LiTrack: A FAST LONGITUDINAL PHASE SPACE TRACKING CODE WITH GRAPHICAL USER INTERFACE [*]

K.L.F. Bane, P. Emma [‡], *SLAC*, Stanford, CA, USA 94309

Abstract

Linac-based light sources and linear colliders typically apply longitudinal phase space manipulations in their design, including electron bunch compression and wakefield-induced energy spread control. Several computer codes handle such issues, but most also require detailed information on the transverse focusing lattice. In fact, in most linear accelerators, the transverse distributions do not significantly affect the longitudinal, and can be ignored initially. This allows the use of a fast 2D code to study longitudinal aspects without time-consuming considerations of the transverse focusing. *LiTrack* is based on a 15-year old code (same name) originally written by one of us (KB), which is now a *Matlab* [1] code with additional features, such as graphical user interface, prompt output plotting, and functional call within a script. This single-bunch tracking code includes RF acceleration, bunch compression to 3rd order, geometric and resistive short-range wakefields, aperture limits, synchrotron radiation, and flexible output plotting. The code was used to design both the LCLS [2] and the SPPS [3] projects at SLAC and typically runs 10^5 particles in $<$ 1 minute. We describe the features, show some examples, and provide free access to the code.

INTRODUCTION

The initial design of linac-based accelerators often places the highest priority on precise manipulation of longitudinal phase space. For example, in a linac-based FEL, the goal is to produce a high peak current (typically several kA) and, simultaneously, a small relative energy spread ($\ll 0.1\%$). For such a task, a computer program is needed that can track longitudinal phase space accurately, including effects such as nonlinearities and wakefields. Existing 6D particle tracking codes can be slow and require that a transverse focusing lattice be in place before tracking can begin. This makes it difficult, in design studies, to try bold variations in the longitudinal optics without also wasting time adjusting the transverse focusing. Since in most accelerators longitudinal phase space can be studied independently of transverse phase space, there can be a major advantage, in such cases, in using a program that tracks longitudinal coordinates only.

LiTrack is such a program. It is a macro-particle tracking program, that follows longitudinal position z (bunch head at $z < 0$) and relative energy deviation $\delta(\equiv \Delta E/E_0$; E_0 is reference energy) of the particles. It can be used to track ultra-relativistic beams, where space charge can be ignored, through linacs and transport lines, where initial conditions and the beamline elements determine the final phase space. *LiTrack* handles linac acceleration and bunch compression. Acceleration is applied as a sinusoidal variation, without approximation. Bend systems are described simply by their 1st, 2nd, and 3rd-order path length *vs.* energy coefficients (R_{56}, T_{566}, and U_{5666}) and the nominal energy. Wakefields (of accelerating structures or the resistive wall wake of a beam pipe) are included by convolving a point charge wake with the evolving temporal (z) distribution. Note that in a linac region, since particles are frozen in z, it is necessary to apply wakefield only once, at the end of the region.

With such a simple treatment it is possible to represent, for example, the full LCLS [2] longitudinal design in as few as 10 lines of input, with each line containing 6 or fewer parameters. Note that this 1.8-km long X-ray FEL comprises 1-km of S-band linac RF, a short section of X-band RF, two bunch compressors, and a final dog-leg bend.

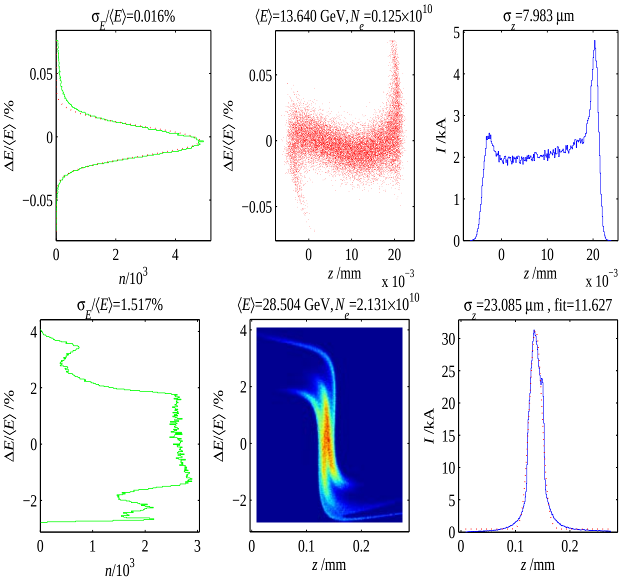


Figure 1: *LiTrack* output plots for LCLS (top) and SPPS (bottom) with energy profile (left), longitudinal phase space (center), and bunch length profile (right). Optional color contour is used at bottom and scatter plot at top.

LiTrack has been and still is being used extensively for analysis on several different projects at SLAC, such as the SLC, the LCLS, the SPPS [3], and the plasma-wakefield acceleration experiment E164X [4, 5]. Example *LiTrack* output plots for the LCLS (at low charge of 0.2 nC) and the SPPS are shown in Fig. 1. An SLC comparison of measurement with *LiTrack* simulations is given in Fig. 2 [6].

[*] Work supported by US Department of Energy contract DE-AC02-76SF00515.

[‡] Emma@SLAC.Stanford.edu

0-7803-8859-3/05/$20.00 ©2005 IEEE

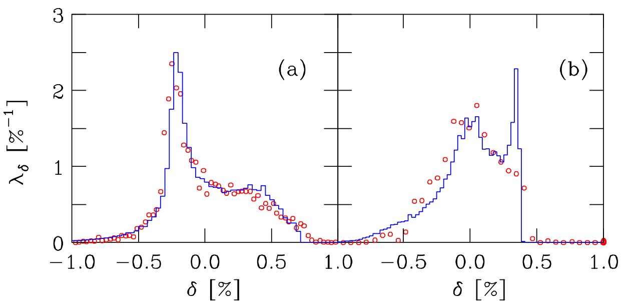


Figure 2: Measured (circles) and simulated (histogram) energy spectrum at the end of the SLC linac, for RF phase settings 7.5° (a) and 2.8° (b) ahead of crest; bunch charge is 3 nC (from Ref. [6]).

TRACKING CONTROL

An ASCII *Matlab* M-file is the input to the (non-GUI) program, including the initial conditions such as energy, charge, bunch length, and energy spread. The beamline transport is represented by a simple numerical table (matrix). The first column is a function-code defining the calculation to be done, such as a linac section with wakefield, a bunch compressor, or fixed aperture. Table 1 lists the most commonly used function-codes.

Table 1: List of function-codes and their parameters. All units are *mks* except energy, voltage, and phase, which are GeV, GV, and degrees, respectively; $\delta \equiv \Delta E/E_0$. Blank parameter slots are not used by the code.

descrip.	code	p_1	p_2	p_3	p_4	p_5
mark	1					
dump	2					
compr.	6	R_{56}	T_{566}	E_0	U_{5666}	
chicane	7	R_{56}	E_0			
linac	10	E_{final}	ϕ	λ	i_{wake}	L
linac	11	V_0	ϕ	λ	i_{wake}	L
rw-wake	15	r	L	σ_c	τ	1/0
ISR	22	σ_δ				
E-cuts	26	δ_{min}	δ_{max}			
z-cuts	36	z_{min}	z_{max}			
STOP	99					

The 1-code simply allows a plot of the present tracking, taking no other action. The 2-code dumps the results into an ASCII 2-column file. The 6-code is a general bunch compressor with 1st, 2nd, and 3rd order compression factors and nominal beam energy. The 7-code is a chicane, identical to a 6-code except that the 2nd and 3rd order terms are implicit assuming the small-angle approximation $T_{566} \approx -3R_{56}/2$ and $U_{5666} \approx 2R_{56}$. The 10-code is a linac section accelerating to E_{final} at an RF phase ϕ ($\phi = 0$ is crest phase and $\phi < 0$ puts bunch ahead of crest), wavelength λ, and linac length L. The pointer i_{wake} is an integer as a row-pointer to an ASCII array of wakefield file names, included in the input M-file. The 11-code is a linac section, like the 10-code, except the crest voltage is given rather than final energy. This is more useful for error testing, since the 10-code always produces E_{final}, whereas the 11-code's final energy is phase, voltage, and wakefield dependent. The 15-code is a resistive-wall wakefield, including AC conductivity, with half-aperture r, pipe length L, (DC) conductivity σ_c, material relaxation time τ, and a flag (=0 or =1) for round or flat pipe, respectively [7]. The 22-code adds random energy spread to the beam due, for example, to incoherent synchrotron radiation (ISR) in a bend magnet. The 26-code is a fixed energy aperture where all particles outside the limits $\delta_{min} < \delta < \delta_{max}$ are lost. The 36-code is the same, but for longitudinal limits. The 99-code, wherever placed, signals the end of the beam line. Any code given a negative sign will produce a full plot at that point (*e.g.*, as shown in Fig. 1). The 99-code always generates a plot, independent of sign.

LONGITUDINAL WAKEFIELDS

The longitudinal short-range wakefield of a periodic RF structure induces an energy variation, or voltage, $V(z)$, along the bunch calculated with the convolution integral

$$V(z) = -eNL\int_{-\infty}^{z} W(z-z')f(z')dz', \qquad (1)$$

with e the electron charge, N the number of electrons in the bunch, L the length of the structure, W the point-charge wake, and $f(z)$ the normalized line charge distribution.

The distribution function $f(z)$, is calculated internally in *LiTrack* by binning the particles in z, with the number of bins chosen by the user. The point-charge wake function, $W(z-z')$, is provided by an external file, generated independently. The range of the wake is usually limited to a few centimeters, covering the short-range wake of a single bunch only. Linear interpolation within the bunch is used to apply the integrated wake to all particles smoothly.

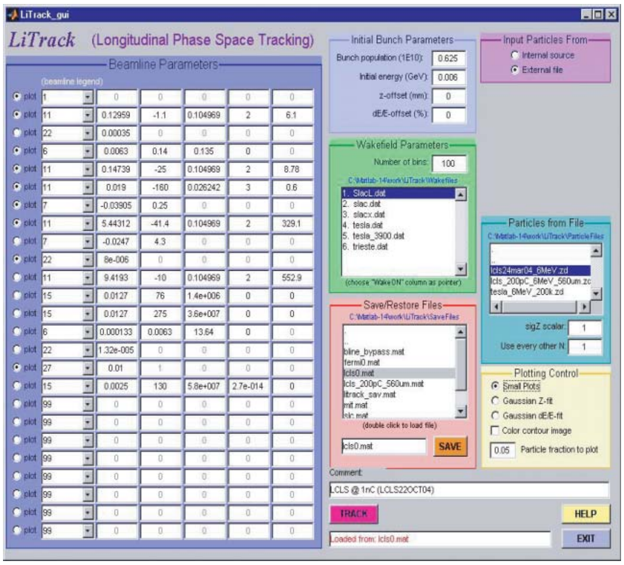
Multiple RF structure types can be represented in one input file by tagging each section with an appropriate integer pointer (i_{wake}) to its point-charge wake file name. In this way, the linac can contain, for example, S-band, L-band, and X-band RF structures all in one run, if necessary.

GRAPHICAL USER INTERFACE

A Graphical User Interface (GUI) is also available as a convenient control environment. Either the non-GUI or GUI version can be run. The GUI panel is shown in Fig. 3. The beamline input M-file described above is modified using a text editor, but the GUI version allows parameter adjustments on a simple user panel, without a text editor. The beamline input M-files are not used by the GUI; another format is used in this case. There are GUI sub-panels for:

Beamline Parameters

Each row is a beamline element with a function-code from a pull-down list, as defined in Table 1. In the GUI, the



LiTrack_gui

LiTrack (Longitudinal Phase Space Tracking)

Beamline Parameters

(beamline legend)

● plot	1	0	0	0	0	0
● plot	11	0.12959	-1.1	0.104969	2	6.1
○ plot	22	0.00035	0	0	0	0
● plot	6	0.0063	0.14	0.135	0	0
● plot	11	0.14739	-25	0.104969	2	8.78
● plot	11	0.019	-160	0.026242	3	0.6
● plot	7	-0.03905	0.25	0	0	0
● plot	11	5.44312	-41.4	0.104969	2	329.1
○ plot	7	-0.0247	4.3	0	0	0
● plot	22	8e-006	0	0	0	0
○ plot	11	9.4193	-10	0.104969	2	552.9
○ plot	15	0.0127	76	1.4e+006	0	0
○ plot	15	0.0127	275	3.6e+007	0	0
○ plot	6	0.000133	0.0063	13.64	0	0
○ plot	22	1.32e-005	0	0	0	0
● plot	27	0.01	1	0	0	0
○ plot	15	0.0025	130	5.8e+007	2.7e-014	0
○ plot	99	0	0	0	0	0
○ plot	99	0	0	0	0	0
○ plot	99	0	0	0	0	0
○ plot	99	0	0	0	0	0
○ plot	99	0	0	0	0	0
○ plot	99	0	0	0	0	0
○ plot	99	0	0	0	0	0
○ plot	99	0	0	0	0	0

Initial Bunch Parameters

Bunch population (1E10): 0.625

Initial energy (GeV): 0.006

z-offset (mm): 0

dE/E-offset (%): 0

Input Particles From

○ Internal source

● External file

Wakefield Parameters

Number of bins: 100

C:\Matlab-14\work\LiTrack\Wakefiles

1. SlacL.dat
2. slac.dat
3. slacx.dat
4. tesla.dat
5. tesla_3900.dat
6. trieste.dat

(choose "Wake ON" column as pointer)

Particles from File

C:\Matlab-14\work\LiTrack\ParticleFiles

.
..
lcls24mar04_6MeV.zd
lcls_200pC_6MeV_560um.zc
tesla_6MeV_200k.zd

sigZ scalar: 1

Use every other N: 1

Save/Restore Files

C:\Matlab-14\work\LiTrack\SaveFiles

.
..
bline_bypass.mat
fermi0.mat
lcls0.mat
lcls_200pC_560um.mat
litrack_sav.mat
mit.mat
slc.mat

(double click to load file)

lcls0.mat

Plotting Control

● Small Plots

○ Gaussian Z-fit

○ Gaussian dE/E-fit

☐ Color contour image

0.05 Particle fraction to plot

Comment:

LCLS @ 1nC (LCLS22OCT04)

Loaded from: lcls0.mat

Figure 3: *LiTrack* GUI panel.

function-code is always positive. A plot is requested with adjacent radio-button toggle (see Fig. 3). When a change is made to any row in this table, a brief help header appears at the top of the table describing each entry in that row and its units (for the function-code presently selected).

Initial Bunch Parameters

This sub-panel, near the top of the GUI, allows entry of the bunch population (units of 10^{10}), and the initial beam energy (in GeV) at the start of the tracking. Initial centroid offsets are also available here with "z-offset" as an initial effective 'timing' error (in mm) and "dE/E-offset" as an initial relative energy error (in %).

Wakefield Parameters

Available longitudinal point-charge wakefield files are listed here. The list simply displays the available files with assigned index number. This index should be entered in the 10 and 11-codes for the i_{wake} parameter (see Table 1). New wake files can be added to the directory shown here. The number of z-bins used in the convolution of Eq. (1), and the distribution plotting, is adjustable in this sub-panel.

Save/Restore Files

All of the parameters for a particular run can be saved on this sub-panel by typing a file name (*e.g.*, 'myrun.mat') and clicking the "SAVE" button. Any file can be restored by double-clicking the file name in the directory list.

Input Particles From

The initial particle coordinates can be generated internally by *LiTrack*, or input from an external file. The radio-buttons will toggle open either the sub-panel "Particles from File" or "Internal Particles" (not shown in Fig. 3).

Internal Particle With "Internal source" selected, a pull-down menu for 'Gaussian' or 'Uniform' profiles is available. Entry boxes also allow setting initial rms bunch length, rms relative energy spread, the number of macro-particles to track, and an asymmetry parameter, which adds skew to the Gaussian z-distribution.

Particles from File With "External file" selected, a list of available particle files is shown. Single-clicking allows the selection of any file. A factor "sigZ scalar" allows arbitrary scaling of the file's bunch length. For faster tracking the "Use every other N" box selects every other n^{th} particle (randomly) from the file (set $n = 1$ for all).

Plotting Control

Small plots (3 across) can be selected (as in Fig. 1), or 3-4 full plots over the entire figure. Gaussian fits for z and/or $\Delta E/E$ profiles can be toggled, and the fraction of particles in the scatter plot chosen (0 to 1). A color contour can be selected to replace the phase space scatter plot (see Fig. 1).

RUN *LITRACK* FROM A SCRIPT

LiTrack (non-GUI) can also be run from a *Matlab* script by calling it as a function. The call sequence is evident by typing 'help LiTrack' at the *Matlab* prompt. In this way, *Litrack* can be called repetitively while scanning or randomizing input parameters. An example is shown in Fig. 4 where the LCLS is tracked 1000 times with many beam and machine parameters (*e.g.*, RF phase and voltage) randomly varied in order to test final bunch arrival time stability.

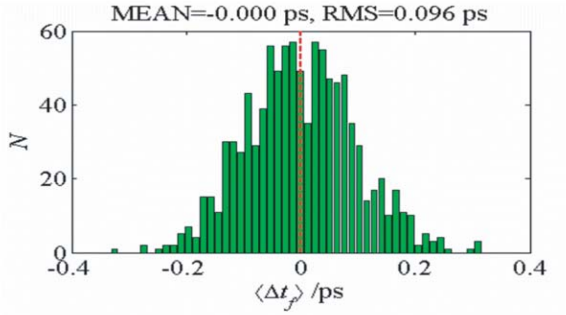


Figure 4: LCLS bunch arrival time jitter simulation.

DOWNLOAD *LITRACK*

LiTrack requires *Matlab*, and can be downloaded free at: http://www.slac.stanford.edu/~emma/codes.shtml, where further instructions are available.

REFERENCES

[1] http://www.mathworks.com/products/matlab/

[2] *LCLS CDR*, SLAC Report No. SLAC-R-593, 2002.

[3] P. Emma *et al.*, PAC'01, Chicago, IL, June 18-22, 2001.

[4] M. Hogan *et al.*, to be published in PRL.

[5] C. Barnes *et al.*, these PAC'05 proceedings.

[6] K.L.F. Bane *et al.*, PAC'99, New York, NY, March, 1999.

[7] K.L.F. Bane and G. Stupakov, these PAC'05 proceedings.

0-7803-8859-3/05/$20.00 ©2005 IEEE