Application of Kernel Density Estimation to Achieve Automated Near Real-Time Antenna Pattern Data Processing and Analysis in an Anechoic Chamber

Emily Mendoza
412 TW EWG 772 TS
Benefield Anechoic Facility (BAF)
Edwards AFB, CA, United States

Shea Watson
Air Force Research Laboratory (AFRL)
Wright-Patterson AFB, OH, United States

***Abstract*—The Benefield Anechoic Facility (BAF) at Edwards Air Force Base is the world's largest known anechoic chamber. Due to its unmatched size and equipment inventory, the BAF hosts far-field pattern measurements at all azimuth angles and multiple simultaneous elevations of installed antennas on large aircraft typically across the BAF's operating frequency range of 0.1 – 18 GHz. Antenna tests at the BAF rapidly produce copious data, which often require immediate analysis to allow system owners to make relevant improvements. Historically, the BAF had accomplished quality assurance manually. Analysis was accomplished post-test by customers and the BAF team. Today, the BAF team has developed scripts that use kernel density estimation and basic machine learning to automatically check incoming data and highlight anomalies for review. During a 2019 test of installed antennas on a B-1B bomber, the BAF team used these scripts to process antenna patterns in near real-time and bring unusual results to the customer's attention fast enough to allow modifications to be applied and re-tested during the same test event – highly significant as aircraft and BAF schedule times are limited and may be a one-time opportunity to gather required data.**

This paper will explore the algorithm used to evaluate antenna patterns, as well as the expected characteristics of patterns that enable the selection of relevant data. Development of this algorithm found that using kernel density estimation to calculate the maxima in a pattern's distribution of gain values, then performing this recursively over the main lobe, can identify problems such as incorrect switching, mismatched transmission lines, and multipath. Algorithm optimization was achieved using generated data, then the algorithm was implemented as a prototype during the B-1B test by searching for data that deviated from a sample pattern. Finally, this paper will discuss the application and impact of this algorithm during a live test.

All plotted data and numerical results included in this paper are based on training data generated in Python to represent generic directive antenna patterns, and do not correspond to the real-world conditions or results of the B-1B test.

I. Introduction

Typical modern antenna measurements take place within near-field anechoic chambers, which use spherical, cylindrical, or planar scanning and accompanying processing techniques to reproduce the far-field coverage patterns of antennas or systems under test. Often limitations in size or available test equipment leads to chambers dedicated to testing one specific type of antenna or system [1]. However, the uniquely large chamber at the BAF grants the 772 Test Squadron (772 TS) the capability to take immediate far-field measurements of antennas, including even simultaneous measurements of multiple installed antennas on massive systems such as bomber-sized aircraft. These qualities grant the BAF extreme testing flexibility, and as a result, the BAF hosts a wide range of systems that vary in their expected antenna coverage, intended purpose, or required equipment setup to integrate with the BAF antenna pattern architecture [2]. Engineers within the 772 TS thus face the challenge of processing, approving, and delivering thousands of antenna patterns for an unfamiliar system during any given test. Previously, the BAF team has surmounted these challenges by providing a live feed of an uncalibrated subset of collected data to the customer and the test team, and by performing thorough manual review during a pre-established time period after test completion. While these quality assurance measures have not been discarded, the increasing demands for test time and data quantities have created the motivation to make quality assurance more thorough and efficient.

To accomplish this, a simplified approach to characterizing antenna patterns was developed. This began by reducing patterns to distributions of their gain values. The characteristics of such a distribution then correspond to common features of functional antenna patterns, such as a main lobe made up of higher gain values contrasted by a noise floor. The choice of features to extract was driven by the ability to associate these features with the distribution of gain values, and by the common elements of pattern data that merited test team intervention based on experiences such as dealing with equipment failure or unexpected multipath effects. In this paper, this feature extraction approach will be accomplished via kernel density estimation and

applied to both generated data and previous test data. The analysis will optimize the approach based on these results, then apply the optimized technique during a live test event.

The approach was implemented as a prototype algorithm during a 2019 test of installed antenna patterns on the B-1B bomber in the BAF. To develop this prototype prior to the test event, training data generation scripts were created to represent generic directive antenna patterns over a common general BAF operating range of 2-18 GHz. All sets of plotted data and numerical results arise from this generated sample data set, and do not reflect the conditions or represent the results of B-1B testing in the BAF.

II. Pattern Structure and Software Parameters

Automated quality assurance functions by identifying which sets of antenna pattern data warrant manual review by looking for characteristics that correspond to equipment issues or unexpected system under test performance. This requires establishing a baseline description of a standard antenna pattern, which can be altered for specific tests, and then mapping types of gain value distributions to elements of this baseline. The following sections describe the characteristics of the baseline pattern, and how the algorithm analyzes the gain value distribution to check for every characteristic.

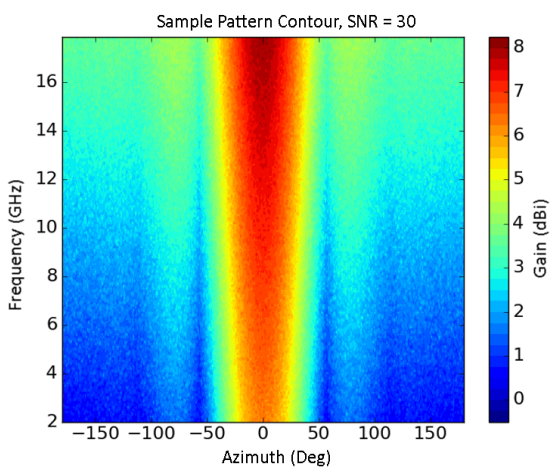

Fig. 1: Generated antenna pattern with standard characteristics and a high signal to noise ratio (SNR).

A. Main Lobe Detection

A typical directional antenna will have a main lobe of higher gain values, in contrast to the lower value side lobes, back lobe, and nulls that tend to lie at or just above the noise floor of a pattern measurement. In the patterns considered for this paper, this structure produces a bimodal distribution of gain values. This distribution has two peaks: one corresponds to all of the lower gain values that fall outside the main lobe, and the other to the higher gain values within the main lobe. Exceptions to this trend often arise from setup

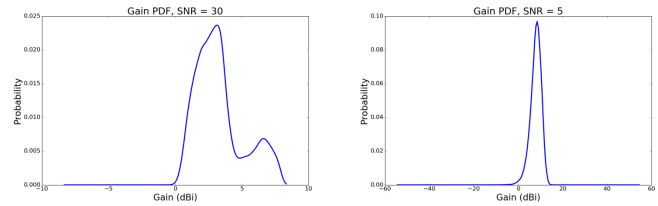

Fig. 2: Estimated PDFs of two antenna patterns. Left: PDF of pattern with an SNR of 30, with two peaks to differentiate main lobe from noise floor. Right: PDF of pattern with an SNR of 5, with only one peak.

failures that produce noise instead of a discernible pattern, which need to be resolved quickly during a test event. Examples include an improperly connected transmission line, an incorrect switch state within an RF network, or insufficient dynamic range in the pattern measurement design. Other issues could potentially produce other types of distributions as well, such as a prominent back lobe or side lobe creating a third peak. This may not represent an issue with the test setup, but rather a valid set of data that the test team may want to prioritize for analysis [3].

Although the aforementioned distribution of gain values in a pattern can be computed directly as a histogram, kernel density estimation offers an easily-implemented alternative approach that conveniently smooths the distribution as it approximates the probability density function (PDF) of the distribution [4]. The creation of a smooth PDF simplifies the peak search process that the algorithm uses to detect anomalies, and the level of smoothing serves as an adjustable parameter to tune the sensitivity of the algorithm. Critically, optimizing this degree of smoothing can be relatively straightforward but is a well-explored and potentially complex but powerful process [5]. The quality assurance algorithm begins by employing the ”gaussian_kde” function included with the SciPy Python package [6] to use a Gaussian kernel to approximate the PDF of the gain values in an antenna pattern. Here, one ”pattern” is defined as the set of gain values across a two dimensional range of azimuth and frequency values. Then the number of peaks in this function is calculated by searching for values that are larger than a given number of neighboring values. Both the amount of smoothing within the KDE process and the sensitivity of the algorithm to local maxima are parameters that can be changed by the operator to optimize for a given test. The effect of these parameters and the optimization process will be discussed further in the Simulated Results and Optimization section.

After determining the number of peaks in the distribution, the algorithm flags the pattern and reports it to the user if the distribution is anything but bimodal. Otherwise, the algorithm proceeds to the next step in the QA process.

B. Beam Width Estimation

Searching for irregularities within the main lobe of the antenna pattern requires first estimating the size and location of the main lobe. Since computing power has already been dedicated to recreating the PDF of the pattern, the algorithm reuses this information to approximate the half-power beamwidth [3]. This approximation works by comparing the heights of the two peaks in the distribution. A wider main lobe results in the overall pattern data set having a higher ratio of large gain values to small ones, so the ratio of the high gain value peak to the low gain value peak increases roughly proportionally. The actual equation used for this estimate is as follows:

$$Beamwidth \approx \frac{h_p}{h_p + h_n} \times Az_{tot} \tag{1}$$

Here h_p is the height of the peak that corresponds to the pattern's main lobe values, h_n is the height of the peak that corresponds to the noise floor values, and Az_{tot} is the total range of azimuth values present in the data set. This equation estimates the beamwidth roughly but quickly by assuming that the peaks can be reduced to triangles of equal widths, and whose relative areas are thus only dependent on the height of each peak. Although this tends to underestimate the half-power beamwidth, this underestimate helpfully avoids false positives by isolating the section of the main lobe that is typically the smoothest. Since the total area of these two peaks should encompass every gain value in the data set, this makes the proportion of azimuth values spanned by the main lobe a function of the ratio of the heights of these peaks. The algorithm centers the main lobe at the azimuth angle of the largest gain value. Truncating the data set down to the area encompassed by this estimated beamwidth allows the algorithm to move to the next QA step. The software also visually reports the estimated beamwidth if a pattern is flagged after this point so that a user can verify the accuracy of this process.

C. Discontinuities in Main Lobe

After the data set has been reduced to only the estimated main lobe of the entire pattern, the algorithm repeats the KDE calculation over this smaller set of data. Here the QA process is similar to that performed over the whole pattern. Since the average main lobe is a relatively constant collection of higher gain values, this new PDF will be unimodal unless a discontinuity exists within the main lobe. Pockets of lower gain in a main lobe often arise from improperly connected RF equipment, unwanted reflections within the test chamber, or performance deficiencies with the system under test itself [7]. If the software detects more than one peak in the PDF of a main lobe, it flags the pattern for engineer review and visually displays the area estimated to be the main lobe.

The result of a successfully flagged pattern with a main lobe discontinuity is illustrated in Fig. 3. The QA algorithm also adds lines to the contour plot to display where it has estimated the location and edges of the pattern's main lobe, so that the operator can determine whether the pattern has been flagged due to an error from the main lobe estimation process.

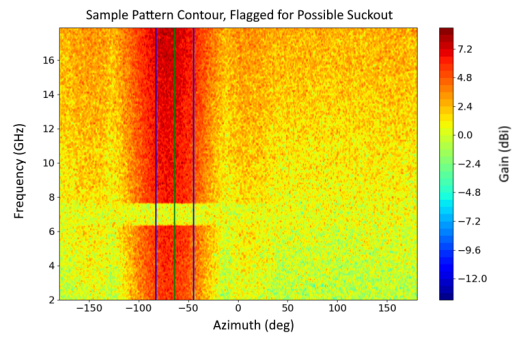

Fig. 3: Generated antenna pattern with a discontinuity in the main lobe.

III. Simulated Results and Optimization

The algorithm was developed and trained using Python scripts to generate sample data to simulate generic directive antennas. These scripts acted as the training data set for the data QA algorithm. Generating data sets trains QA algorithms efficiently because the data can be created alongside a truth table that already lists whether every piece of data is intended to pass or fail the QA test. Additionally, the generation script can be modified to change the pass or fail criterion, or how "good" a set of data has to be to pass the algorithm. The primary disadvantage is that there is no guarantee the algorithm training data represents a real world scenario, which is why the test team implemented the algorithm as a malleable prototype when first applied to a live test. All quantitative results in this paper were created from sampled data during the training process.

Data generated for this project was controlled by the following input parameters: the maximum and minimum signal to noise ratio (SNR) of the data set, the location (in azimuth, anywhere from -180 to 180 degrees) of the center of the main lobe, the width (across a range of azimuth values, along the x-axis) of the main lobe, whether or not the pattern has a discontinuity in the main lobe, and, if it does, the width of said discontinuity (across a range of frequency values, along the y-axis). Though not part of the data itself, a user-input threshold SNR tells the algorithm whether or not to flag each pattern as too noisy. A threshold SNR offers a simple, quantifiable basis for predicting if a set of pattern data suffers from unacceptable measurement inaccuracies [8]. Adjusting this value, in a sense, allowed the user to adjust the difficulty of the data set. The QA script could be tested either on its ability to only pick out nearly white noise patterns amongst a set of otherwise very clear data, or its ability to pick out slightly lower quality patterns

with more precision than an observing engineer. The main lobe discontinuity worked much more simply by comparison. Although the difficulty of finding a discontinuity could be tuned by reducing the width of any added discontinuities, whether or not to add one to each pattern is a binary option.

For an algorithm to use a set of its training data, it must have adjustable parameters that can be tuned as a user attempts to train it. The KDE-based algorithm offers a wide range of potential trainable parameters that could be explored beyond this paper, such as alternate choices for the kernel function or applying weighting to multiple estimates [9], but in this case only two parameters were made adjustable for the sake of simplicity. The two adjustable parameters were the KDE bandwidth h and the sensitivity e of the peak search function to extrema. The KDE bandwidth specifically refers to the width of the sample statistical distribution (in this case, a Gaussian function) that the estimator uses to build the PDF. Effectively the bandwidth sets the level of smoothing of the resulting function, and in the extreme cases of $h \to 0$ and $h \to \infty$ the result is, respectively, equal to the unaltered histogram or to the sample Gaussian function centered on the mean of the data set [4]. The extrema sensitivity specifically controls how many nearby values a point must be greater than in order to be considered a peak. Ideally, correct settings for h and e will produce a function that smooths out small noise effects but retains peaks for large scale behavior, then correctly identifies peaks without reporting small local maxima that survived the smoothing process. These parameters are closely related, since smoothing reduces the appearance of local maxima.

A set of 200 patterns was generated during training, with a range of SNRs between 14 and 19, and a threshold SNR of 16. Approximately 50% of the patterns were free of issues, 25% were generated without a clear main lobe, and 25% were generated with a frequency discontinuity that replaced the main lobe with low gain noise values. This replacement overwrote between 100MHz and 300MHz of the 2-18 GHz range of frequency values. When operated over this data set, with the optimized input parameters, the QA algorithm correctly flagged 89.3% of the flawed patterns and incorrectly flagged 9.1% of the correct patterns. The process was iterated 30 times over a set of input parameters that were predicted to contain the optimal values. The range of tested KDE bandwidths spanned 0.04 to 0.16, centered on the roughly 0.1 value calculated by Silverman's Rule [10]. The range of maxima search widths began at 1 (any value with lower neighboring values would be considered a maximum) and ended at 10. These attempts are plotted in Fig. 5 as a receiver operating characteristic (ROC) curve [11]. The discontinuities in this "curve" arise from creating the data by adjusting two separate parameters in discrete steps. The overall behavior illustrates that this QA method tends to achieve a high detection rate with most input values but must be tuned to avoid an unacceptably high false positive rate.

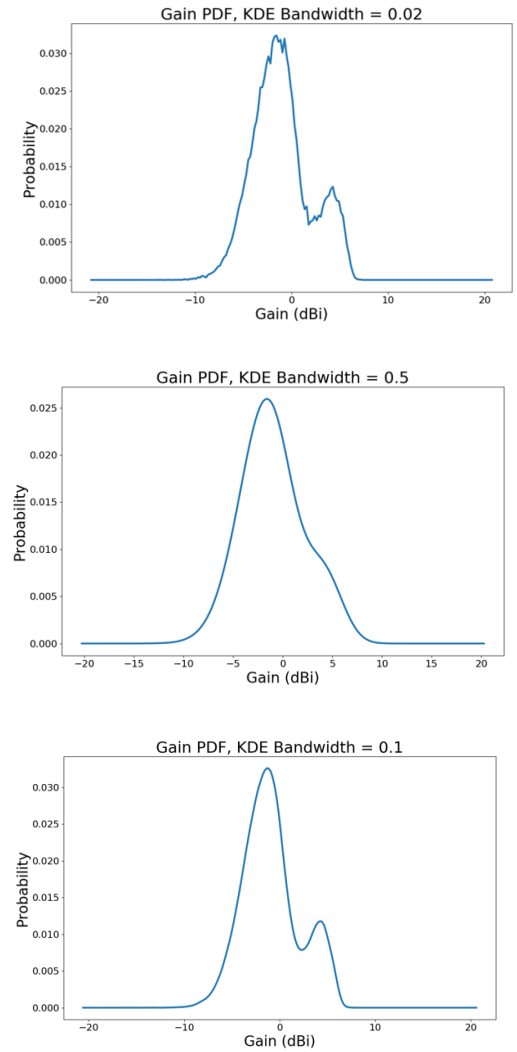

Fig. 4: PDFs of one sample pattern that should not be flagged by the QA algorithm, using multiple KDE bandwidths. Top: Below optimal bandwidth. Middle: Above optimal bandwidth. Bottom: Near optimal bandwidth.

For this relative training effort, the best pair of parameter values will land as close to the upper left corner as possible, minimizing the root sum square of the false positive rate and missed error rate. For the generated set of data this occurred when $h = 0.10$ and $e = 5$, although Fig. 6 suggests that within this range of parameters, algorithm performance minimally depended upon the selection of KDE bandwidth. These values do not represent the optimal values for every application of this technique, but rather a promising starting point before tuning the script based on incoming data.

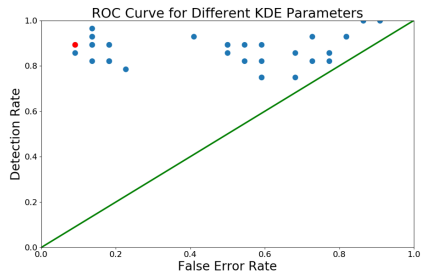

Fig. 5: ROC curve that represents the variation in algorithm performance in terms of ability to correctly flag flawed patterns (Detection Rate) and tendency to incorrectly flag other patterns (False Error Rate).

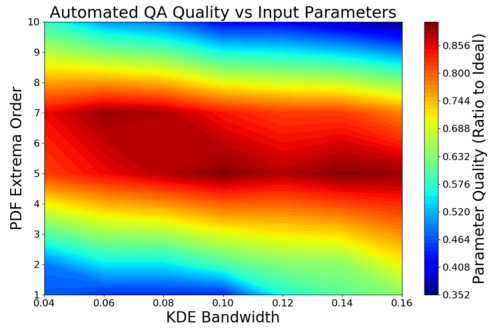

Fig. 6: Contour plot that represents algorithm performance (in terms of relative distance from the center line in Fig. 5 that represents random classification) specifically as a function of the KDE bandwidth and the peak search width.

IV. Application and Results

Following the development of this algorithm, the BAF team added these scripts to the data processing chain for the 2019 B-1B antenna pattern test. The operator would run every batch of approximately 100 patterns immediately after collection, then examine the flagged patterns. Engineers could then decide if the flagged patterns actually merited inspection, or if the sensitivity of the algorithm needed to be reduced. Additionally, engineers would manually review other random subsets of the data near the start of test execution and after significant changes to test conditions to verify that the algorithm did not need to be more sensitive.

Though the team designed the QA algorithm with the intent to automatically detect issues with the measurement setup, during test execution the algorithm flagged several patterns that included main lobe discontinuities that did not correspond to the predicted performance. Eventually, after the test team verified that the behavior was not due to a problem with the test equipment, the results were brought to the attention of the owners of the system under test. Subsequent analysis by both the test team and the system owners revealed that the algorithm had highlighted a true characteristic which only affected coverage in a small subset of the data. Without the addition of the QA algorithm, this characteristic would likely have gone unnoticed until the customer performed thorough analysis of the data set after the test. Since the discovery can now occur during test execution, the owners can implement a prototype change to the system and re-run the relevant test points, confirming a path forward to the desired performance - within the valuable test window at the BAF and not afterwards when it would require another return for retest.

V. Conclusion

In this paper a kernel density estimation based antenna pattern quality assurance algorithm has been developed, applied to generated training data and optimized, then applied to antenna pattern data during a test of the B-1B bomber. The algorithm assumes a baseline pattern with a distinct main lobe that does not exhibit significant discontinuities along its entire range of frequency values. After accepting an input KDE bandwidth and sensitivity to maxima, the algorithm approximates the distribution of all gain values in the pattern and determines whether a clear main lobe exists based on whether or not the distribution has only two peaks. Then, the pattern beam width is estimated and used to truncate the data set to only the main lobe, over which the algorithm repeats the KDE process and flags the data if more than one peak exists.

This algorithm achieved a 89.3% true detection rate with only a 9.1% false error rate when run using optimal parameters over an artificial set of pattern data. When improved based on these results and applied to live test data of the B-1B bomber, the algorithm successfully isolated critical but subtle valuable characteristics of the system under test well before they would have been uncovered by manual review. While the algorithm does perform better when applied to the generated data that was designed alongside it, successful application of the software has also revealed the potential capability of this technique to rapidly isolate, flag, and characterize a wide variety of unexpected behaviors in copious antenna pattern data.

References

[1] D. Gray, "How to choose an antenna range configuration", 24th Symposium of the Antenna Measurement Techniques Association, November 2002.

[2] E. F. Ali, "Electronic warfare testing at the Benefield Anechoic Facility", 1997 IEEE Autotestcon Proceedings. IEEE Systems Readiness Technology Conference. Systems Readiness Supporting Global Needs and Awareness in the 21st Century, Anaheim, CA, USA, 1997, pp. 232-243.

[3] C. A. Balanis, "Fundamental Parameters of Antennas" in *Antenna Theory*, 3/tsrd ed. Hoboken, New Jersey, USA: John Wiley & Sons, 2005,pp. 27-104.

[4] D.W. Scott, "Kernel Density Estimators" in *Multivariate Density Estimation: Theory, Practice, and Visualization*. New York, New York, USA: John Wiley & Sons, 1992, pp. 125-193.

[5] B.A. Turlach, "Bandwidth selection in kernel density estimation: a review", *CORE and Institut de Satistique*.

[6] P. Virtanen, R. Gommers, T.E. Oliphant, *et al*. "SciPy 1.0: fundamental algorithms for scientific computing in Python", *Nat Methods*, vol. 17, 2020, pp. 261-272.

[7] J.W. Moffat, G.A. Morin, M.E. Kelly, C.B. Brochu, "Development of standardized procedures for antenna measurement ranges", 20th Symposium of the Antenna Measurement Techniques Association, November 1998.

[8] A. Geise and R. Geise, "Error probabilities for amplitude and phase measurements derived from signal-to-noise ratio", 37th Symposium of the Antenna Measurement Techniques Association, Long Beach, California, October 2015.

[9] J. Kim, C.D. Scott, "Robust kernel density estimation", *Journal of Machine Learning Research*, vol. 13, pp. 2529-2565, September 2012.

[10] B. W. Silverman, "Density estimation for statistics and data analysis", Monographs on Statistics and Applied Probability, vol. 26, Chapman and Hall, London, 1986.

[11] T. Fawcett, "An introduction to ROC analysis", Pattern Recognition Letters, vol. 27, pp. 861-874, 2006.