QUINDI - A CODE TO SIMULATE COHERENT EMISSION FROM BENDING SYSTEMS

D. Schiller, S. Reiche, M. Ruelas, UCLA, Los Angeles, CA 90095, USA

Abstract

We present a newly developed code, QUINDI, to address the numerical challenge of calculating the radiation spectra from electron bunches in bending magnet systems. This code provides a better tool for designing diagnostic systems such as bunch length monitors in magnetic chicanes. The program calculates emission on a first principle basis, combining the dominant emission processes in a bending magnet system – edge and synchrotron radiation. The core algorithm is based on the Lienard-Wiechert potential and utilizes parallel computer architecture to cover complete electron beam distributions with a high resolution spatial grid. The program models the coherence level of the emitted radiation from the electron bunch, focusing on long frequency components.

INTRODUCTION

In beam physics, many diagnostics are based on the radiation from an electron bunch, however, not many codes have been developed to model these instruments. Most codes support self-interaction by radiation between particles, but not the radiation as seen by a detector fixed in space. QUINDI addresses this lack of numerical codes, providing a simulation tool that mimics a detector or windowed port in an experiment. QUINDI was originally coded to simulate the ATF compressor experiment at Brookaven National Laboratory[1].

QUINDI was designed to be as modular as possible, so that alternate modules can be used for different physics models. There are three major steps to the computational process of QUINDI: tracking, electric field calculation, and radiation processing (see Fig.1).

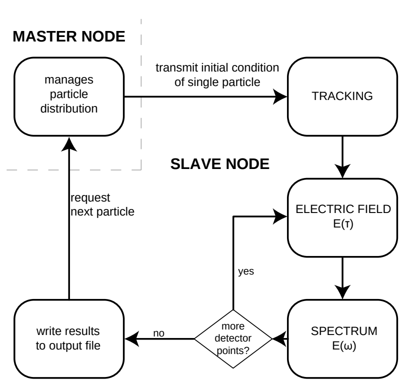

Figure 1: Flow of processes for QUINDI.

After a particle's initial state is transmitted by the master node to a slave node, its trajectory is determined, and the positions, velocities, and accelerations at each step are recorded. The tracking algorithm approximates the trajectory by a circle for each step, assuming the presence of only a magnetic field.

After tracking, the slave node begins calculating the observed electric field for each grid point on the detector plane while the electron moves along its trajectory. The code utilizes the acceleration field of the Lienard-Wiechert potential, which makes use of the position, velocity, and acceleration data gathered in the tracking module.

Finally, the electric field data is passed on to the last part of the code. The spectrum is calculated individually for each grid point. Two main steps occur in the spectral routine. First, we perform a Fourier transform on the electric field data, which transforms the data into the frequency domain. Second, the radiation from each individual particle is accumulated to obtain the complete spectrum for each grid point; both the amplitude and phase are accounted for to maintain the coherence of the signal.

TRACKING

QUINDI performs the particle tracking based on the Lorentz force. Tracking is done once for each particle and the result is saved. QUINDI takes an object-oriented approach to describe the magnetic lattice. Many codes require each particle to sequentially move through every magnet. QUINDI, instead, defines each magnet as an object and determines if the particle at that time is inside of it, inside of the fringe, or not inside the magnetic field at all. This approach allows for a more sophisticated geometry of the magnets, such as slanted faces. For calculation, each trajectory step represents a curve which can be approximated as the arc of a circle. (see Fig.2)

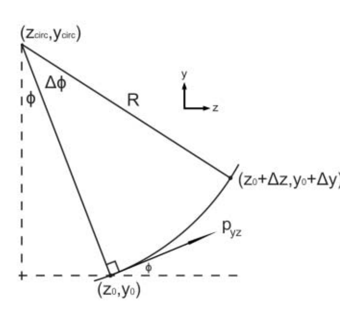

Figure 2: Geometry used for calculating position and momentum.

1-4244-0917-9/07/$25.00

We assume, for each step, the particle begins at (x_0, y_0, z_0). QUINDI models the trajectory in two dimensions: the z-axis is the longitudinal direction and we ignore acceleration effects in the x-direction because the magnetic field component points in the same direction (the fringe field is modeled in 2D). The longitudinal step size, Δz, is given as an input parameter.

We assume that since the motion is split up in steps in z (and not time),

$$z(z_0 + \Delta z) = z(z_0) + \Delta z. \tag{1}$$

The y-position, which lies in the plane of deflection, is calculated geometrically from

$$y(z_0 + \Delta z) = y_{circ} - R\cos(\phi + \Delta\phi), \tag{2}$$

where (z_{circ}, y_{circ}) is the origin of the circle (see Fig.2) with the bend radius $R = \frac{p_\perp}{eB}$, where $p_\perp = \sqrt{p_z^2 + p_y^2}$ is the momentum in the bending plane. We obtain ϕ from the trigonometrical relationship $\phi = \arctan\left(\frac{p_y}{p_z}\right)$. According to our geometry, we can also make use of

$$z(z_0 + \Delta z) = z_{circ} + R\sin(\phi + \Delta\phi). \tag{3}$$

Subtracting $z(z_0)$ from $z(z_0 + \Delta z)$ and comparing to Eq.1 yields

$$\Delta z = R[\sin(\phi + \Delta\phi) - \sin\phi], \tag{4}$$

and by the Taylor series approximation

$$\Delta\phi = \frac{\Delta z}{R\cos\phi}. \tag{5}$$

The new x-position is trivially calculated as

$$x(z_0 + \Delta z) = x(z_0) + \Delta t \cdot v_x, \tag{6}$$

with $v_x = c\beta_x$. We obtain Δt by calculating the time of flight along the arc with

$$\Delta t = \frac{R \cdot \Delta\phi}{c\beta_\perp}. \tag{7}$$

We record the time, t, by incrementing the previous time by Δt. The remaining problem is the calculation of the new momenta. Because we are ignoring acceleration in the x-direction, the x-momentum is preserved:

$$p_x(z_0 + \Delta z) = p_x(z_0). \tag{8}$$

The y-momentum is obtained geometrically:

$$p_y(z_0 + \Delta z) = \sqrt{p_y(z_0)^2 + p_z(z_0)^2} \cdot \sin(\phi + \Delta\phi). \tag{9}$$

We obtain the z-momentum through conservation of momentum in the y-z plane:

$$p_z(z_0 + \Delta z) = \sqrt{p_y(z_0)^2 + p_z(z_0)^2 - p_y(z_0 + \Delta z)^2}. \tag{10}$$

QUINDI possesses the ability to accept external tracking files which are passed on to the remaining calculations. For example, QUINDI currently supports using an external TREDI[2] tracking file, which is a more self-consistent approach since TREDI calculates self forces in the bunch. Tracking data from other codes could easily be imported if a variation on the tracking module is produced; any code that outputs position, momentum, and time data and is parsable can be used.

ELECTRIC FIELD

To calculate the emitted electric field from the trajectory, we use the acceleration field of the Lienard-Wiechert potential[3]:

$$E(\tau) = \frac{e^-}{c} \cdot \frac{\vec{n} \times [(\vec{n} - \vec{\beta}) \times \dot{\vec{\beta}}]}{R \cdot (1 - \vec{n} \cdot \vec{\beta})^3}, \tag{11}$$

where $\vec{n}$ is the unit vector and points from the particle's position along the trajectory to the detector point, R is the distance, and $\tau = t + \frac{R}{c}$. While $\vec{\beta}$ and $\dot{\vec{\beta}}$ remain unchanged, $\vec{n}$ changes with each grid point since no far field approximation is assumed. Due to the discreteness of the trajectory, the total electric field is described by a series of step functions, χ:

$$E(\tau) = \sum_j E(\tau_j) \cdot [\chi(\tau_j - \frac{\Delta\tau_j}{2}) - \chi(\tau_j + \frac{\Delta\tau_j}{2})], \tag{12}$$

which requires that $\Delta\tau_j$ is much smaller than the characteristic time of the analytical signal.

SPECTRUM

We obtain the field spectrum by performing a Fourier transform on the emitted electric field from each particle. We exclude an FFT algorithm because we are only interested in a relatively small frequency domain, and instead opted to perform a series of discrete transformations:

$$\begin{aligned} E(\omega) &= \int_{-\infty}^{+\infty} E(\tau) e^{i\omega\tau}\, d\tau; \\ &= \sum_j E(\tau_j) \int_{\tau_j - \frac{\Delta\tau_j}{2}}^{\tau_j + \frac{\Delta\tau_j}{2}} e^{i\omega\tau'}\, d\tau'; \\ &= \sum_j E(\tau_j) e^{i\omega\tau_j} \frac{2}{\omega} \sin\left(\omega \frac{\Delta\tau_j}{2}\right); \end{aligned} \tag{13}$$

where $\Delta\tau_j$ is the time difference between two "observed" field amplitudes. Because we focus only on low frequency components, the integration method of a step-like function yields valid results as long as the condition $\omega\frac{\Delta\tau}{2} << 1$ is fulfilled. The highest frequency to which we integrate is defined as a parameter in the main input file, as well as the total number of frequency cuts that are made.

INPUT/OUTPUT

QUINDI uses simple ASCII text files for the input. Relevant parameters are given their own lines in these files. The input files are parsed internally to extract the parameters. The parser is set up so that not all parameters are required to be defined in the main input file; if a parameter is not specified, then its default value will be used.

QUINDI output uses the HDF5[5] file format, which is both a fast and portable approach. HDF5 is a binary format that stores data in a hierarchal format, which allows selective parsing and the ability to manipulate data at any point in the program. These files aim to be self-describing, with units where appropriate in the data properties and with a clear, concise format. The QUINDI output file contains, first and foremost, the spectrum data which is divided up by grid point and frequency cut. We also store the initial and final phasespace of the particle distribution, as well as the trajectory for the very first particle so it can be easily analyzed for proper trajectory. Finally, we store some metadata about the run including the input parameters and the length of the run.

POST-PROCESSING

The HDF5 output file that QUINDI produces can be post-processed to extract further information. We have created a Matlab script called specGUI which analyzes the HDF5 file and extracts information in an easy-to-use fashion.

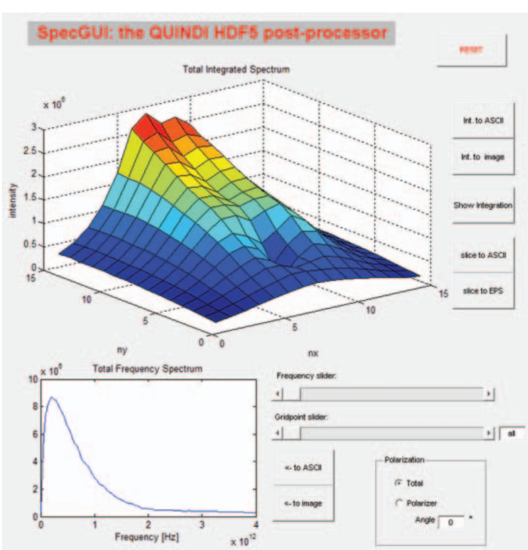

Figure 3: QUINDI output viewed with specGUI script.

The 3D plot at the top (see Fig.3) represents the field intensity as seen on the various grid points of the detector. The 2D plot at the bottom shows the total spectrum, integrated over the entire detector. This software can be used to view the intensity by frequency cut and is also capable of applying a polarizer of arbitrary angle to the spectrum.

BENCHMARKS

QUINDI runs have been completed to simulate the ATF compressor at BNL, and the results have been compared to the experimental data.

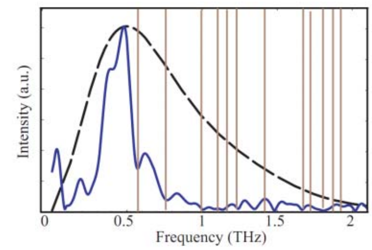

Figure 4: The dashed curve is the QUINDI results. The solid curve is the experimental data from BNL. The vertical lines are water absorption lines, corresponding to local minima in the data.[1]

Figure 4 shows the QUINDI data superimposed on the experimental data; the peaks of these graphs line up closely. The spectrum gathered from the actual experiment is affected by a few factors, including water vapor being present in the transport. These effects contribute to the finer variation in the graph of the experimental data, and QUINDI makes no attempt to model these environmental variables. However, the general character of the spectrum seems to align nicely. The 3D plot of Figure 3 shows the hollow structure which indicates the presence of edge radiation[4]. This structure has also been verified in the measurements of the ATF compressor experiment.

OUTLOOK

QUINDI is currently useful for modeling the radiation from electron bunches moving through bending magnets, utilizing a two-dimensional geometry for the trajectory calculation. In the future, we hope to increase the functionality of QUINDI in a variety of ways. First, we will include the ability to use three-dimensional magnetic lattices with more accurate fringing effects. There is interest in increasing the number of types of magnets that QUINDI can utilize, adding quadrupoles and sextapoles as possible options.

REFERENCES

[1] G. Andonian, et al., ”Chicane Radiation Measurements with a Compressed Electron Beam at the BNL ATF”, Presented at These proceedings (2007)

[2] ”TREDI”’, http://www.tredi.enea.it/

[3] J.D. Jackson, *Classical Electrodynamics*, 3rd ed. (Wiley, NY, 1999) p. 664.

[4] O.V. Chubar and N.V. Smolyakov, J. Optics (Paris) **24**, 117 (1993)

[5] ”Hierarchical Data Format”, http://hdf.ncsa.uiuc.edu/HDF5/

