

How to Use CSEC

M. Blaskiewicz

November 2006

Collider Accelerator Department
Brookhaven National Laboratory

U.S. Department of Energy

USDOE Office of Science (SC)

Notice: This technical note has been authored by employees of Brookhaven Science Associates, LLC under Contract No. DE-AC02-98CH10886 with the U.S. Department of Energy. The publisher by accepting the technical note for publication acknowledges that the United States Government retains a non-exclusive, paid-up, irrevocable, world-wide license to publish or reproduce the published form of this technical note, or allow others to do so, for United States Government purposes.

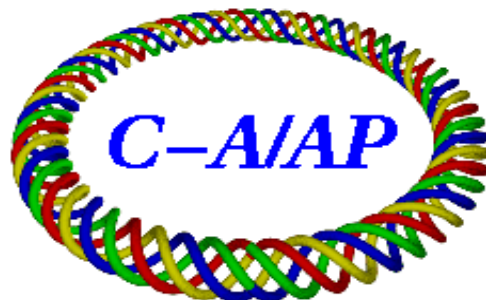
DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or any third party's use or the results of such use of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof or its contractors or subcontractors. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

C-A/AP/#260
November 2006

How To Use CSEC

M. Blaskiewicz and U. Iriso



**Collider-Accelerator Department
Brookhaven National Laboratory
Upton, NY 11973**

How To Use CSEC

M. Blaskiewicz and U. Iriso

November 2, 2006

Abstract

This document guides possible users of CSEC, a *Cylindrically Symmetric Electron Cloud* computer code that simulates the electron cloud build up. Input parameters, basic working principles, and output files are explained.

1 Introduction

Beam induced multipacting accelerates electrons (produced by field emission, photo-emission, residual gas ionization, etc) towards the system's chamber wall. Depending on the Secondary Electron Yield – SEY – of the wall surface (number of emitted electrons per impinging electron), the primary electrons produce secondary electrons when they hit the chamber wall, which in turn are accelerated by the beam electric field. If the appropriate conditions are met, the electron density inside the vacuum chamber increases exponentially creating the so-called *electron cloud*. The cloud growth is limited by its own electric fields, resulting in a configuration that varies periodically at the revolution frequency.

Since the 1990s, several computer simulation codes have been written to model the circumstances in which the build up of an electron cloud occurs [1]. This note describes how to use the *Cylindrically Symmetric Electron Cloud* (CSEC) computer simulation code, originally written for the PSR and SNS [2], and subsequently used for RHIC [3–5]. Physical descriptions of the code can be found in Refs. [2–5]. The phenomenon is quite sensitive to a host of accelerator parameters including beam bunch intensity, beam bunch spacing, beam dimensions, chamber geometry, and SEY.

The CSEC input parameters are separated in different input files, which are used by different executable programs. All programs are written in FORTRAN, and can be compiled using the Portland Group “pgf90” or the Intel “ifort” compilers. Table 1 lists the programs and goals, together with the input and output files associated to the executable files. These files can all be downloaded at:

<http://www.agsrhichome.bnl.gov/People/ubaldo/>

For all but the simplest bunch patterns CSEC requires the beam's bunch train to be read as an input file. There are several ways to create it. We provide here IRTRAIN. However, this part can be relatively straight forward and possible users might find more appropriate to write their own executable. Appendix B shows, for example, another executable that allows to create homogenous bunch patterns. The code SEY_SIM provides for a quick check of the secondary emission and reflection parameters used by CSEC.

Next, the input parameters associated to each input file and program are detailed. An example applied to the Relativistic Heavy Ion Collider (RHIC) case is also shown.

Table 1: List of CSEC programs to simulate an electron cloud build up. The common variables are listed in a separate file, called “ncsecz0h_common.f”.

Program	Goal	Input Files	Output Files
IRTRAIN	Create beam’s bunch pattern	btrain.info	ir.dat straight.dat
CSEC	Compute multipacting	ncsec.file ncsecz0_hist.in	ncsecz0.sum pot.dat hist.out fort.66
SEY_SIM	compute SEY	ncsecz0_hist.in	sey.sim

2 Create the beam’s bunch train – IRTRAIN

The first thing we need to do is to put the bunch train in a turn ready to be used by CSEC. This is done with IRTRAIN. Its goal is to develop arbitrarily inhomogenous bunch patterns, i.e. a bunch train in which bunches are separated by different spacings, as it is the case at RHIC. It is possible to simulate the situation at both the straight sections and the Interaction Regions of a collider.

2.1 Input file: “btrain.info”

This input file provides longitudinal information about the beam. It just consists on 12 parameters and was developed to reproduce the RHIC case of two uneven bunch trains colliding at any point of an Interaction Region.

1. “switch for IR section (kIR) - dist from IP (deltaz, in m)”
The first number should only be “1” if one wants to study the case in an Interaction Region (with two beams circulating in opposite directions). Otherwise, it is always “0”. If ‘kIR=1’, the parameter ‘deltaz’ indicates the distance (in m) from the Interaction Point where the multipacting study is focused.
2. “total collider length (circ, in m)”
Self explanatory.
3. “# posible. bchs, not counting abort gap limitations (nb)”
Self explanatory.
4. “bunch spacing (sb, in m)”
Self explanatory. In case of uneven bunch patterns, the minimum distance between two consecutive bunches is given here. The program considers that only multiples of the minimum bunch spacing are allowed.
5. “abort gap length (abgap, in m)”
Self explanatory.
6. “switch for arb. bunch patterns ”a la RHIC” (kEQSPB)”
This switches on (if ‘kEQSPB=1’) the possibility of uneven bunch patterns. That is, it tells to the program to read the bunch train input file (see next).
7. “name of input file”
This file states, out of the possible number of bunches in parameter ‘nb’ (see bullet # 3), which bunches are filled in. This input file finishes with a commented line that gives an “awk” command that can be used from the prompt of any Unix pc. The output of

this script provides the bunch pattern you want to study in a two column text file. The first column corresponds to the bunch number (up to the maximum possible bunches in a bunch train, 120 for the RHIC case), and the second column states whether the bunch is filled or not with 1 (filled) or 0 (empty).

8. “parabolic bunch profile switch, power”
The parabolic bunch profile is expressed as

$$N(t) = N_{\text{peak}} \left[1 - \frac{(t - t_0)^2}{bl^2} \right]^{\text{xorder}} \quad (1)$$

Since this is the only bunch profile developed up to date, only the parameters “1” with the option 'xorder=3' are allowed in this line.

9. “9: full parabolic bunch length (bl, in m) ”
Self explanatory. It corresponds to the head to tail bunch length (bl in Eq. 1).

10. “total # of slices/turn”
This number gives the time steps at which the programs IRTRAIN and CSEC recompute all electromagnetic forces. The bunch passage must be well resolved to properly compute all the forces. It is a compromise between the accuracy the user requires and the CPU speed the user disposes. A typical number for RHIC is 108000 for injection bunch lengths, around 312000 for transition bunch lengths.

2.2 Output file: “ir.dat” and/or “straight.dat”.

The output file “straight.dat” gives the normalized bunch intensity as a function of the angular phase (from 0 to 2π). This is the only output file if option 'kIR' is switched off.

If the option 'kIR' was switched on ('kIR=1'), the output file “ir.dat” is created with the bunch pattern that an observer at a distance 'deltaz' would see – see Ref. [5]. The program does not considers the opposite directions both beams travel.

3 Tracking the multipacting electrons – CSEC

CSEC assumes an electron cloud that is cylindrically symmetric within a round, straight vacuum chamber. The code simulation includes the transverse electric field of the beam and the electrons, transverse and longitudinal electron motion, and solenoid fields. The longitudinal electron motion is due solely to the fact that secondary electrons can have a longitudinal velocity component. Longitudinal electric fields, mainly due to the electrons for relativistic beams, are ignored. The common variables are listed in a separate file, called “ncsecz0h_common.f”.

As other electron cloud computer simulations [7], CSEC does not track single electrons, but electron “macroparticles”, which group between 10 and 10^5 electrons. This number changes depending on the balance about the accuracy the user requires, and the CPU speed the user disposes.

3.1 Input file 1: 'ncsecz0_hist.in'

The input file consists in 7 lines. Each line includes first several parameters, and next to them the parameter name in the program adopted in CSEC. Next, we detail the meaning of the parameter at each line with an example. The example corresponds to the input file

used to simulate the beam injection conditions at RHIC, shown later on in Section 4. All the parameters use MKS system of units unless otherwise stated.

1. *12.82e-6 16e-9 0 3 0.99876 0 trev,tbunch,fgap,xorder,beta,bsol*

- 'trev': revolution time
- 'tbunch': bunch length (head to tail) in units of time. This parameter is active only when no bunch train input file is used.
- 'fgap': fraction of beam in the gap. This is a simple parameterization for debunched beam. This parameter is active only when no bunch train input file is used.
- 'xorder': exponent for the bunch shape (see Eq. 1). This parameter is active only when no bunch train input file is used.
- 'beta': beam relativistic velocity
- 'bsol': solenoid magnetic field

2. *0.10725 1.0e+5 15 0.0001 0.0 avgcurr,flambdae,fracmax,fracmin,xoff*

- 'avgcurr': total beam intensity (in Amps)
- 'flambdae': initial line density of electrons. It denotes the number of electrons per meter.
- 'fracmax': maximum charge fraction (in terms of the initial charge per macroparticle). When macroparticles hit the wall and emerge with a charge greater than 'fracmax' they are divided into daughter particles of charge smaller than 'fracmax'. Care is taken to make sure that the daughter trajectories are sufficiently different to guarantee statistical independence.
- 'fracmin': minimum charge fraction (in terms of the initial charge per macroparticle). When a macroparticle leaves the wall with a charge less than 'fracmin' it is destroyed. This keeps the number of macroparticles manageable.
- 'xoff': beam offset. This option must be updated. Always set xoff=0.

3. *108000 25 .001 3 10 10 np,ne,rade,nturns,nskp,iskip*

- 'np': number of slices (updates) per turn
- 'ne': initial number of macroparticles
- 'rade': transverse electron smoothing length. With imposed cylindrical symmetry a macroparticle creates a charge distribution in the form of a cylindrical shell. The effective thickness of this shell is 'rade'. This keeps the electric fields smooth, especially near the beam axis.
- 'nturns': number of turns to be simulated
- 'nskp': number of updates before writing average quantities to the output file
- 'iskip': number of proton bins between creation of macroparticles. The macroparticle generation rate is proportional to the instantaneous beam current. If the bunch is very long, or if 'np' is very large it is possible that a large fraction of the created electrons will have a charge less than 'fracmin'. By increasing 'iskip' a smaller number of larger macroparticles are created.

4. *.0024 .06 0 1 radbeam,radpipe, ibeamtype, ipz*

- 'radbeam': rms beam radius. (The beam is always assumed to be round).

- 'radpipe': beam pipe radius
- 'ibeamtype': A round beam of uniform density for 'ibeamtype' =1. For 'ibeamtype' =0 the radial electric field varies as $E \propto r/(r^2 + radbeam^2)$
- 'ipz': when 'ipz'=1, electrons created at the wall are allowed to have a velocity component parallel to the beam direction. for 'ipz' = 0 the electron motion is strictly transverse. Setting 'ipz'=1 improves agreement between CSEC and POSINST [6] and is recommended.

5. *1.95 305 8.925 0.5 60 1.813 1 0.5 1 0.1 Stainless*

This line sets the parameters defining the SEY, and their meaning is shown in Appendix A. The order is:

- 'deltamax': δ_{\max}^* in Eqs. 3 and 4.
- 'epeak': E_{\max} in Eqs. 3 and 4 (in eV).
- 'esecond': E_{sec} in Eq. 6 (in eV). It determines the energy distribution of the secondary electrons as shown above.
- 'preflect0': δ_0 in Eqs. 3 and 4
- 'erefect': E_r in Eqs. 3 and 4 (in eV)
- 'sfurman': s in Eqs. 3 and 4. Typical values for this parameter are 1.54 (Copper) or 1.813 (stainless steel).
- 'pang': This is the parameter μ in Eq. 8, which is used to determine the angular distribution of secondary and reflected electrons. According to Ref. [6], we strongly encourage to set this parameter ot 1.
- 'prediffuse': This is the conditional probability that a reflected electron is re-diffused (emitted with an energy between between 0 and the impinging energy, E)
- 'alfsey': α in Eq. 7. According to Ref. [6], we strongly encourage to set this parameter ot 0.5.
- 'preflectinf': δ_{∞} in Eqs. 3 and 4

6. *2.4e6 12898643 0.06 flambdaperturn idum rborn*

- 'flambdaperturn': electron line density in number per meter created per turn. This sets the primary electron creation.
- 'idum': seed for random numbers
- 'rborn': radius where the electrons are created. This should never be larger than the beam pipe radius. Setting it to the pipe radius corresponds to loss created or photo electrons. Setting it to something like 'radbeam' corresponds to primaries created by ionizing residual gas.

7. *12.82e-6 100.e-9 1000 100 thist,dthist, emaxhist, nhist*

These parameters control the accumulation of data for the macroparticle phase space density plot in "hist.out" and the average electric fields and potentials in "pot.out".

- 'thist': central time for data accumulation
- 'dthist': total time over which the data are accumulated
- 'emaxhist': Upper energy limit (in eV) for the histogram. Particles with energy larger than 'emaxhist' are put in the largest energy bin.
- 'nhist': The output array for "hist.out" is binned from -nhist to nhist in each dimension.

3.2 Input file 2: 'ncsec.file'

It gives the location and name of the train file, and other computer parameters to be taken into account by the program.

- *ir.dat*

The first line is the filename for the bunch train. That is, the output of either IRTRAIN (“straight.dat” or “ir.dat”) or MAKETRAIN (“fort.30”). If the simple bunch patterns generated by the code are going to be used any file name will do.

- *0 108000 2 1 nskipin,nptin,ncol, igetprof*

If 'igetprof' is 0 no bunch train file is opened and the bunch parameters on line 1 of “ncsecz0_hist.in” are used. Assuming that a file is used: 'nskipin' tell the code how many lines of the input file to skip (for comments, etc), 'nptin' is the number of points in the array containing one turn of profile data, and 'ncol' is the column containing the profile data. It is not necessary for 'nptin' to equal 'np'. However, the code does assume that $nptin \leq np$.

3.3 Output file 1: 'ncsecz0.sum'

The output file variables are listed in Table 2.

Table 2: List of all output variables shown in “ncsecz0.sum”.

column number	parameter	unit
1	time	μs
2	rms radius	cm
3	rms energy	keV
4	electron line density	nC/m
5	beam current	A
6	electron flux to the wall	mA/m
7	electron flux in a pick-up	$\mu A/cm^2$
8	number of electron macroparticles	-
9	beam line density	nC/m
10	integrated charge	pC/(cm \times 10deg)
11	electron density within the beam	nC/m
12	statistic number of macroparticles	-
13	average kinetic energy of electrons	eV
14	$E > 100eV$ electron flux into wall	mA/m

3.4 Other output files

3.4.1 Energy spectrum: 'fort.66'

It provides the electron energy spectrum. The first row correspond to the electron energy, the second row to the number of electron-wall collisions produced at that energy.

3.4.2 Phase space: 'hist.out'

This file gives the time average over $[thist - dthist/2, thist + dthist/2]$ of the radial electron phase space density. There are three (3) columns: radial position (meters), radial velocity (meters per second), total charge in the bin (unnormalized). Using the command-line plotting utility “Gnuplot”, a two dimensional plot is obtained via

```
gnuplot> sp 'hist.out' u 1:2:3 w l
```

3.4.3 Output 'pot.dat'

This file gives the time average over $[thist - dthist/2, thist + dthist/2]$ of the electric fields and potentials due to the beam and the electron cloud. There are five (5) columns, r, ehist, phist,pote,potp. The radius at which the time average is calculated is 'r'. The average radial electric fields due to the electrons and ions are 'ehist' and 'phist' respectively. They are both in volts per meter. The average electrostatic potentials due the the electrons and ions are 'pote' and 'potp', respectively. Both are in Volts.

4 Example

In the Interaction Region of a collider, two counter-rotating beams (i.e., blue and yellow for RHIC) circulate in opposite directions and collide in the middle of the beam pipe. This peculiar situation is addressed in detail in Ref. [5]. Consider s_b as the single beam bunch length (blue or yellow). At a distance Δz from the collision point, a blue bunch is separated by $2|\Delta z|$ from a yellow bunch, which in turn is at $s_b - 2|\Delta z|$ from the next yellow bunch passage - s_b is the bunch spacing for a single beam (blue or yellow). Note that at the collision point ($\Delta z = 0$), the bunch intensity is doubled and the bunch spacing is the same as in the single beam pipes.

Table 3: List of input parameters for electron cloud simulations at IR12. The actual bunch ion population of 10^9 gold ions ($Z = 78+$) is converted to a charge equivalent number of protons per bunch.

parameter	unit	injection / transition
bunch population	protons	7.8×10^{10}
full bunch length	ns	16.6 / 4
single beam bunch spacing	ns	214
bunches/ring	...	56
rms beam radius	mm	2.4 / 2.0
lorentz factor	...	10.5 / 100
pipe radius	cm	6.0
distance from IP, Δz	m	7.0
# of turns	...	3
# of slices/turn ('np')	...	108000 / 312000
# of updates ('nsp')	...	10 / 30
# of creation bins ('iskip')	...	10 / 20
max. SEY, δ_{\max}^*	...	1.95
energy for max. SEY, E_{\max}	eV	305
SEY for $E \rightarrow 0$, δ_0	...	0.5
SEY for $E \rightarrow \infty$, δ_∞	...	0.1
"reflection" energy, E_r	eV	60
energy for secondary electrons, E_{sec}	eV	8.9
re-diffusion probability ('prediffuse')	...	0.5
probability distribution parameter, μ	...	1.0
non-normal incidence parameter, α	...	0.5

It is very important for CSEC to update the electric fields during the electron-bunch interactions. This is controlled by the number of slices in both IRTRAIN (when the bunch train is created) and CSEC. Besides the changes in the beam parameters (see Table 3), we need to change few computer parameters to update our needs to the "computer" requirements:

- Since during transition crossing the bunch length is significantly reduced (from 16.6ns to 4 ns), we have increased the number of slices from 108000 (injection case) to 312000 (transition crossing case) to properly update the electric fields during the bunch passage.

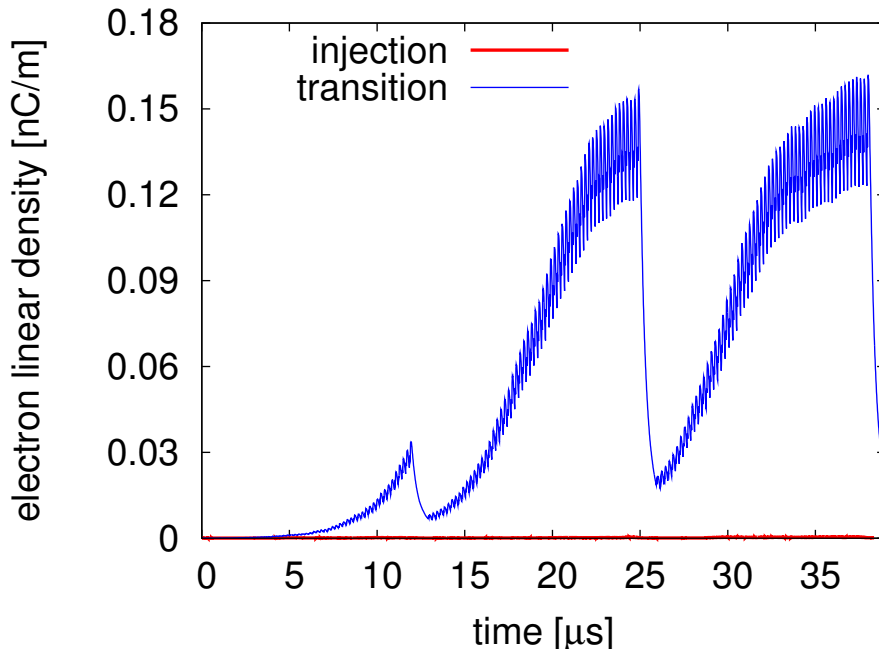


Figure 1: Electron density during during beam injection conditions and while transition crossing. The plot is obtained using columns 1 (time) and 4 (electron density) of the output file 'ncsecz0.sum'.

- In order to not create a huge output file while studying the transition case, we tell CSEC to write in the output file every 30th slice instead of every 10th slice (parameter 'nsp').
- The parameter 'iskip' also changes from 10 (at injection) to 20 (at transition) to avoid numerical noise.

In this example we simulate the situation at RHIC IR12. It is a stainless steel vacuum chamber of 6 cm in radius. Beam and chamber parameters are listed in Table 3. Note none of the chamber parameters changes between both simulation cases.

Figure 1 compares the build up of an electron cloud during injection and “transition” conditions, where the main difference is in the longitudinal bunch length. One can see that for the injection conditions ($bl = 5$ m) the electron cloud does not build up. The bunch length when the beams cross the transition energy is reduced to $bl = 1.2$ m. This increases the electron energy gain, which triggers the multipacting. The electron density in this case saturates at about 0.14 nC/m.

Figure 2 compares the electron energy spectrum (file 'fort.66') during the two cases.

Figure 3 shows a zoom of the four bunch passages (blue - yellow - blue - yellow) during transition conditions.

The input files used for these two examples can be downloaded at the abovementioned website.

A Secondary Emission Yield parameterization

Currently, the SEY used by CSEC is based on the Furman-Pivi parameterization [6]. An electron hitting the wall with energy E , can undergo elastic reflection, rediffused reflection, or produce true secondaries. The reflection probability as a function of energy is denoted

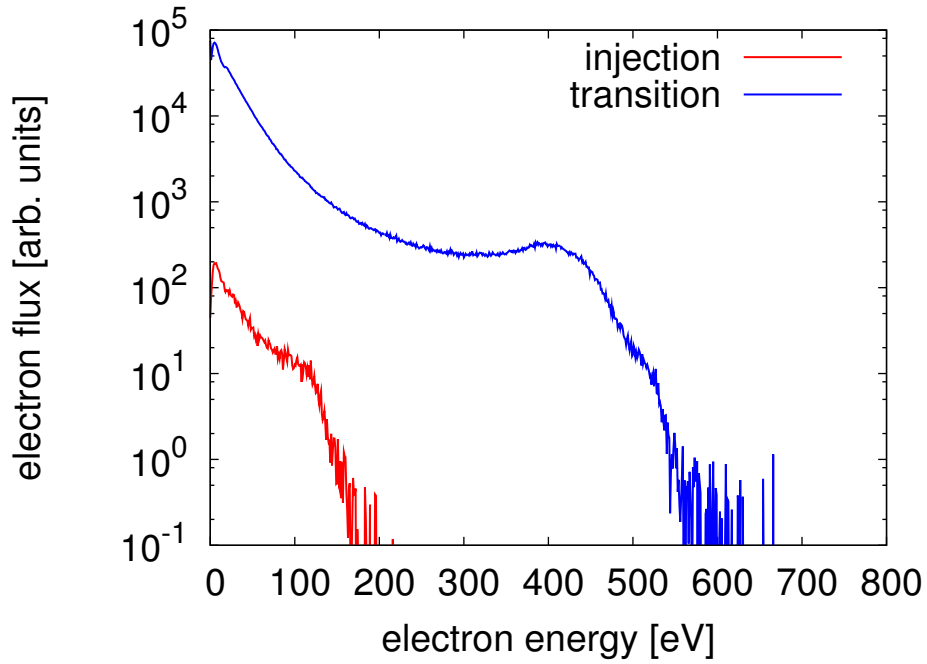


Figure 2: Electron energy spectrum during beam injection conditions and while transition crossing. The plot is obtained from file 'fort.66'.

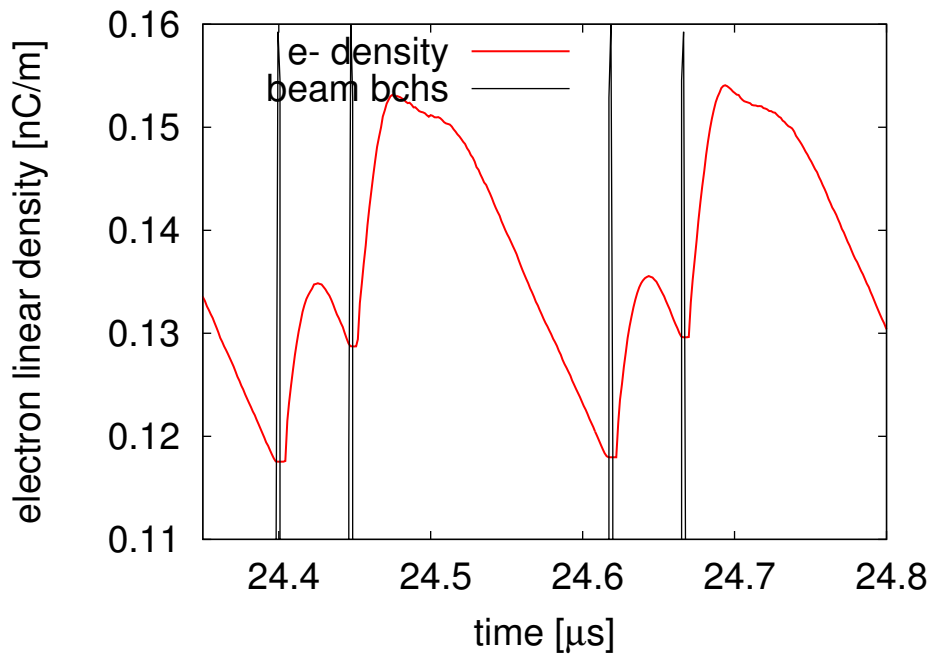


Figure 3: Electron energy spectrum during beam injection conditions and while transition crossing. The plot is obtained using columns 1 (time), 4 (electron density), and 5 (bunch intensity) from the output file 'ncsez0.sum'.

by δ_r and the number of true secondaries per incident electron is denoted by δ_t . The total number of electrons leaving the surface per incident electron is

$$\delta = \delta_r + \delta_t \quad (2)$$

For normal incidence, the equation for the SEY as a function of the energy E of the impinging electron is:

$$\delta_r(E) = (\delta_0 - \delta_\infty)e^{-E/E_r} + \delta_\infty, \quad (3)$$

$$\delta_t(E) = \delta_{\max}^* \frac{s(E/E_{\max})}{s - 1 + (E/E_{\max})^s} \quad (4)$$

Rediffused electrons are handled in a very simple way. A single parameter 'prediffuse' gives the conditional probability that a reflected electron is rediffused. Rediffused electrons have an energy distribution

$$P(E_{\text{out}}) = 1/E \quad \text{for } E_{\text{out}} < E. \quad (5)$$

True secondaries have an energy distribution

$$P(E_{\text{out}}) \propto E_{\text{out}} / (E_{\text{out}}^2 + E_{\text{sec}}^2)^2. \quad (6)$$

For non-normal incidence angle, the entire $\delta(E)$ is multiplied by

$$\delta(E, \theta) = \delta(E) \exp^{\alpha(1-\cos\theta)}, \quad (7)$$

where the angle θ is taken to the surface's normal ($\theta = 0$ indicates perpendicular incidence), and α is a fitting parameter [6]. If the user wishes to keep the reflection probability independent of θ , comment out line 403, and uncomment line 405. This change might be buggy.

The electrons leave the surface with an arbitrary angle. True secondaries and reflected electrons are treated in the same way. The probability distribution as a function of θ is parameterized as

$$P(\theta)d\theta = (1 + \mu) \cos^\mu \theta \sin \theta d\theta \quad (8)$$

Imagine centering a hemisphere on the emission point of the electron. Setting $\mu = 0$ with 'ipz'=1 gives a uniform distribution over the hemisphere. For $\mu > 0$ there is a tendency to concentrate trajectories near the top of the hemisphere. The maximum entropy distribution corresponds to $\mu = 0$ when 'ipz'=1.

A.1 Program SEY_SIM

A program to visualize the SEY curve is also provided in the website

<http://www.agsrhichome.bnl.gov/People/ubaldo/> .

Running the program SEY_SIM with the input file "ncsecz0_hist.in" creates the output file "sey.sim". The code will prompt the user for the cosine of the angle at which to calculate to total secondary yield, including reflections. There are four (4) columns: incident electron energy (eV), true secondary electron yield for perpendicular incidence, reflection probability for perpendicular incidence, sum of true secondary yield and reflection probability at the angle requested.

B Train of equispaced bunches – MAKETRAN

This valid was originally written to create a train of equispaced bunch trains. This program can also be downloaded at the aforementioned website:

<http://www.agsrhichome.bnl.gov/People/ubaldo/> .

B.1 Input file: “train.in”

It only consists of two lines. Each line includes several parameters, and next to them the parameter name in the program MAKETRAN.

- *360 2519 1 0 nh1 nh2 v1 v2* This line lists the first harmonic number, second harmonic number, and the respective normalized voltages (1 or 0).
- *3 110 300 1.5 3 nskp nbunches nsamp phi_hat power* The first two numbers determine the number of buckets between bunches (“3”) and the total number of bunches to fill in (“110”). The third number (“300”) determines the number of samples in the file, while the fourth and fifth numbers (“1.5” and “3”) determine the bunch profile – similar to Eq. 1.

B.2 Output file: “fort.30”

Azimuthal phase between samples and bunch density in arbitrary units. As IRTRAIN, the bunches follow a parabolic profile.

References

- [1] F. Zimmermann et al., in Proceedings of EPAC’04, THPLT017, Lucerne, 2004. See also <http://wwwslap.cern.ch/collective/ecloud02/ecsim>.
- [2] M. Blaskiewicz et al. “Electron cloud instabilities in the PSR and SNS”, PRST-AB 6, 014203. Jan. 2003.
- [3] W. Fischer, J.M. Brennan, M. Blaskiewicz, and T. Satogata, “Electron Cloud Measurements and Simulations for the Brookhaven Relativistic Heavy Ion Collider”, Phys. Rev. ST Accel. Beams 5, 124401 (2002).
- [4] U. Iriso and W Fischer. “Electron induced molecular desorption from electron clouds at the Relativistic Heavy Ion Collider”, Phys. Rev. ST Accel. Beams 8, 113201 (2005).
- [5] U. Iriso, Ph.D. thesis, “Electron clouds at the Relativistic Heavy Ion Collider”. University of Barcelona, 2006. Also BNL Technical Note, C-A/AP/228, http://www.agsrhichome.bnl.gov/AP/ap_notes/cad_ap_index.html.
- [6] M.A. Furman and M. Pivi, “Microscopic probabilistic model for the simulation of secondary electron emission”, Phys. Rev. ST Accel. Beams 5, 124404 (2002).
- [7] G. Rumolo and F. Zimmermann, “Practical User Guide for ECloud”. CERN-SL-2002-016 AP, 2002.