A Small Guide to the NUTMEG installation at DESY

P. Pierini, INFN - Milano, LASA
E.T. Scharlemann, LLNL

December, 14, 1994

Abstract

This short note describes the installation of NUTMEG, the LLNL 2D FEL code, on the HP Cluster at DESY.

1 Implementation of NUTMEG at DESY

The version of NUTMEG running on the HP Cluster at DESY originates from the LLNL Cray sources. The code has been ported to double precision and, through the use of the standard Unix **m4** preprocessor, conditional directives have been added in order to have both the Cray and the HP versions in a single maintanable file.

NUTMEG is a 2D FEL simulation code developed at LLNL, authored primarily by E.T. Scharlemann. The code can model either a microwave device in which the emitted radiation is confined within a waveguide, or a short wavelength device taking into account the diffraction of the optical beam.

References [1]-[7] describe in some detail many characteristics of the LLNL set of FEL codes: FRED, GINGER and NUTMEG.

2 User setup

The executable **xnut** (NUTMEG program) and **xpltnut** (NUTMEG graphics postprocessor), as well as the**m4** files, are placed in the following directory of the HP-Cluster:

```
-rwxr-xr-x   1 pierini  mpy  283272 Dec 12 12:41 /usr/users/pierini/src/xnut
-rwxr-xr-x   1 pierini  mpy  799368 Dec 12 09:59 /usr/users/pierini/src/xpltnut
```

and the read/execute access is granted to all users of the HP cluster.

In order to have access to the code the user should put the following symbolic links in one of the directories in his/her path (in the example below **~/bin**:

```
[ips109] /usr/users/pierini cd bin
[ips109] bin $ ln -s /usr/users/pierini/src/NUTMEG/xnut xnut
[ips109] bin $ ln -s /usr/users/pierini/src/NUTMEG/xpltnut xpltnut
```

After this a `ls -l` command should produce the following output:

```
total 510
     ...
lrwxr-xr-x   1 ...  34 Dec  1 17:28 xnut -> /usr/users/pierini/src/NUTMEG/xnut
lrwxr-xr-x   1 ...  37 Dec  8 17:34 xpltnut -> /usr/users/pierini/src/NUTMEG/xpltnut
     ...
```

3 How to run NUTMEG

The code requires an input string (max. 5 character long) on the command line, with which it builds the names of the input and output files needed for the simulation.

The minimum set of files needed/produced in a single simulation is the following (*name* is the variable containing the input string):

in*name* The input file, containing the FORTRAN Namelist with the input data for the run. Input file.

plt*name* The "plotting" file, where most of the output needed by the postprocessor is written. Output file.

prl*name* An auxiliary file for parameter check and reports from the integrator. Output file.

phs*name* The file containing the phase space particle plots, used by the postprocessor. Output file.

By switching appropriate flags in the input file other files may be produced or needed for writing the output beam or restarting from a previously saved beam.

4 Input Namelist and meaning of the variables

The input file contains a single NAMELIST, `inlist`, reported below with the DATA statements that set default values for the input data not explicitely appearing in the input file.

```
      namelist /inlist/ aw0init,awdmult,awmin,bright,bw0,
     . campmode,current,delpsi,
     . dgamdz0,dgamfac,dgamma,dgbyg,dpsir0,dt,dzsteer,zlion,
     . emit0,em8,energy,eps,fharm,focusfac,gammad0,gammar0,gap,
     . ghz,gmg,hlb,hub,hubmeter,idesign,idsteer,ierode,
     . ierpde,indpc2,inecho,irstfld,irstphs,irstrd,irstwr,
     . iseed,iqseed,iwseed,jmg,lanl,lssrl,lawaxis,lbetapar,
     . lchnglw,lcnstwgl,lconstaw,lconstbw,lconstkw,lconstgp,lconstkh,
     . lequil,lenddrft,lfixfld,lgain,lincrdes,lkwtaper,llinear,
     . lnouperp,loval,lphase10,lquiet,lrefract,lspacech,lswharm,mswharm,
     . luncoupl,lvacfld,lvarygap,lwavegd,lxvxfix,lte21,
     . mharm,mstep,maxstep,mode,modestr,modetest,
     . nax,nnd,nnew,nmg,ncurve,nharms,nprint,nsmooth,nspec,ntestp,
     . omg0,omgj,omg0fac,psirinit,psir0,plaser,phlaser,phite0m,
     . phaserra,phaserrl,qlaser,quad0,rdesign,rkxkw,rdesign,
     . rmaxsim,rlinear,rdesfac,rmaxcur,te0m,
     . upj,wavelw,wavels,wavelx,wiglfluc,wlwinit,xwidth,xoff,xbscale,
     . ywidth,yoff,ybscale,zfocus,zfcmeter,zmaxsim,zmxmeter,
     . zsteer,zdiag,zspec,zqstart,zqend,z1wlw,z2wlw,z1psi,z2psi

c laser field parameters
      data nmg/2/,                ! laser field profile index, 2 - Gaussian
     . omg0/maxharms*-1.0/,       ! input laser field beam radius (1/e**2)
     . omg0fac/0.8/,              ! if omg0<0 then omg0 = omg0fac*omgj
     . wavels/1.06e-5/,           ! laser wavelength (m)
     . phaserrl/1.0/,             ! scale length of phase abberations in beam
     . phaserra/0.0/,             ! amplitude of phase abberations in beam
     . zfocus /maxharms*0.0/, ! laser focus pos. in Rayleigh ranges (past z=0)
     . zfcmeter/maxharms*-1.0/, ! zfocus in meters.
     . ghz/-1.0/,                 ! signal frequency in GHz, for waveguide runs
     . lfixfld/.false./,          ! fixes laser field to input field
```

```
     . lvacfld/.false./,          ! electrons evolve in a vacuum field
     . lrefract/.true./,          ! refactive guiding switched off if .false.
     . lgain/.true./,             ! gain switched off if .false.
     . plaser /1.0e6,0.0/,        ! initial laser power (W)
     . phlaser/maxharms*0.0/,     ! initial laser phase (radians)
     . campmode/(1.0,0.0),        ! complex amplitude of Gauss-Laguerre modes
     .          mampdim*(0.0,0.0)/
     . nharms/maxharms/,          ! number of odd harmonics (2 or 3)
     . mharm/1,0/,                ! which harmonics used (i.e. 1st, 3rd, etc.)
     . fharm/maxharms*1.0/,       ! harmonic source term multiplier
     . nsmooth/3/                 ! radial smoothing in sourcejs

c electron beam parameters
      data jmg/-1/,           ! electron beam profile index (2=Gaussian)
     . gmg/-1/,               ! electron energy spread prof. index (2=Gaussian)
     . omgj/0.3/,             ! electron beam radius
     . gammar0/-1.0/,         ! initial electron gamma
     . energy/50.0/,          ! initial electron energy (MeV)
     . current/2.0e3/,        ! initial beam current (A)
     . modetest/0/,           ! switch for betatron motion 0 = on
     . ntestp /2057/,         ! number of particles (2**n + 9)
     . bright /-1.0/,         ! electron beam brightness A/(m-rad)**2
     . emit0 /-1.0/,          ! electron beam normalized rms emittance mm-mrad
     . psir0/0.4/,            ! resonant psi(radians)
     . dpsir0/-1.0/,          ! half-width of initial psi around psir0
     . delpsi/0.0/,
     . gammad0 /0.0/,         ! initial gamma for design particle
     . dgamma /0.0/,          ! half width of initial electron energy distrib.
     . dgbyg/0.0/,            ! fractional initial electron energy spread (rms)
     . dgamfac/-1.0/ ! fraction of effective energy spread due to emittance
       data zlion/-1.0/       ! external harmonic focusing betatron wavelength

      data wavelx/-1.0/,      ! x betatron wavelength in m.
     . xoff/0.0/,             ! initial electron beam offset in x
     . yoff/0.0/,             ! initial electron beam offset in y
     . xbscale/1.0/,          ! multiplier of initial beam size in x
     . ybscale/1.0/,          ! multiplier of initial beam size in y
     . loval/.true./,         ! see settestp: oval test particle beam
     . rmaxcur/-1.0/,         ! radius at which gaussian e-beam is truncated
     . lnouperp/.false./,     ! leaves out gam**2 betaperp**2 term in dpsidz eqn
     . luncoupl/.false./,     ! uncouples particles from field for debugging
     . nax/2/,                ! particles leaded at 2**nax in quiet start
     . lquiet/.true./

c restart parameters
      data irstrd/.false./, ! restart with "rst" file
     .  irstwr/.false./,      ! write an "rst" file for subsequent restart
     .  irstfld/.false./,     ! read field values for "rst" file
     .  irstphs/.false./,     ! restart with scrambled phases (theta)
     .  indpc2/.false./,      ! run with extern. generated particle file
     .  lswharm/.false./,     ! if .true., switch mswharm harmonic of
     .                        ! previous run to fundamental of restarted
     .  mswharm/3/            ! number of harmonic to switch to fundamental
```

```
c wiggler parameters
       data wavelw/0.08/,    ! wiggler wavelength (m)
     . rdesfac/0.707107/,    ! design radius as a fraction of beam radius
     . rdesign/-1.0/,        ! wiggler design radius (cm), for tapering
     . z1psi/0.0/,           ! more elaborate self design parameters
     . z2psi/0.0/,           !     for variable psir: see setaw
     . psirinit/0.0/,        !
     . lchnglw/.false./,     ! switch for variable wiggler wavelength
     . z1wlw/0.0/,           ! initial z for wiggler wavelength
     . z2wlw/0.0/,           ! final z for wiggler wavelength
     . wlwinit/0.0/,         ! initial wiggler wavelength (wavelw is final)
     . idesign/1/,           ! wiggler self design flag (1) self, (0) input
     . lcnstwgl/.true./,     ! forces constant wiggler when true
     . awdmult/1.00/,        ! dimensionless multiplier for aw
     . lincrdes/.false./,    ! switch to let design radius increase adiabaticall
     . lkwtaper/.false./,    ! wiggler tapered in aw (.f) or kw (.t)
     . lconstaw/.false./,    ! if true, taper wavelw at constant aw
     . lconstkw/.true./,     ! if true, taper aw0 at constant wavelw
     . lconstbw/.false./,    ! if true, taper wavelw at constant bw = aw0*kw
     . lconstgp/.false./, !if .t, kw and aw tapered with aw*kw/(1+aw**2) const
     . lconstkh/.false./,    ! if true, wiggler designed to max field for hybrid
     . gap/2.0/,             !   at given gap
     . lvarygap/.false./,    ! if true, gap is varied to be = 4*w0 (vacuum)
     . aw0init/-1.0/,        ! if > 0 and lconstaw, initial value for aw0
     . awmin/0.0/,           ! minimum aw value: wiggler ended when aw<awmin
     . bw0/-1.0/,            ! peak wiggler magnetic field (kG) if lconstbw
     . lanl/.false./,        ! modifies emittance defn.; see setbeam
     . lssrl/.true./,        ! modifies emittance defn.; see setbeam
     . llinear/.false./      ! linear wiggler (as opposed to helical)

      data zmaxsim/2.0/,     ! run length in Rayleigh ranges
     . zmxmeter/-1.0/,       ! run length in meters.
     . rmaxsim /9.0/,        ! maximum radius for the simulation (cm)
     . quad0/0.0/,           ! strength of external quadrupole focusing(G/cm)
     . qlaser/0.0/,          !
     . rkxkw/-1.0/,          ! ratio between kx for wiggler and kw
     . focusfac/-1.0/,       ! tapered quadrupole focusing (if idesign.ge.1)
     . lawaxis/.false./,     ! if .true., wiggler field on axis is used
     . lequil/.true./,
     . lenddrft/.true./,     ! if .f uses specified end of drift space
     . lphase10/.false./,    ! .t prevents dphidz > 10% in wiggler design
     . lbetapar/.false./,    ! 2 or (betapar*(1+betapar)) in dpsi/dz eqn.
     . zqstart/nqsex*-1.0/,  ! sets up beginning of quadrupole focusing in z
     . zqend/nqsex*0.0/      ! sets up end of quadrupole focusing in z

      data wiglfluc/0.0/,      ! random wiggler errors are generated if .ne.0
     . zsteer/maxsteer*-1.0/,  ! locations of steering stations in z
     . modestr/0/,             ! interlaces steering and diagonstics stations
     . iwseed/0/,              ! random number seed for wiggler errors
     . zdiag/maxsteer*-1.0/,   ! locations of beam position monitors
     . dzsteer/-1.0/,          ! if > 0, steering stations separated by dzsteer
     . dgamdz0/0.0/            ! longitudinal accel field (in meters**-1)

c solver and I/O parameters
```

```
      data nnd/60/,         ! number of radial grid points for laser field
     . ierpde/3/,           ! error in partial differential equation solver
     . ierode/3/,           ! error in ordinary differential equation solver
     . nnew/5/,             ! max number of times main particle loop is iterated
     . ncurve/50/,          ! number points where plotter data is written
     . nspec /5/,           ! number points where phase plot data is written
     . mstep/1000/,
     . maxstep /10000/,     ! maximum number of steps in run
     . upj/.3/,
     . eps/1.0e-4/,         ! error control of Gear integrator
     . em8/1.0e-5/,
     . dt/1.0e-9/,          ! initial step size in rayleigh lengths
     . hlb/1.0e-11/,        ! lower bound on step size
     . hub/-1.0/,           ! upper bound on step size
     . hubmeter/-1.0/       ! upper bound on step size in meters
     . iseed/0/,            ! seed for the random number generator
     . iqseed/0/,
     . rlinear/-1.0/,       ! r value out to which grid is roughly linear
     . idsteer/2/,          ! switch for writing or reading steering file
     . nprint/4/            ! number of times to write to prl file

c Space charge and waveguide parameters
      data xwidth/9.834/,   ! full width of waveguide parallel to field (cm)
     . ywidth/2.908/,       ! full width of waveguide in direstion of variation
     . lwavegd/.false./,    ! flag for waveguide (.t) or free space(.f)
     . te0m/0.0/,           ! fraction of input laser power not in TE01 mode
     . mode/1/,             ! mode number of te0m power dependent on lte21
     . phite0m/0.0/,        ! phase of TE0m mode in waveguide
     . lte21/.false./,      ! waveguide field assumed cos(pi*x/xwidth) dependence
     . lspacech/.false./,   ! switch to include space charge
     . lxvxfix/.false./
```

Of all these input data only a few are relevant for the optical simulations needed for the TTF FEL, and are described below, with the units in which they are expressed. In general, variables referring to transverse lengths are in cm and variables referring to longitudinal lengths are in m.

4.1 Electron Beam Parameters

current The beam peak current (A).

emit0 Beam rms normalized emittance (mm mrad). Defined as $\epsilon_0 = \gamma(\langle x^2\rangle\langle x'^2\rangle - \langle xx'\rangle^2)^{1/2}$.

energy Initial beam energy (MeV).

dgbyg Fractional rms energy spread.

dgamma Rms energy spread of initial electron distribution. Use **either dgbyg or dgamma**. If **dgbyg** is not set to zero then **dgamma=dgbyg*gammar0**, where **gammar0** is the electron γ computed from **energy** (**gammar0=energy/0.511+1.0**).

jmg Electron beam profile index. Use 2 for a transverse gaussian beam.

gmg Electron energy spread profile index.

gmg=2 Gaussian energy spread, σ_γ/γ =**dgbyg**.

gmg=-1 Uniform energy spread $\pm\sqrt{2}$**dgamma**.

ntestp Number of simulation particles. There is no code limit to this number, since memory is dynamically allocated at execution.

xoff Initial electron beam offset in x (horizontal direction).

yoff Initial electron beam offset in y (vertical direction).

xbscale Multiplier of initial beam size in x (to allow mismatching).

ybscale Multiplier of initial beam size in y (to allow mismatching).

4.2 Laser Beam Parameters

wavels The optical wavelength (m).

nharms Number of odd harmonics to consider.

mharm Harmonic number (odd) of the `nharms` to consider.

plaser Initial laser power (W). It is an array containing the values for the harmonics.

nmg Laser field profile index. 2 for a purely Gaussian mode.

omg0fac The initial relative ratio of the radiation spot size with respect of the electron beam radius.

zfcmeter Position of the optical waist (m) past the wiggler entrance (zfcmeter ¡ 0 would correspond to a focus before the wiggler entrance). It is an array containing the values for the harmonics.

4.3 Wiggler and focusing Parameters

Wiggler

wavelw The wiggler period (m).

lcnstwgl Forces constant (untapered) wiggler if `.true.`. Logical.

llinear Choose between linear or helical wiggler. Logical.

llinear=.true. Linear wiggler.
llinear=.false. Helical wiggler.

zmxmeter Wiggler length (m).

awdmult Dimensionless multiplier for a_w.

rkxkw Ratio k_x/k_w for tedpole focusing.

Focusing

zlion External harmonic focusing betatron wavelength (in both planes).

zlion=-1.0 Turned off.
zlion=λ_β Betatron wavelength of the continuous external focusing channel.

quad0 Strength of external quadrupole focusing (G/cm).

Wiggler Errors

wigfluc If $\neq 0$ random wiggler errors are generated at each pole, with an rms value of `wigfluc`.

iwseed Seed number for the wiggler errors.

zsteer Array containing the locations of the steering stations.

4.4 Solver and I/O Parameters

rmaxsim Maximum radius for the simulation (cm).

ncurve The number of plotted points along the wiggler.

nnd Number of radial gridpoints.

nspec Number of points in the wiggler where the phase plot data are written.

4.5 Restart Flags

irstrd The restart flag. If `.true.`, the particles are read by a restart file.

irstwr The flag used to save a "restart" file. If `.true.`, the particles are written to a restart file at the end of the simulation.

indpc2 Flag to allow the beam startup from an external particle file.

4.6 Additional Flags

lwavegd `.false.` for optical FEL case.

5 Sample input file for the 6 nm FEL at TTF

Here is a sample input file for the 6 nm FEL at TTF.

```
nutmeg : FEL output at 6 nm for TTF SASE FEL. J.R. data table
 &inlist
 jmg = 2,                           gmg = 2
 current = 2490.0                   energy = 1000.0
 dgbyg = 1.0e-3                     emit0 = 1.0
 ntestp = 4105                      wavelw = 0.0273
 zmxmeter = 25.0                    rkxkw = 0.0
 llinear = .t                       lcnstwgl = .t
 delpsi = 0.0                       zlion = 31.4159
 nmg = 2                            nharms = 1
 mharm = 1                          wavels = 6.421e-9
 plaser = 1.0e3, 0.0d0, 0.0d0       zfcmeter = 0.0
 idesign = 2                        psir0 = 0.0
 rdesfac = 1.5                      awdmult = 1.0000
 rmaxsim = 0.05                     nnd = 63
 nsmooth = 1                        dt = 1.0e-6
 hlb = 1.0e-11                      ncurve = 51
 nspec = 1                          zspec = 10.0, 19.999
 eps = 2.0e-04                      iseed = 77775501
 lspacech = .f                      lbetapar = .f
 irstwr = .f
 &end
```

6 Screen output for the above input file

The code, at startup, writes on the standard output (i.e. the screen if not redirected in a file) some parameters of the simulation, like beam current, brightness and emittance, beam and radiation spotsizes, wiggler parameters (derived from the resonance condition) and initial laser power.

At the end of the simulation some relevant output parameters are writtent on the screen: Saturation power and length and energy conservation tests.

```
 xnut executed with command line httf
Source file: nut1205   --  run name: httf
nutmeg : FEL output at 6 nm for TTF SASE FEL. J.R. data table

Current = 2.49E+03 Amps    :  Brightness =  7.50E+12 A/(m-rad)**2
Normalized emittance =  1.00E+00 mm-mrad    :  Beam edge = -1.00E+00 cm.
omgj =  6.72E-03  :  omg0 = 5.38E-03  cm.
Calculated rms emit: x = 9.29E-01 mm mrad, y = 9.15E-01 mm mrad

Initial laser power (n = 1) =   1.0017E+03  1.0017E+03
Initial Bw(peak) on axis    =  4.970E+00 kiloGauss
Initial aw(rms) on axis     =  8.958E-01
Initial wiggler wavelength  =   2.730 cm
Final laser power for  1st harmonic =  2.1220E+09


Maximum laser power for  1st harmonic = 2.7810E+09 W, at z =  1.650E+01 m


Laser power from electrons = 2.1284E+09
Final electron beam power  = 2.48780E+12
Trapping fraction = 72.36 %
Extraction        =   .09 %
Bw(peak) on axis = 4.970E+00 kiloGauss
aw(rms) on axis  = 8.958E-01

wiggler wavelength =  2.730 cm
at z =    25.000 m
max wiggler design iterations = 0

R.m.s. wiggler error =  .000E+00 over    0 poles
Weighted wiggler errors =  .000E+00 cm
sin term =  .000E+00     cos term =  .000E+00

Energy conservation:
Initial total power = 2.48993E+12
Final total power   = 2.48992E+12
Total power gain    =-6.466E+06

Average step size   = 2.106E-02
```

7 Output produced by NUTMEG

The postprocessor output is run with the command **xpltnut** *name*, where *name* is the run flag used in **xnut**. The graphic output is provided by CERNLIB HIGZ[8] calls. On execution the HIGZ library asks for the graphics workstation type. If **xpltnut** is executed from an X terminal, the default workstation type (1) can be choosen pressing Return. As the user goes through a display of the postprocessor plots, a PostScript file, **nutmeg.ps**, is created in the current directory. This file can then be later viewed with **ghostscript** or printed.

The first three pages of output contains the input data specified in the input file.

Then, the code plots the following quantities as a function of the undulator length:

- Total emitted power (See Fig. 1), both in a linear and semilog scale.

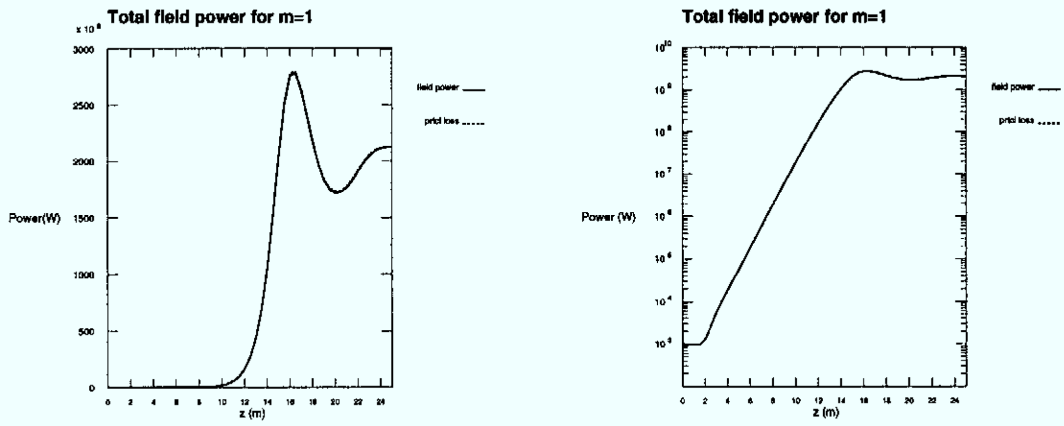

Figure 1: The emitted intensity along the undulator for the input deck presented here.

- Instantaneous and average gain (See Fig. 2).

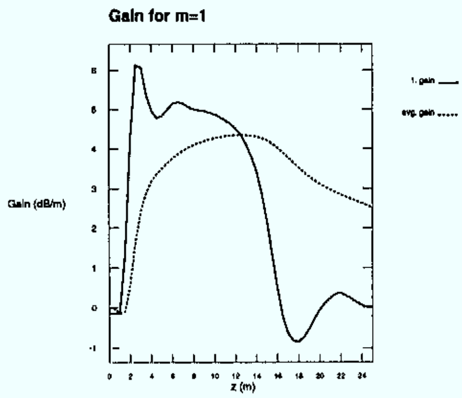

Figure 2: Instantaneous and average gain (in dB/m) along the undulator.

- The bunching parameter (See Fig. 3):
  - Linear scale. The real part, imaginary part and absolute value of the bunching parameter $b \equiv \langle \exp i\theta \rangle$ are shown.
  - Logarithmic scale. Only the absolute value of b is displayed.
- The mode contents of the radiation field (See Fig.4), both in a linear and semilog plot, showing the TEM00, TEM01 and TEM02 modes.
- The rms beam envelopes (x and y).
- The rms energy spread.
- The trapped particle percentage.

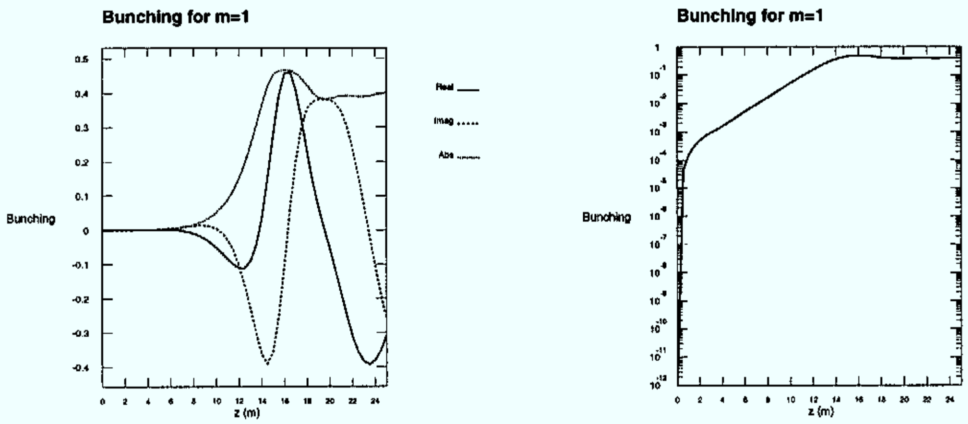

Figure 3: The bunching parameter along the undulator.

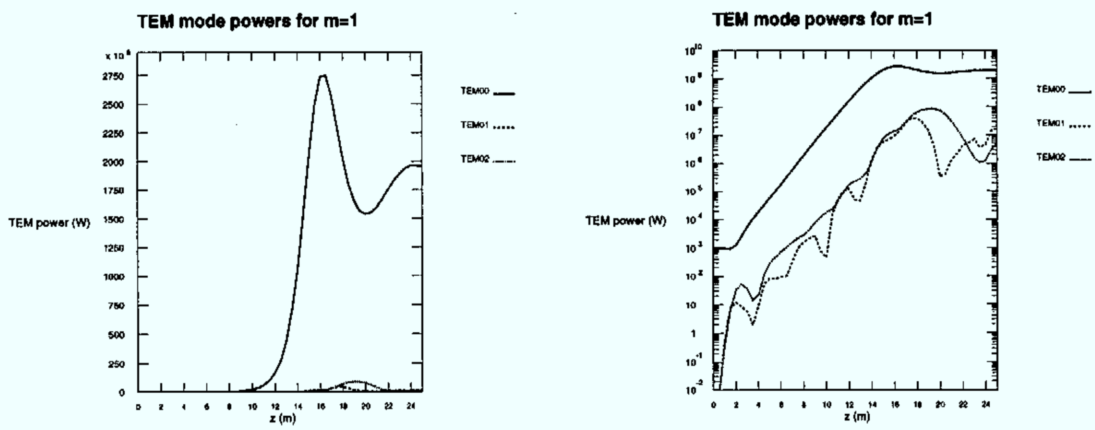

Figure 4: The modal contents of the radiated fiels along the undulator.

After these graphs the program displays the longitudinal phase space plots at some fixed position along the undulator. (See Fig. 5). The particles are binned in three phase space diagrams according to their radial position. The leftmost diagram (the most populated) shows the inner region of the electron beam, the rigthmost the outer.

References

[1] E.T. Scharlemann and W.M. Fawley, in *Modeling and Simulation of Optoelectronic Systems*, SPIE, vol. 642 (1986), p. 1;

[2] R.A. Jong, W.M. Fawley and E.T. Scharlemann, in *Modeling and Simulation of Laser Systems*, SPIE, vol. 1045 (1989), p. 18;

[3] R. Bonifacio, L. De Salvo, P. Pierini, E.T. Scharlemann, *Nucl. Instrum. and Methods*, **A296** (1990), pp. 787-790;

[4] E.T. Scharlemann, *Jour. of Appl. Physics*, **58** (1985), p. 2154;

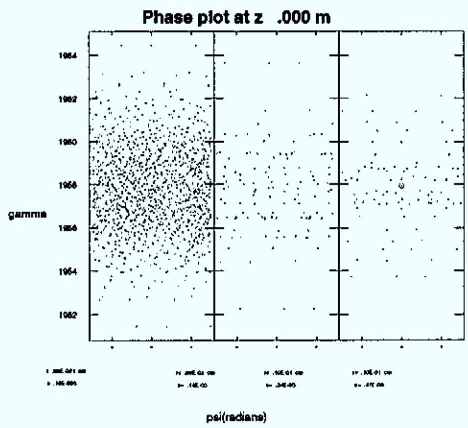

Figure 5: Beam longitudinal phase space at the beginning of the undulator. Three radial regions of the beam are shown.

[5] E.T. Scharlemann, W.M. Fawley, B.R. Anderson, T.J. Orzechoswski, *Nucl. Instrum. and Methods*, **A250** (1986), p. 150;

[6] W.M. Fawley, D. Prosnitz, E.T. Scharlemann, *Phys. Rev. A*, **30** (1984), p. 2472;

[7] E.T. Scharlemann, in *Laser Handbook*, Vol. **6**, edited by W.B. Colson, C. Pellegrini and A. Renieri.

[8] *"HIGZ", High Level Interface to Graphics and Zebra*, User's Guide, CERN Program Library Long Writeups Q120 and Y251, CN Division, CERN Geneva, Switzerland.