



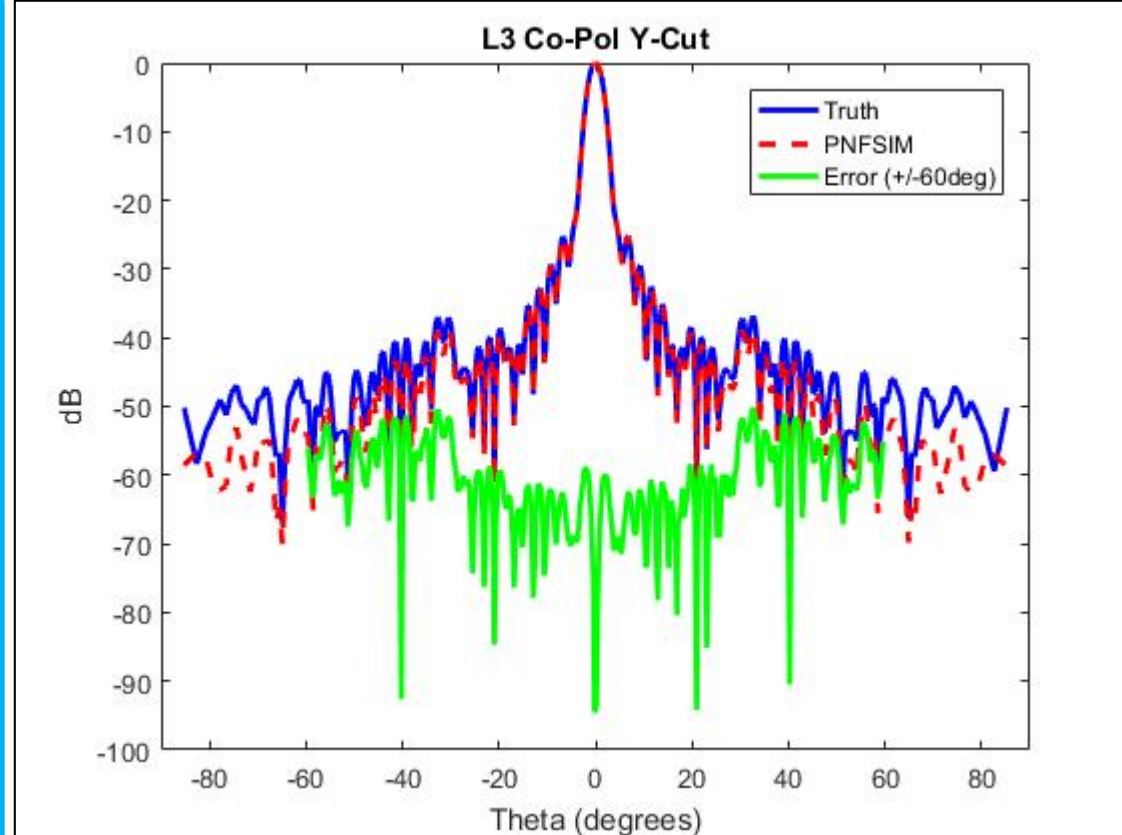
## ***AMTA 2021 Regional Event:*** **Simulating Near-Field Measurement** **Errors for Spherical & Planar Geometries**

Ryan T. Cutshall  
Jon D. Lawrence, Ph.D.

April 22, 2021

# Motivation

- Any given **measurement system** will result in a finite amount of uncertainty in the measured value.
  - A planar near-field (PNF) or spherical near-field (SNF) facility is our measurement system.
- **What is the uncertainty** in any given value reported by a near-field antenna measurement system?
- Moreover, which **source of error** is the largest contributor to the measurement uncertainty?
- The second question is almost impossible to answer via measurement, since more than one error source is usually present during a measurement.
- To predict and quantify the measurement uncertainty induced by any error source(s), we use a **simulated** near-field antenna measurement, add the contribution of the error source(s), and transform to the far-field.

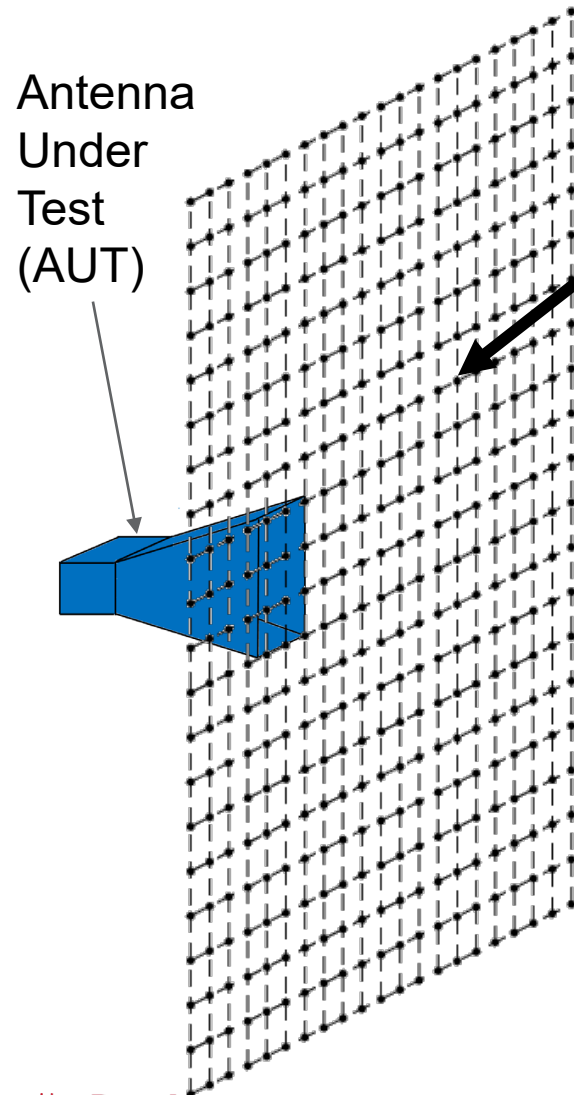


Example of the results generated by the pnfSim tool

## How do we build a near-field measurement simulator?

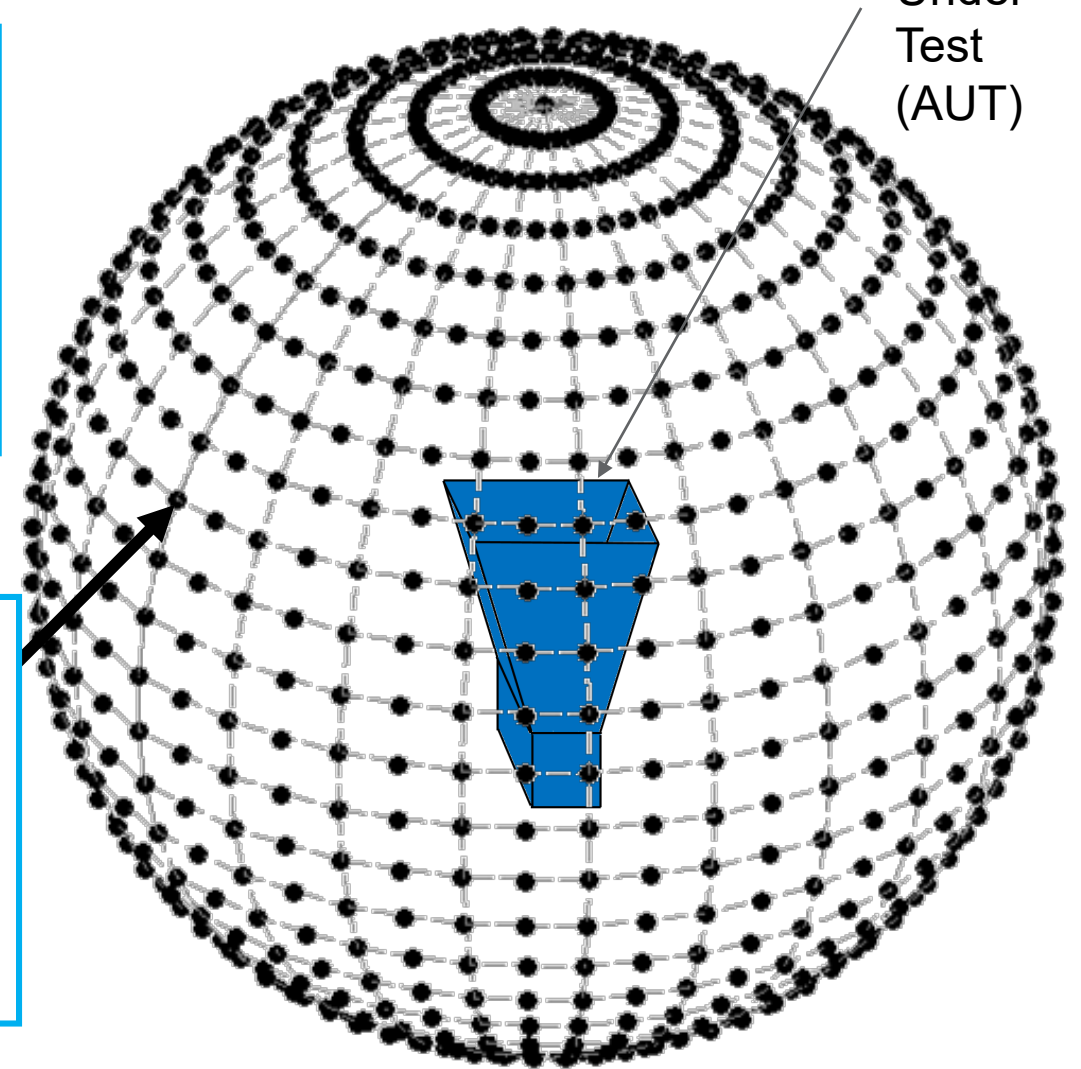


# Planar Near-Field and Spherical Near-Field Geometries



In a planar near-field (PNF) measurement, the tip of the probe transverses a plane, ideally set at a fixed  $z$ -distance from the aperture of the AUT, and moves in pre-determined steps  $\Delta x$  and  $\Delta y$

In a spherical near-field (SNF) measurement, the tip of the probe transverses a sphere, ideally set at a fixed  $r$ -distance from the center of the AUT, and moves in pre-determined steps  $\Delta\theta$  and  $\Delta\phi$



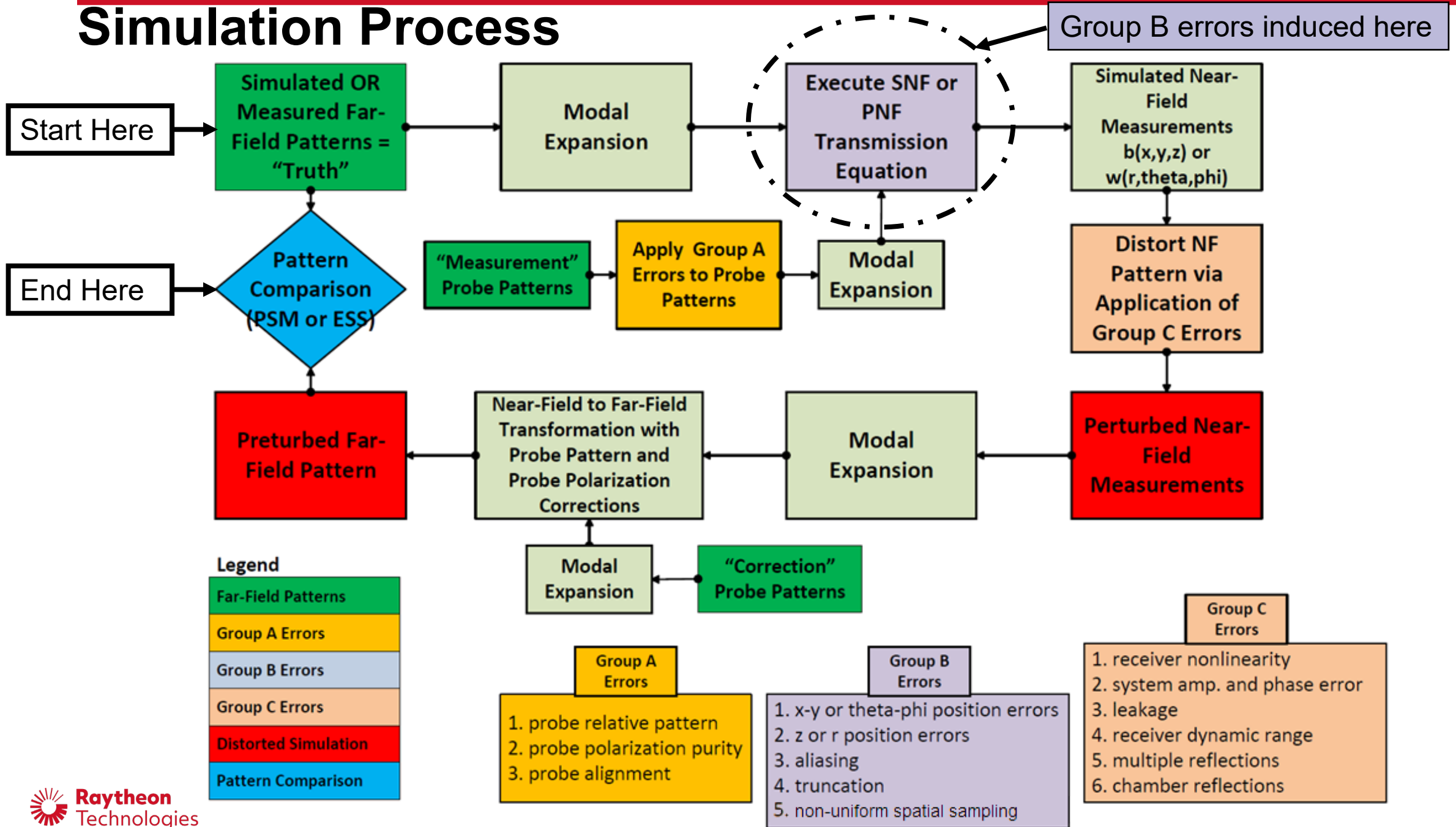
## Near-Field Measurement Error Sources (1 of 2)\*\*\*

Error Term	Comments
Probe relative pattern	For SNF, includes presence of higher-order $\mu$ -terms in “measurement” probe pattern, and neglect of higher-order $\mu$ -terms in “correction” probe pattern
Probe polarization purity	PNF includes probe polarization information in modal expansions, whereas SNF requires a special probe polarization correction step during the near-field to far-field transformation
Probe alignment	Probe alignment errors can be induced in the “measurement” probe pattern errors when simulating
AUT alignment	Not required if simulating relative uncertainties
Probe transverse positioning	PNF = $(x, y)$ positioning , SNF = $(\theta, \phi)$ positioning
Probe axial positioning	PNF = $z$ positioning , SNF = $r$ positioning
Gain reference or probe gain uncertainty	Not required if simulating relative uncertainties
Normalization constant	Not required if simulating relative uncertainties
Impedance mismatch	Not required if simulating relative uncertainties

## Near-Field Measurement Error Sources (2 of 2)\*\*\*

Error Term	Comments
Aliasing	Typically avoidable if adhering to spatial Nyquist criteria ( $\Delta x, \Delta y \leq \lambda/2$ for PNF, $\Delta\theta, \Delta\phi \leq 180^\circ/k r_o$ for SNF)
Truncation	Can be avoided in SNF by measuring a full sphere; however, truncation is always a source of error in PNF
Receiver linearity	Typically avoidable if receiver is kept in linear regime during measurements
Systematic phase and amplitude variations	For example, phase and amplitude variations vs. position due to cable flexing errors, phase and amplitude drift vs. time due to temperature and humidity variations, etc.
Leakage	Near-field measurements are particularly susceptible to errors caused by leakage
Receiver dynamic range	Also known as signal-to-noise ratio (SNR)
Multiple reflections	Also known as probe-AUT multiple interactions
Chamber reflections	Also known as room scattering
Non-Uniform Spatial Sampling	Interpolation required prior to executing NF-to-FF transform algorithm

# Simulation Process



# Near-Field Simulated Measurements for SNF (SNFsim)

## Inputs:

- AUT far-field (FF) pattern
- “Measurement” Probe FF Pattern
- Spatial  $\theta$ - $\varphi$  grid
- Radial measurement distance
- Frequency

Outgoing voltage    Incoming voltage    Transmission coefficients of AUT    Probe response constants

$$w(\theta, \phi, A, \chi) = v \sum_{\substack{smn \\ \mu}} T_{smn} e^{im\phi} d_{\mu m}^n(\theta) e^{i\mu\chi} P_{s\mu n}(kA)$$

**SNF Transmission Equation**

## Near-Field (NF) Pattern Simulation Steps:

- Calculate “measurement” probe coefficients in receive mode. Calculate translation coefficients based on frequency & radial measurement distance.
- Compute probe response constants by summing the product of translation coefficients & probe coefficients over all appropriate dimensions. SNFsim **not limited to  $\mu = +/-1$**  for NF pattern generation.
- Calculate rotation coefficients based on  $\theta$  &  $\varphi$  sampling angles
- Express AUT FF pattern as spherical modes, which have a discrete spectrum.
- Multiply rotation coefficients by AUT modes and sum over all appropriate dimensions to get new rotation coefficients.
- To execute the transmission equation, multiply new rotation coefficients and probe response constants, then sum over all appropriate dimensions to obtain the final complex voltages for the near-field (NF) pattern.
- Perform these steps for each polarization component ( $\theta$  &  $\varphi$ ) to get a dual-pol simulated NF pattern.

**When running the tool on a computer with modern hardware (12+ cores @ 2+ GHz & > 64 Gb of RAM), we can obtain simulation outputs in a few hours.**

NF Patterns Simulated Using the SNF Transmission Equation

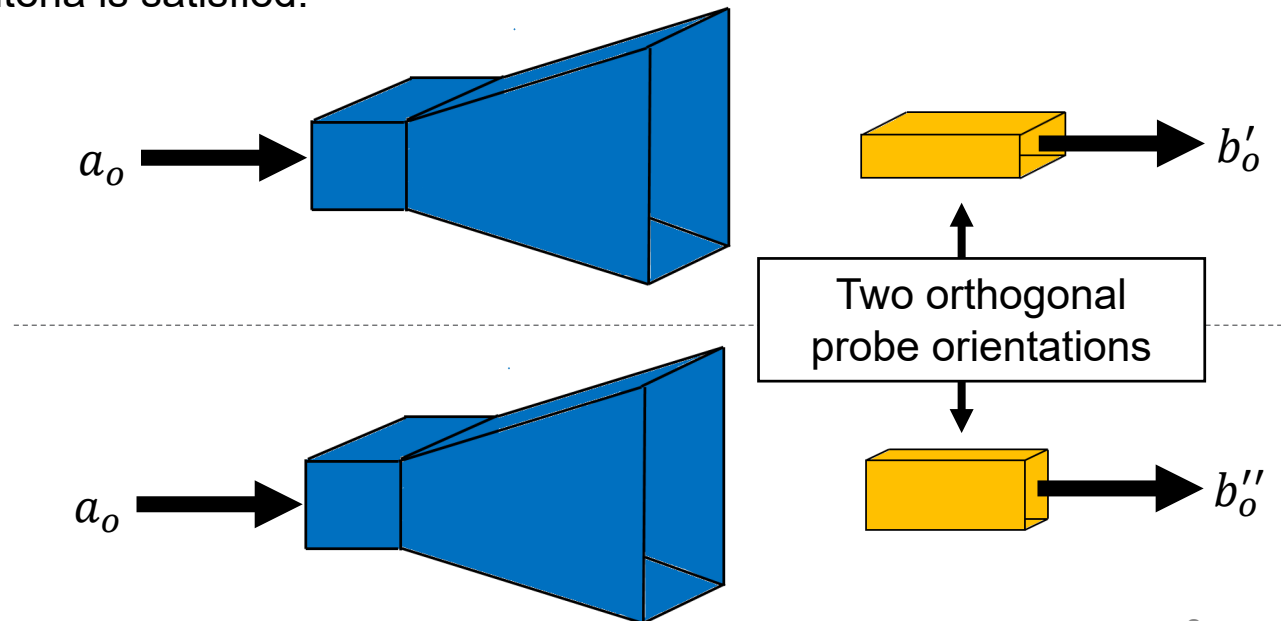
# Near-Field Simulated Measurements for PNF (pnfSim)

- Just like SNF, we simulate the PNF measurements using the PNF transmission equation for two orthogonal orientations of the probe (or two ports of a dual-polarized probe)
  - Orientation #1 is indicated with the “prime” symbols, and orientation #2 is indicated with the “double-prime” symbols
  - Software referred to as *pnfSim*, stands for Planar Near-Field Simulator
- However, since the plane-wave spectrum is continuous (not discrete, like the spherical-wave spectrum), we iteratively converge on the simulated measurement results
  - First, run the simulation using a pre-determined discrete number of  $k_x$  and  $k_y$  values, and **numerically integrate** the transmission equation. Then, re-run the simulation using smaller  $\Delta k_x$  and  $\Delta k_y$  values, **compare to previous iteration**. Repeat until user-defined convergence criteria is satisfied.

$$b'_o(\vec{P}) = a_o \int \vec{t}_{10}(\vec{K}) \cdot \vec{s}'_{02}(\vec{K}) e^{jk_z d} e^{j\vec{K} \cdot \vec{P}} d\vec{K}$$

↑  
 Outgoing voltage  
 Incoming voltage  
 Transmission coefficients of AUT  
 Receiving coefficients of probe

$$b''_o(\vec{P}) = a_o \int \vec{t}_{10}(\vec{K}) \cdot \vec{s}''_{02}(\vec{K}) e^{jk_z d} e^{j\vec{K} \cdot \vec{P}} d\vec{K}$$

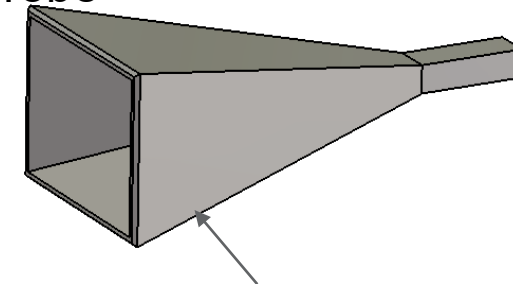
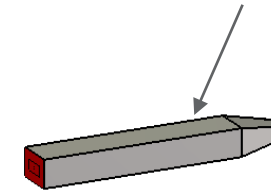




# A Quick Validation: Comparing Simulated Near-Field Measurements with Actual Measurements

- We modeled a probe and an AUT in CST to acquire far-field patterns, then used these patterns within pnfSim to simulate the near-field measurement
- Performed the same measurement using the real-world probe and AUT
- Very good agreement obtained between simulated and measured near-fields
- The pnfSim uses the Weyl identity to handle singularities encountered when executing the PNF transmission equation (a  $1/\cos\theta$  term appears when computing AUT transmission coefficients from far-field patterns)

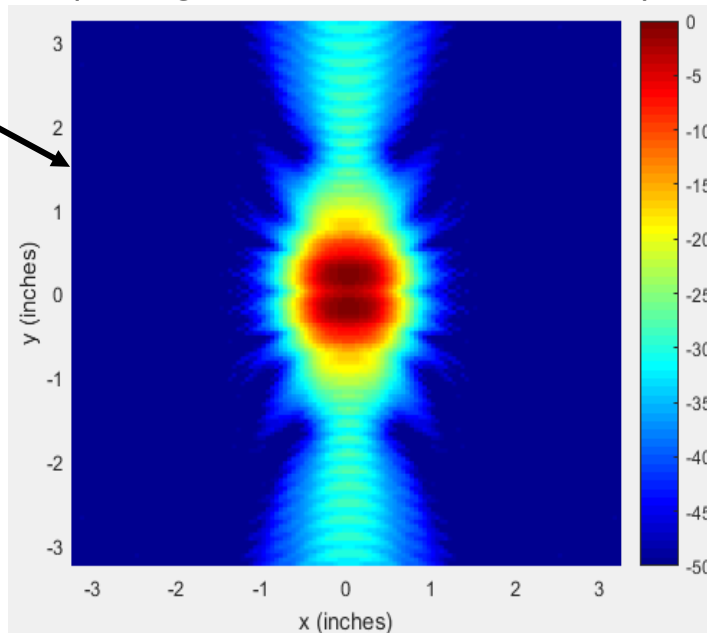
Rectangular Open-Ended  
Waveguide Probe



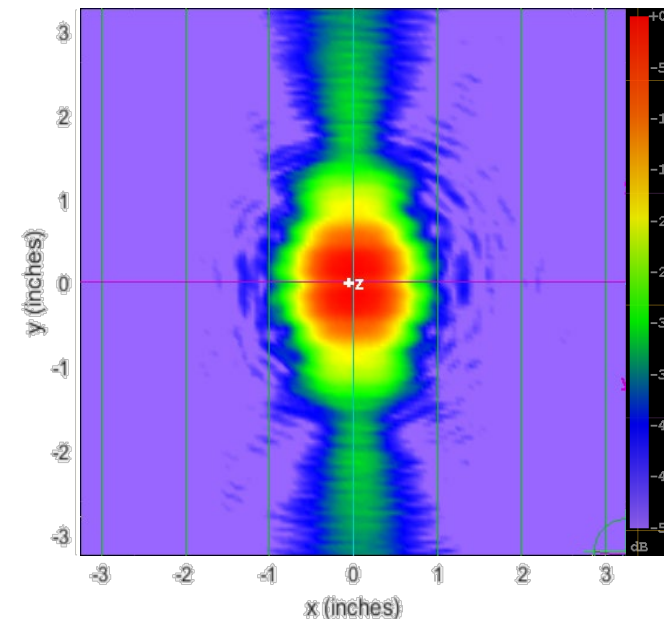
AUT = Pyramidal Horn

It takes pnfSim **1 to 5 minutes** to compute all near-field measurements using the modal expansions + transmission equation approach. A full-wave solver (like CST or HFSS) would take **1 to 5 days** to compute a comparable data set (depending on computational hardware available, of course).

pnfSim, after convergence  
(plot generated in MATLAB)



Measurement  
(plot generated in NSI2000)



# Measuring and Incorporating Errors

- We don't have quite enough time today to describe how to measure and simulate each error source
- Therefore, we've picked two case-studies that are applicable to both simulation packages:
  1. Receiver dynamic range, also known as signal-to-noise ratio (SNR)
  2. Probe positioning errors, transverse and axial
- We will demonstrate how to isolate each error source via measurement, and how to incorporate these particular error sources in the simulations

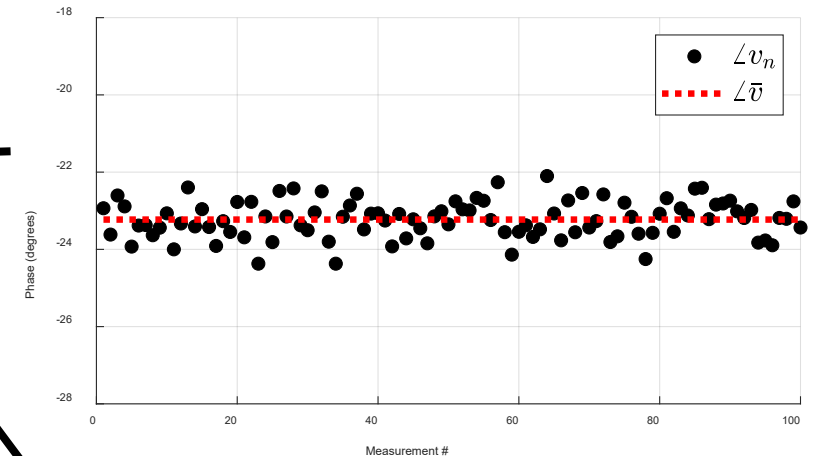
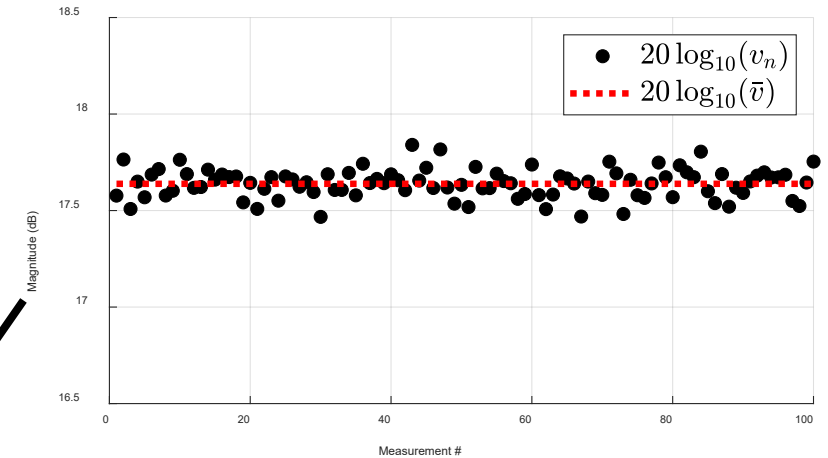


# Raytheon's Preferred SNR Measurement Method

- Dynamic Range is defined as the dB difference between the peak near-field signal and the noise floor (SNR = signal-to-noise ratio).
  - **Note:** The peak near-field value may not be at AUT boresight! Requires searching for peak value.
- Raytheon's recommend procedure for measuring and computing SNR:
  1. With the probe oriented for a peak near-field, co-polarized measurement, a total of  $N$  measurements are collected with the receiver. Each measurement is a complex value (it has magnitude and phase). Each measurement can be represented as  $v_n$ .
  2. Compute the coherent near-field SNR as follows:

$$\text{Coherent SNR (dB)} = 20 \log_{10} \left[ \frac{|\bar{v}|}{\sqrt{\frac{1}{N} \sum |v_n - \bar{v}|^2}} \right], \text{ where } \bar{v} = \frac{1}{N} \sum_{n=1}^N v_n$$

- Increasing the number of averages or decreasing the receiver's IFBW will improve SNR, but will lead to longer measurement times
- There's a trade-off between obtaining a statistically significant number of samples, and unintentionally measuring both SNR and system drift. In practice, we recommend taking as many repeated measurements as possible within about 30 seconds to keep drift to a minimum.



SNR value used in simulator

# How to Incorporate Receiver Dynamic Range in the Simulator

```
% Near-Field measurements are bp and bpp (for PNF).
% Assuming that bp and bpp are the same size:
[nRows,nCols] = size(bp);

% Coherent SNR, in dB, defined from peak signal:
coherentSNR_dB = 40;

% Column vector with all near-field measurements:
allNearFieldMeasurements = [bp(:) ; bpp(:)];

% Find the peak near-field value:
peakNearFieldValue = max(allNearFieldMeasurements);

% Calculate sigma, standard deviation for complex noise:
sigma = abs(peakNearFieldValue)*10^(-coherentSNR_dB/20);

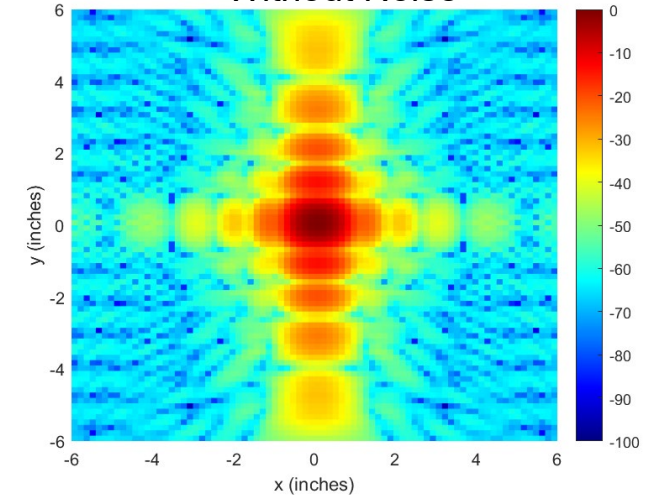
% Generate the random complex noise. Division of real and imaginary parts
% by sqrt(2) required for conservation of total noise power.
noise1 = random('norm',0,sigma,[nRows,nCols])/sqrt(2) + ...
    1i*random('norm',0,sigma,[nRows,nCols])/sqrt(2);
noise2 = random('norm',0,sigma,[nRows,nCols])/sqrt(2) + ...
    1i*random('norm',0,sigma,[nRows,nCols])/sqrt(2);

% Add the noise to the measured near-field values:
bp_withNoise = bp + noise1;
bpp_withNoise = bpp + noise2;
```

**MATLAB code for adding noise to simulated measurements.**

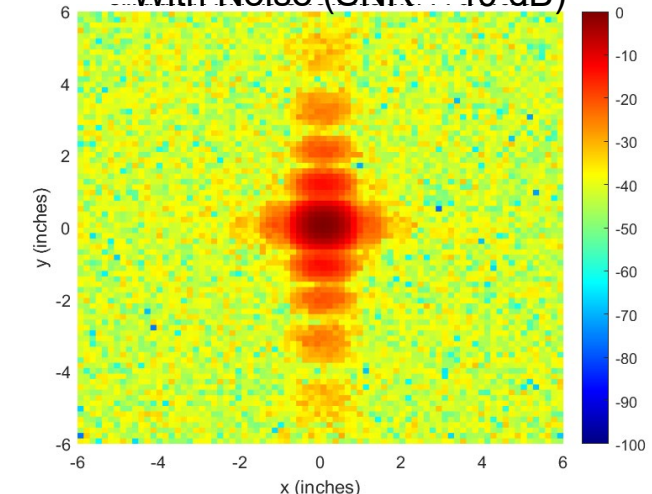
Simulated Near-Field Measurements

Without Noise



Simulated Near-Field Measurements

With Noise (SNR = 40 dB)

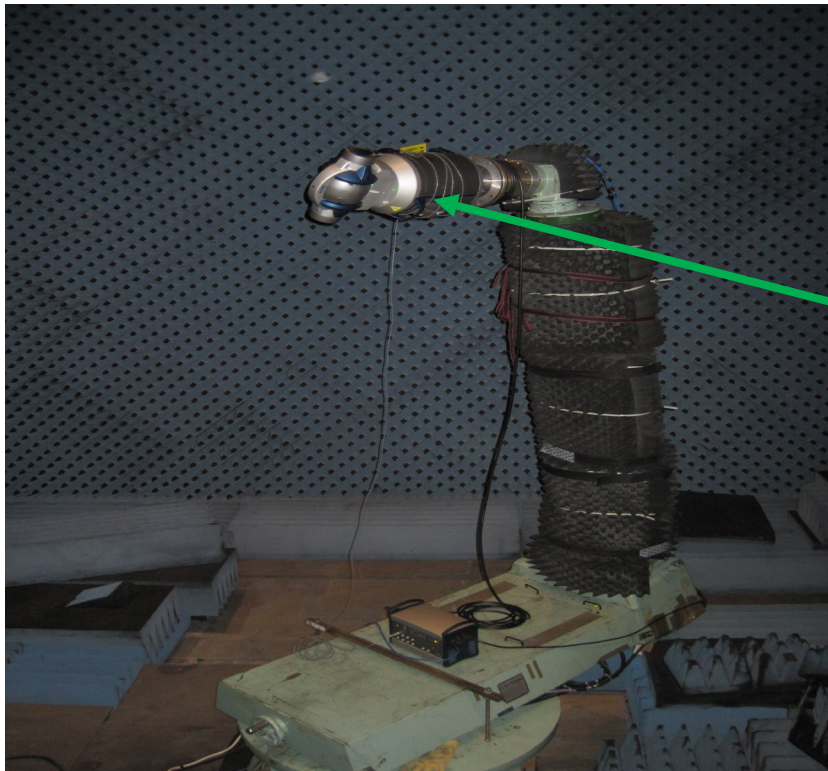




# Measurements of Positioner Errors

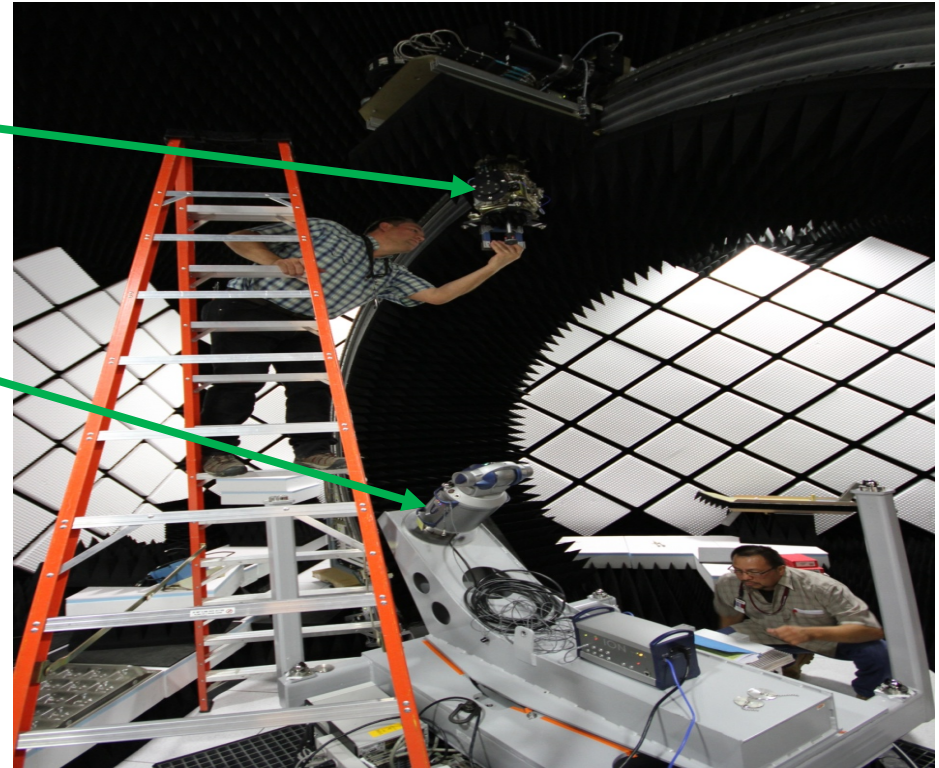
$r, \theta, \phi$  position errors for SNF, x,y,z position errors for PNF :

The achieved positions for the measurement probe are not exactly what is commanded. This creates a non uniform sample grid for the measured data. In order to quantify these positioner errors, we use a laser tracker to measure the actual positions as a function of commanded positions for our system.



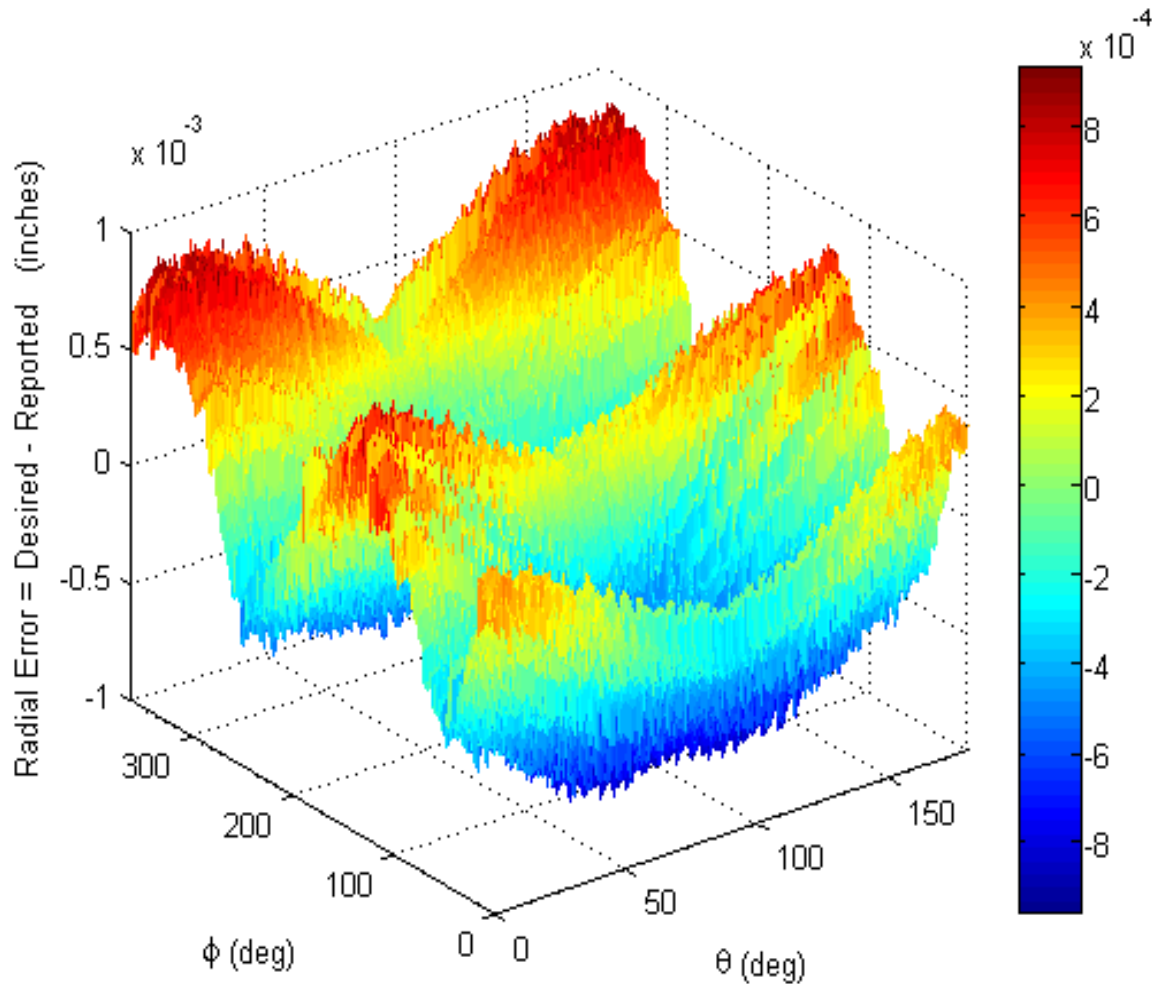
NF Measurement Probe

Laser tracker

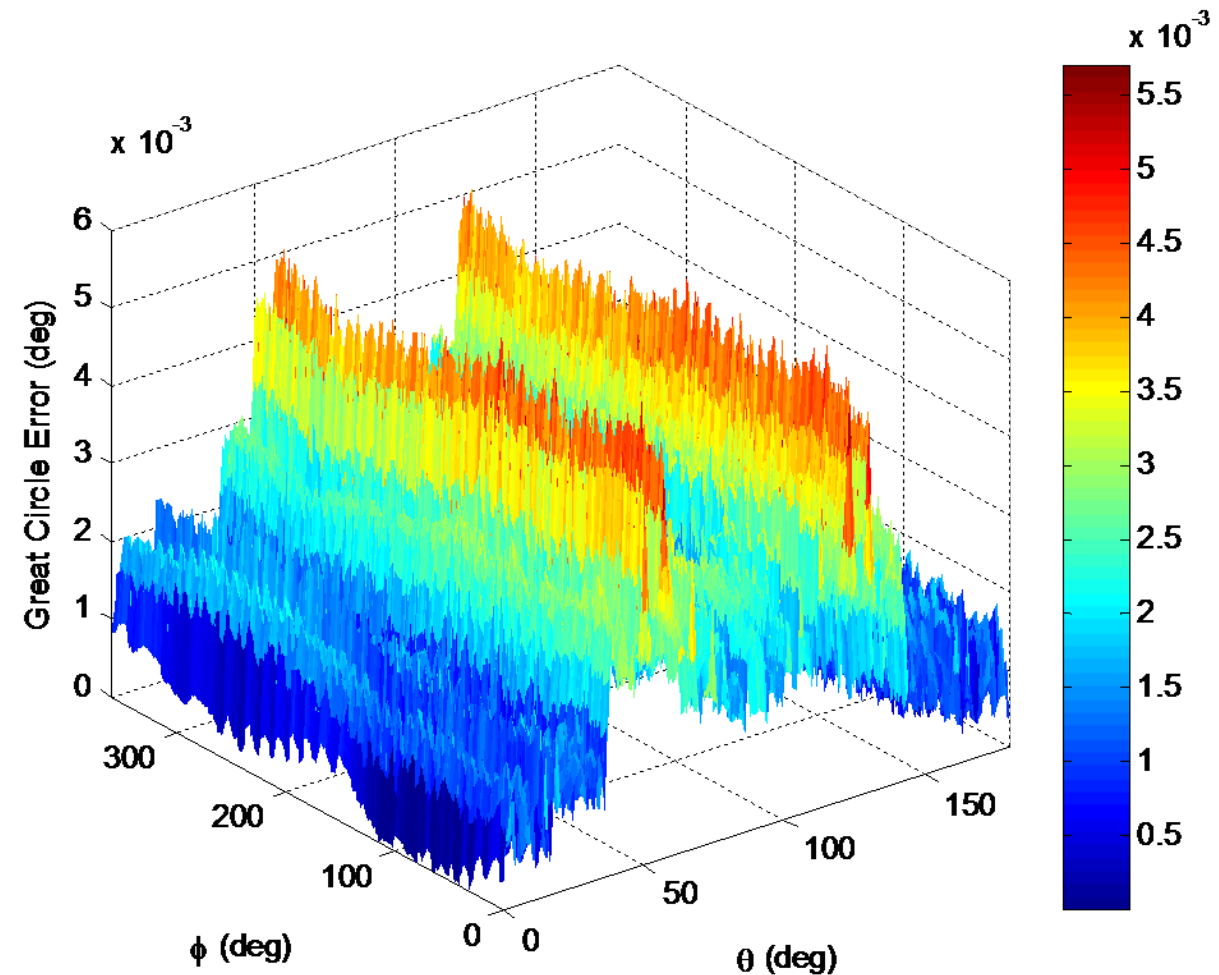


# SNFsim Positioner Errors

Radial position errors, worst case = 0.00096"



Great circle position errors, worst case = 0.0057°



SNF positioning errors can be minimized via careful calibration w/ error correction maps applied in real-time during the measurement. In SNF, radial real-time correction relies on an automated radial stage for the probe.

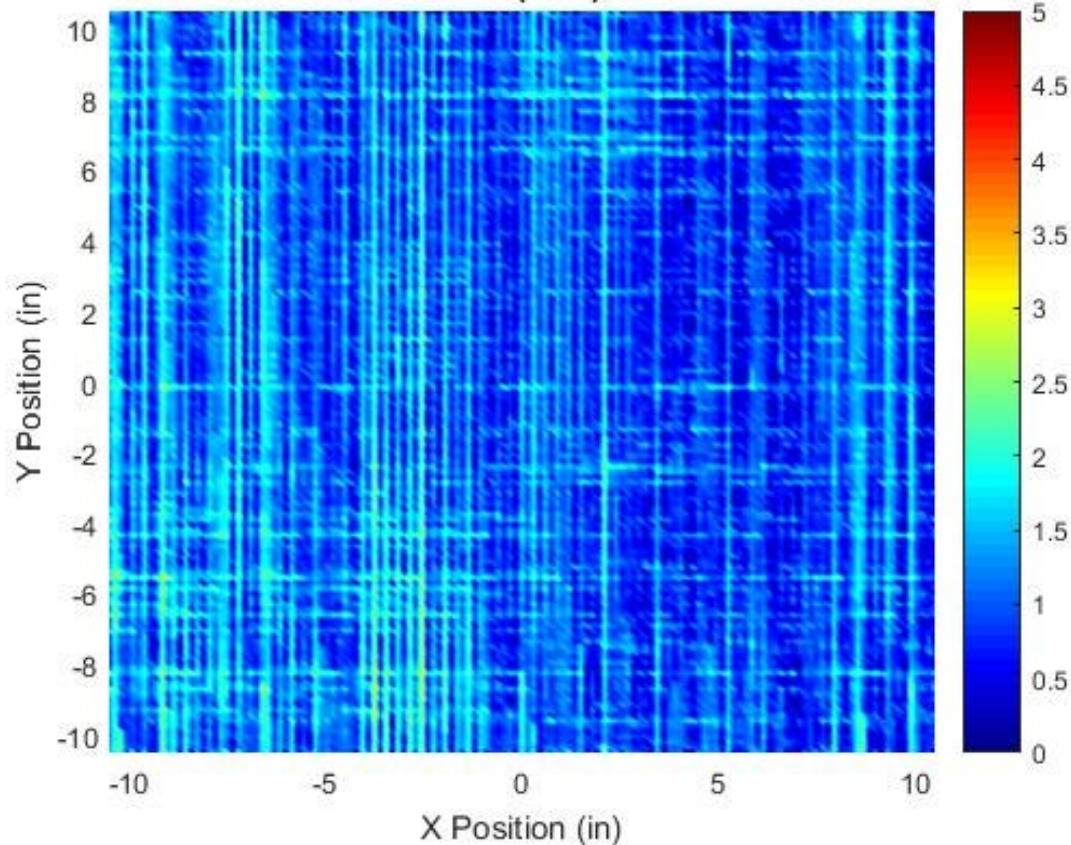


# PNF Transverse (Translational) and Axial (z) Position Errors

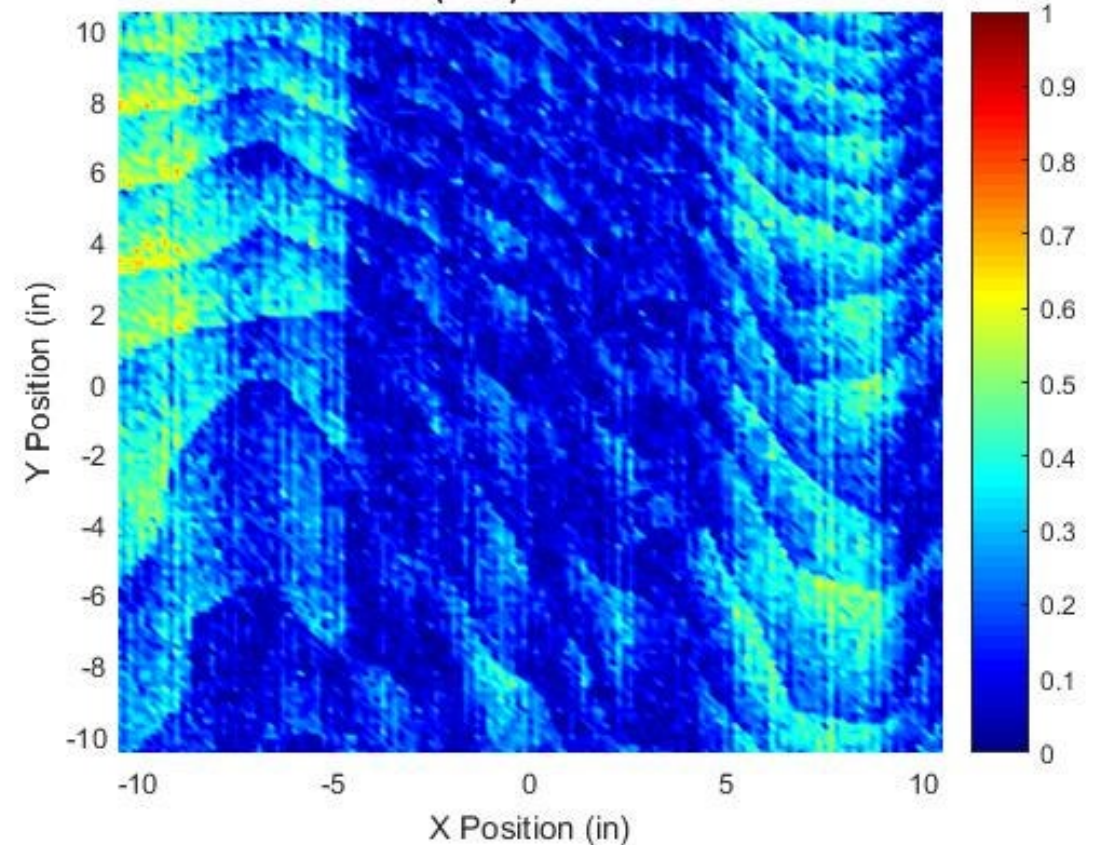
$$\text{translational error} = \sqrt{(x_m - x_i)^2 + (y_m - y_i)^2}$$

$$z \text{ position error} = |z_m - z_i|$$

Translational Position Error (mils) in Ka-Band Scan Plane



Z-Position Error (mils) in Ka-Band Scan Plane



PNF positioning errors can be minimized via careful calibration w/ error correction maps applied in real-time during the measurement. Residual errors measured while using the real-time corrections are shown in the above plots. In PNF, z-axis real-time correction relies on an automated radial stage for the probe.

# Positioner Error Simulation Inputs

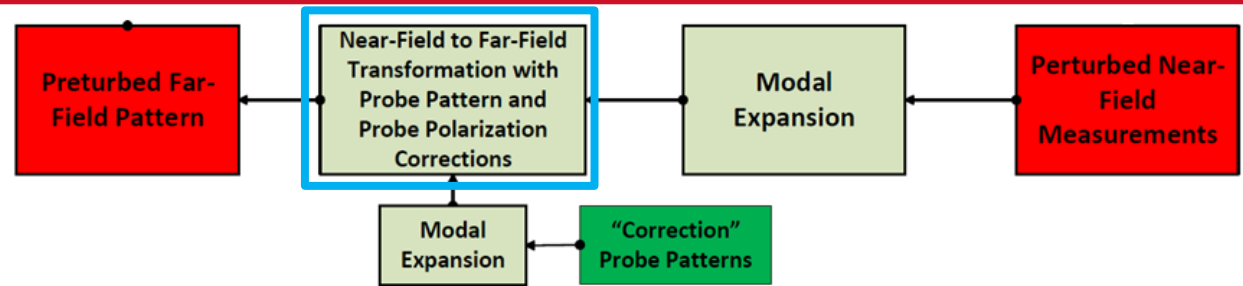
- The user inputs to the simulations are a tabulated grid of measured points (plots from the previous slides) that indicate the commanded and achieved positions of the probe for a NF data collect.
- The simulated collection then takes place on a grid that is perturbed from uniform in  $r, \theta, & \phi$  (for SNFsim) or  $x, y, & z$  (for pnfSim).

NF simulations use actual measured data



# SNF NF-FF transform

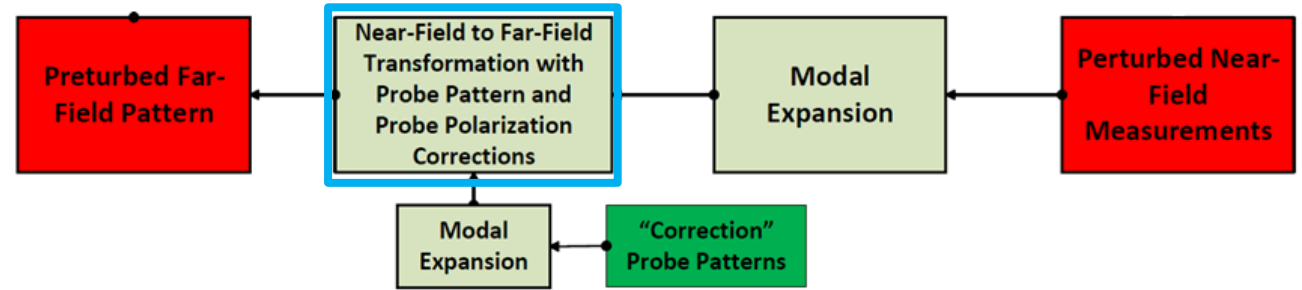
$$\vec{E}(r, \theta, \phi) = \frac{k}{\sqrt{\eta}} \sum_{smn} Q_{smn}^{(3)} \vec{F}_{smn}^{(3)}(r, \theta, \phi)$$



**NistSphere1.0.6:** This MATLAB package was derived from a government SNF – FF transform code that was developed by Ron Wittman (ret.) and supplied by Randy Direen.

- Calculate probe corrections: Linear polarization ratios and amplitude-phase factor for the dual-polarized SNF probe.
- Apply the complex linear polarization ratios and complex amplitude phase factor of the measurement probe to the  $\theta$ - $\phi$  components of the NF pattern.
- Calculate the translated probe modes based on frequency & radial measurement distance.
- Condition the NF AUT data so that it can have Fourier series operations performed ("double-sphere").
- Transform the NF AUT data from pattern space into mode space.
- Correct the AUT's modes with the translated probe modes to remove probe effects.
- Calculate antenna pattern Fourier coefficients using corrected AUT modes.
- Convert from fourier series to final FF pattern (on a  $\theta$ - $\phi$  spatial grid) with a 2D inverse FFT.

# PNF NF-to-FF Transform



1. The angular spectra  $D'(\vec{K})$  and  $D''(\vec{K})$  are computed from the error-perturbed simulated near-field measurements. Due to the band-limited nature of the measurements, the integrals can be implemented without error using 2D discrete Fourier transforms.

$$D'(\vec{K}) = \frac{e^{-jk_z z}}{4\pi^2} \iint \frac{b'_o(\vec{P})}{a_o} e^{-j\vec{K} \cdot \vec{P}} d\vec{P} \quad , \quad D''(\vec{K}) = \frac{e^{-jk_z z}}{4\pi^2} \iint \frac{b''_o(\vec{P})}{a_o} e^{-j\vec{K} \cdot \vec{P}} d\vec{P}$$

2. Probe pattern correction is executed, using the “correction probe patterns” provided by the user. The “correction probe patterns” should match the patterns that will be used during the real-world NF-to-FF transformation. For PNF, no extra step is required for probe polarization correction, it is built-into the probe pattern correction algorithm. AUT transmission coefficients are obtained.

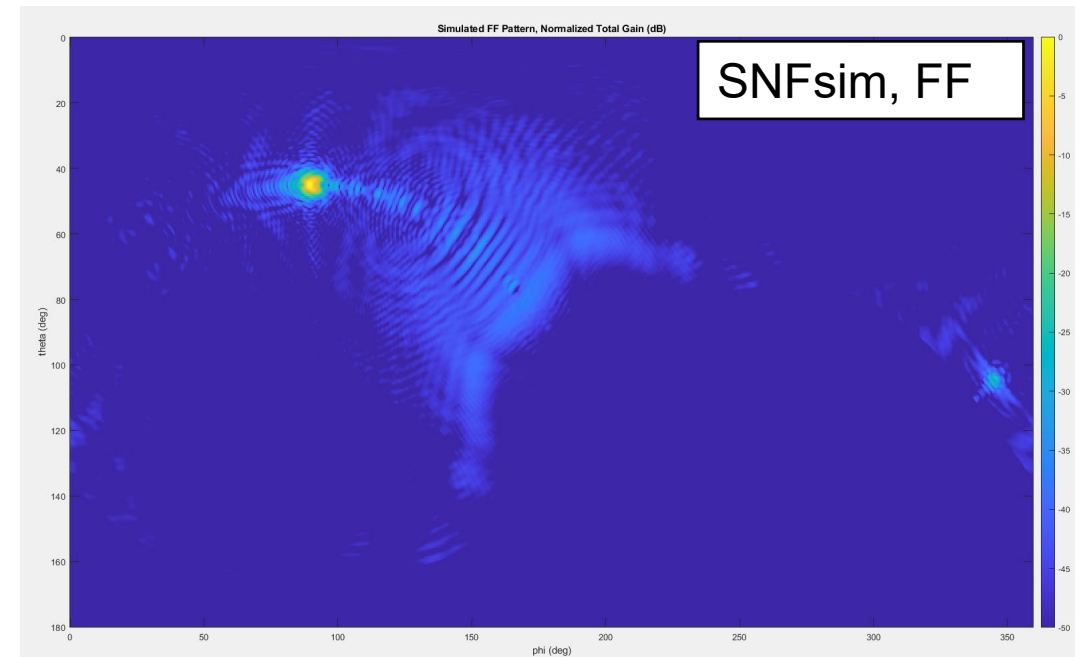
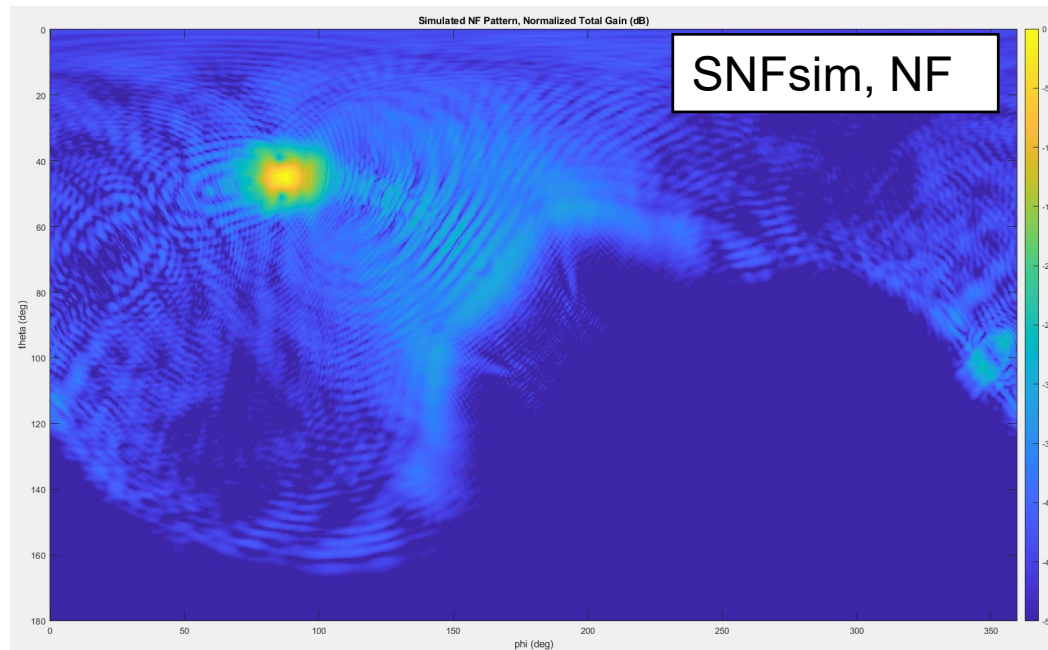
$$t_m(\vec{K}) = \frac{D''(\vec{K})s'_c(\vec{K}) - D'(\vec{K})s''_c(\vec{K})}{s''_m(\vec{K})s'_c(\vec{K}) - s'_m(\vec{K})s''_c(\vec{K})} \quad , \quad t_c(\vec{K}) = \frac{D'(\vec{K})s''_m(\vec{K}) - D''(\vec{K})s'_m(\vec{K})}{s''_m(\vec{K})s'_c(\vec{K}) - s'_m(\vec{K})s''_c(\vec{K})}$$

( $m$  = Main Polarization,  $c$  = Cross Polarization, assumes two orthogonal transverse components)

3. AUT transmission coefficients are used to compute the AUT far-field pattern. Very easy with PNF, since  $\vec{E}_{FF}(\theta, \phi) \sim \cos \theta \vec{t}(\vec{K})$ .

# Simulation Results/Outputs

- Main output from simulation packages are the Near-Field & Far-Field patterns.
- Perform comparisons & analysis on these field quantities using the post-processing tools of your choice.
- Our comparison metrics are Pattern Similarity Metric (PSM) & Equivalent Stray Signal (ESS).
- PSM compares the co-pol field components of 2 appropriately scaled patterns within the 3 dB BW.
  - Create a common grid, then RSS the differences between the 2 patterns to condense into 1 value.
- The Equivalent Stray Signal (ESS) is a metric used to compare the magnitudes of two antenna patterns
  - For more details on ESS (E/S), see Ch. 10 of *Theory and Practice of Modern Antenna Range Measurements*, 2<sup>nd</sup> ed.

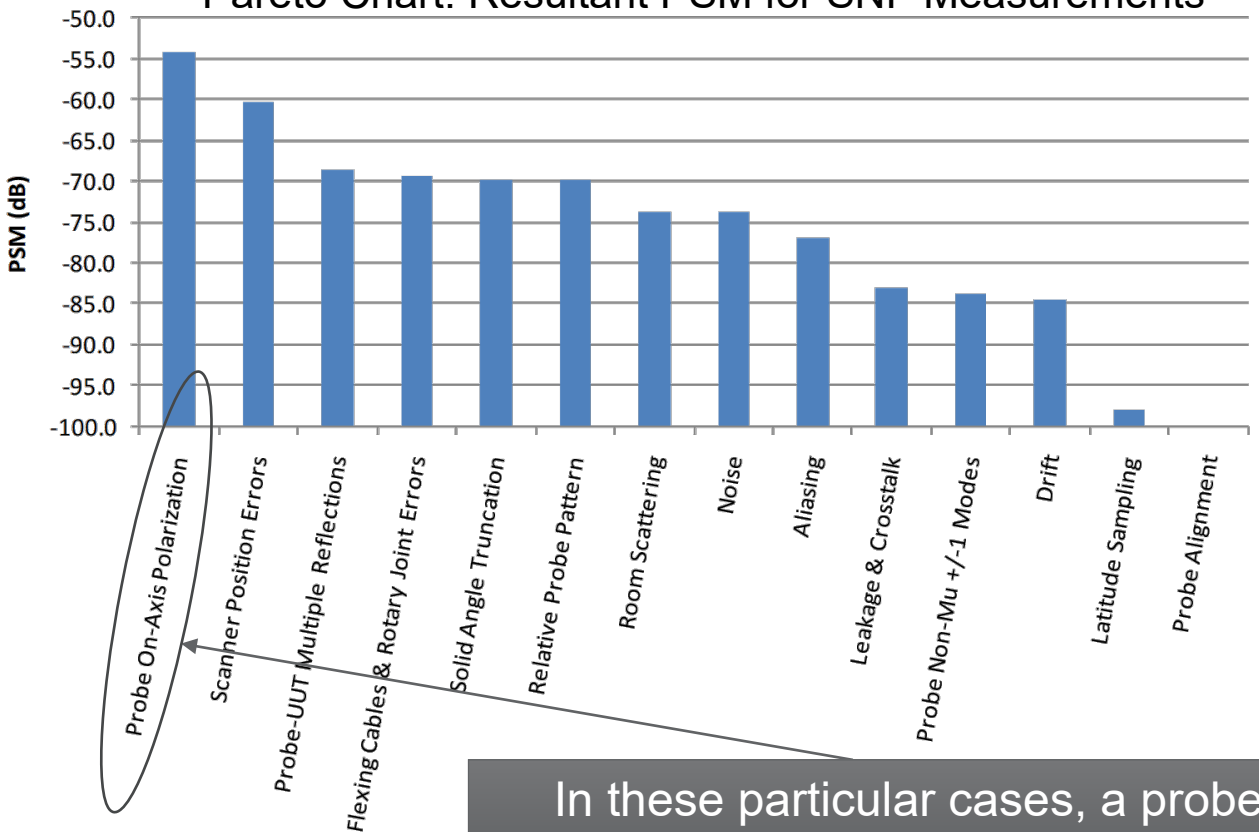


Simulation gives insight into both NF & FF patterns

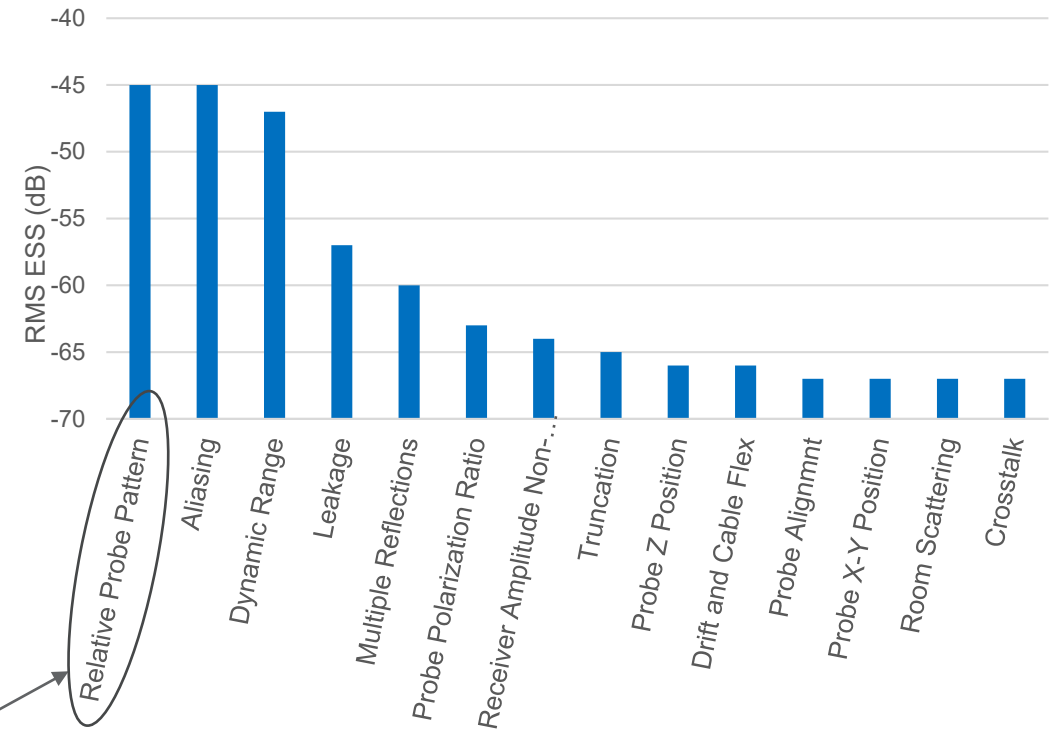
# Pareto Charts for SNF and PNF Results

- Placing the resulting PSM or ESS values on a Pareto chart allows us to evaluate the error source that causes the most measurement uncertainty
- If there are simple ways to reduce the leading error sources, this can be implemented for reduced measurement uncertainties (e.g., increase averaging to improve SNR)

Pareto Chart: Resultant PSM for SNF Measurements



Pareto Chart: Resultant RMS ESS for PNF Measurements



In these particular cases, a probe-related error source was the leading cause of uncertainty. Scanner position errors was next-largest source of error for SNF, whereas aliasing and dynamic range (SNR) were next-largest error sources for PNF



# Two Approaches for Combination of Errors

## 1. Root Sum Square (RSS) each error's value for cumulative error value:

- Run the simulation N times, each with only 1 of the individual errors.
- Assumes that each error is orthogonal to the others.
- Take the square root of the sum of the squares (RSS) of each linear value to get a cumulative value.

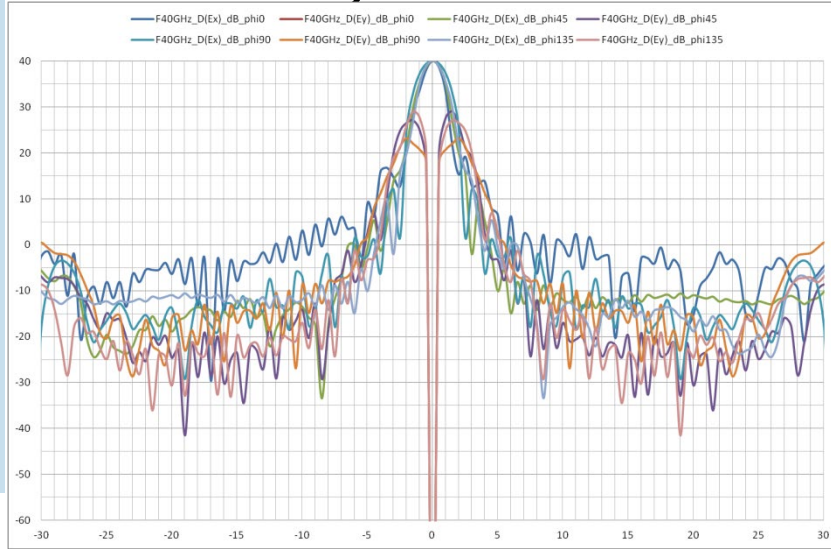
## 2. Simulate once with all errors as inputs to get a cumulative error value:

- Run the simulation once with all of the individual errors.
- Allows for capturing interactions among errors without assumptions of orthogonality.
- Use the same inputs for each individual error as in method 1, noting that the effect of SNR errors on the NF pattern will not be exactly the same as in method 1 (see slide 12).
- The final simulation result is the cumulative value.

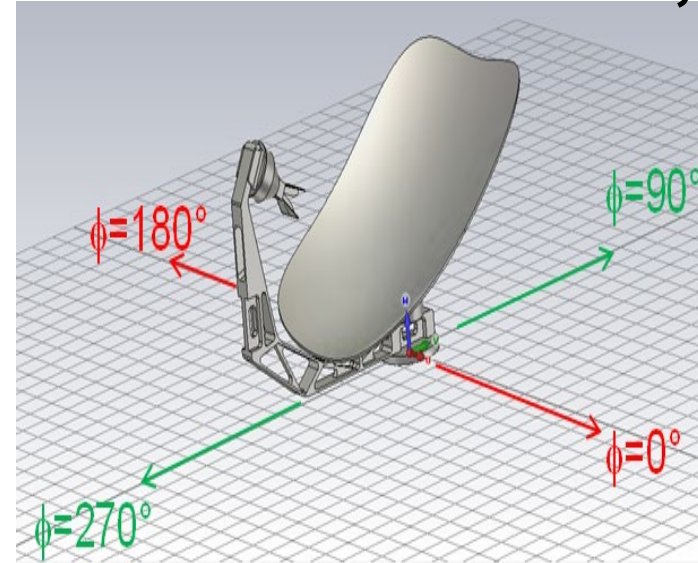
Can be done for any number of combination of errors.

# Simulation Validation, SNFsim

## SR-40A Reflector, measured



## SR-40A Reflector, simulated

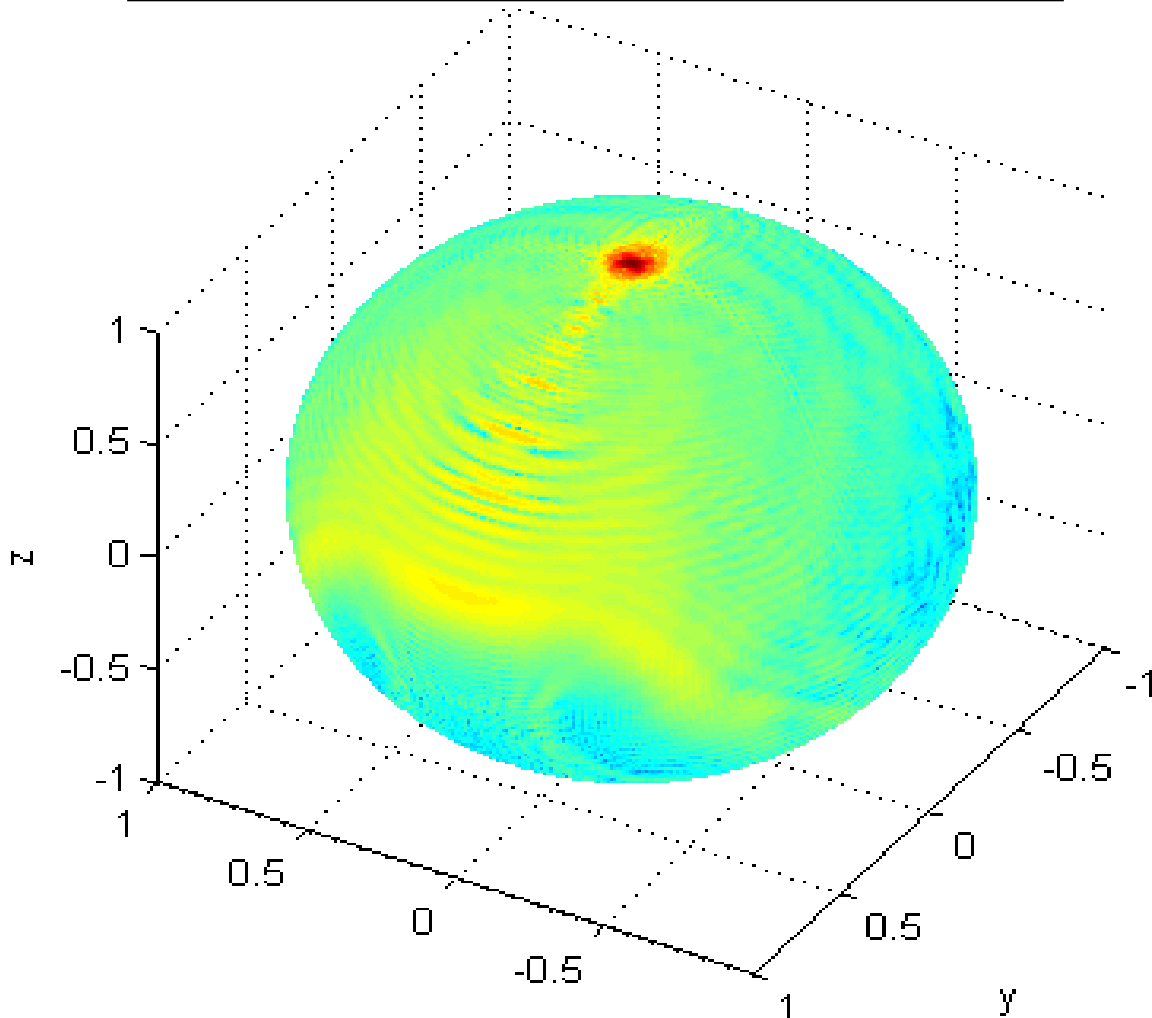


- A high fidelity, 3D model in CST Microwave Studio was provided by Satimo
- Model run with 5 billion mesh cells at Ka band

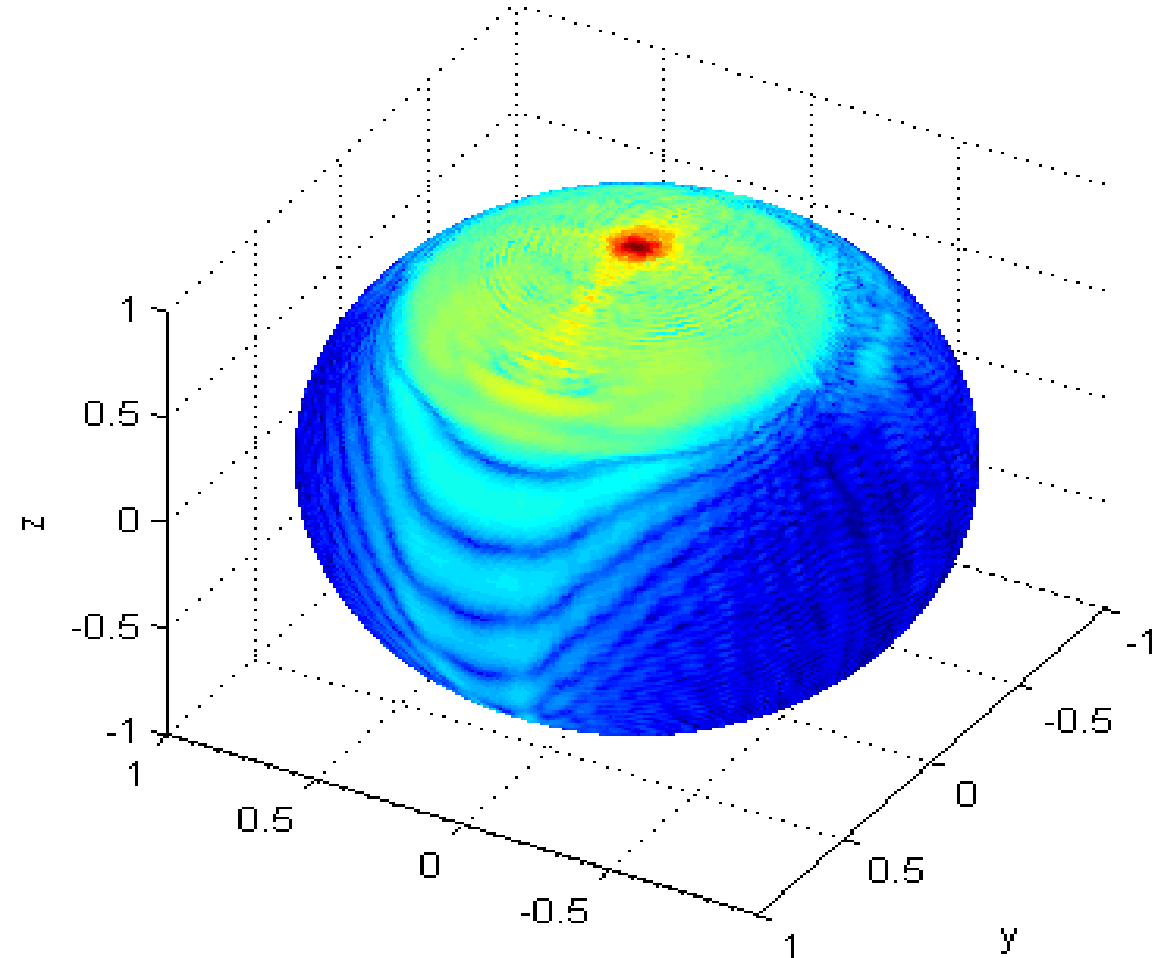
- We compared a measured FF pattern (latitude sampling grid & truncated at 40° around the beam pointing direction) to a simulated FF pattern output from SNFsim that had incorporated all appropriate error sources.
- Some translations & rotations necessary to get both FF patterns in the same coordinate system.

# Comparing Measured to Simulated, SNF

Simulated with SNFsim



Measured in SNF chamber



Sum of RMS of differences on 3 dB BW grid  $\sim 6.3 \times 10^{-3}$  (-44 dB PSM)  
Within the measured truncation radius ( $40^\circ$ ), there is good agreement between the patterns as well.

# Conclusions & References

- **Questions that the PNF and SNF simulators can answer:**

1. Is it possible to build a measurement system that can meet pre-determined measurement uncertainty requirements based on anticipated error source magnitudes?
2. How sensitive is a particular AUT to the various potential error sources?
3. What is the magnitude allowable from each error source to develop error budget requirements?

- **For Raytheon Technologies, pnfSim and SNFsim have been invaluable tools:**

- Used to assess the quality of our antenna measurement systems.
- Used to quantify how individual error sources of our antenna measurement systems impact far-field pattern uncertainty.
- Used to validate that our systems are able to meet the requirements for far-field pattern accuracy.

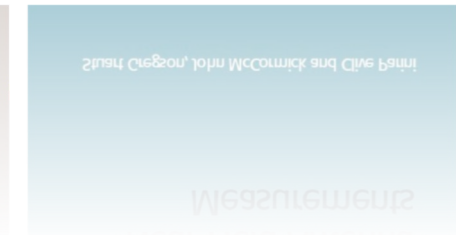
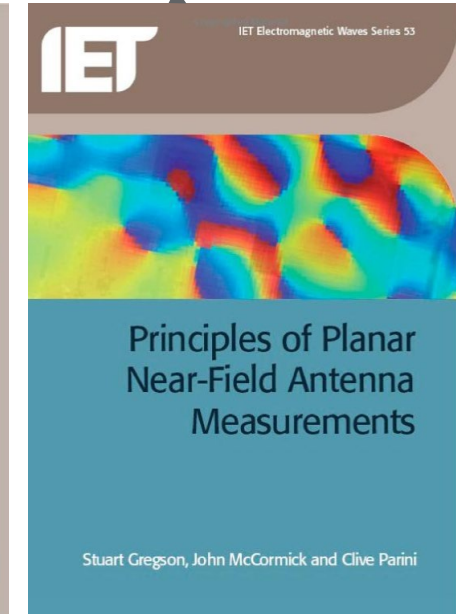
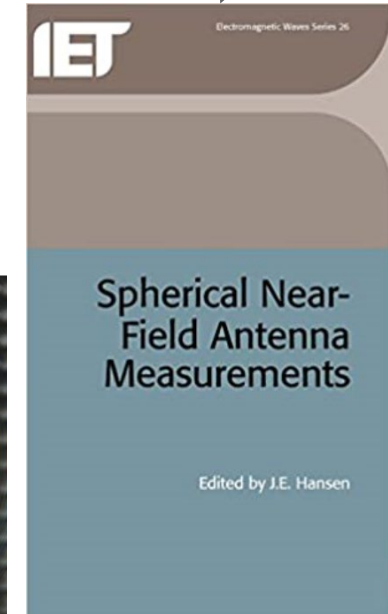
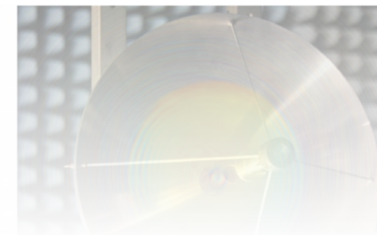
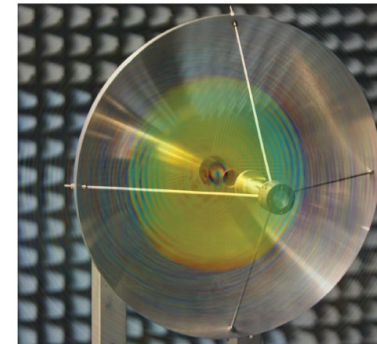
- We did not have the time today to discuss **the nuts-and-bolts of the math and code** behind a near-field simulation tool, but hopefully this presentation provides enough “breadcrumbs” to serve as a starting point for those interested in building their own near-field measurement simulation tools.

More info on  
near-field range  
assessments

99% of the equations needed to  
develop a near-field pattern simulator  
can be found in these two books

**IET** The Institution of  
Engineering and Technology  
**Schtech**  
PUBLISHING  
an imprint of the IET  
**Theory and Practice of Modern  
Antenna Range Measurements**  
2nd Expanded Edition, Volume 2

Clive Parini, Stuart Gregson, John McCormick,  
Daniël Janse van Rensburg and Thomas Eibert





# Thank you.

