1 COMSOL X-ray Generation

1.1 Algorithm outline

1. User inputs:
   - Bending magnet source and beam parameters.
   - No. of release points, no. of rays released at each point, no. of iterations. The product of these three quantities is equal to the total number of rays in the simulation.
2. A uniform distribution of release points on beam trajectory is randomly assigned either once or upon each iteration.
3. Each release point emits a cone of x-rays with randomly sampled normal distributions (horizontal and vertical) in the polar angle about the tangential vector to the local beam direction.
4. A Gaussian power distribution approximates the modified Bessel functions of the analytic expressions. The distribution is a function of the polar angle about the local tangential vector with a mean of zero and a standard deviation of $0.608\gamma^{-1}$. The total power produced by the bending magnet is disseminated via the Gaussian over all simulated x-rays.
5. Deposited power densities upon surfaces are calculated via

   $$P_d[W/cm^2] = \frac{1}{A_i}\sum Q_j \tag{1}$$

   where A_i is the area of the i^{th} element with j incident rays of power Q_j.

1.2 Release points

Simulated rays are released according to a randomly sampled uniform distribution of points on an analytically determined curve for the beam trajectory in the xz-plane. The curve is parameterized by the bending angle, $\Delta\theta$, which is related to the arc length Δl and bending radius ρ through the expression

$$\Delta\theta = \frac{\Delta l}{\rho} \tag{2}$$

In addition to the initial and final coordinates, the user must input the total bending angle, θ, or arc length, l, in order to properly define the beam trajectory.

Let the beam propagate in the +x-direction with a perpendicular acceleration in the +z-direction induced by a dipole field in the -y-direction. Then, with the

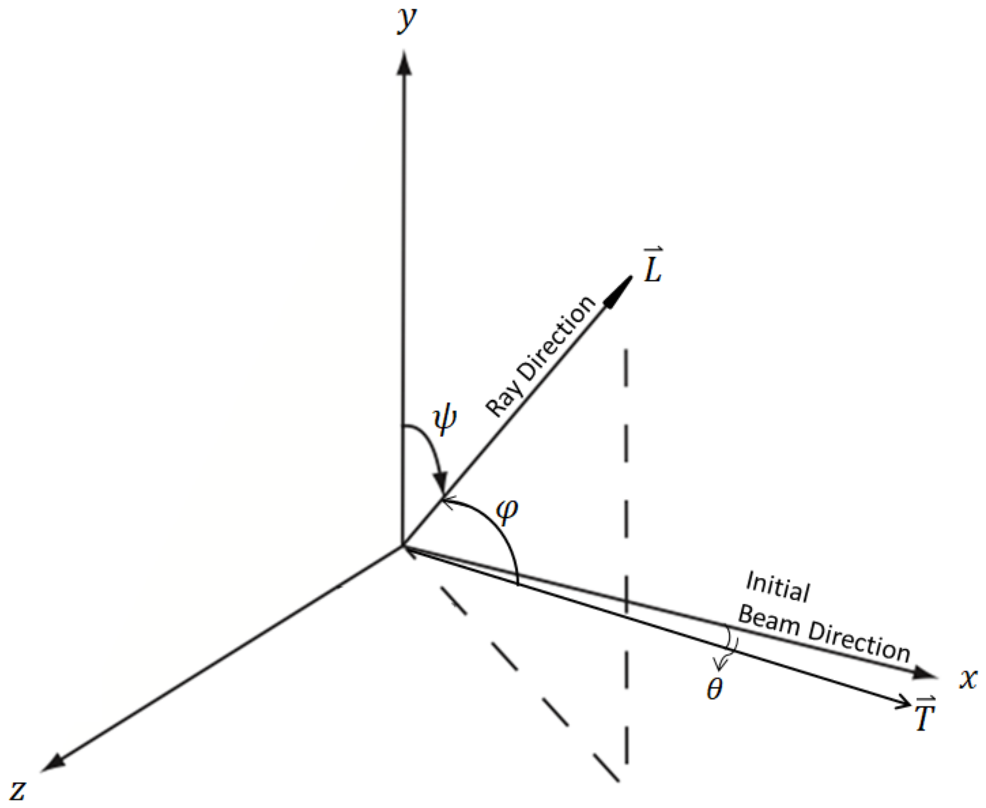

Figure 1: Local coordinate system of emitted x-ray.

initial and final trajectory points, we can randomly sample a uniform distribution of x values between $x_{initial}$ and x_{final}. From the equation of a circle, the corresponding z-values are defined as a function of x:

$$z(x) = \sqrt{\rho^2 + (x + x_c)^2} - \rho \tag{3}$$

where x_c is the x-component of the center coordinate of the circle traced by the beam trajectory. These randomly sampled coordinates in the xz-plane define the release points for the x-ray generation algorithm.

1.3 Virtual photon release

The accurate release of synchrotron x-rays requires a normal distribution of rays about the polar angle, ψ, with respect to the tangential direction vector, $\vec{T}$. We choose to release rays in bunches with conical distributions at each previously selected point along the beam trajectory. Each conical release is defined by a cone axis and angle.

The cone axis components are given by

$$\begin{aligned}
&\hat{x} : \cos\theta\cos\psi + \sin\theta\sin\psi\cos\phi \\
&\hat{y} : \psi \\
&\hat{z} : -\sin\theta\cos\psi + \cos\theta\sin\psi\cos\phi
\end{aligned}$$

where θ is a function of the x-coordinate:

$$\theta(x) = asin(((x - x_c)/\rho)), \tag{4}$$

ψ is defined by a normal distribution with standard deviation of $3\gamma^{-1}$ in the vertical direction and γ^{-1} in the horizontal direction (bending plane), and ϕ is uniformly distributed between $0, 2\pi$. The cone angle is defined by ψ.

For a user-defined number of iterations N_i, release points N_r, and photons per release (or per cone) N_c, each distribution of ϕ, and ψ are randomly sampled. The total number of photons in a given simulation, N_{tot}, is then

$$N_{tot} = N_i \cdot N_r \cdot N_c \tag{5}$$

1.4 Angular power distribution

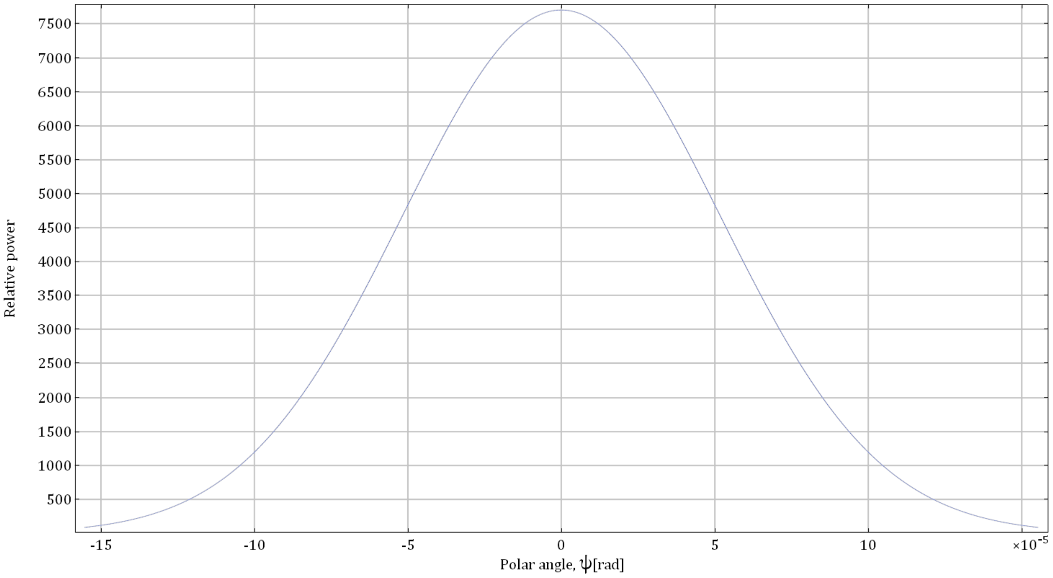

Figure 2: Angular power distribution function Pd1(ψ).

The angular distribution of power from bending magnet radiation is implemented in COMSOL as an expression of the power per ray or source power, P_{src}.

P_{src} = <Total BM power> * <Scaling factor (Gaussian function)> / <Average of scaling factor over all rays in the distribution>. We define P_{src} with the equivalent expression:

$$P_{src} = P_{tot} * P_{coeff}/P_{norm} \tag{6}$$

$$P_{tot}[W] = 14079 \cdot l[m]I[mA]E_e^4[GeV^4]\rho^{-2}[m^{-2}] \tag{7}$$

The scaling factor, P_{coeff}, is defined as the Gaussian function from figure 2, Pd1(*), with argument ψ and standard deviation $0.608\gamma^{-1}$.

$$P_{coeff} = Pd1(\psi) \tag{8}$$

Averaging the scaling factor over all rays is achieved with the built-in "Average over rays" operator gop.gopaveop1(*).

$$P_{norm} = gop.gopaveop1(P_{coeff}) \tag{9}$$

2 SynRad X-ray Generation

SynRad defines the beam trajectory in the same way as the COMSOL implementation. SynRad chooses a uniform distribution of points along this trajectory where each point releases many photons; these points are chosen by a user-defined step size.

The SynRad photon generation process then proceeds according to the following steps:

1. **Photon energy/power**
   At the beginning of each simulation, the energy range between $1x10^{-10}E_c$ and $100E_c$ is sampled on 5000 logarithmically increasing intervals. For each interval, the number of photons and the power it contains is calculated using a fast numerical approximation method.[1] The power in a given interval is calculated as the number of photons in the interval multiplied by the mean interval energy. Summing the interval fluxes and powers generates a cumulative distribution function over the entire energy range. This CDF is randomly sampled to assign photon energies at the pre-calculated release points.

   Each simulated photon represents a real flux and power according to the following recipe:

   - The total SR flux for an electron beam of a full revolution in practical units is 8.0841017*E[GeV]*I[mA] photons/sec.

[1] This is similar to the approximation method using Chebyshev Polynomials.

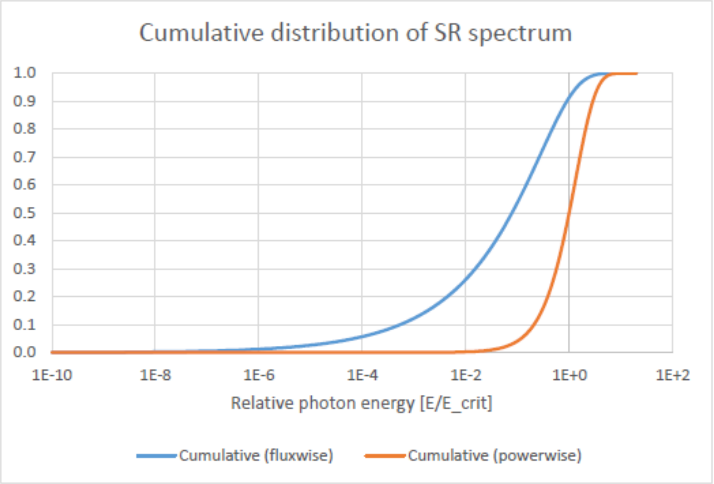

Figure 3: Relative photon energy CDF

- Each pre-calculated trajectory point represents a short dipole source which itself represents a fraction of the full revolution according to the ratio
$$revolutionRatio = dL/(2\rho\pi) \tag{10}$$
where dL is the trajectory step length and ρ is the radius of curvature.
- A starting point is chosen uniformly from the pre-calculated trajectory points. A "full scan" is defined as having emitted a photon from each trajectory point exactly once. As starting points are chosen randomly, a full scan can be represented statistically as having generated the same number of test photons as the number of trajectory points. At this point, the number of scans is equal to 1. Here, SynRad introduces the following factor.
$$No.scans = \frac{No.SimulatedPhotons}{No.TrajectoryPoints} \tag{11}$$
- Using the quantities defined above, one simulated photon can be represented by
$$K_{real/virtual} = revolutionRatio \cdot \gamma \cdot 4.131x10^{14} \cdot I[mA]/No.Scans \tag{12}$$

2. **Vertical angle (between bending plane and vector normal to bending plane)**
Knowing the relative photon energy (E/Ec) from the previous step and using a preloaded vertical angle/energy distribution, the two lines digitized for the energies just below and above the searched energy are looked up by

binary search. From the values of the cumulative distribution functions stored in these two lines (for a slightly lower and slightly higher energy), an interpolated distribution is calculated for the searched energy. Then a reverse interpolation is performed in this new distribution constructed on-the-fly, to assign a vertical angle to a pseudo-random number. If the beam's emittance isn't considered to be zero, this natural vertical angle is offset by a σ'_y width Gaussian-distributed random number, to account for the divergence.

3. **Horizontal angle (in bending plane)**
   The photon's horizontal angle is chosen according to the same method as in step 2 except that now a preloaded horizontal angle/*vertical angle* distribution function is used. In other words, given the vertical angle, a corresponding horizontal angle is chosen.

3 COMSOL vs SynRad Power Density Verification

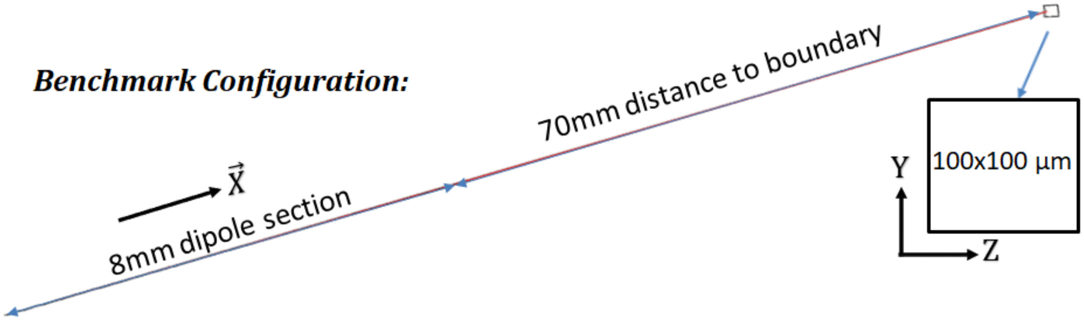

Figure 4: Benchmark simulation configuration.

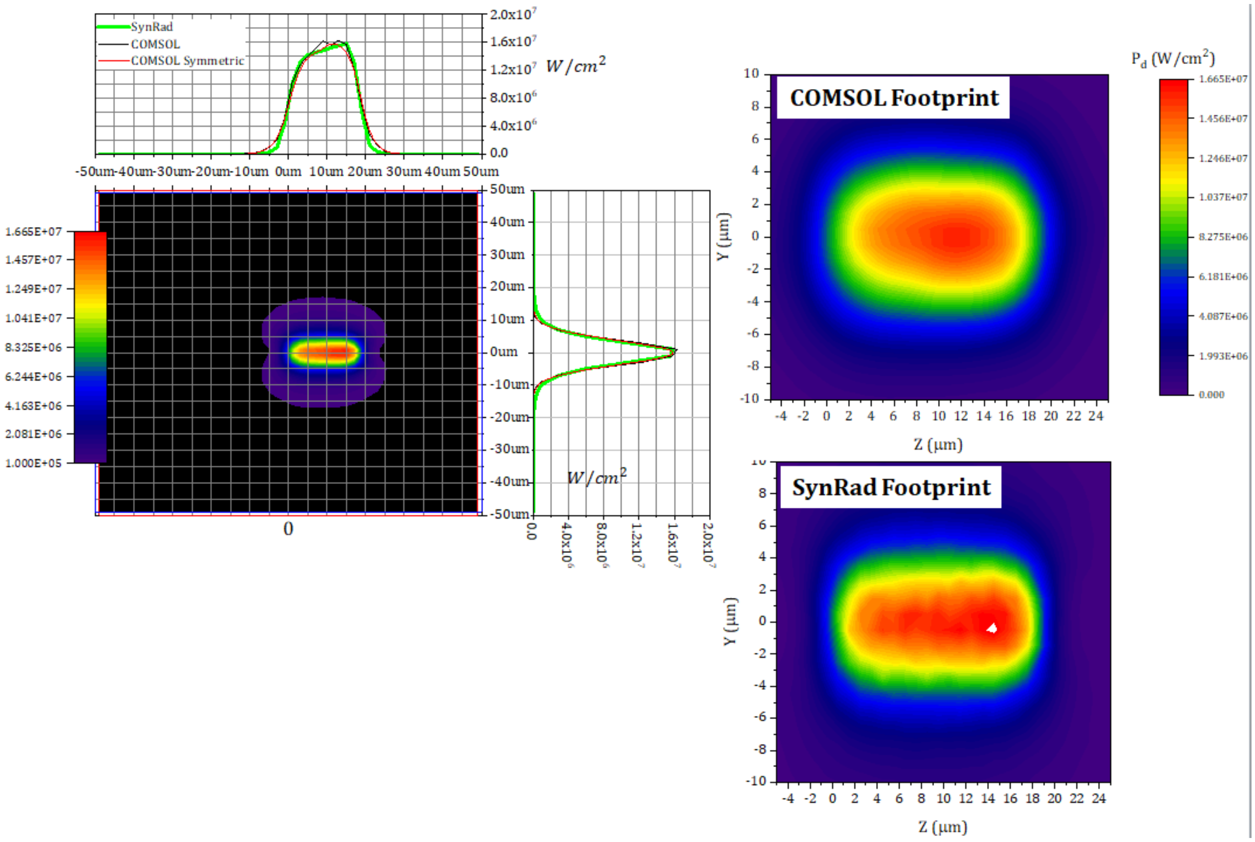

Figure 5: Footprint comparison. Red profile plot indicates COMSOL photon generation with the same trajectory points used on each iteration. Black profile plot indicates simulation with different trajectory points for each iteration.

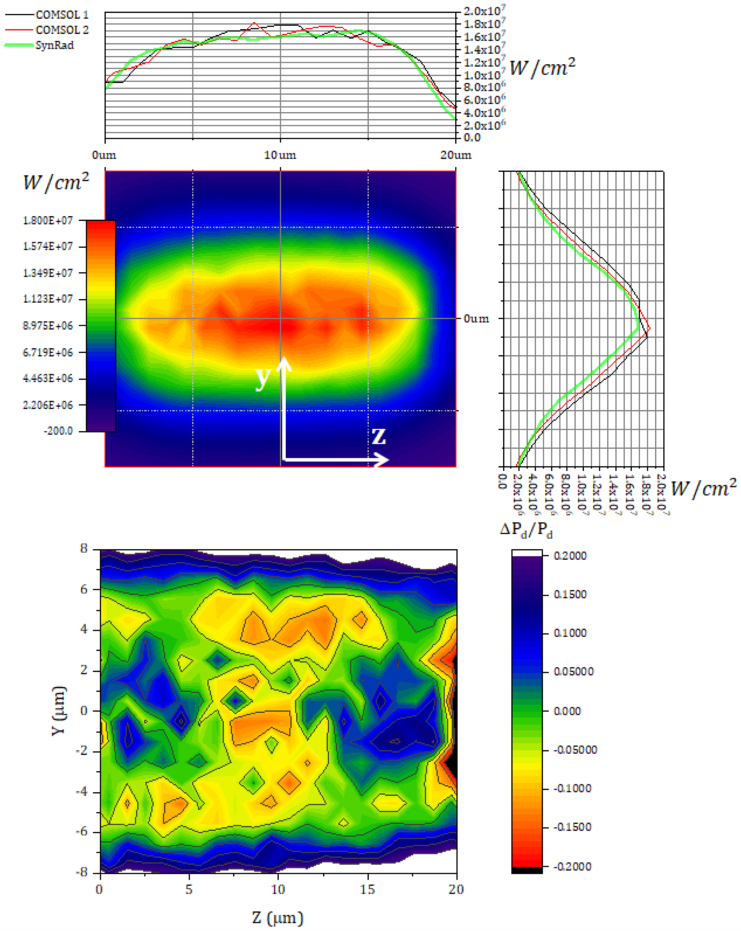

Figure 6: Heat load profile comparison and difference plot for a SynRad simulation and two COMSOL simulations using varying particle distributions in number of trajectory points and particles released per point.