BROOKHAVEN
NATIONAL LABORATORY

BNL-77074-2006-IR

The TRANFT User's Manual

Michael Blaskiewicz

August 2006

Collider-Accelerator Department

Brookhaven National Laboratory
P.O. Box 5000
Upton, NY 11973-5000
www.bnl.gov

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.

The TRANFT User's Manual

M. Blaskiewicz*
BNL, Upton NY 11973, USA

The Fortran program TRANFT simulates transverse instabilities in circular accelerators using fast Fourier transform algorithms. It may be used for any particle type. Forces from transverse wakefields, longitudinal wakefields, and transverse detuning wakes are included, with linear transverse space charge forces included as a special case. This note describes the algorithms and their implementation in TRANFT.

I. INTRODUCTION AND THEORY

Coherent instabilities are of significant concern for a wide variety of planned and existing accelerators. The theory of these phenomena has been advancing steadily for decades [1–14] and, quite recently, a crucial piece of the puzzle for transverse instabilities was found [15]. A theoretical treatment involving all the relevant pieces appears very difficult whereas simulation using particle tracking is conceptually straightforward [16].

The algorithm involves single particle evolution and multi-particle kicks. First consider the single particle motion. The single particle longitudinal update for one turn is given by

$$\bar{\epsilon} = \epsilon + \frac{q}{mc^2}[V(\tau) - V_s] + \delta\epsilon - T_0\epsilon/T_r \tag{1}$$

$$\bar{\tau} = \tau + \frac{T_0\eta}{\beta^2\gamma_0}\bar{\epsilon} \tag{2}$$

where τ is the arrival time of the particle with respect to the synchronous phase, $\epsilon = \gamma - \gamma_0$ is proportional to the energy deviation, γ_0 is the reference Lorentz factor for a particle of mass m and charge q, $V(\tau)$ is the RF voltage, V_s is the synchronous voltage due to both acceleration and radiation, $\beta = v/c$, T_0 is the revolution period, η is the frequency slip factor, $\delta\epsilon$ is a quantum excitation random kick, T_r is the longitudinal radiation damping time, and the updated variables are $\bar{\tau}$ and $\bar{\epsilon}$.

Only one transverse variable is considered and it will be referred to as x. The single particle transverse update, without radiation, for one turn is

$$\bar{x} = x\cos\psi + p\sin\psi \tag{3}$$

$$\bar{p} = -x\sin\psi + p\cos\psi \tag{4}$$

$$\psi = \psi_0 + \frac{2\pi\xi}{\beta^2\gamma_0}\epsilon \tag{5}$$

where p is the transverse momentum variable, ψ_0 is the on-momentum phase advance, and ξ is the chromaticity. Transverse radiation damping and quantum excitation are also included,

$$\bar{x} = -\frac{T_0}{T_x}x + \delta x \tag{6}$$

$$\bar{p} = -\frac{T_0}{T_x}p + \delta p, \tag{7}$$

where T_x is the transverse radiation damping time, and δx and δp are random variables. While equations (1) though (7) are written for one turn, TRANFT allows the user to choose the number of updates per turn.

The multiparticle forces are associated with three Green's functions that are referred to as wake potentials. The longitudinal voltage is

$$V_s(t) = -\int_{-\tau_b}^{\tau_b} W_s(\tau) I_b(t-\tau) d\tau, \tag{8}$$

*Electronic address: `blaskiewicz@bnl.gov`

where τ_b is the bunch length, $W_s(\tau)$ is the longitudinal wake potential, and $I_b(t)$ is the instantaneous beam current. Note that $I_b(t)$ is the linear superposition of the current impulses from each of the individual macro-particles. The transverse voltage is driven by two terms. The short range term is

$$V_x(x,t) = \int_{-\tau_b}^{\tau_b} \left[xW_d(\tau)I_b(t-\tau) + W_x(\tau)D_x(t-\tau)\right] d\tau, \tag{9}$$

where $W_d(\tau)$ will be called the detuning wake [17–19], $W_x(\tau)$ is the usual transverse wake potential, and $D_x(t)$ is the instantaneous dipole density. Note that $D_x(t)$ is the product of the instantaneous current and the instantaneous value of x.

A second term in the transverse force is included to account for multi-bunch effects. Each particle in the bunch receives the same transverse kick

$$V_x = Re\left\{[X+iP]\sum_{k=1}^{\infty} W_x(kT_0/M)e^{i(\psi_0 - 2\pi s)k/M}\right\}, \tag{10}$$

Where $X = \int D_x(t)dt$ and $P = \int D_p dt$ are the total dipole moments in x and p. It is assumed that there are M identical, equally space bunches interacting with coupled bunch mode number s. In actual accelerators there is usually a gap in the bunch train. For rigid modes the growth rate for a symmetric fill is never smaller than the growth rate for a partial fill[20]. For an uneven fill and arbitrary modes, one can prove that the largest magnitude tune shift for the symmetric fill is never smaller than the largest magnitude tune shift in an uneven fill[21]. With the typical error bars associated with accelerator impedances the error incurred by using (10) is probably benign.

II. ALGORITHMS

All calculations, but wakefields, are done to machine precision using straighforward implementations of the equations already introduced. As an example of the wakefield calculations consider the longitudinal voltage, equation (8). Taking the instantaneous current to be a series of delta functions one obtains the first order approximation

$$V_{s,1}(t) = -\sum_{k=1}^{N} \hat{q}W_s(t-\tau_k), \tag{11}$$

where there are N macroparticles of charge $\hat{q}$. There are two problems with using (11) as it stands [22]. Firstly, since N is small compared to the actual number of particles within the bunch, there can be large statistical fluctuations in the applied voltage. This is especially worrisome since short range wake potentials tend to be very large. The net effect is that one can have a significant, unphysical, blow-up in the longitudinal emittance. The second problem, not fully unrelated to the first, is caused by the discrete time steps between updates. A typical particle makes a step $2\pi Q_s\sigma_\tau$ each turn, where Q_s is the synchrotron tune and σ_τ the rms bunch length. When length scales less than $2\pi Q_s\sigma_\tau$ are important in the wake potential then it is possible for macro-particles to pass each other without interacting via the short range wake. Both of these problems can be alleviated by convolving (11) with a smoothing function of characteristic scale $\Delta\tau \gtrsim 2\pi Q_s\sigma_\tau$ and, since convolution is commutative and associative, we may consider a smoothed wake potential $\hat{W}_s(\tau)$. This leads to a second approximation for the voltage that is physically reasonable

$$V_{s,2}(t) = -\sum_{k=1}^{N} \hat{q}\hat{W}_s(t-\tau_k). \tag{12}$$

To update the particles equation (12) needs to be evaluated for $t = \tau_1, \ldots \tau_N$ and a naive algorithm requires $O(N^2)$ operations. Instead of incurring this computational penalty, it was decided to use an approximate technique. First, a uniform grid of points spaced by $\delta t \lesssim \Delta\tau/5$ is generated. Next, the macroparticles are placed on the grid via linear interpolation. A fast Fourier transform (FFT) is applied, multiplied by the FFT of $\hat{W}_s$, and an inverse FFT completes the calculation of $V_{s,2}$. The total number of grid points is a power of 2 and the total grid length is at least twice the total bunch length to eliminate "phantom" of "ghost" forces [23]. There are two sources of error involved with this computation. The first is due to the application of linear interpolation in gridding the system and the second involves using numerical integration (via FFT) to evaluate the sums. The net effect is easily tested by cutting δt in half and rerunning the simulation until the answer converges.

III. IMPLEMENTATION AND USE

The fortran source code resides in the file `tranft.f` and is liberally commented. Common blocks and some parameters are in `tranft_c.f`. The code employs some routines from numerical recipes[24]. Some small modifications were needed for the code to compile. The top of subroutine `SORT2.FOR` needs to be changed from

```
      SUBROUTINE SORT2(N,RA,RB)
      DIMENSION RA(N),RB(N)
```

to

```
      SUBROUTINE SORT2(N,RA,RB)
      DIMENSION RA(N)
C NEXT DECLARATIONS FOR INTEGER TAG-ALONG ARRAY
C BASED ON QUICKSORT
      INTEGER RB(N),RRB
```

The function `ran3(idum)` needs to be changed from

```
      FUNCTION RAN3(IDUM)
C         IMPLICIT REAL*4(M)
C         PARAMETER (MBIG=4000000.,MSEED=1618033.,MZ=0.,FAC=2.5E-7)
      PARAMETER (MBIG=1000000000,MSEED=161803398,MZ=0,FAC=1.E-9)
      DIMENSION MA(55)
```

to

```
       FUNCTION RAN3(IDUM)
c         IMPLICIT REAL*4(M)
c         PARAMETER (MBIG=4000000.,MSEED=1618033.,MZ=0.,FAC=2.5E-7)
      PARAMETER (MBIG=1000000000,MSEED=161803398,MZ=0,FAC=1.E-9)
      common/ran3stuff/ MA(55),inext,inextp
```

The subroutine `four1.for` compiled without difficulty. The code may be obtained by contacting the author `blaskiewicz@bnl.gov`. The code has been tested with the intel fortran compiler `ifort` and the generic linux compiler g77. The compilation commands for ifort and g77 are

```
ifort  -o tranft numrec.f tranft.f
g77  -o tranft numrec.f tranft.f
```

where the three numerical recipes routines are in `numrec.f` and the executable is `tranft`.

TRANFT uses three input files. The file `tranft.in` contains data with regard to the beam and the lattice while `imped3.in` contains data with regard to collective forces. Numerically calculated wakefields can be incorporated via `impedfile.dat`. The three files are not entirely independent. A typical `tranft.in` is given by

```
 1500  100   50   12763 2 nturns,ndim,nwrite,iseed,nperturn  nsls2 16 july 06
52.13 780.3 5871.  -3.7e6 1300 1. 3 gammat,circ,gamma0,vrf,nharm,radharm2,nharm2
16.28 3.e-6  1.e-6  5. 5.  0  tunex,ampx,xinject,chrom_init,chrom_final,dispavg
15.e9 5.446e-4 1 5.e-12 4 1.e-10 -0.322 1000 pnumber,aatom,qatom,taupart,power,tauhat,phisynch,nturnon
4000 1300 17  6.e-10  14.e-10  nresamp,mbunch,mode,tlob,thib
 13.e-3  3.e-6  6.5e-3 5.4     tradperp,sigperp,tradlong,siglong  trev = 2.60 us
```

The code reads only the numbers on the lines (free format) and the character strings to the right are the corresponding variables in the source code. No text is required, but the author finds it very helpful. For the case above `nturns = 1500` is the total number of turns simulated. The parameter `ndim=100` controls the number of macro-particles. Writes to the screen and output files are done every `nwrite=50` turns. Setting `nwrite` to a negative number calculates more beam properties every `|nwrite|` turns and increases computional time. The random number generator is `ran3` from numerical recipes[24] and `iseed =12763` is the random seed. Radiation damping and quantum excitation are always applied once per turn. All other algorithms are applied `nperturn=2` times per turn. The comments `nsls2 16 july 06` and `trev = 2.60 us` are reminders to the user. Additional lines after the input file are allowed too.

The transition Lorentz factor is `gammat`, the machine circumference is `circ` meters. The central Lorentz factor of the beam is `gamma0`. The primary RF voltage amplitude is `vrf` volts and `nharm` is the primary harmonic number. For stable motion, the product vrf*qatom is positive below transition and negative above. Since electron clouds are not included, switching the signs of vrf and qatom lead to identical dynamics. The ratio of higher harmonic voltage to primary is radharm2 and nharm2 is the ratio of the higher harmonic to the main harmonic. For radharm2=0 there is no higher harmonic voltage and for radharm2=1. the RF voltage is cubic to leading order in τ about the stable fixed point. For values between 0 and 1 the amplitude of the second harmonic is modified but its phase is unchanged.

The betatron tune is tunex and the average beta function is defined to be circ/(2*pi*tunex) with pi = 3.141.... The rms transverse amplitude at average beta is ampx meters and the particle is (eventually) given a kick of amplitude xinject. The unnormalized initial and final chromaticities are chrom_init,chrom_final and the simulation linearly interpolates in time between the two. This mimicks the rapid lattice changes that can accompany transition and one should set chrom_init=chrom_final to emulate steady state systems. The average dispersion referenced to the average beta is dispavg. It is best left 0.

The actual, physical bunch contains pnumber particles of atomic mass aatom and atomic number qatom. Set aatom $= m_e/m_p$ for electrons. The wake potential smoothing is controlled by the parameter taupart. The smoothing employs a gaussian function with equivalent length taupart $= \sqrt{2\pi}\sigma$ seconds. The initial bunch has half length tauhat seconds and its shape is controlled by power. The initial bunch is matched to a linear rf force with shape $[1 - (t/\texttt{taupart})^2]^{\texttt{power}}$. The synchronous phase is phisynch radians. Over the course of nturnon turns the rf is smoothly changed from linear to a sum of appropriate sinusoids. The longitudinal wake field is smoothly turned on during this same time. Then, a transverse kick of amplitude xinject is given and both the transverse wakes are turned on. With radharm2 =1 a surprisingly large number of turns can be required for nearly adiabatic rf turn on.

The number of grid points used for the FFT is the smallest power of 2 that is at least as large as nresamp. The number of bunches used for coupled bunch modes is mbunch and mode is the mode number. This parameter is s in equation (10). To minimize problems with truncation the stable fixed point is at half an rf period $T_{rf}/2$. The interval used for the entire FFT domain is [tlob,thib]. The code will generate unmistakable warnings if the bunch leaves the calculation interval, or if the bunch gets too long. One gains no increase in accuracy by having thib − tlob greater than twice the bunch length, but a factor of 3 leaves a good safety margin without introducing a large computational overhead.

The transverse radiation damping time is tradperp, called T_x in equation (6). The rms, equilibrium beam size at average beta is sigperp. The longitudinal radiation damping time is tradlong, called T_r in equation (1) and sigperp is the rms energy spread in units of γ. Setting tradperp < 0 turns off radiation damping and quantum excitation.

A sample imped3.in is

```
20. 212.   1.7e-8    0.0025  0. 0.5  slenx,slens,rhoe,bpipe,wstep,detunefrac  uses beta weighting
0   1.e-12 0. 0. wallinduct,twall,scimped,scabrat
1 1   npolex,npoles
0 2.17e17  0.94e11  1.33e11  transverse 1.e6 Ohm/m, 30 GHz, Q=1
5.654e+15 -3.264e+15 9.425e10 1.632e11 longitudinal 30.e3 Ohm/m, 30 GHz, q=1
```

The first line defines all the resistive wall quantities. The effective length for the transverse impedance and transverse detuning wake is slenx meters. Remember that all quantities are referenced to the average beta function $\texttt{circ}/(2*\texttt{pi}*\texttt{tunex})$ and that the relevant quantity for stability calculations is the beta weighted transverse impedance. The effective length for the longitudinal resistive wall impedance is slens meters. The electrrical resistivity of of the wall material is rhoe ohm-meter. The transverse wake potential is calculated assuming a round pipe of radius bpipe meters. The transverse detuning wake is a fixed fraction detunefrac of the longitudinal wake. For vertical instabilities with a small vertical aperture and large horizontal aperture detunefrac = 0.5[18]. The parameter wstep is the value for the transverse step function wake in volts per coulomb per meter.

The broad band wall inductance is wallinduct in Ohm-seconds. The very short range nature of this force warrants additional smoothing[25] and the rms length of the additional smoothing is twall seconds. The net voltage kick per turn is given by

$$V = -\texttt{wallinduct}\,\frac{dI_{beam}}{dt}\bigg|_{\texttt{twall}},$$

where the subscript on the time derivative of the beam current denotes smoothing with a gaussian pulse of rms length twall seconds. For longitudinal space charge[26]

$$\texttt{wallinduct} = -\frac{T_0}{2\pi}\left|\frac{Z}{n}\right|_{sc} = -\frac{T_0}{2\pi}\frac{Z_0}{\beta\gamma^2}\ln\left(\frac{b}{1.5\sigma_b}\right), \tag{13}$$

where the beam has relativistic parameters β, γ, an rms (round) transverse size σ_b in a round pipe of radius b and $Z_0 = 377\Omega$. The magnitude of the direct, transverse space charge impedance is `scimped` in ohms per meter [26, 27],

$$\texttt{scimped} = \frac{Z_0 \texttt{circ}}{2\pi\beta^2\gamma^2 a^2} \tag{14}$$

where $a = 2\sigma_b$ is the radius of a uniform equivalent beam[27]. The parameter `scabrat` $= a/b$ is the ratio of the equivalent beam radius to the beam pipe radius and is used to calculate the coherent space charge force. The last components defined in `imped3.in` are the resonators. There are `npolex` transverse resonators followed by `npoles` longitudinal resonators. Each resonator is defined by 4 real (2 complex) numbers. The input parameters on each line are w_r, w_i in Ohms per second; and α and ω_r in inverse seconds. The wake potential vanishes for $t < 0$ and is $Re(w_r + iw_i)\exp(-\alpha t - i\omega_r t)$ for $t > 0$. One may set either or both of `npolex,npoles` to zero and neglect resonator contributions to that plane.

The final input file is `impedfile.dat`. The top few lines of such a file are

```
  3.137e-12 -242 114    dtfile,nptfile,idex0
 -3.545098E-10  8.046247E+12  0.000000E+00  4.023123E+12  filex(k),files(k),filed(k)
 -3.513726E-10  4.601705E+12  0.000000E+00  2.300852E+12
```

The file defines three arrays of `nptfile` elements. The wakepotentials defined in these arrays are added to those calculated above. Setting `nptfile < 0` causes the subroutine to exit and no lines after the first are read. The array elements are spaced in time by `dtfile` seconds. The index corresponding to time equals zero is `index0`. It is strongly suggested that at least one negative time be included, even if all wake potentials vanish at that time. The second line on contain the array elements. The first column is the time lag in seconds. While this column is ignored by the code the author finds it useful for comparison purposes. columns 2,3 and 4 are the additional values of W_x, W_s and W_d.

Making the substitution `nwrite = -5` in `tranft.in` and leaving all else unchanged produces the following on the screen

```
[blaskiew@blaskiew r04]$ ../tranft
 nresamp,nfft        4000        4096
 eta =   3.6795126E-04
 synchronous voltage =    1170918.
  amp of de/e =   6.1927512E-03
  synchrotron frequency =    3626.553
  revolution period =   2.6027351E-06
 reference beta function = circ/(2*pi*tunex) =    7.628292
 tbin/sigtsmooth=  9.7915158E-02
  zcbm =  (3.5463145E+15,-4.6984043E+15)
  Im(Q) for cold bunches =   2.2852253E-03
  dpwake for 1 meter offset with step wake = 1
  6.1044420E-18
  Oide-Yokoya intensity factor for electrons I =   1.4685191E-14
 avgchrom, chrom, avgtune =  -5.8565885E-03   5.000000       5.6935042E-02
kturns,np,log10(csfull),coherecs        0     31428 -1.074E+01  2.323E-14
 avgchrom, chrom, avgtune =    4.956137       5.000000       0.2799785
kturns,np,log10(csfull),coherecs        5     31428 -1.074E+01  2.449E-14
```

The code was run in the subdirectory `r04` with the executable residing in the directory above. All input and output files are in `r04` making this an easy way to keep track of several runs. The first line of output shows that the number of samples used for the FFTs has been raised to 4096. The second line is the frequency slip factor $\eta = 1/\gamma_T^2 - 1/\gamma^2$. Line 3 gives the first harmonic synchronous voltage in volts. Line 4 is the maximum, fractional energy deviation for the initial distribution. Next comes the synchrotron frequency in Hz and the revolution period in seconds. Next is the reference beta function for transverse coherent kicks. Next is the ratio of bin length to the rms of the gaussian smoothing length. The parameter `zcbm` is the sum of the wakes in equation (10). Next is the imaginary part of the tune one expects if the bunches are rigid. The variable `dpwake` is the kick in p from transverse wakefields. The last message before entering the update loop is the Oide-Yokoya intensity factor I[13] in MKS units.

With `nwrite < 0` the betatron tunes of the individual particles are calculated. To simplify notation consider a single particle and let x_k, p_k be its coordinates at the end of turn k. For purely linear motion $x_{k+1} = ax_k + bp_k$ and

$p_{k+1} = cx_k + dp_k$, where a, b, c, d are constants. Now consider the sum

$$\Gamma(a,b,c,d) = \sum_{k=m+1}^{m+\ell-1} \left(x_{k+1} - ax_k - bp_k\right)^2 + \left(p_{k+1} - cx_k - dp_k\right)^2 . \tag{15}$$

Minimizing Γ simultaneously with respect to a, b, c and d gives a least squares fit to the transfer matrix best describing the particle between turns $m+1$ and $m+\ell-1$. Given the transfer matrix the single particle tune q satisfies $2\cos(2\pi q) = (a+d)/(ad-bc)$. The sign of q is the same as the sign of b. In TRANFT $\ell = |\texttt{nwrite}| - 1$ and m is an integer multiple of `nwrite`. Taking correlations between ϵ and q yields the chromaticity. On the first turn, 0, the arrays are not full so the data are meaningfull only for `kturns > 0` . The setpoint chromaticity is also displayed.

In all cases the turn number (`kturns`) and number of macroparticles (`np`) are displayed. There are also two quantities which allow the user to monitor the progress of a transverse instability. The parameter `csfull` is the average over particles of $x^2 + p^2$. This parameter includes information about the emittance as well as any coherent motion. The parameter `coherecs` is tailored to be a sensitive indicator of instability. Let $\bar{x}(t)$, and $\bar{p}(t)$ be smoothed average values of x and p as the bunch passes and let $I(t)$ be the smooth current pulse, then

$$\texttt{coherecs} = \frac{\int_{bunch} dtI(t) \left[\bar{x}^2(t) + \bar{p}^2(t)\right]}{\int_{bunch} dtI(t)}. \tag{16}$$

Along with writes to the screen there are 8 output files. The wake potentials in various forms are in `w.out`, `z.out`, and `ws.out`. The raw wake potentials before smoothing are in `w.out`. There are 4 columns: the time, $W_x(\tau)$, $W_s(\tau)$, and $W_d(\tau)$. All are in MKS units. The file `z.out` contains the Fourier transforms of the wake potentials and the frequency window used for smoothing. Defining Fourier transforms as

$$\tilde{F}(f) = \int_{-\infty}^{\infty} F(t)\exp(2\pi i f t)dt,$$

the columns are f in Hz, $Re(\tilde{W}_x)$, $Im(\tilde{W}_x)$, $\exp(-\pi[f\texttt{taupart}]^2)$, $Re(\tilde{W}_s)$, $Im(\tilde{W}_s)$, $Re(\tilde{W}_d)$, and $Im(\tilde{W}_d)$ all in MKS. The final wake field file is `ws.out`. This file contains the smoothed wake potentials in the same format as `w.out`.

A note concerning the resistive wall quantities is in order. Only the simplest low frequency approximations for resistive wall quantities have been used, resulting in continuum forms that have singularities of various types. The transverse resistive wall wake is taken to be

$$W_x(\tau) = H(\tau)\frac{cL_x}{\pi b^3}\sqrt{\frac{Z_0\rho}{\pi c\tau}}, \tag{17}$$

where $Z_0 = \mu_0 c = 376.74\Omega$ is the impedance of free space, $L_x = \texttt{slenx}$, $b = \texttt{rpipe}$, $\rho = \texttt{rhoe}$, and $H(\tau)$ is the Heaviside function. The value of W_x assigned to time $n\delta t$ is

$$W_{x,n} = \frac{1}{\delta t}\int_{(n-1/2)\delta t}^{(n+1/2)\delta t} W_x(t)dt. \tag{18}$$

This guarantees that the low frequency impedance behaves correctly and that the smoothed quantities converge rapidly to their continuum values. When the pipe radius or beta function vary with machine azimuth the appropriate value for $L_x = \texttt{slenx}$ satisfies

$$\bar{\beta}\frac{L_x\sqrt{\rho}}{b^3} = \oint \beta(s)\frac{\sqrt{\rho(s)}}{b(s)^3}ds, \tag{19}$$

where $\bar{\beta} = \texttt{circ}/(2*\texttt{pi}*\texttt{tunex})$ is the average beta function. The longitudinal resisitve wall wake potential it taken to be

$$W_s(\tau) = \frac{d}{d\tau}\left\{H(\tau)\frac{Z_0L_s}{2\pi b}\sqrt{\frac{\rho Z_0}{\pi c\tau}}\right\}, \tag{20}$$

where $L_s = \texttt{slens}$. $W_s(\tau)$ has a functional near $\tau = 0$, but it is trivial to integrate as long as $\tau = 0$ is not one of the end points. The code does this in exact analogy to equation (18). The analog of equation (19) is

$$\frac{L_s\sqrt{\rho}}{b} = \oint \frac{\sqrt{\rho(s)}}{b(s)} ds. \tag{21}$$

Two output files describe the instantaneous dynamics of the beam. The file `res.den` is over written every |`nwrite`| turns and contains gridded data. The file `tran.full` is over written every |`nwrite`| turns if `nwrite < 0`. Both are always written when the simulation ends. There are 8 columns of data in `res.den`. The first column is arrival time in seconds, with $T_{rf}/2$ corresponding to the stable synchronous phase. The second column is the instantaneous line density normalized to be equal to the number of macroparticles in $[t - \texttt{taupart}/2, t + \texttt{taupart}/2]$, which is useful for judging the statistical accuracy of the simulation. Columns 3 and 4 are $\bar{x}(t)$ and $\bar{p}(t)$ in meters, as defined in equation (16). Column 5 is the instantaneous current in amperes. Column 6 is the value of the kick to p due to W_x, column 7 is the longitudinal kick due to W_s in units of γ, and column 8 is the detuning kick. Multiplying column 8 by $-\texttt{nperturn}/4\pi$ yields the tune shift due to W_d as a function of longitudinal position within the bunch.

The second file describing instantaneous dynamics is tran.full. The first 4 columns are x, p, t, ϵ for each of the macroparticles, They are ordered in t. Column 5 is the betatron tune of the particle. Trust this only if `nwrite < 0` and $|\texttt{nwrite}| \ll 1/Q_s$. The last column is the kick in γ due to the RF voltage.

There are 3 files containing a summary of the beam dynamics over the whole simulation. The file `tim.out` has three columns. The first is the time, the second is the average value of x and the third is the average value of p, all in MKS. The file `spec.out` contains the Fourier transform of `tim.out`. The first column is frequency in tune units, the second and third columns are the real and imaginary parts of the FFTs of `tim.out`. The FFT is of $< x + ip >$ so the tune is resolved on the full span between 0 and 1. The last output file is `csmon.out`, has 8 columns. Column 1 is the turn number. Column 2 is $< x^2 + p^2 >$ averaged over the beam. Column 3 is the log base 10 of column 2. Column 4 is `coherecs` defined in equation (16). Column 5 is the rms bunch length in seconds. Column 6 is the rms energy spread in units of γ. Column 7 is the effective synchrotron tune defined using the ratio of the rms energy spread to the rms bunch length. Column 8 is $< t >$ averaged over the bunch.

Acknowledgements

Thanks to Michael Borland, Yong-Chul Chae, Sam Krinsky, Ryutaro Nagoaka, Thomas Roser, and Todd Satogata for useful discussions and encouragement.

[1] F.J. Sacherer, CERN/SI-BR/72-5, (1972).
[2] F.J. Sacherer, IEEE-TNS **NS-24**, # 3, p1393, (1977).
[3] A.W. Chao & C.Y. Yao, PEP-Note 321, (1979).
[4] J.L. Laclare, CERN 87-03, p 264 (1987).
[5] J.M. Wang, BNL 39057, (1986).
[6] A.W. Chao *Physics of Collective Beam Instabilities in High Energy Accelerators* Wiley, (1993).
[7] Y. Chin, K. Satoh & K. Yokoya, *Particle Accelerators* **13** , p45 (1983).
[8] Y. Chin, NIMA , **215** p501, (1983)
[9] K. Satoh & Y. Chin, *Nuclear Instruments and Methods* **207**, p309 (1983).
[10] G. Besnier, D. Brandt & B. Zotter, *Particle Accelerators* **17**, p51 (1985).
[11] S. Krinsky & J.M. Wang *Particle Accelerators* **17** , p109 (1985).
[12] Y. Chin, CERN SPS/85-2 (1985).
[13] K. Oide & K. Yokoya, KEK 90-10 (1990).
[14] M. Blaskiewicz, W.T. Weng PRE **50** p4030, (1994)
[15] P. Kernel, R. Nagaoka, J.-L. Revol, & G. Besnier p1133, EPAC00.
[16] R.H. Siemann, AIP Conf 127, p368, (1983).
[17] R.L. Gluckstern, J. van Zeijts, B. Zotter, PRE **47** # 1, p656 (1993).
[18] K. Yokoya *Particle Accelerators* **41** , p221 (1993).
[19] A. Burov & V. Danilov, PRL **82** , #11 p2286 (1999).
[20] R.D. Kohaupt DESY-85-139 (1985).
[21] J.S. Berg, CERN SL 97-72 (AP) (1997).
[22] Shane Rupert Koscielniak, PhD Thesis, University of Oxford, (1987).
[23] R.W. Hockney, J.W. Eastwood *Computer Simulation using Particles* Hilger, Bristol, (1988).

[24] W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery *Numerical Recipes in Fortran 77* Cambridge University Press (1992).

[25] J. Wei *PhD thesis* State University of New York at Stony Brook (1990).

[26] A.W. Chao, M. Tigner *eds Handbook of Accelerator Physics and Engineering* World Scientific (1999).

[27] M. Blaskiewicz PRSTAB 4, 044202 (2001).