Highly Portable Open Source Array & Phased Antenna Simulator

1st Merve Tascioglu Yalcinkaya
Barkhausen Institut
Dresden, Germany
merve.tascioglu@barkhauseninstitut.org

2nd Padmanava Sen
Barkhausen Institut
Dresden, Germany
padmanava.sen@barkhauseninstitut.org

3rd Mehrab Ramzan
Barkhausen Institut
Dresden, Germany
mehrab.ramzan@barkhauseninstitut.org

4th Gerhard P. Fettweiss
Barkhausen Institut
Dresden, Germany
gerhard.fettweiss@barkhauseninstitut.org

***Abstract*—Phased array antennas are key components in many applications such as radar, communication, 6G/5G networks, in-band full duplex (IBFD) operations, and future joint communication/sensing (JC&S) applications. These advanced applications require different performance characteristics in terms of gain, beamwidth, and scanning range. In this paper, a tool, which meets the needs of rapid observation of the array factor (AF) pattern by assigning the control variables such as inter-element spacing, geometrical array configuration, beam steering angle, and beam steering technique, is presented. The developed tool is further capable of capturing the gain pattern of multi-antenna structures and calculation of half power beamwidth (HPBW) with respect to the defined control variables. The performance of the tool is shown against measurements for state-of-the-art antenna structures.**

***Index Terms*—Phased array, Array factor (AF), Gain pattern**

I. Introduction and Simulator Objectives

The increasing proliferation of advanced devices and applications demand an antenna array which can handle huge data rates, provides high gain, wide scanning coverage, and stable radiation pattern. Therefore, antenna arrays play a vital role in today's wireless communications to enable the deployment of low-cost high-performance systems since they provide the best solution in terms of directional gain enhancement and allocation of power to beams. In order to shape the overall pattern of the antenna arrays, there are four control variables which have direct impact on key performance indicators (KPIs) of an antenna array system including but not limited to phased array systems as shown by Fig. 1. To the best of the authors' knowledge, the presented tool is the only such open-source simulator available with these functionalities. The geometrical configuration determines if the antenna array elements will be placed either in a linear or planar configuration. A linear array structure is a one-dimensional array that is only capable of scanning in a single plane. On the other hand, a planar array provides both elevation and azimuth resolution. The layout parameter determines whether the antenna array elements are placed uniformly or non-uniformly. Non-uniform array layouts, e.g. unequal spacing between adjacent elements, provide a viable option to potentially higher gain and reduced side lobe levels (SLLs). Moreover, non-uniform placement could be considered to obtain a given beamwidth with fewer elements than a uniformly spaced array [1]. The excitation amplitude of each array element is another significant parameter which can be uniform weighting or non-uniform weighting for each radiator. The SLLs can be considerably degraded by the 2nd option which can be implemented via different methods such as Dolph-Tschebyscheff, Taylor, Binomial, Blackman, Hamming, Gaussian and Kaiser-Bessel [2], [3], [4]. The relative phase of each element is also decisive for controlling the shape and direction of the array's radiation pattern towards a desired angular sector. Once an array is designed to focus towards a particular direction, the next step is to steer the beam towards other directions by controlling the progressive phase difference between the elements. Thus, in phased array systems, relative phases determines the beam direction. Phased arrays consisting of phase shifters have been the conventional way of steering beams electronically over years. However, the phase shifting systems are not suitable for some mmWave applications while using variable passive components that require amplifiers to compensate for the gain variation [5]. Besides this, in case of a wide bandwidth requirement, the use of phase shifters is challenging because of their frequency-dependent characteristics. This results in beam squint which implies the change in beam direction across the RF signal's frequency. It limits the applications at mm-wave frequencies demanding large bandwidths. As a result of the beam squint

This work is financed on the basis of the budget passed by the Saxon State Parliament.

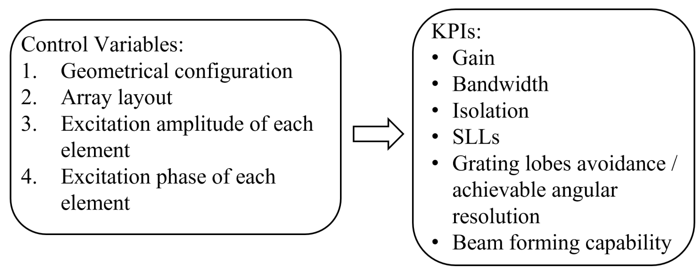


Fig. 1. Control variables and KPIs of an antenna array

effect, phase compensation based on center frequency is replaced by True Time Delay (TTD) circuits to compensate for the wave path difference in arrival and to obtain enhanced bandwidth. The familiar way to provide time delays is to insert incremental lengths of transmission line (delay lines) of length $L_n = d_{n-1} \cdot \sin\theta \; or \; L_n = d_{n-1} \cdot \cos\theta$ depending upon the axis of the antenna placement [6]. The developed simulator rapidly integrates both phase shifting and TTD methods.

In this work, considering all control variables, a Python-based phased array antenna pattern simulator aligned with the theory of antenna arrays has been developed. The user of the simulator can swiftly observe the desired array characteristics as a prior investigation to time-consuming electromagnetic (EM) simulations by changing essential parameters (e.g. steering angle, number of elements, geometrical configuration, array layout, beam steering technique) which have significant impact on the results. There are limited number of publicly available MATLAB based tools with a Graphical User Interface (GUI) [7], [8] for array antenna simulations. However, none of them covers all essential input parameters in a single GUI and do not provide a comparison of the results of multiple antenna arrays on the same plot. The developed tool gives the user a setup to analyze the phased array performance for various scenarios and applications not only for uniform but also for non-uniform symmetric/asymmetric element placements. The tool includes a user-friendly GUI that updates the simulation results (plots) as the user changes input parameters. Thus, it becomes a very interactive tool, supporting user to obtain valuable insight about the topic of interest. Full access to scripts from the GitHub repository [9] gives the user the ability to customize and extend the tool for a particular application. Furthermore, a Python library [10] named 'arrayfactor' has also been created in parallel with the simulator to calculate normalized array factor numerically with the range of -180° to +180° for desired arrays without the GUI and can be included as part of different simulations.

The paper is organized into six sections. Section II will describe the theory behind the simulator. Section III will give a brief overview of the simulator and the GUI. A validation of the results obtained by the simulator and and the comparison of 3D EM simulations and measurements is demonstrated in Section IV. Limitations and related discussions is described in section V. Section VI provides the conclusion of this work.

II. Theory Behind the Simulator

There are two key factors utilized to represent the total radiation pattern of an antenna array in the far-field as $E(\theta,\phi)$: Element Factor $EF(\theta,\phi)$ and Array Factor$AF(\theta,\phi)$. Element factor which depends on the physical dimensions and electromagnetic characteristics represents the radiation pattern of an individual element in the array. Array factor which depends on the amplitude, phase and position of the elements, represents the radiation pattern of an array composed of identical elements. The functional relationship of these factors is given by equation 1 [11].

$$G(\theta,\phi) = EF(\theta,\phi) + AF(\theta,\phi) \quad \text{in dB} \tag{1}$$

where $G(\theta,\phi)$: Array's radiation pattern, $AF(\theta,\phi)$: Array Factor, $EF(\theta,\phi)$: Element Factor

In order to synthesize the overall array pattern, the major focus should be on the AF which significantly determines the effect of combining radiating elements. In general, array factor can be written as [12]:

$$AF = \sum_{n=0}^{N-1} I_n e^{j\psi_n} \tag{2}$$

$$x-axis : \psi_n = kd_n \cos\phi \sin\theta + \beta_n, \tag{3}$$

$$y-axis : \psi_n = kd_n \sin\phi \sin\theta + \beta_n, \tag{4}$$

$$z-axis : \psi_n = kd_n \cos\theta + \beta_n \tag{5}$$

where n: n^{th} element from the reference, I_n: Excitation of the radiator, N: Total number of elements, k:wave number, d_n:inter-element spacing ψ: Phase function, the real part of the exponential function that varies depending upon the axis of the element placement. Phase function is the most crucial parameter while creating the phased array system which generates radiation pattern and provides the flexibility to steer transmitted signal to desired angle electronically. According the mathematical expression, phase delay, β_n , inserted into the simulator. The beamsteering with respect to the phase shifting and TTD was applied as given by the following equations:

$$\beta_n = -k_0 d_n u \quad \text{For phase shifting} \tag{6}$$

$$\beta_n = -k d_n u \quad \text{For TTD} \tag{7}$$

where u may be $\cos\phi_0, \sin\theta_0, \sin\phi_0$ depending on the axis and the plane, $k_0 = 2\pi/\lambda_0, k = 2\pi/\lambda$, λ: The wavelength where TTD is based on, λ_0: The wavelength the phase shift is based on.

The design principles and the derivation of array factor for planar arrays are similar to linear ones and thus, calculated as the multiplication of the array factors of linear arrays for the relevant axis. In this work, uniform weighting was considered and the normalized array factor was obtained in logarithmic scale. Besides the ability of the normalized array factor pattern simulations, the developed simulator furthermore enables to plot the gain pattern of desired phased array antenna. Accordingly, the gain pattern of an array in the simulation was created by considering the single antenna gain and the directivity factor (DF) which can be associated with the square of the array factor [1], [13]. Furthermore, in the case of uniform weighting, the peak directivity of a broadside linear array can be expressed in terms of half power beamwidth (HPBW) [3] as given by the equation 8 to simplify the calculation in particular for non-uniformly spaced array elements. It should be noted that the directivity of an end-fire array is two times higher than a broadside array [2]. Accordingly, a relation between the end-fire array's directivity and HPBW was derived as expressed via equation 9.

$$D_{broadside\ array} = \frac{101.5}{HPBW[°]} \tag{8}$$

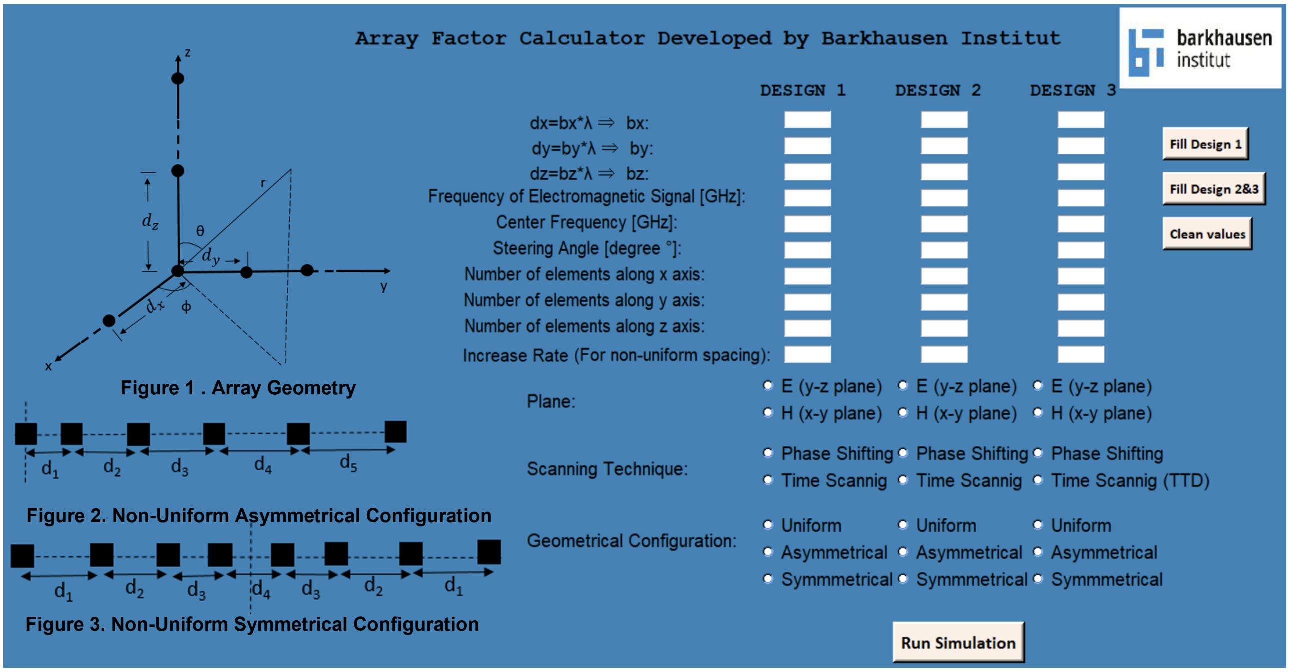


Fig. 2. GUI of Python-based Simulator's input window

$$D_{endfire-array} = \frac{46535}{(HPBW)^2[°]} \quad (9)$$

Stutzman [13] already illustrated the alteration of the directivity as a function of uniform element spacing with uniform excitation for a linear broadside array with respect to various element numbers. It is explicitly seen that directivity is exposed to a sharp decline in case the spacing exceeds one wavelength. When the inter-element spacing is adequately large compared to the wavelength, the in-phase addition of the radiated field occurs in more than one direction. This causes multiple maxima, known as grating lobes, of the far-field radiation pattern of the array which results in degraded array performance. Therefore, developed Python-based simulator performs gain pattern simulations for the spacing values smaller than one wavelength which is the most preferred case in the practical applications. Furthermore, non-uniformly spaced arrays with uniform current excitation are also applicable in this simulator. Accordingly, two methods used for the implementation of non-uniformly spaced arrays: Non-uniformly asymmetrical and non-uniformly symmetrical placed arrays as illustrated in Fig 3. The relation of each element distance for non-uniform configurations are associated with the 'increase rate' which is mathematically expressed as the difference between the positions of two successive array elements:

$$increase\, rate = d_n - d_{(n-1)} \quad (10)$$

Apparently, this simulator enables to analysis non-uniform spaced array which is significant for SLL reduction.

III. Overview of the Simulator

To help users' comprehension of the concepts behind the theory as summarized in the previous section, a Python-based simulator was developed with a user-friendly GUI as presented in Fig 2. The scripts were developed by Spyder 4.1 which is an open-source integrated development environment (IDE). The tool was written in Python 3.6 which eases to define the underlying inputs and presenting the graphical outputs in terms of either 2D cartesian or polar plots as shown in Fig 4. It also provides comparison of different arrangements up to 3 designs on the same plot.

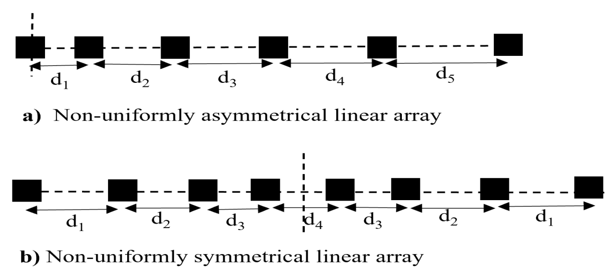


Fig. 3. Non-uniformly geometrical configurations

IV. Validation and Comparison

The success of the simulator has been already tested within the scope of the master thesis conducted at Barkhausen Institut (BI) [12] that covers the comparison of the results obtained via developed simulator, a 3D EM tool (HFSS) as well as measurements. Moreover, all results obtained by the tool are also consistent with the results already reported in the literature [6], [11], [14]–[19]. Within the scope of this work, uniformly and non-uniformly spaced various linear and planar arrays consisting of rectangular patch antennas with the element numbers of 2,4,8,16 were designed in X-band (at 10 GHz) and simulated with uniform current excitation. Accordingly, non-uniformly symmetrical 4x1 linear and 4x2 planar arrays were fabricated as shown by Fig 5 and the simulation results were

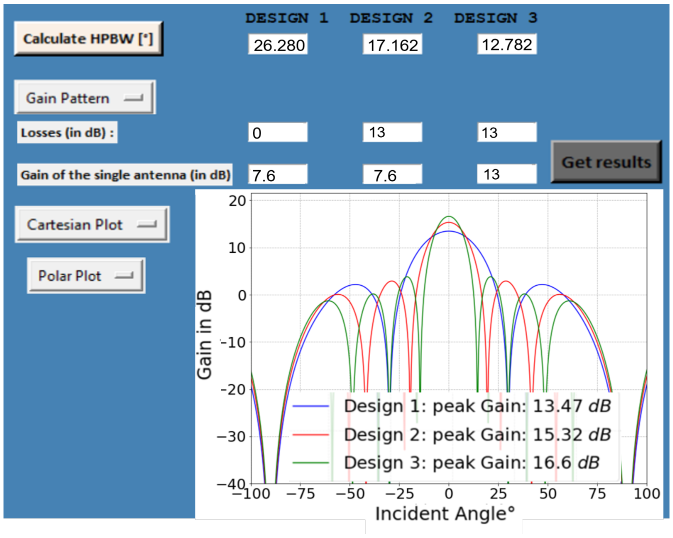


Fig. 4. GUI of Python-based Simulator's output window

validated by measurements. The non-uniform symmetrical placement implies that the inter-element spacing between the antennas located on the left- and right-hand sides of the array centre constantly increases by a value associated with the wavelength while placing additional elements as described in [12]. In this work, the inter-element spacing of linear array was chosen as half wavelength in the centre of the array and increased by 0.1λ for other elements. For planar array, the same strategy were applied for the elements along the y- axis and the spacing of the elements along x- axis was chosen 0.58λ to avoid superposition of patches and the feed lines.

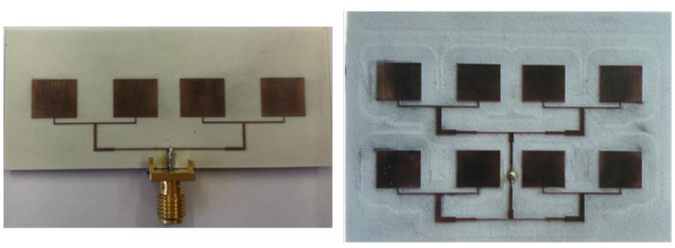
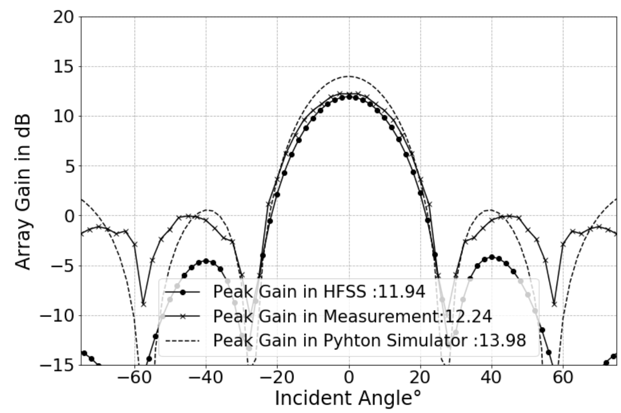
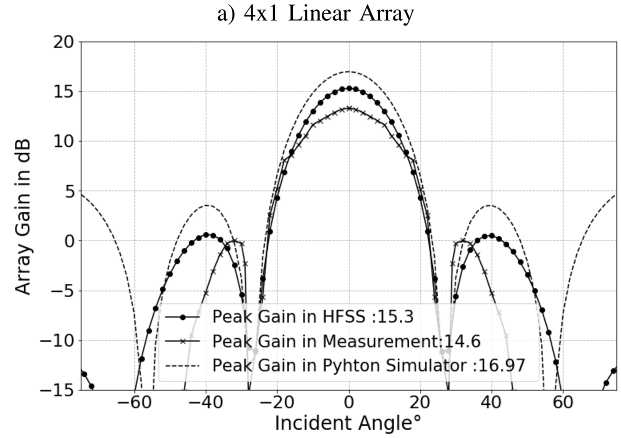


Fig. 5. Fabricated antenna arrays a) 4x1 Linear b) 4x2 Planar

Fig 6 presents the comparison of the simulation and measurement results. The losses (e.g., radiation loss, impedance mismatch loss, dielectric loss, conducting loss, and mutual coupling between adjacent elements) were ignored and the gain of the single antenna (7.6 dB) is added into Python-based simulator while actualizing the comparison. However, a user can manually add both the potential losses and the single antenna's gain value in the simulator because of its flexibility feature. According to this comparison, it can be deduced that Python-based simulator provides a useful outcome in terms of obtaining the expected behaviour of array pattern including exact HPBW rapidly considering relevant factors which are array layout, geometrical configuration, beamforming techniques, steering angle and inter-element spacing. Furthermore, the SLL reduction with non-uniform asymmetrical and symmetrical linear & planar array configurations has been proven as shown in the example of Fig 7. Beside this, another outstanding advantage of non-uniform symmetrical array can also be observed by the simulator as illustrated in Fig 8. Accordingly, non-uniform symmetrical array can be utilized to get comparable beamwidth with fewer element compared to the uniform array.



a) 4x1 Linear Array



b) 4x2 Planar Array

Fig. 6. Comparison of Gain Patterns

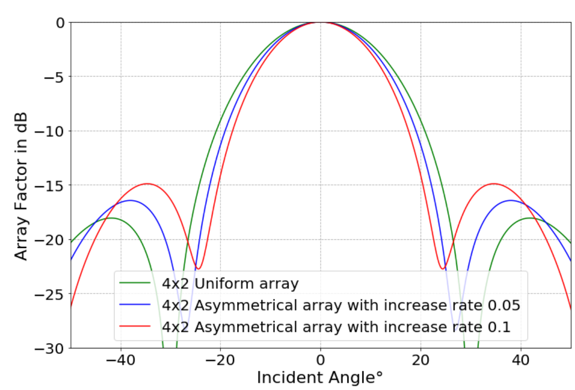


Fig. 7. Normalized Array factor patterns of 4x2 Uniform and Non-uniform Asymmetrical planar arrays

V. LIMITATIONS AND DISCUSSIONS

There are slight differences between the classical approach, which is based on antenna array theory, and Floquet Model

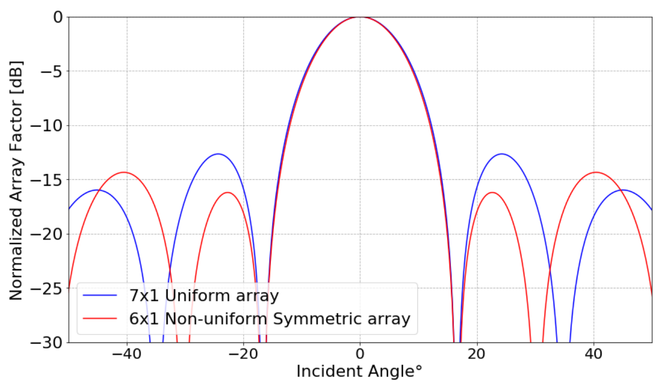


Fig. 8. Different Array Layouts

approach, which is the basis of HFSS. The aforementioned losses considered in HFSS depends upon multiple influencing factors such as feed mechanism, materials of substrate and conductor etc. At the same time, calculations based on the theory requires a complicated process, long simulator time even with parallel processing, and huge user efforts. Therefore, it is recommended to use a loss correction factor which can be approximately assumed as 1.5-2 dB with respect to previous studies on antenna arrays [17]–[24]. Additionally, it should be noted that the gain of the single antenna element, or element factor, must be written manually as an input on the simulator to obtain the relevant gain pattern since it depends heavily upon the antenna type (e.g., bowtie, dipole) and its dimensions, material and thickness of the substrate, and the feeding technique. The slight differences between the measurement and simulations could be explained by the reasons such as environment effect, insertion loss of connectors as well as the fabrication tolerance.

VI. Conclusions

This paper presents an extensive study on the design of antenna arrays taking significant parameters such as array architectures and element placement techniques into account. Accordingly, a fast and user-friendly simulator has been developed to provide the potential array pattern observations as an initial impression prior to time-consuming computational EM simulations. Furthermore, the performance of phased array antennas can also be examined by this tool which is capable of presenting how the array pattern alters with respect to common beam-steering techniques such as phase shifting and TTD. Additionally, the aforementioned non-uniform weighting methods could be implemented within the developed simulator. This attractive solution for future work would considerably facilitate the users to suppress the SLLs and thus, to achieve better antenna array performances.

References

[1] M. I. Skolnik, *Introduction Radar Systems,*, 2nd ed. McGraw-Hill,Chapter 8., 1981.
[2] C. A. Balanis, *Antenna Theory*, 3rd ed. John Wiley&Sons, Chapter 6., 2005.
[3] R. Elliot, *Antenna Theory and Design*, revised ed. ed. John Wiley&Sons, Chapter 5., 2003.
[4] F.B.Gross, *Smart Antennas with MATLAB*, 2nd ed. McGraw-Hill Education,Chapter 4., 2015.
[5] S. Ghosh and D. Sen, "An inclusive survey on array antenna design for millimeter-wave communications," *in IEEE Access*, vol. 7, pp. 83 137–83 161, 2019.
[6] M.Longbrake, "True time-delay beamsteering for radar," pp. 246–249, 2012.
[7] "Array factor," 2021, last accessed December 27 2021. [Online]. Available: https://www.mathworks.com/matlabcentral/fileexchange/38332-array-factor
[8] A.Harrison, "Introduction to radar using python and matlab," 2019. [Online]. Available: https://github.com/RadarBook/software
[9] M.Tascioglu, "Antenna array simulator," 2021. [Online]. Available: https://github.com/Barkhausen-Institut/AntennaArraySimulator
[10] ——, "Antenna array simulator," 2021. [Online]. Available: https://pypi.org/project/arrayfactor/
[11] J. K. P. Delos and B.Broughton, "Phased-array antenna patterns (part 1)—linear-array beam characteristics and array factor," 2020.
[12] M.Tascioglu, "Theoretical analysis,simulation and development of antenna arrays for mm-wave systems," 2021. [Online]. Available: https://www.barkhauseninstitut.org/en/results/publications
[13] W.L.Stutzman and G. Thiele, *Antenna Theory and Design*, revised ed. John Wiley&Sons, Chapter 1., 1981.
[14] J. P. Delos and B.Broughton, "Phased array antenna patterns—part 2: Grating lobes and beam squint," 2020.
[15] M. O. Manu and A.Graur, "Analysis of beamforming in phased antenna arrays," 2010.
[16] M. T. A. M. N. M. Tan, S. K. A. Rahim and T. A. Rahman, "Smart antenna:weight calculation and side-lobe reduction by unequal spacing technique," *2008 IEEE International RF and Microwave Conference*, 2008.
[17] B. A. S. A. R. Nair and S. S. Thakur, "Design of rectangular microstrip 4x2 patch array antenna at 2.4 ghz for wlan application," *Second International Conference on Advances in Computing and Communication Engineering*, 2015.
[18] Y. E. Hasnaoui and T. Mazri, "Study, design and simulation of an array antenna for base station 5g," *International Conference on Intelligent Systems and Computer Vision (ISCV)*, 2020.
[19] N. T. S. Singh and N. Sinha, "Design and analysis of single patch, 2×1 and 4×1 microstrip antenna arrays," *International Conference for Convergence for Technology*, 2014.
[20] W. N. W. M. N. Ab Wahab, Z. Bin Maslan and N. Hamzah, "Microstrip rectangular 4x1 patch array antenna at 2.5ghz for wimax application," *2nd International Conference on Computational Intelligence*, 2010.
[21] H. M. M. Makkawi and H. A. Ali, "Design and performance analysis of 1×8 patch antenna array for x-band applications," *International Conference on Computing, Control, Networking, Electronics and Embedded Systems Engineering (ICCNEEE)*, 2015.
[22] A. M. T. Reynalda and E. Bharata, "Characterization of 4×4 high gain microstrip array antenna for 3.3ghz wimax application," *6th International Conference on Telecommunication Systems, Services, and Applications (TSSA)*, 2011.
[23] A. A. S.H. Kinani, K. Mahmood, "A linear array for short range radio location and application systems," *International Journal of Advanced Computer Science and Applications*, vol. 9, no. 4, 2018.
[24] V. V. Y. D.A.Naik, K.P. Padiyar, "Design analysis of microstrip patch antenna comparison between the arrays," *International Journal for Innovative Research in Science Technology*, vol. 4, no. 12, 2017.