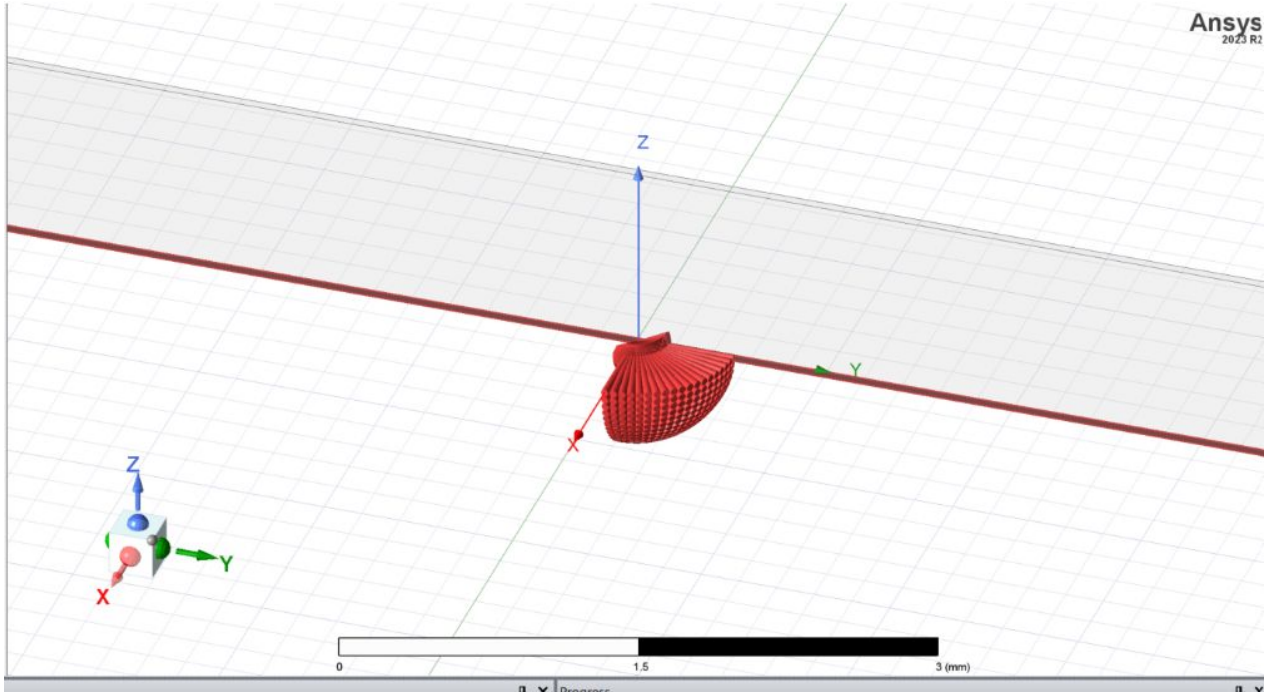
Blackbody Simulations

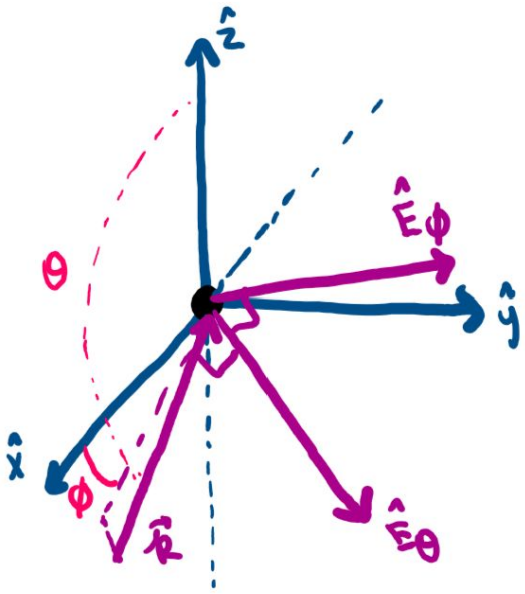
7/24/2025

Sweep Over Quarter of Input Angles



*Output CS is defined differently so that the sweep is more natural

Conventions for E_φ and E_θ

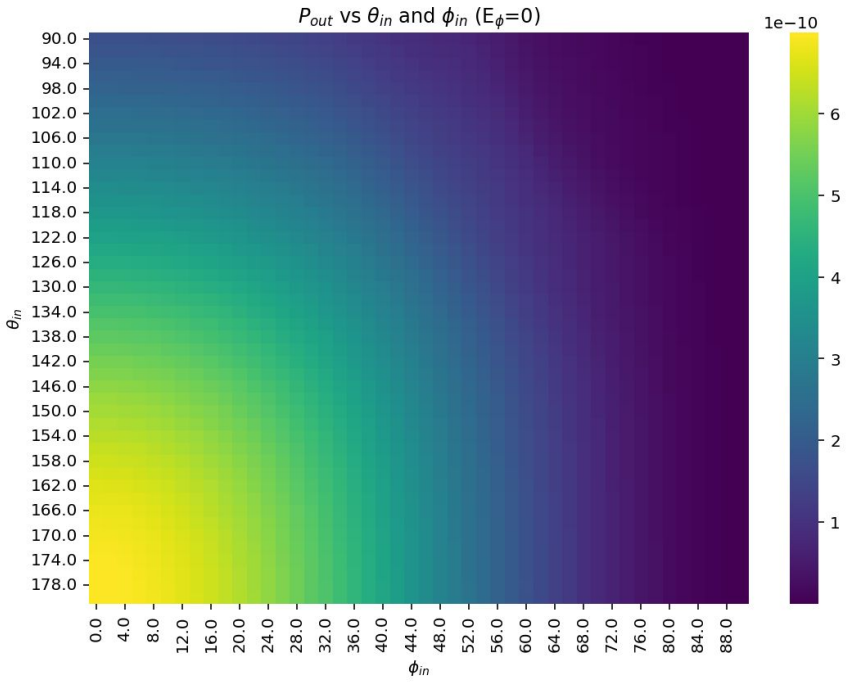


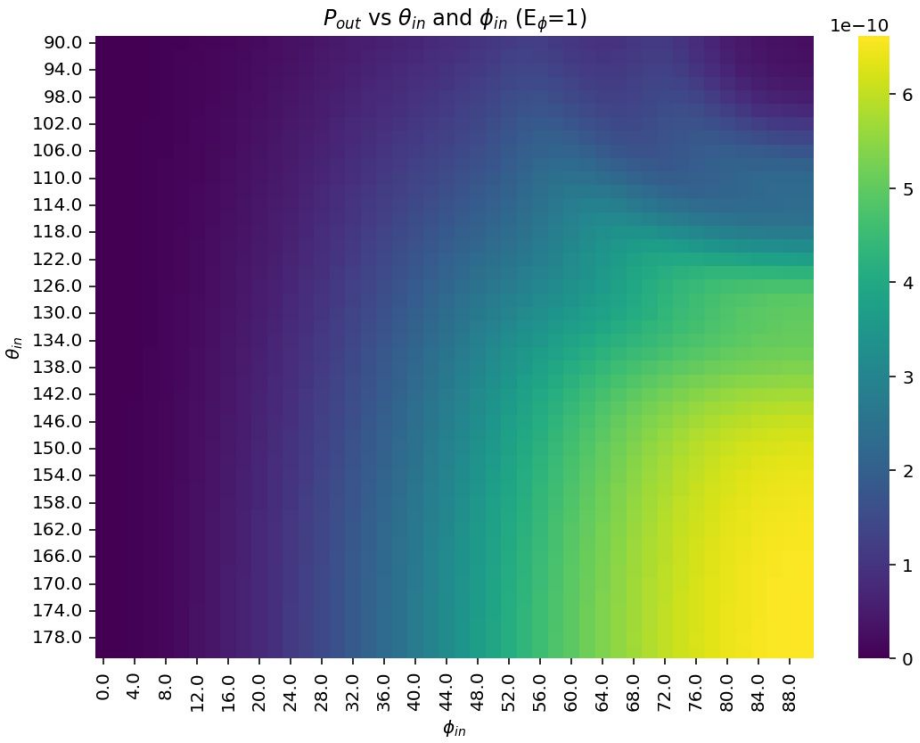
- θ-direction is $(\cos\theta\cos\varphi, \cos\theta\sin\varphi, -\sin\theta)$
- φ-direction is $(-\sin\varphi, \cos\varphi, 0)$

Initial 2 degree sweep in theta/phi (≈4 hours)

- Use absolute magnitude of the power difference (instead of fractional difference) and compare to the incoming power
  - May miss or not sample finely enough diffraction side lobes, but not important for this project
- Check: at 2 degree input angle, maximum absolute power difference is 3.64%, which is similar to the 100% * 2 *π/180 ≈ 3.49 expected (derivative of sine/cosine is bounded by unity)

Outgoing Power as a Function of Ingoing Angles (Smooth)



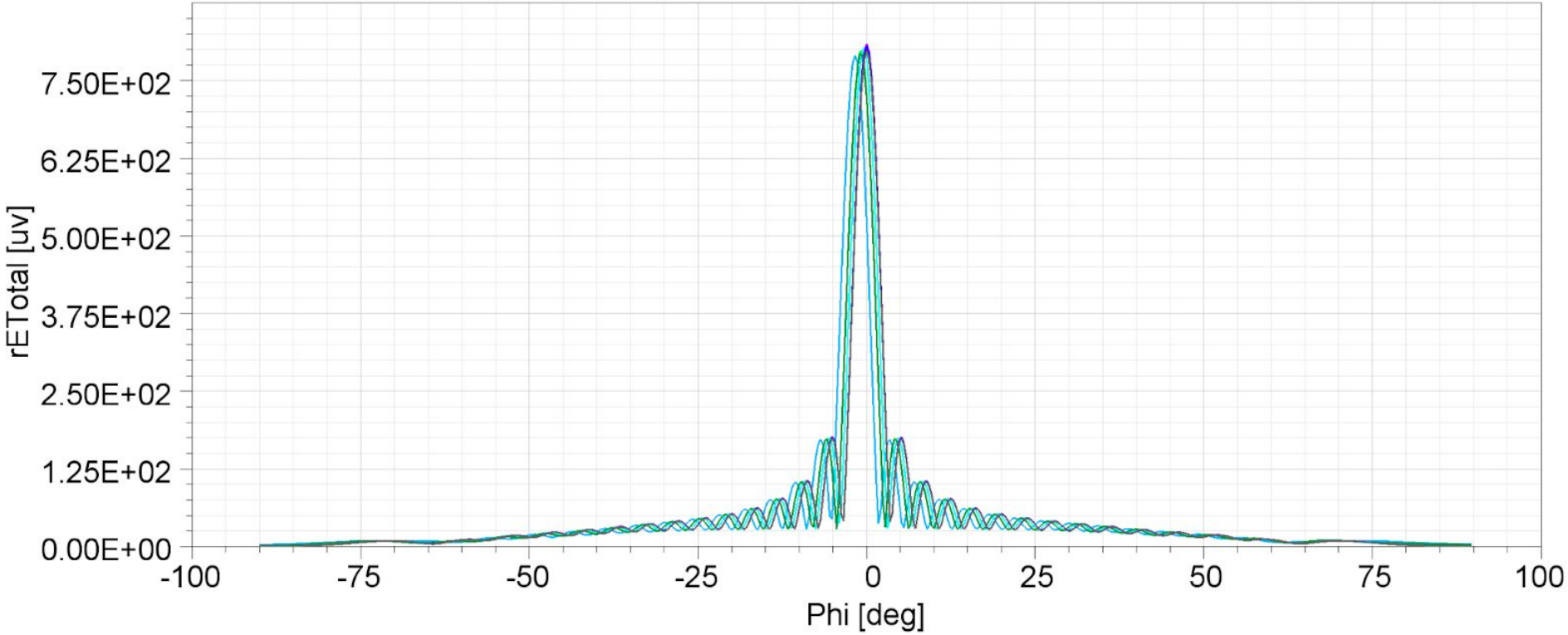


*All plots made with E_φ =0

Variable Input Angles

bbsim12

Ansys
2023 R2

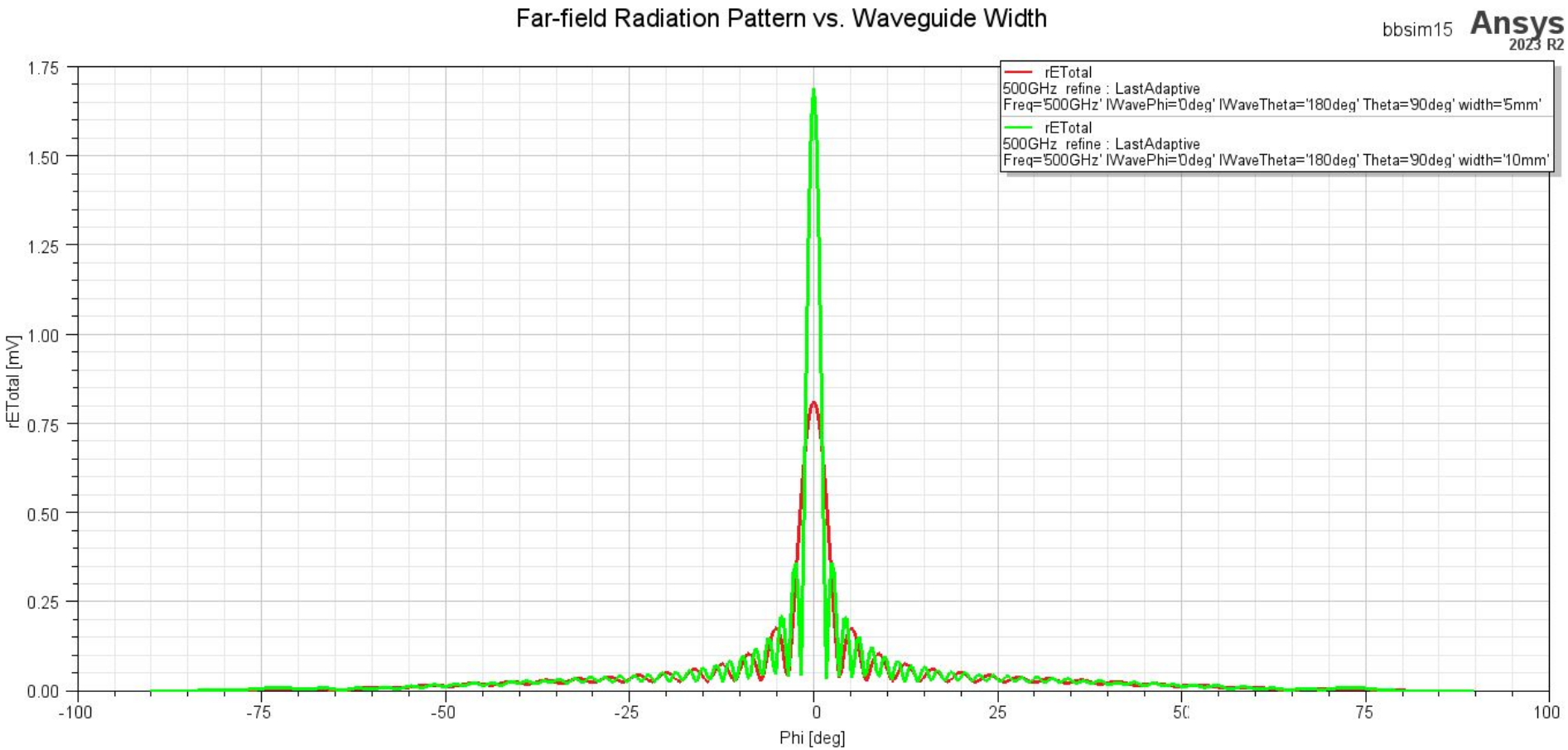


Comments

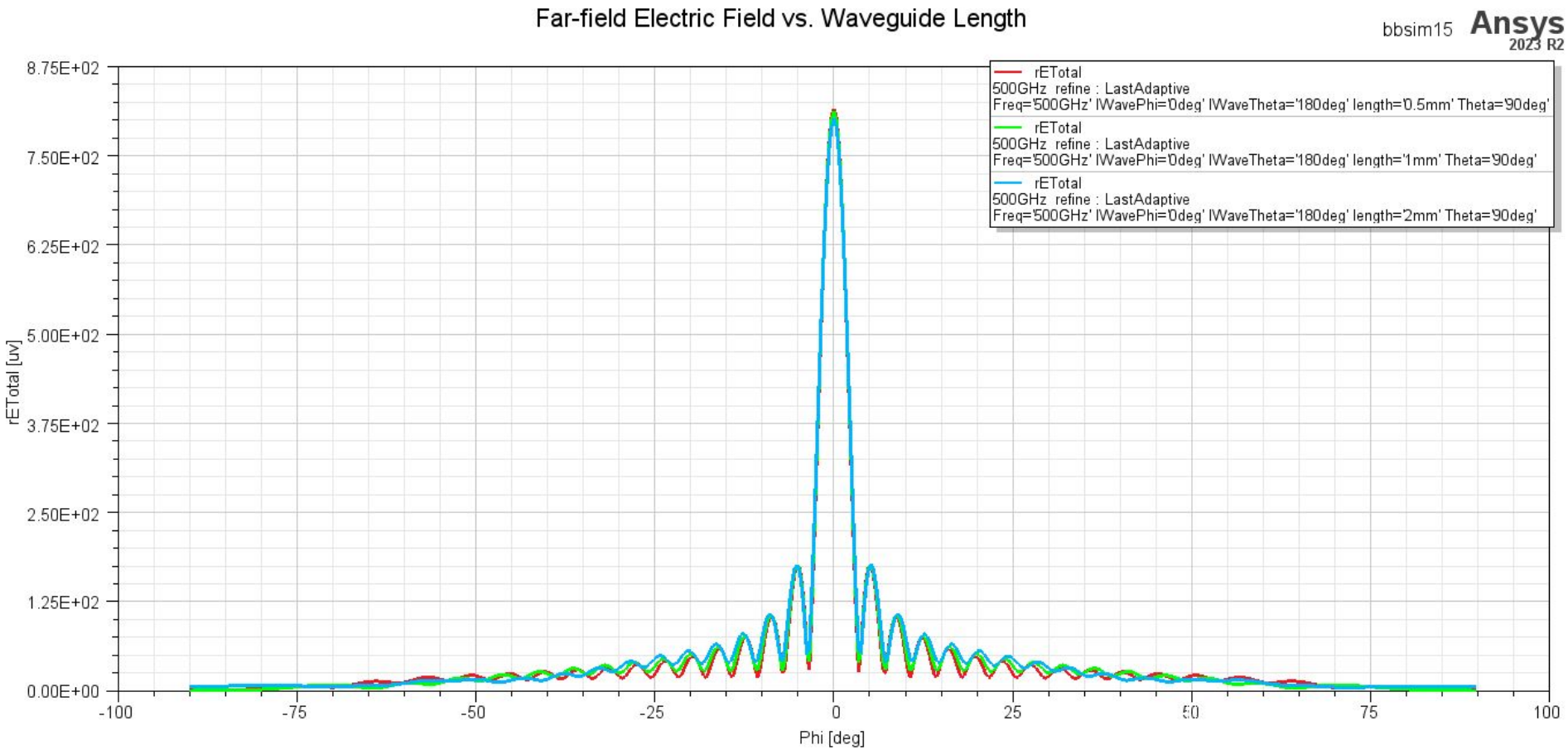
- Far-field electric field is smooth as a function of input angle
- Prof. Golwala suggests that a 5% angle-to-angle difference should be sufficient
  - Interpolation for intermediate points
- May have to sample far-field pattern more finely because of small but significant side lobes in the diffraction pattern
  - Takes longer to export, but made up for by less % angle-to-angle difference needed

Blackbody Simulations

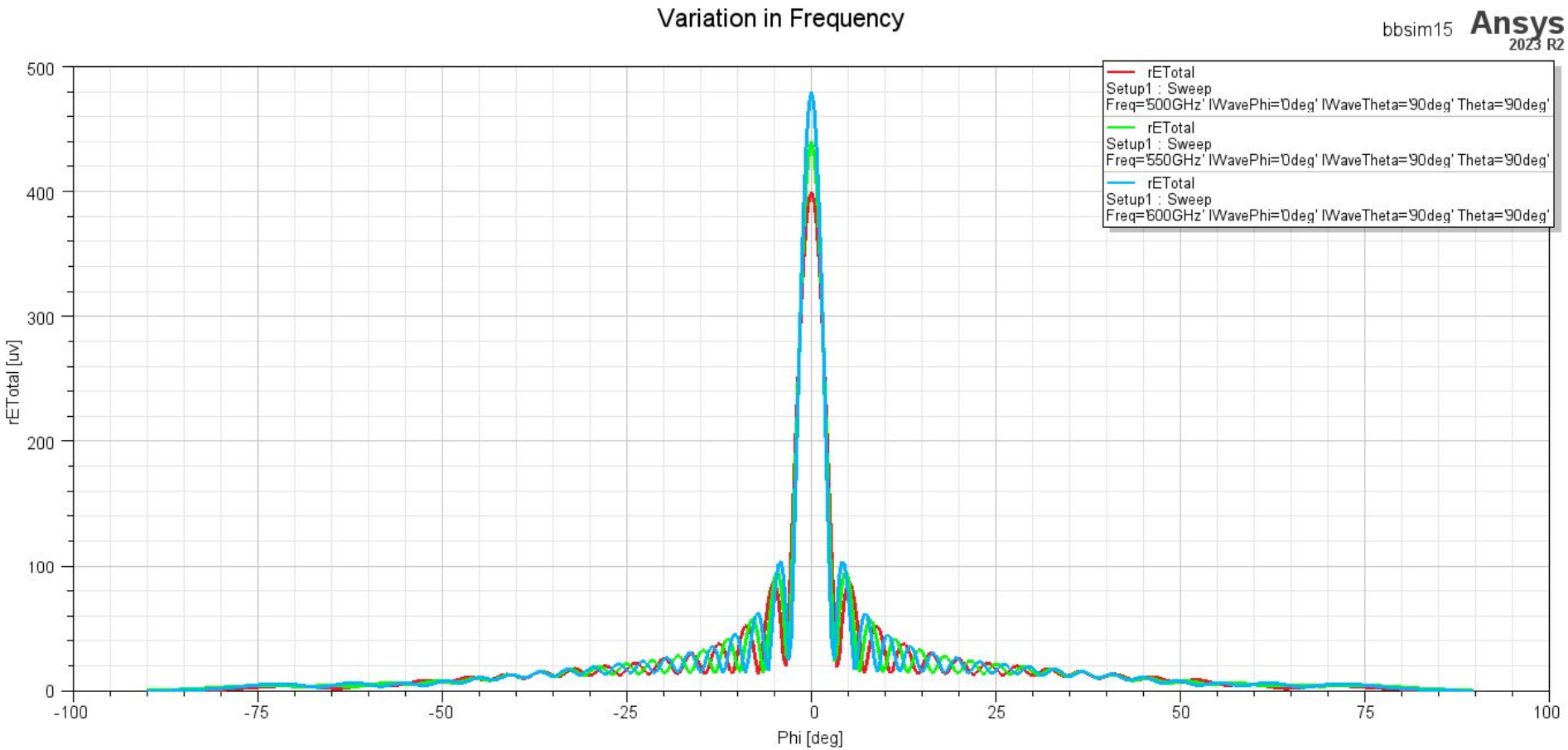
7/31/2025



*As expected, with a larger width, the peak is sharper and the width is smaller

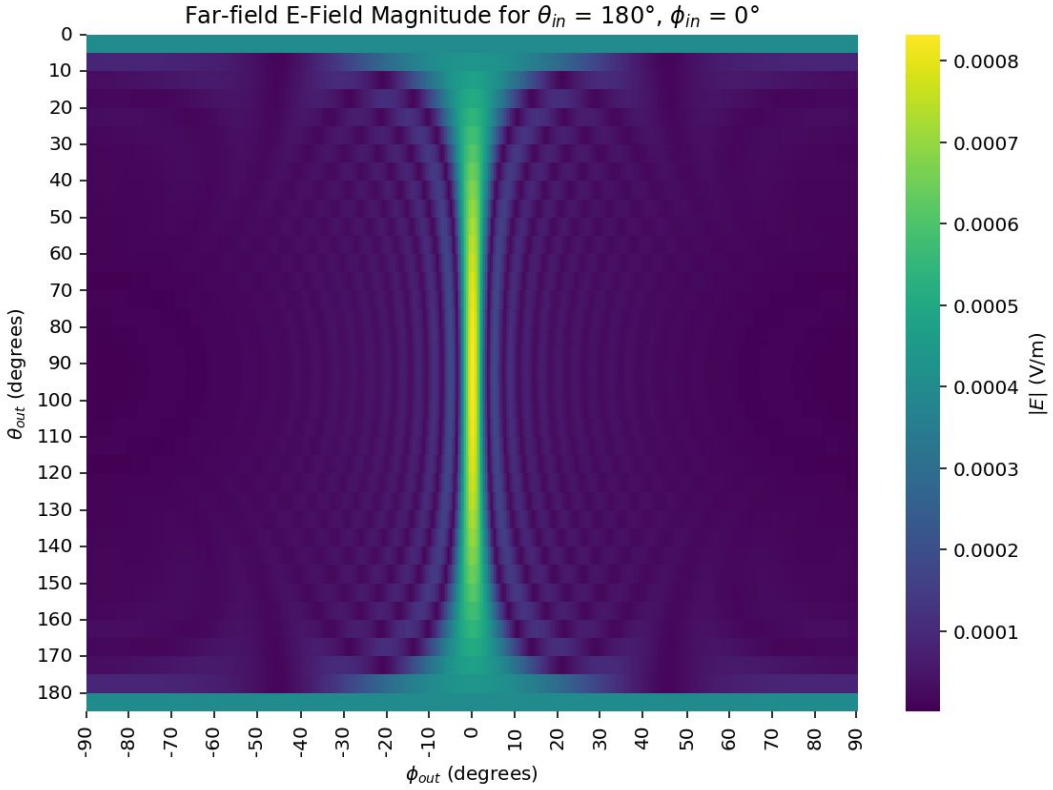


*With a longer length, the peaks are approximately the same width but the electric field magnitude at the troughs rises, especially near the center lobe



*Change in the far-field diffraction pattern as frequency is varied

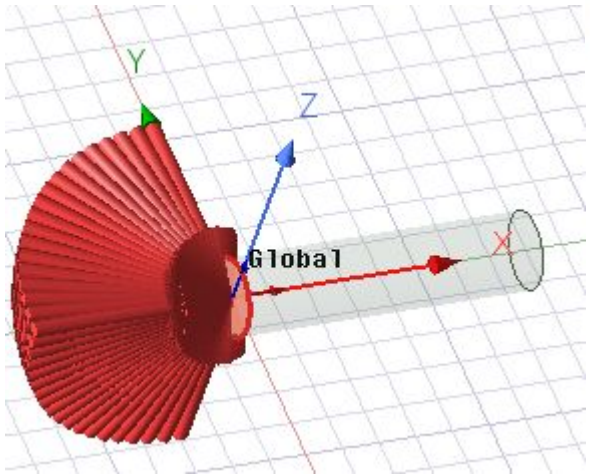
Far-Field E-Field (as a function of output angles)



Comments:

- Will run 0.8 degree input sweep; expect point-to-point output power variation (relative to ingoing power) to be < 1.5%.
- Try on different model geometries?
- Frequency sweep?
  - Will take too long. HFSS has an option to run an analysis over a range of frequencies instead of a single frequency, but it still solves each frequency individually and will have to export each frequency individually
  - Run in parallel (HPC?)

Some Cylindrical Geometry Plots (f=2 THz, r = 0.05mm)

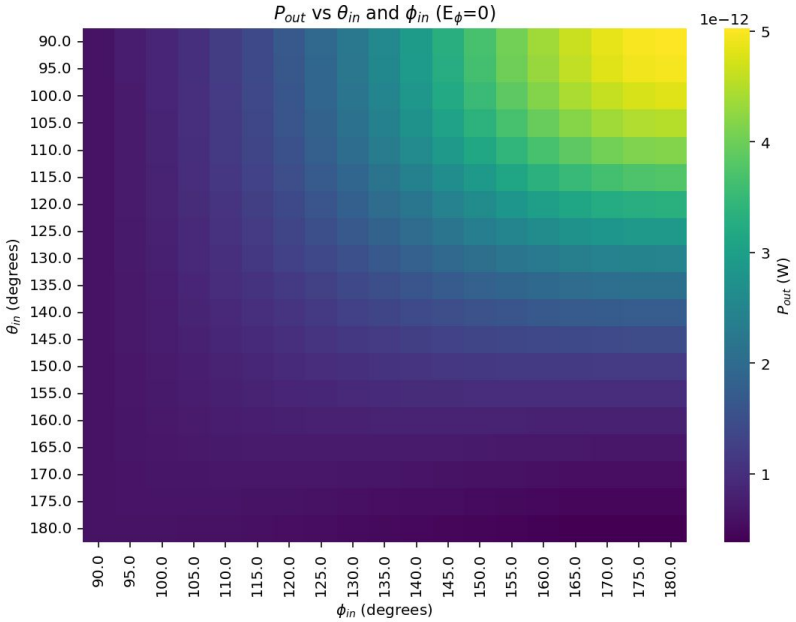


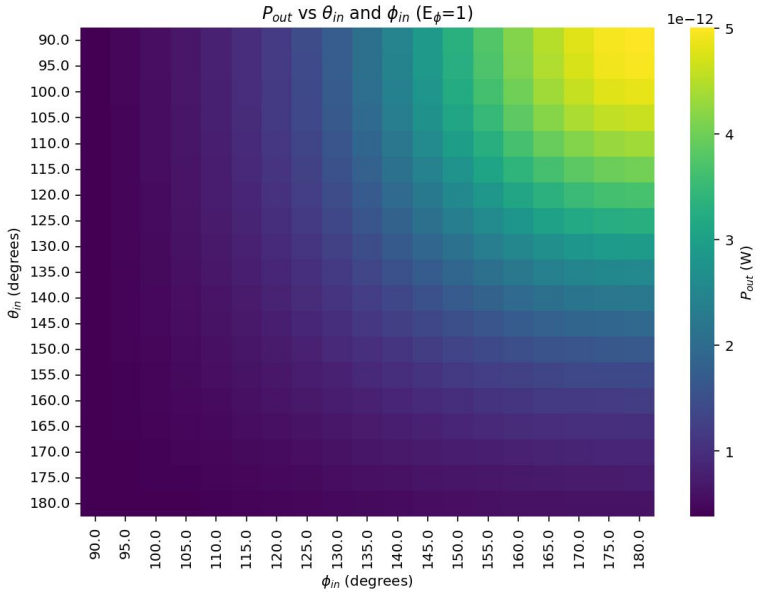
$$f_{c,\nu n} = \frac{c}{2\pi}\frac{p'_{\nu n}}{a}$$ Corresponding to a cutoff frequency ≈ 1.76 THz

*Output CS is defined the same way, just shifted to the output face

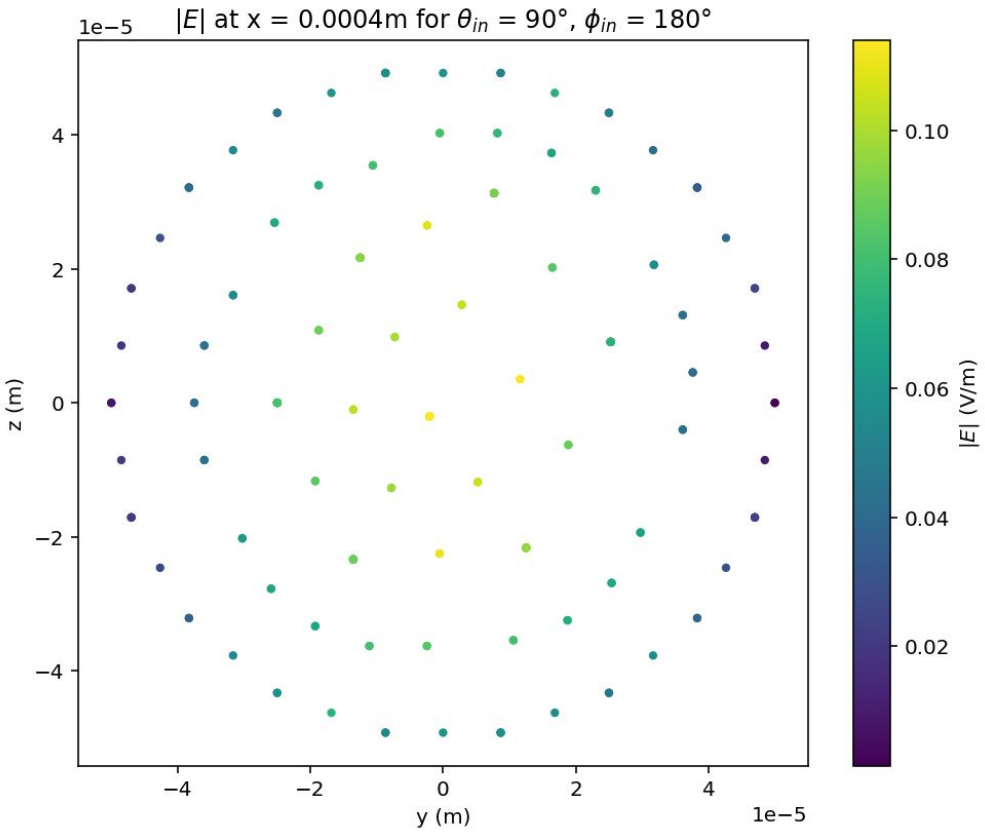
*Also ran a height perturbation geometry

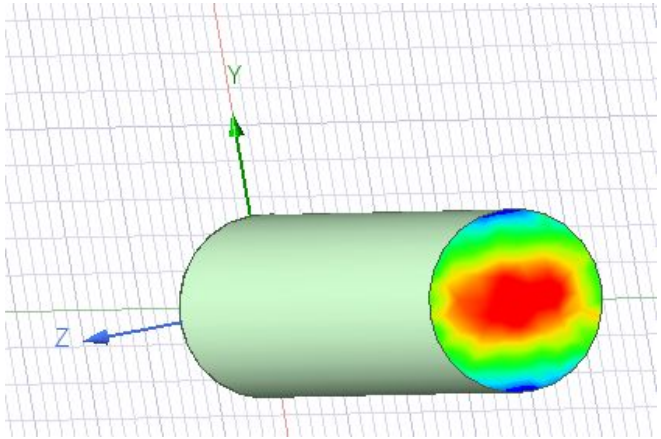
Outgoing Power as a Function of Ingoing Angles (Smooth)

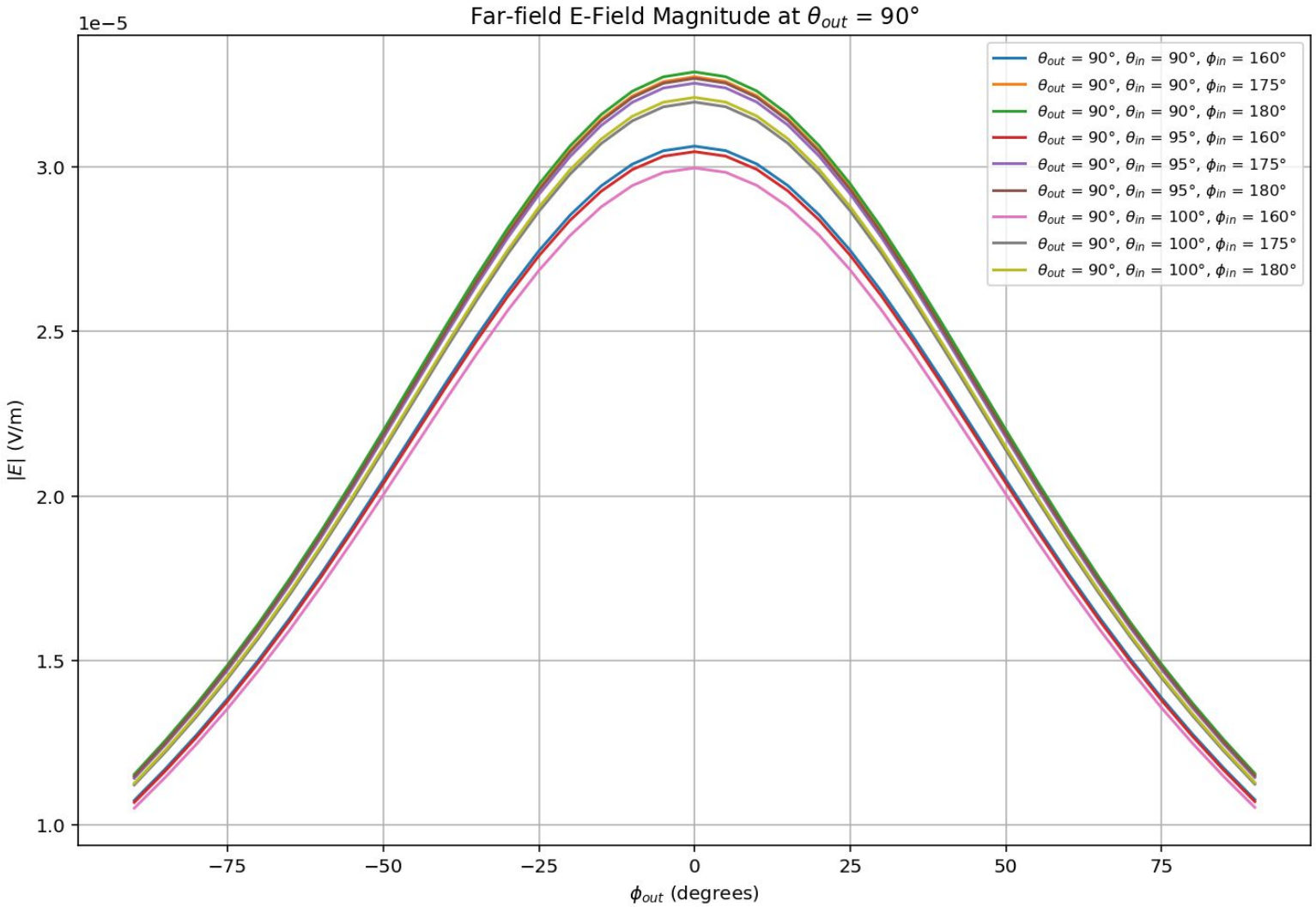


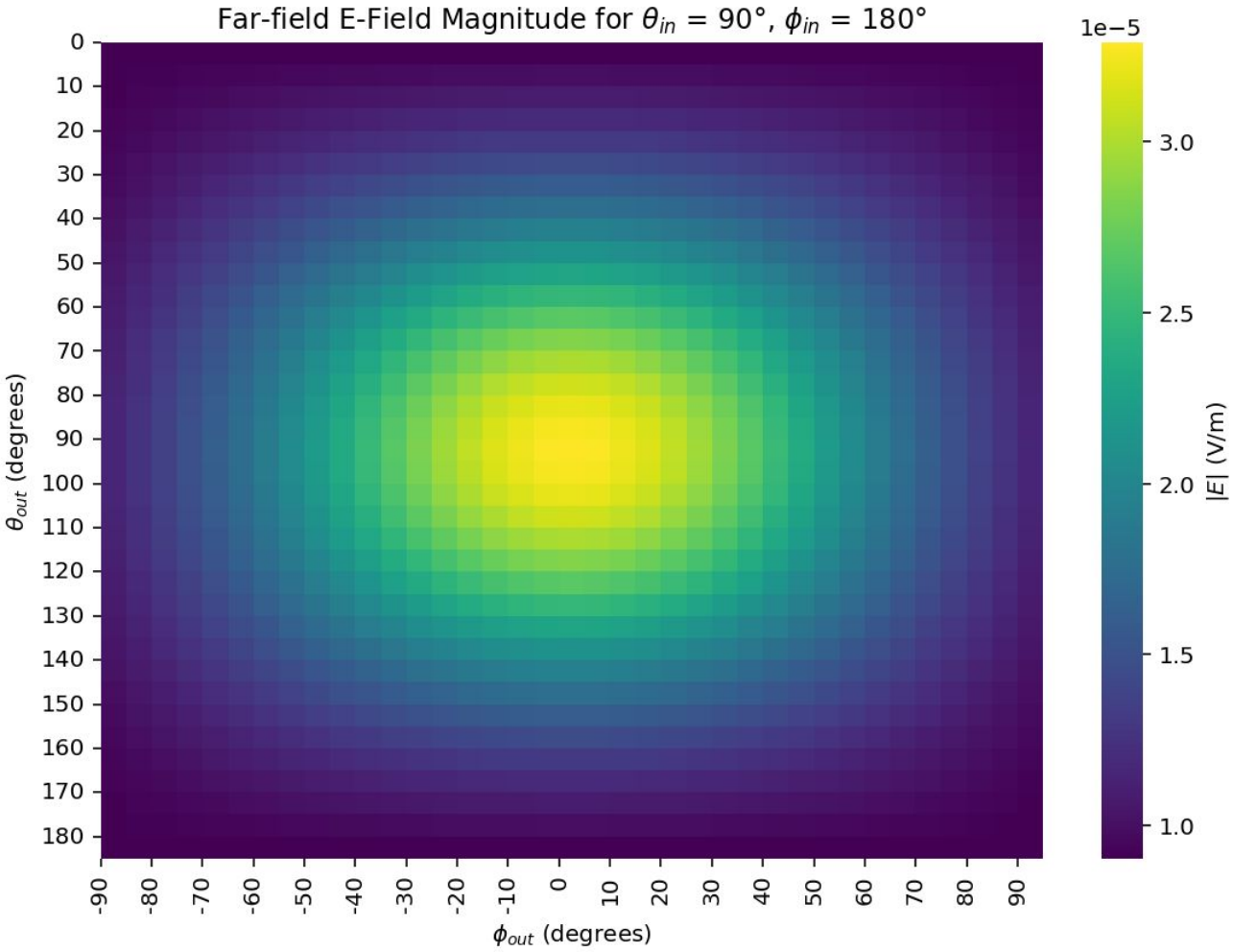


Electric Field at Outgoing Face






Far-field E-Field Magnitude at θ_{out} = 90°
1e−5
3.0
2.5
2.0
1.5
1.0
|E| (V/m)
−75
−50
−25
0
25
50
75
ϕ_{out} (degrees)
θ_{out} = 90°, θ_{in} = 90°, ϕ_{in} = 160°
θ_{out} = 90°, θ_{in} = 90°, ϕ_{in} = 175°
θ_{out} = 90°, θ_{in} = 90°, ϕ_{in} = 180°
θ_{out} = 90°, θ_{in} = 95°, ϕ_{in} = 160°
θ_{out} = 90°, θ_{in} = 95°, ϕ_{in} = 175°
θ_{out} = 90°, θ_{in} = 95°, ϕ_{in} = 180°
θ_{out} = 90°, θ_{in} = 100°, ϕ_{in} = 160°
θ_{out} = 90°, θ_{in} = 100°, ϕ_{in} = 175°
θ_{out} = 90°, θ_{in} = 100°, ϕ_{in} = 180°


Far-field E-Field Magnitude for θ_{in} = 90°, ϕ_{in} = 180°
θ_{out} (degrees)
0
10
20
30
40
50
60
70
80
90
100
110
120
130
140
150
160
170
180
ϕ_{out} (degrees)
-90
-80
-70
-60
-50
-40
-30
-20
-10
0
10
20
30
40
50
60
70
80
90
1e−5
3.0
2.5
2.0
1.5
1.0
$|E|$ (V/m)

Blackbody Simulations

8/7/2025

ComplexMag_E vs. Mag_E

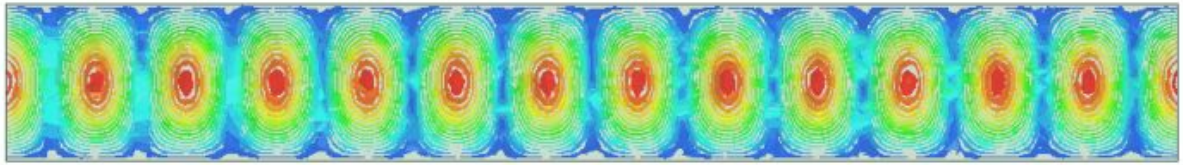
- To average over time, we probably need ComplexMag_E
- Originally, I was used Mag_E

Mag_E	Mag(AtPhase(Smooth(<Ex,Ey,Ez>), Phase))

$$|\vec{E}| = |Re[\tilde{E}]| = \left| \sum_{i \in \{x,y,z\}} \hat{i}\, E_i(x,y,z)\, cos(\varphi_i + \phi) \right|$$

$\tilde{E}$ = <Ex,Ey,Ez> - complex vector (named CVc in *Calculator*) is what HFSS computes. You can't plot this in real space. That's why HFSS provides the short cuts to the 3 real quantities.

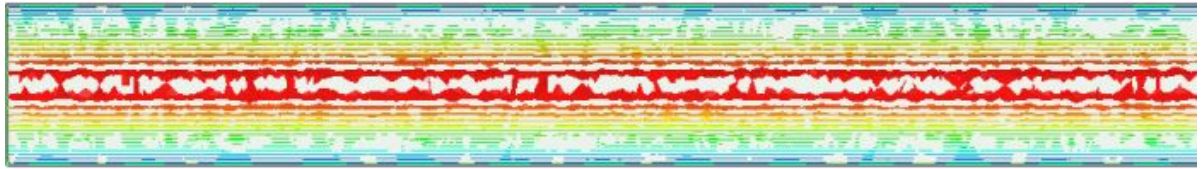
Smooth(<Ex,Ey,Ez>) – averages the fields calculated at all elements in contact to give a continuous fields.
AtPhase() – calculates at certain value of Φ.
Mag() – finds the length of the vector (magnitude of the vector).



ComplexMag_E	Mag(Smooth(<Ex,Ey,Ez>))

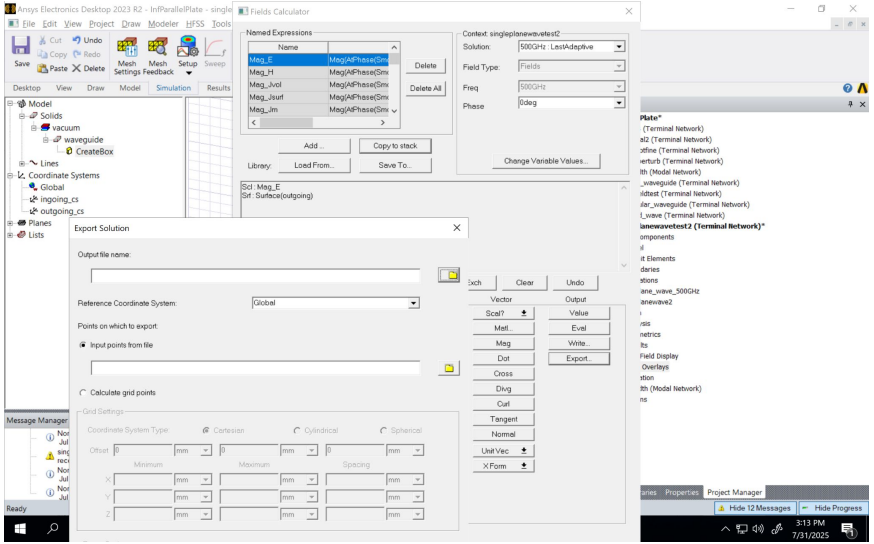
$$|\tilde{E}| = \sqrt{\tilde{E}.\tilde{E}^*} = \sqrt{\sum_{i \in \{x,y,z\}} |E_i|^2}$$

This kills out the $e^{j\Phi}$.



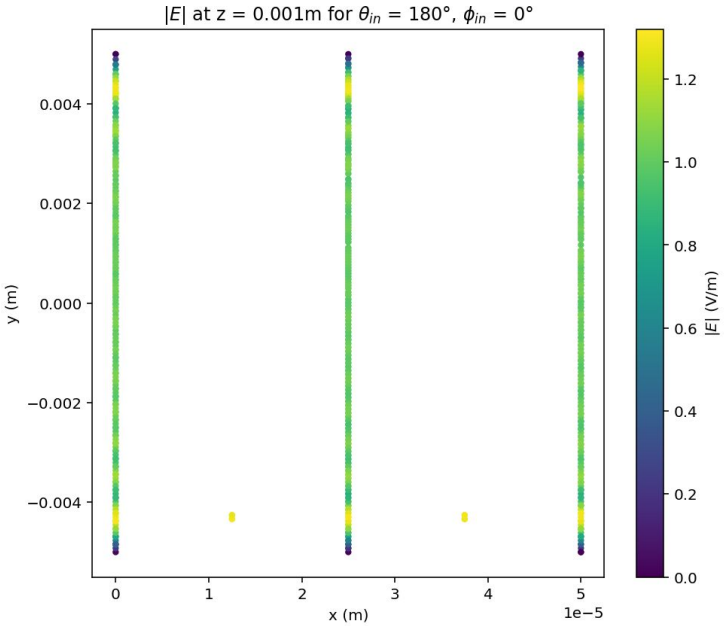
Exporting Fields

- For exporting the field at the export face, current using Write command, instead of Export (check Fields Calculator cookbook)
  - Write is primarily meant for data intended for use in the future in HFSS calculator
  - Export is mean for 3rd party processing
    - Requires either a file of input points or a grid of specified inputs
    - Added a manual option to specify $[x_1, y_1, z_1]$, $[x_2, y_2, z_2]$, $[x_{res}, y_{res}, z_{res}]$

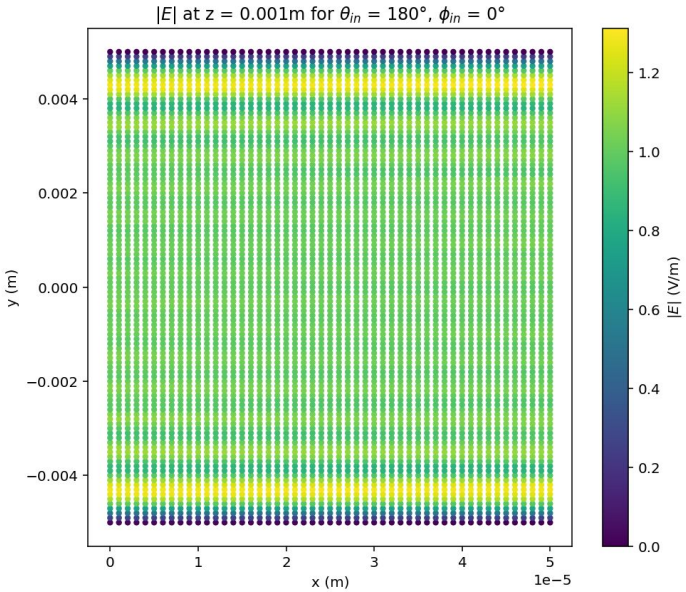


New Field Export

Before



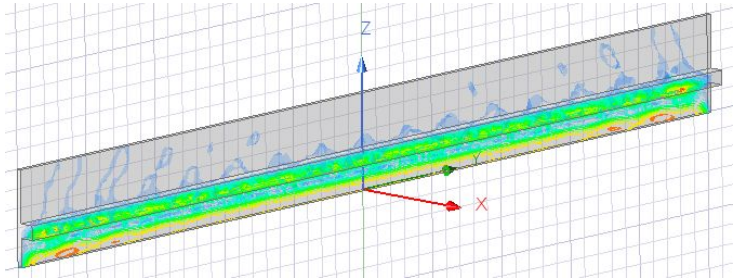
After

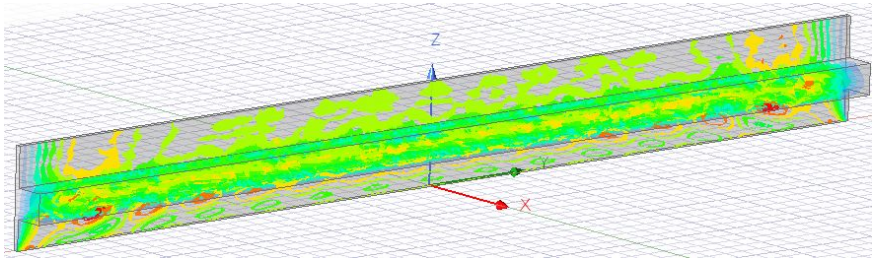


*Rectangular waveguide geometry
*Seems to not take much longer to run

Stub Filters

$$\lambda_g = \frac{\lambda}{\sqrt{1 - \left(\frac{\lambda}{\lambda_c}\right)^2}},$$

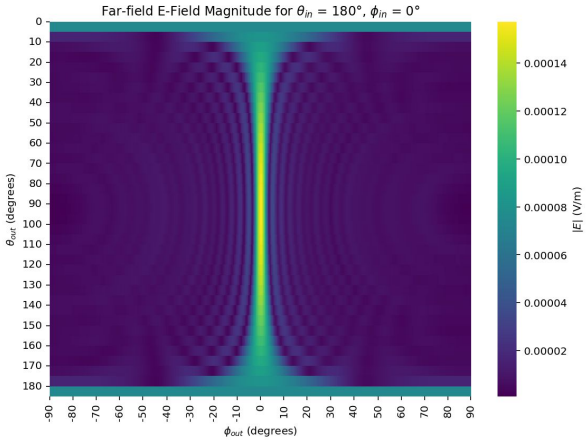




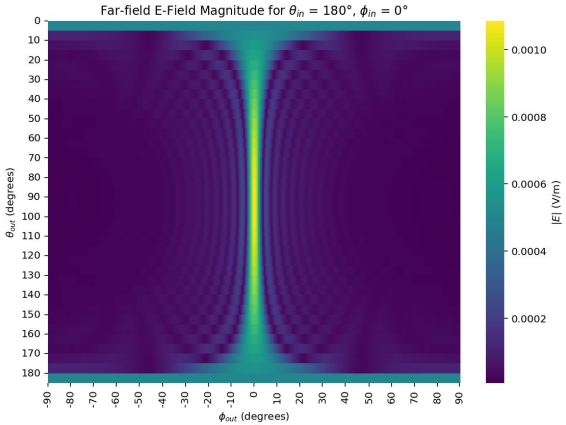
Stub length = $\lambda_g/4$ (Band-stop)
|S21|^2 = 0.04

Stub length = $\lambda_g/2$ (Band-pass)
|S21|^2 = 1.00

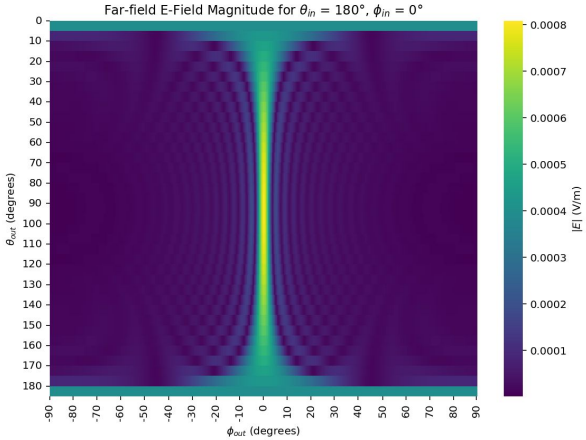
Aside: Far-Field Radiation Patterns for Different Geometries are Similar!



Stub design



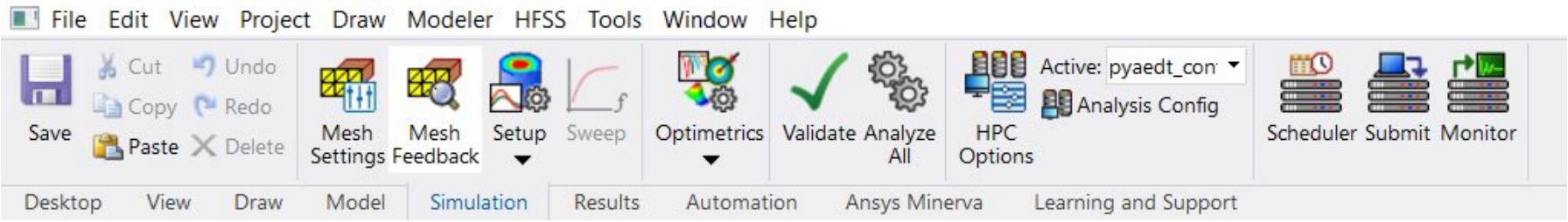
Height perturb



Standard

*But outgoing power is different, electric field at exit is somewhat different

HFSS HPC (High-Performance Computing)



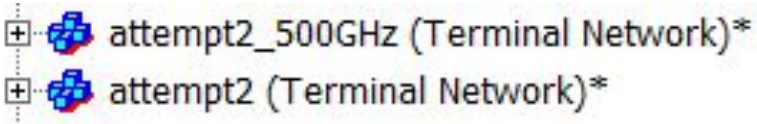
- Parallelize sweep over many frequencies
- Needs to run HFSS at many different frequencies (many different instances of HFSS)
  - Run single bbsim.py file with frequency sweep specified in the code
  - Run bbsim.py multiple times, and restrict each run of bbsim.py to 1 frequency

Blackbody Simulations

8/14/2025

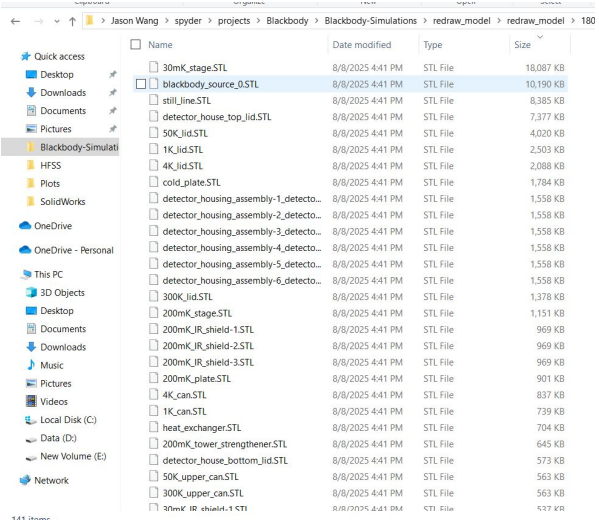
One Frequency

- Originally, code looped over multiple frequencies
  - Changed code to accept one frequency at a time via command line argument
  - E.g. python bbsim1freq.py 500 runs the simulation at 500GHz
- The simulation creates a COPY of the existing design in HFSS, where the new design name is {old_design_name}_{frequency}GHz,
  - Makes it easier to run the different designs in parallel using HPC



Other Updates

- Solidworks Geometries -- can’t seem to import into Solidworks as an assembly

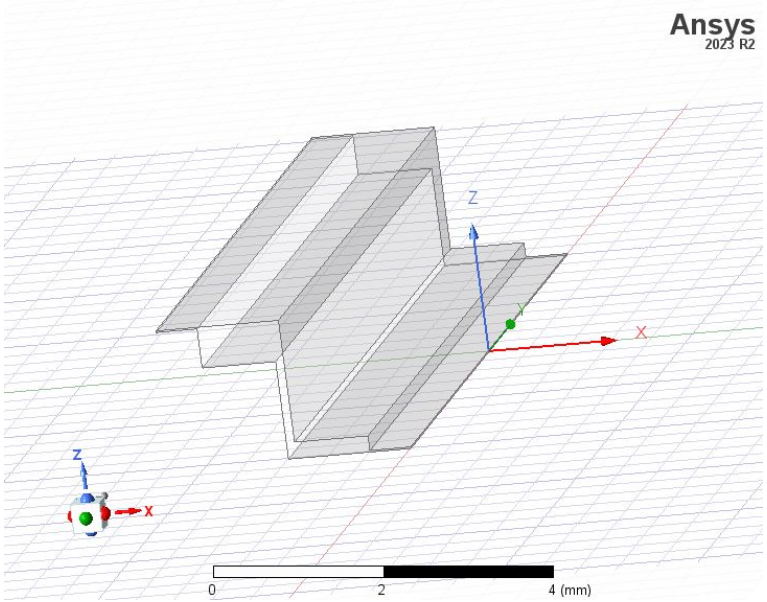


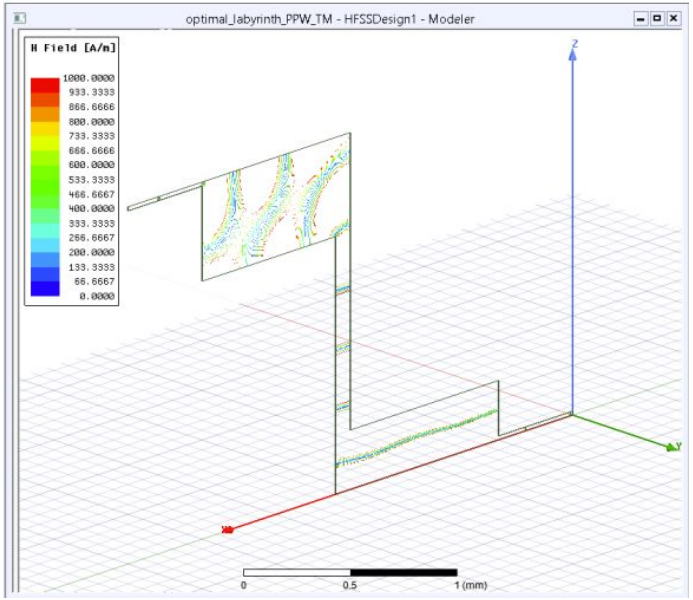
- HPC -- still a work in progress

Blackbody Simulations

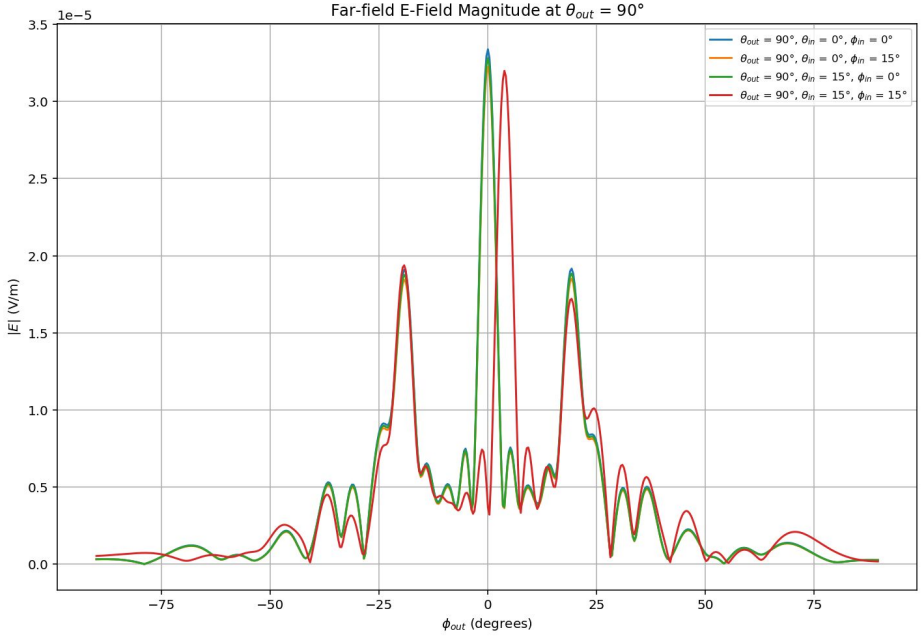
8/21/2025

Yen-Yung's Geometry



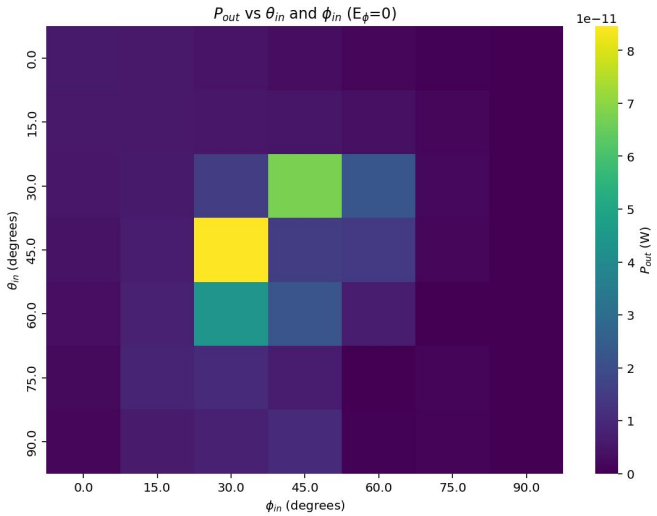


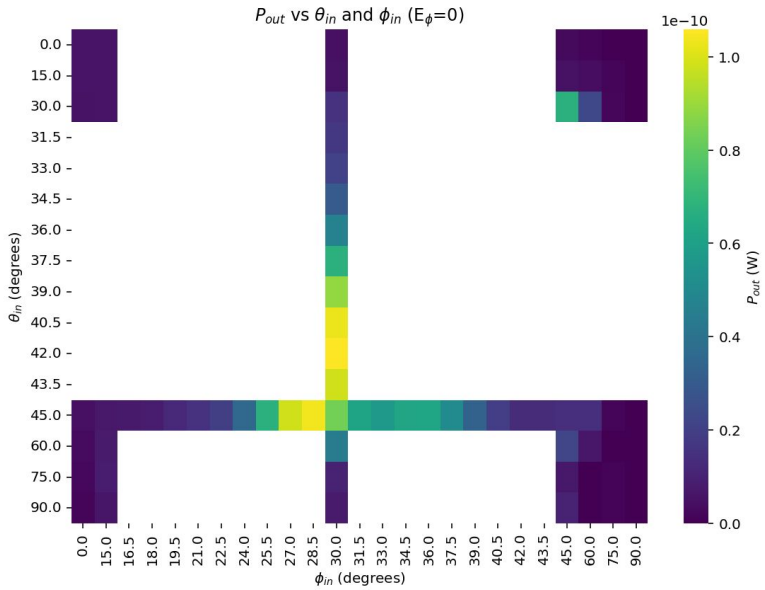
Yen-Yung's Geometry



*Magnitude of E is less than before, indicating successful blocking
*Relatively smooth dependence on ingoing angles

Yen-Yung's Geometry





*Outgoing power is somewhat unsmooth, but the adaptive refinement counters that

Determining the Probability Distributions

The mechanism for calculating the necessary output probability distributions is modularized over three steps.

(i) Firstly, an incident plane wave is propagated through the waveguide via an incident plane wave excitation in HFSS. The time-averaged power incident on the waveguide can be calculated as

$$P_{in} = \int_I |\mathbf{S_{in}}| dA, \tag{52}$$

where $\int_I$ refers to an integral over the outgoing face and $\mathbf{S_{in}} = \frac{1}{2}\frac{\mathbf{E}\times\mathbf{B}}{\mu_0}$ indicates the time-averaged Poynting vector of the incoming wave. Notice that in this case, we are not interested in the flux of the Poynting vector ($\int_I \mathbf{S_i n} \cdot \mathbf{dA}$) through the input face, because the propagation of the wave is not necessarily aligned with the direction of the waveguide.

From the simulation, we obtain a value of the time-averaged transmitted power by integrating the Poynting vector $\mathbf{S_{out}}$ over the outgoing face, i.e.

$$P_{out} = \int_O |\mathbf{S_{out}}| dA, \tag{53}$$

where $\int_O$ refers to an integral over the input face. As for the input face, we are concerned not about the flux of the Poynting vector through the outgoing face but rather the total power coming out of the output face regardless of directionality. This is because photons leaving the waveguide reenter free space, and we wish to account for the full power carried away, even if it is not perfectly aligned with the waveguide axis.

For the first stage, then, the probability that the photon propagates to the other end of the waveguide without being reflected or absorbed is given as

$$|S_{21}|^2 = \frac{P_{out}}{P_{in}}, \tag{54}$$

where $|S_{21}|^2$ represents the forward transmission coefficient.

Determining the Probability Distributions (cont.)

(ii) Secondly, given that the photon arrives at the output face, the probability distribution of output locations on the output face to feed back to Geant4 is given by $P(x,y) \propto |\mathbf{E}(x,y)|^2$, i.e. the probability for the photon to be emitted from any given point on the output face is proportional to the electric field squared at that point. We use the complex magnitude squared of the electric field ($|\mathbf{E}(x,y)|^2 = |E_x|^2 + |E_y|^2 + |E_z|^2$, where E_x, E_y, $E_z \in \mathbb{C}$), rather than the magnitude squared of the instantaneous electric field ($|Re[\mathbf{E}(x,y)]|^2$) because we are interested in the time-averaged intensity of the electric field over the outgoing face independent of the phase of the wave. Then, the numerical probabilities can be calculated from the discretized electric field positions $\{(x_i, y_i)\}$ as

$$P(x,y) = \frac{|\mathbf{E}(x,y)|^2}{\sum\limits_{(x_i,y_i)} |\mathbf{E}(x_i,y_i)|^2}. \tag{55}$$

(iii) Thirdly, given that the photon arrives at the output face, the probability distribution of output wavevectors (parametrized by θ_{out} and ϕ_{out}) to feed back to Geant4 can be found by analyzing the far-field radiation pattern, which is calculated in HFSS as the Fourier transform of the electric field at the output face. We can then use the same method as equation 55, i.e. from the discretized output angles $\{(\theta_i, \phi_i)\}$ as

$$P(\theta,\phi) = \frac{|\mathbf{E}(\theta,\phi)|^2}{\sum\limits_{(\theta_i,\phi_i)} |\mathbf{E}(\theta_i,\phi_i)|^2}. \tag{56}$$

where again $|\mathbf{E}(\theta,\phi)|^2$ denotes the complex magnitude squared of the far-field electric field at outgoing angles θ and ϕ. Given an outgoing angle parametrized by (θ,ϕ), the polarization of the outgoing photon can be inferred through the far-field radiation pattern as well:

$$\hat{e}_{out} = \frac{\mathbf{E}(\theta,\phi)}{|\mathbf{E}(\theta,\phi)|}. \tag{57}$$

Determining the Probability Distributions (cont.)

The main difficulty in creating these lookup tables lies in the need to evaluate many combinations of input parameters, including frequency, wavevector direction, and input electric field polarization. One simplification is to use symmetry to simulate only a subset of the possible input angles, as shown in figure 3. Another simplification is to simulate only two orthogonal polarization directions. We denote the two orthogonal directions by $\hat{\mathbf{E}}_\theta$ and $\hat{\mathbf{E}}_\phi$, where the unit vector in the θ direction is $(\cos\theta\cos\phi, \cos\theta\sin\phi, -\sin\theta)$, and the unit vector in the ϕ direction is $(-\sin\phi, \cos\phi, 0)$. A depiction of the two polarization states is given in figure 4. Results for any arbitrary polarization can then be reconstructed by linearly combining the outcomes from this basis due to linearity of Maxwell's equations. In particular, if the ingoing electric field polarization is $\hat{e}_{in} = E_\theta\hat{\mathbf{E}}_\theta + E_\phi\hat{\mathbf{E}}_\phi$, then $\mathbf{E}(x,y)$ and $\mathbf{E}(\theta,\phi)$ become

$$\begin{aligned}\mathbf{E}(x,y) &\to E_\theta\mathbf{E}_{E_\theta=1,E_\phi=0}(x,y) + E_\phi\mathbf{E}_{E_\theta=0,E_\phi=1}(x,y)\\ \mathbf{E}(\theta,\phi) &\to E_\theta\mathbf{E}_{E_\theta=1,E_\phi=0}(\theta,\phi) + E_\phi\mathbf{E}_{E_\theta=0,E_\phi=1}(\theta,\phi)\end{aligned} \tag{58}$$

where the subscripts on $\mathbf{E}$ indicate one of the two orthogonal polarization directions. Equations 55, 56, and 57 then proceed as before with the replacements given in equation 58.

Interpolation

Finally, let us discuss methods for interpolating simulation data for reintroduction into Geant4. The electromagnetic outputs – scalar transmission coefficient $|S_{21}|^2$, complex electric field at the outgoing face, and far-field electric field – are sampled over a multi-dimensional parameter space of frequency, input and output angles, and spatial coordinates. For each of the two orthogonal polarization directions, the input parameters and corresponding outputs of the three relevant quantities are shown in table 1.

Quantity	**Input Variables**	**Output Variables**
$\|S_{21}\|^2$	f, θ_{in}, ϕ_{in}	$\|S_{21}\|^2(f, \theta_{in}, \phi_{in})$
E-field at Exit Face	f, θ_{in}, ϕ_{in}, x, y, z	$\mathbf{E}(f, \theta_{in}, \phi_{in}, x, y, z)$
Far-field E-field	f, θ_{in}, ϕ_{in}, θ_{out}, ϕ_{out}	$\mathbf{E}(f, \theta_{in}, \phi_{in}, \theta_{out}, \phi_{out})$

Table 1: Input and output variables for the three quantities of interest. The output variables directly correspond to the probability distributions described in section 3.1, so this table describes the input and output variables of the desired lookup table.

Classical approaches such as radial basis function interpolation, tensor-product splines, and kriging provide smooth, physically consistent estimates but can be computationally expensive for large datasets. Reduced-order modeling mitigates this by projecting the fields onto a small set of modal basis functions (e.g., waveguide or spherical harmonics) and interpolating only the modal coefficients. A particularly effective strategy combines this with machine learning: the fields are first compressed via proper orthogonal decomposition (POD) into parameter-independent, dominant modes, and neural networks are trained to map input parameters (frequency, angles, and spatial coordinates) to the corresponding modal coefficients [7]. The full fields can then be efficiently reconstructed from these coefficients, yielding smooth, accurate, and physically consistent predictions across the multi-dimensional parameter space. Once the fields are obtained, they can be sampled from as described section 3.1.