreasing part of Gaussian function,	$\alpha_b(\xi) = 0.5 \alpha_0 \exp(-18 \delta\xi^2/l_s^2).$

**espread**, *float* (0): Auxiliary value of the longitudinal momentum,  $p_a$

**eshape**, *choice* (m): Longitudinal momentum distribution of beam particles,  $f_{\parallel}(p_{bz})$

Available options:

‘m’ or ‘monoenergetic’: Monoenergetic,	$f_{\parallel}(p_{bz}) = \delta(p_{bz} - p_{b0}).$
‘u’ or ‘uniform’: Uniform over the interval,	$f_{\parallel}(p_{bz}) = (p_{b0} - p_a)^{-1}$ if $p_{b0} > p_{bz} > p_a,$ $f_{\parallel}(p_{bz}) = 0$ otherwise.
‘l’ or ‘linear’: Linear energy growth from $p_a$ to $p_{b0}$ ,	$f_{\parallel}(p_{bz}) = \delta(p_{bz} - p_{b0} \delta\xi/l_s - p_a(1 - \delta\xi/l_s)).$
‘g’ or ‘gaussian’: Gaussian distribution,	$f_{\parallel}(p_{bz}) = \frac{e^{-(p_{bz} - p_{b0})^2/(2p_a^2)}}{\sqrt{2\pi}p_a}$
‘N’ or ‘N’: $N$ monoenergetic fractions ( $N$ is a digit, 2 ... 9),	

$$f_{\parallel}(p_{bz}) = \sum_{i=0}^{N-1} \delta(p_{bz} - p_i), \quad p_i = p_a + \frac{i}{N-1}(p_{b0} - p_a)$$

**m/q**, *float* (1): Absolute value of mass-to-charge ratio, compared to the electron

## 6.2 Example of beam profile

```
default:  angspread=3e-3, energy=2000
xishape=cos, ampl=1, length=6PI
xishape=len, ampl=0, length=1e6
```

The first line redefines the default values for three options.

The second line defines the beam segment of the length  $6\pi$ , current  $I_b(\xi) = 0.5 I_{b0}(1 - \cos(|\xi|/3))$ , and Gaussian transverse distribution (hardcoded default value) with  $\sigma_r = 1$  (hardcoded default value) and  $\alpha_b = 0.003$  (hardcoded default  $\xi$ -dependence, new default value). All particle in the segment have the same longitudinal momentum (hardcoded default distribution) that equals 2000 (new default value). Particles are either electrons or positrons, depending on the sign of  $I_{b0}$ .

The second line defines the space extending from  $\xi = -6\pi$  to  $\xi = -1000000 - 6\pi$  with zero current and thus no particles.

## 7 Format of other input and output files

### 7.1 beamfile.bin

This file contains information about beam macro-particles in the binary form. For each particle, 8 values are written, 8 bytes each (type double). These values are:

1. Longitudinal position,  $\xi_b$ .
2. Transverse position,  $r_b$  (cylindrical geometry) or  $x_b$  (plane geometry).
3. Longitudinal momentum,  $p_{bz}$ .
4. Transverse momentum,  $p_{br}$  (cylindrical geometry) or  $p_{bx}$  (plane geometry).
5. Angular momentum  $M_b$  (cylindrical geometry) or third momentum component  $p_{by}$  (plane geometry).
6. Absolute value of the charge-to-mass ratio, compared to electron.
7. Current carried by the macro-particle, in units of  $mc^3/e$ .
8. Ordinal number of the macro-particle.

### 7.2 beamfile.bit

Contains the current time and serves as the indicator of the continuation mode. If this file is present, the code tries to continue a previous run. If you want to start the run from zero time, remove (`beamfile.bit`) from the working folder.

### 7.3 plasma.bin

File (`plasma.bin`) contains information about plasma macro-particles (one line for one macro-particle). It has 6 columns:

1. Transverse coordinate of the macro-particle ( $r$  or  $x$ ) **counted from the bottom of the simulation window**.
2. First component of the transverse momentum ( $p_r$  or  $p_x$ ) **of the macro-particle** (not of a single electron or ion). It equals the particle velocity times relativistic factor times “mass” of the macro-particle.
3. Second component of the transverse momentum ( $p_\varphi$  or  $p_y$ ) of the macro-particle.
4. Longitudinal momentum ( $p_z$ ) of the macro-particle.
5. “Mass” of the macro-particle. For uniform electrons of the unit density (i.e., of the density  $n_0$ ), the “mass” is just the **cross-section** corresponding to this macro-particle (it is a ring in the cylindrical geometry or a rectangle in the plane geometry). For heavier particles, the “mass” is correspondingly greater.
6. “Charge” of the macro-particle. It is the “mass” times the real charge-to-mass ratio for particles of this sort.

When the run is normally terminated or interrupted in the regimes of a long plasma, the final parameters of all macro-particles are automatically saved to (`plasma.bin`). If the run is then continued, the initial plasma parameters are taken from (`plasma.bin`). The same happens if a new run is started with an arbitrary initial plasma profile.

### 7.4 fields.bin

File (`fields.bin`) contains information about the fields and plasma currents. Each line in (`fields.bin`) corresponds to a grid point. Number of lines is  $N_r + 1$  (number of grid steps plus one). Each line contains 8

values:

$E_r, E_\varphi, E_z, B_\varphi, B_z, p_r, p_\varphi, p_z$  for cylindrical geometry and fluid plasma model,  
 $E_x, E_y, E_z, B_y, B_z, p_x, p_y, p_z$  for plane geometry and fluid plasma model,  
 $E_r, E_\varphi, E_z, B_\varphi, B_z, j_r, j_\varphi, j_z$  for cylindrical geometry and kinetic plasma model.  
 $E_x, E_y, E_z, B_y, B_z, j_x, j_y, j_z$  for plane geometry and kinetic plasma model.

Here  $p_\alpha$  are components of the electron momentum;  $j_\alpha$  are components of the total current density.

When the run is normally terminated or interrupted in the regimes of a long plasma, the final fields are automatically saved to (`fields.bin`). For kinetic plasma models, the run can be then continued with the initial fields taken from (`plasma.bin`). The same happens if a new run is started with an arbitrary initial plasma profile.

## 7.5 `elocf.dat`, `glocf.dat`, `emaxf.dat`, `gmaxf.dat`

Output of wakefield extremum values (section 5.4):

(`elocf.dat`) — local extrema of the on-axis electric field  $E_z$ ,  
(`glocf.dat`) — local extrema of the on-axis wakefield potential  $\Phi$ ,  
(`emaxf.dat`) — absolute extrema of the on-axis electric field  $E_z$ ,  
(`gmaxf.dat`) — absolute extrema of the on-axis wakefield potential  $\Phi$ .

If enabled, a line of 5 values is appended to the corresponding file immediately after an extremum is found:

1. Current time  $t$
2. The maximum value of the quantity
3.  $\xi$ -coordinate of this maximum
4. The minimum value of the quantity (earlier found in case of local extrema)
5.  $\xi$ -coordinate of this minimum

## 7.6 `hystog.dat`

Normalization of the output histograms (section 5.5.4). The line of 3 values is appended to file (`hystog.dat`) each time a histogram picture is produced. These values are:

1. Time
2. Histogram type (second character of the filename)
3. Height of the highest column.

## 7.7 `n1.dat`

Output of the number of the beam macro-particles (section 5.5.2). If `output-survival-xi` is enabled, then a line of 7 values is appended to file (`n1.dat`) at each time step:

1. Time  $t$ ,
- 2-7. Number of survived beam macro-particles with  $\xi_b > \xi_{si}$

The information is written by one value immediately after passing  $\xi_{si}$ .

## 7.8 `tt*.dat`

Information about beam slices (section 5.5.2). If `output-slices-xi` is enabled, characteristics of 6 beam slices are appended to files (`tt*.dat`), where (\*) is a cross-section number (1...6). When the slice is passed, 6 values are appended to the corresponding file:

1. Time  $t$ ,
2. On-axis electric field  $E_z$  at this  $\xi$ ,
3. Emittance  $\epsilon$  of the slice,
4. Beam transverse displacement  $X_{b0}$  (is nonzero only in plane geometry),
5. Beam radius  $R_b$  or half-width  $X_b$ ,
6. Number of beam macro-particles in this layer.

## 7.9 `r*****.det`

Auxiliary information for subwindow diagnostics (section 5.5.6). If any sub-window diagnostics is on, then 3 values are written (not appended) to file (`f*****.det`), where (\*\*\*\*\*) is the time:

1. Transverse grid step.
2. Lower boundary of the sub-window.
3. Number of  $\xi$ -steps inside the sub-window.

## 7.10 **u\*\*\*\*\*.det, v\*\*\*\*\*.det, u\*\*\*\*\*.swp, v\*\*\*\*\*.swp**

Information about on-axis and off-axis values of various quantities, either in the sub-window (**?\*\*\*\*\*.det**) or in the full window (**?\*\*\*\*\*.swp**). The format is described in section 5.5.2.

## 7.11 **xi\*\*\*\*\*.det**

Coordinates of sub-grid layers in the subwindow, one value per line. The file is created, if **substepping-output-map** diagnostics is on. See section 5.5.6 for details.

## 7.12 **d?\*\*\*\*\*.swp**

Numerical data for histograms (section 5.5.4). The files have 2 columns:

1. Number of the cell
2. Number of macro-particles in this cell.

## 7.13 **??\*\*\*\*\*.swp, ??\*\*\*\*\*.w.swp, ??\*\*\*\*\*.det**

Two-dimensional arrays of values (sections 5.5.1, 5.5.6). In the files, one row is for one  $\xi$ -layer or sub-layer. Columns correspond to different transverse coordinates.

## 7.14 **captured.pls**

File (**captured.pls**) contains information about captured plasma macro-particles (section 5.4).

## 7.15 **\*\*\*\*\*.pls, s\*\*\*\*\*.pls**

File (**\*\*\*\*\*.pls**) contains information about plasma macro-particles (section 5.5.5). File (**s\*\*\*\*\*.pls**) contains information about plasma macro-particles in the subwindow (section 5.5.6).

## 7.16 **fl\*\*\*\*\*.pls, pl\*\*\*\*\*.swp, tb\*\*\*\*\*.swp**

Files (**fl\*\*\*\*\*.pls**), (**pl\*\*\*\*\*.pls**), (**tb\*\*\*\*\*.pls**) are snapshots for (**fields.bin**), (**plasma.bin**), (**beamfile.bin**) at a certain time step.

## 7.17 **partic.swp**

File (**partic.swp**) contains parameters of separate beam macro-particles (section 5.4). The same characteristics of beam particles can be also extracted from (**beamfile.bin**).

## 7.18 **Temporary files**

Files (**aver.swp**), (**digcol.swp**), (**digcol2.swp**), (**bfile.swp**), (**info.swp**), (**??\*\*\*\*\*.swp**) are temporary files and are always deleted in normal regimes