IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 16 June 2023, accepted 3 July 2023, date of publication 6 July 2023, date of current version 12 July 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3292961

APPLIED RESEARCH

Global Complex Roots and Poles Finding Algorithm in $\mathbb{C} \times \mathbb{R}$ Domain

SEBASTIAN DZIEDZIEWICZ, MALGORZATA WARECKA, (Student Member, IEEE), RAFAL LECH, (Senior Member, IEEE), AND PIOTR KOWALCZYK, (Member, IEEE)
Faculty of Electronics, Telecommunications and Informatics, Gdańsk University of Technology, 80-233 Gdańsk, Poland
Corresponding author: Rafal Lech (rafal.lech@pg.edu.pl)

This work was supported in part by the Gdańsk University of Technology under the Ministry Subsidy for Research.

ABSTRACT An algorithm to find the roots and poles of a complex function depending on two arguments (one complex and one real) is proposed. Such problems are common in many fields of science for instance in electromagnetism, acoustics, stability analyses, spectroscopy, optics, and elementary particle physics. The proposed technique belongs to the class of global algorithms, gives a full picture of solutions in a fixed region $\Omega \subset \mathbb{C} \times \mathbb{R}$ and can be very useful for preliminary analysis of the problem. The roots and poles are represented as curves in this domain. It is an efficient alternative not only to the complex plane zero search algorithms (which require multiple calls for different values of an additional real parameter) but also to tracking algorithms. The developed technique is based on the generalized Cauchy Argument Principle and Delaunay triangulation in three-dimensional space. The usefulness and effectiveness of the method are demonstrated on several examples concerning the analysis of guides (Anti-Resonant Reflecting Acoustic Waveguide, coaxially loaded cylindrical waveguide, graphene transmission line) and a resonant structure (Fabry-Pérot open resonator).

INDEX TERMS Complex zeros and poles, numerical computations, root finding algorithm, roots of complex equation, resonators, waveguide structures.

I. INTRODUCTION

The necessity of finding roots for complex functions emerges in many fields of science. It is a common problem in electromagnetism, however, it can be also found in acoustics [1], stability analyses [2], spectroscopy [3], optics [4], [5] and even in elementary particle physics [6] and solving the nonlinear Schrodinger equation [7]. Roots (zeros), depending on the problem, can represent different parameters such as propagation coefficients [8], [9], [10], [11], [12], [13] or resonant frequencies [14], [15], [16], [17] for acoustic and electromagnetic phenomena, or energy in quantum physics [18].

There are numerous algorithms for finding roots in the complex domain, both local and global. Local methods, such as Miller's [19] and Newton's [20], are characterized by their efficiency, ability to find zeros in a few iterations, and numerically undemanding nature. Unfortunately, preliminary knowledge of the root location and the neighboring zero distribution is necessary, which is an undeniable drawback. Alternatively, global methods are able to find all roots inside the fixed domain. Those of them that are based on sampling a function only at the boundary of the domain can also be efficient if they are applied for polynomial or rational functions, for example the ratdisk [21] and AAA (adaptive Antoulas–Anderson) [22] algorithms. Nevertheless, some problems may emerge for more complex functions and the algorithms can generate false solutions and omit some roots, and their accuracy cannot be controlled [23], [24]. Those global algorithms which discretize the entire domain are more reliable and their accuracy can be controlled, however, they are more expensive numerically [23], [25], [26], [27], [28], [29], [30].

The other important aspect is how the position of the root in the complex domain changes in the function of the additional variable ($t \in \mathbb{R}$). In electromagnetic propagation problems, an extra parameter often represents frequency – dispersive characteristics, for instance. On the other hand, in resonant

The associate editor coordinating the review of this manuscript and approving it for publication was Guido Lombardi.

structures, it can be one of the geometric dimensions or a variable describing some material properties. The problem then boils down to the formula.

$$F(z, t) = 0 \tag{1}$$

where the solution $z = \xi(t)$ is a curve in a $\mathbb{C} \times \mathbb{R}$ space.

One possibility is to solve such a problem by using the slicing method, which finds the zeros in the complex domain [23] for every discretized value of the extra parameter $t = \text{const}$. Depending on the complexity of the function, the discretization may be equidistant, or some densification with respect to t may be required. The observed area may contain numerous $\xi(t)$ curves that may fork, intersect, or approach each other, and their curvatures can be arbitrary. Slicing is particularly inefficient when $\xi(t)$ is parallel to the plane $t = \text{const}$. Therefore, the selection of the discretization distribution can be problematic to automate, resulting in the difficulty of creating a general algorithm. In addition, since the procedure is iterative, performing the same operations many times is not only tedious and time-consuming for the user but also numerically expensive. As was shown in [23] and [24], no global algorithm is guaranteed to find all zeros/poles within a few function calls. Sometimes, some of them can be omitted and then may or may not appear in the next slice. In general, it is difficult to make a preliminary estimation of the number of points, and even then, the possibility of omission cannot be ruled out.

To increase the efficiency of the analysis, a tracing algorithm can be utilized [31], [32], [33]. Nevertheless, it requires a precise location of the starting points and traces only one curve per recall. Moreover, if the curve forks or curves intersect, the algorithm following one curve skips some solutions or may even start tracking an incorrect one [34]. If curves are near each other, the algorithm may also start tracking another solution. The risk can be minimized by reducing the resolution step, however, this leads to an increase in the numerical cost.

Even though there are many algorithms dedicated to problem (1) and numerous improvements have been introduced to them [31], [32], [33], [34], no direct global algorithm for root-finding in $\mathbb{C}\times\mathbb{R}$ space has been provided in the literature. In this article, we propose a new algorithm, which constitutes a generalization of the global root and poles finding algorithm (GRPF) [23] to the $\mathbb{C} \times \mathbb{R}$ space.

The aim of this generalization is to obtain an algorithm that creates a full picture of the distribution of roots for function (1) in the entire analyzed domain $\mathbb{C} \times \mathbb{R}$ and avoids the aforementioned problems arising while using tracing or slicing methods. The proposed tool is intended for preliminary analysis in order to determine the number of curves $\xi(t)$ and their complexity. In contrast to the tracing technique, a single call of the proposed algorithm obtains all curves that represent roots/poles in the analyzed area, which significantly reduces the user's involvement and attention. Moreover, no *a priori* knowledge about the distribution of these curves is needed, neither about their number, starting or ending point, nor the distance between them. Therefore, the user does not have to select a resolution step or perform an initial analysis. In contrast to the slicing approach, the level of parameter t discretization does not have to be controlled, which reduces the user's workload and implies higher efficiency. Having solutions $\xi(t)$ parallel to $t = \text{const}$ are no longer a problem. Moreover, the proposed method is more reliable compared to the slicing method, where discontinuity of the domain with respect to the variable t carries the risk of omitting one of the solutions in a given region.

The proposed algorithm is based on the generalized Cauchy Argument Principle (CAP) and discretization of the entire search area $\Omega \subset \mathbb{C} \times \mathbb{R}$ using Delaunay triangulation. The process is iterative and, in subsequent steps, the mesh is refined around the curves representing the zeros/poles. It should be emphasized at this point that the use of the proposed algorithm is intended to give a rough picture of the zero/pole distribution (without any distinction between zeros and poles), in the analyzed area Ω with the accuracy determined by the assumed number of function calls. Once this picture is obtained, other algorithms may be utilized to improve the image in the desired regions with a low numerical cost.

II. FORMULATION OF THE PROBLEM

The proposed method is based on the generalization of CAP to the $\mathbb{C} \times \mathbb{R}$ space, which was used to some extent in tracing roots [32], [33], [34].

Let us start with a one-variable complex (holomorphic) function $f(z)$ and a closed curve C defined on a complex plane. According to standard CAP [35], the integral

$$q = \frac{1}{2\pi i} \oint_C \frac{f'(z)}{f(z)} dz \tag{2}$$

is equal to a sum of all zeros (counted with their multiplicities), minus the sum of all poles (counted with their multiplicities) located inside the region determined by the close contour C. If, for instance, there is only one single root in the region, then $q = 1$ or if there is one single pole in the region, then $q = -1$. In fact, the integral (2) is exactly equal to the total increase in the argument of $f(z)$ over contour C divided by 2π.

If the contour C is composed of P straight line segments based on points $\{z_1, z_2, \ldots, z_P\}$, then CAP can be rewritten in the following discretized form [36], [37]:

$$q = \frac{1}{2\pi} \sum_{p=1}^{P} \Delta\Theta_p \tag{3}$$

where $\Delta\Theta_p = \arg f(z_{p+1}) - \arg f(z_p)$ and $z_{P+1} = z_1$. It must be emphasized that the expressions (2) and (3) are equivalent only if the following discretization condition is met: $|\Delta\Theta_p| < \pi$ for all segments $p = 1, 2, \ldots, P$.

As was shown in [23] and [33], if the considered domain is covered with a sufficiently dense mesh, the detection of a root or pole in this domain does not require the calculation

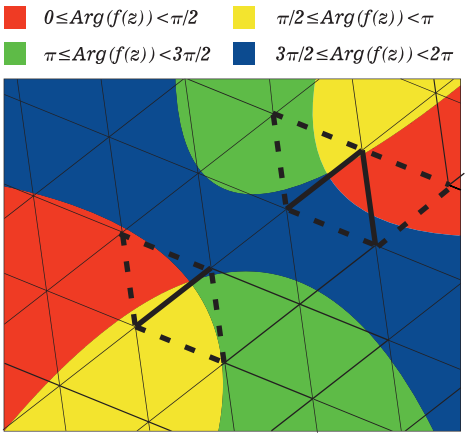

FIGURE 1. Phase portrait of a complex function with a triangular mesh. The single root and the single pole are depicted in the region.

of q – see Fig. 1. The root/pole is then located next to the segment (edge of the mesh), which connects two regions in which the argument of the function differs by two quadrants. Such segments are called candidate edges and are denoted by thick solid lines in Fig. 1 (quadrants of the function argument are denoted by colors: red, yellow, green, and blue). The integral (2) can be evaluated only to verify the existence of the root inside the curve C. It can be obtained using (3) with the segments denoted by thick dashed lines.

The idea of using CAP in the complex plane described above can be easily generalized to the space $\mathbb{C} \times \mathbb{R}$. It leads to searching for roots of the function (1) as curves $z = \xi(t)$ and it was successfully applied in [32] and [33]. In the proposed approach, segments connect nodes in the space $\mathbb{C} \times \mathbb{R}$. Nevertheless, determining of the argument increment along the segments is still possible and (3) can be applied in exactly the same form. However, discretized CAP is not utilized directly in the proposed algorithm. The presented approach focuses on the property regarding the location of the roots/poles in the vicinity of the candidate edges. Additionally, the condition dealing with the candidate edges was generalized; instead of assuming the argument increase along the candidate edge as two quadrants (as in [23]), we suggest that the edge should be considered as a candidate for phase difference greater than $\pi/2$. Therefore, in the proposed approach, the edge is a candidate edge if it fulfills the condition:

$$|\Delta\Theta_k| > \pi/2 \tag{4}$$

This generalization includes the cases when the argument changes by two quadrants [23] but also expands the condition to the cases when the phase is arbitrarily shifted. For instance, if the phase changes from 89° to 209° it produces change by two quadrants, however, if the phase changes from 91° to 211° it produces change by one quadrant, despite in both cases $\Delta\Theta_k = 3\pi/4$ (120°). The proposed generalization does not impose discretization of the function argument with respect to angles $0, \pi/2, \pi, 3\pi/2$ because there is no reason to prefer just such a discretization in relation to another arbitrarily shifted one.

III. ALGORITHM DESCRIPTION

In this article, we propose to utilize and generalize the technique applied in GRPF [23], which is based on the discretization of a region in $\mathbb{C}$ and iterative densification of the mesh with the close proximity of roots/poles (around the candidate edges). The same concept can be used in $\mathbb{C} \times \mathbb{R}$ space to determine the candidate edges. An obtained set of the candidate edges forms searched curves representing zeros/poles of function (1).

A. INPUT PARAMETERS

First, the user has to define the boundaries of $\Omega \subset \mathbb{C} \times \mathbb{R}$ to specify the searched region. The Ω can be of arbitrary geometry and in the simplest scenario, the range of each variable can be set independently ($Re(z)$, $Im(z)$, t) creating a cuboid in three-dimensional space.

As the second input parameter, the maximum number of function calls N_{max} has to be defined. This number can vary depending on the level of function complexity and numerical effort related to its evaluation. As a result, the running time of the algorithm can be estimated in advance. This number undeniably determines the quality of the analysis and its accuracy, however, it can be increased in later stages of the algorithm if necessary. The user has insight into these stages and can decide during the process to increase the number of function calls in order to improve the quality of the results.

B. OPERATION OF THE ALGORITHM

In the first stage of the algorithm, a uniform initial discretization of region Ω is carried out. It requires the utilization of a three-dimensional mesher ($Re(z)$, $Im(z)$, t), and the obtained mesh can be built of node-based tetrahedrons $(z_i, t_i) \in \Omega$. A Delaunay Triangulation is used for this purpose for the domain $\mathbb{R}^3$. The number of nodes of the initial mesh M depends on the previously defined number N_{max} and it is usually one percent of this number. The other ninety-nine percent of the nodes are used in further stages of the algorithm. In special cases, the user can change the relation between the initial M and N_{max}, e.g. when N_{max} is relatively low, M should be increased.

In the next step, function values in the nodes of the initial mesh are calculated $(z_m, t_m) \in \Omega$

$$F_m = F(z_m, t_m) \tag{5}$$

Then, for each edge, the argument change of the considered function along this edge is determined solely based on the knowledge of the phases in the nodes (linear approximation).

$$\Delta\Theta_{i,j} = arg(F_j) - arg(F_i) \tag{6}$$

Analogous to the procedure described in [23] and according to the new condition (4), the edges along which the argument changes by more than $\pi/2$ are distinguished and

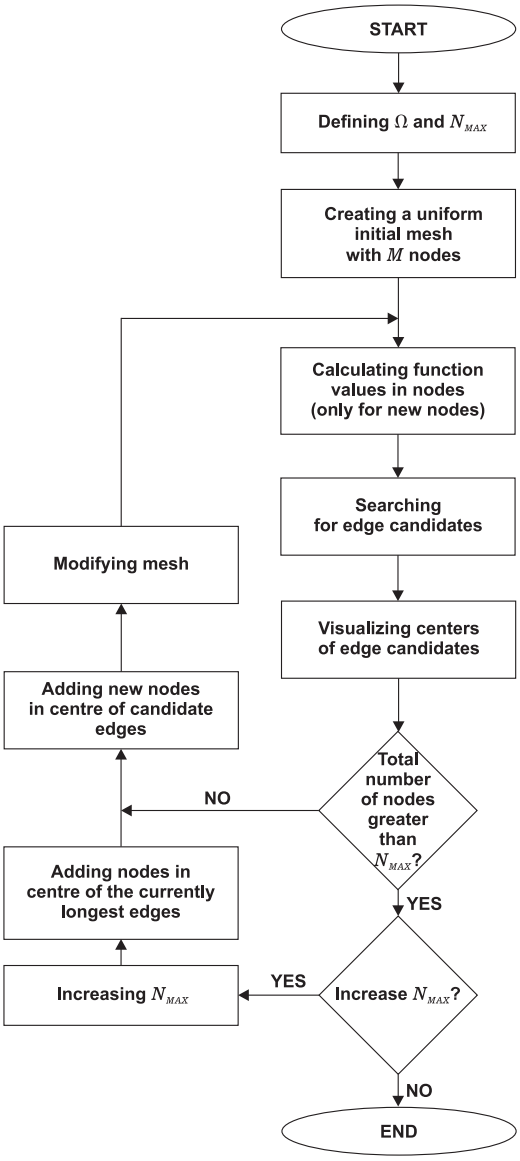

FIGURE 2. Block diagram of the algorithm.

included in the set of candidate edges. It is in the immediate vicinity of these edges where the zeros or poles of the function are located.

All candidate edges are half-split by adding additional nodes at their centers. Adding new nodes requires mesh modification and subsequent function calls. In this way, in successive iterations, the mesh becomes denser around the curves representing the zeros or poles of the function (1). The procedure ends when the number of function calls reaches N_{max}. The user may then decide to increase the number of calls to improve the quality and accuracy of the obtained image. The whole process is also shown in Fig. 2.

Finally, the $\xi(t)$ curves representing zeros/poles (without initial distinction) and constituting a solution to the problem are constructed from the candidate edges (or, more precisely, from their centers). The distribution of the candidate edges and their lengths prove the quality and accuracy of the results (the shorter the edges and the more clustered the distribution, the more accurate the results). The obtained results provide a general, overall picture and form a good basis for a possible detailed analysis for which local algorithms such as Newton or Muller can be used. Local techniques allow one to achieve arbitrarily high accuracy with just a few function calls and can be used in any cross-section (not necessarily $t = \text{const}$).

C. LIMITATIONS

The algorithm may seem numerically expensive at first glance, but the iterative process and triangulation are not particularly demanding in terms of processor speed and memory resources. The computational complexity of the examined function is of the greatest importance. Nevertheless, the size of the area Ω, and the number and complexity of the curves in this area are also significant, knowing that they directly affect the number of calls. As in the case of classical global algorithms that operate only on the complex plane, there is no guarantee of finding all zeros and poles of an arbitrary complex function with a finite number of calls. However, searching in the continuous domain over the variable t reduces the risk of omitting one of the solutions, as the process is more likely to converge for a slightly lower or higher value of t. An increase in N_{max} also reduces the risk of omitting one of the solutions while improving the quality and accuracy of the results obtained. Moreover, it largely eliminates noise, i.e. edges that do meet the condition (4) but are not zeros/poles (usually due to overly long edges). Thus, the main limitation of this method is the limit in the number of calls – for N_{max} increasing to infinity, all zeros/poles curves must be found.

The other aspect worth mentioning is an application of the algorithm to the multiple-valued functions. This can be done in two ways, either by analyzing each branch separately, or by using the pointwise product function [38]. The analysis of a single branch may be less effective if we ultimately study all the branches. Moreover, the algorithm may recognize a discontinuity along the cut (if the change in the function phase is greater than $\pi/2$) and generate candidate edges in the same way as zeros or poles. However, such an operation can be numerically expensive (the cut is represented by a surface in the $\mathbb{C} \times \mathbb{R}$ domain, so involves a huge number of candidate edges). Therefore, it is better to avoid a single branch analysis and utilize the pointwise product function [38].

Moreover, as mentioned before, the proposed method does not distinguish between zero and pole curves in the obtained solution. However, if needed, it can be done in a post-processing stage for any point on $\xi(t)$ with the use of discretized CAP (3) and neighboring nodes for which function values have already been calculated. Only the verification of the essential singularities is problematic because the use of (3) leads to a finite value of the parameter q.

IV. NUMERICAL RESULTS

Four different example structures were analyzed, and the results are presented in this section. The examples include

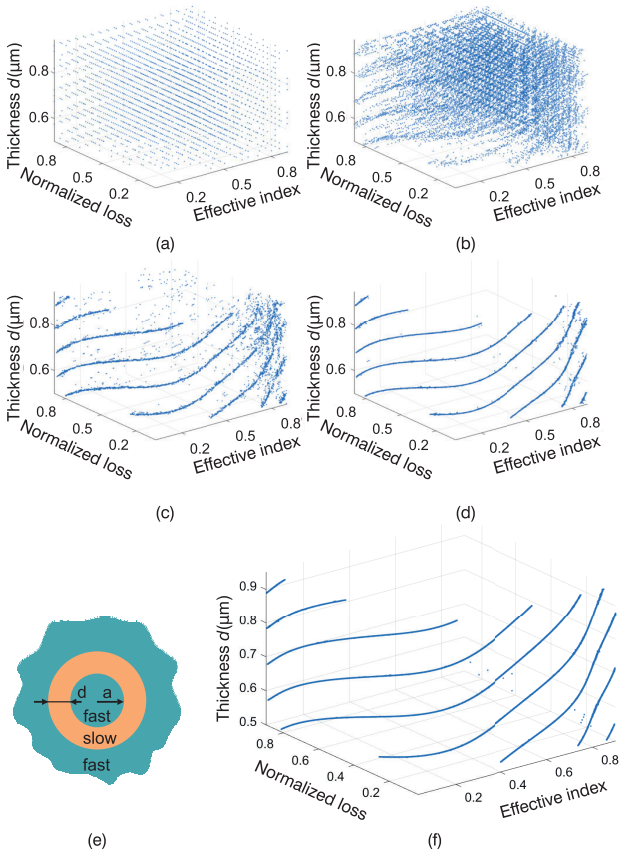

FIGURE 3. Iteration process for the ARRAW (radius of the core $a = 0.5$ μm). Normalized complex wavenumber as a function of the cladding thickness of d (a) iteration 1 (1 000 nodes) (b) iteration 3 (13 233 nodes) (c) iteration 5 (36 027 nodes) (d) iteration 7 (49 525 nodes) (e) cross-section of the structure (f) final iteration (100 000 nodes).

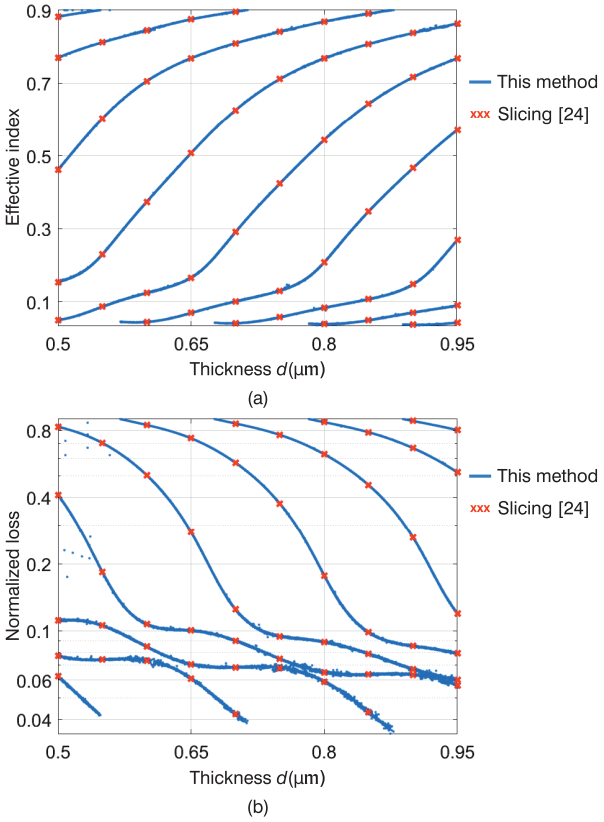

FIGURE 4. Normalized complex wavenumber of the ARRAW as a function of the cladding thickness of d (a) normalized effective index (b) normalized loss.

three guiding structures and a resonator. The algorithm was implemented in the Matlab environment and the code is freely available to the reader. A link to the source code and some additional comments are given in Appendices.

A. ANTI-RESONANT REFLECTING ACOUSTIC WAVEGUIDE

The first example considers the Anti-Resonant Reflecting Acoustic Waveguide (ARRAW) [1], which consists of core and anti-resonant cladding. The cylindrical structure of alternating layers of silicon/silica/silicon is schematically presented in Fig. 3. The structure propagates longitudinal (P), and shear (S) waves with higher velocities than those in the inner cladding. The analysis was performed in the area of the normalized propagation coefficient as a function of cladding thickness: $\Omega = \{z \in \mathbb{C}, t \in \mathbb{R} : 0.01 < Re(z) < 0.9 \wedge 0.01 < Im(z) < 0.9 \wedge 0.5 \cdot 10^{-6} < \mathrm{t} < 0.95 \cdot 10^{-6}\}$ with the use of $N_{max} = 100,000$ (1, 000 initial nodes).

Fig. 3 presents a process of the algorithm performance (for iterations 1, 3, 5, 7 and 11). As can be noticed, the results contain some scattered points that may not be zeros/poles, the so-called noise. As the number of points increases the noise is being reduced. In the last iteration, the assumed limit of 100, 000 function calls is reached and a resultant characteristic involving nine curves is obtained. In Fig. 4, the real axis and imaginary axis projections are presented. For comparison with the reference [1], the imaginary part representing the losses is shown on a logarithmic scale. Moreover, in this figure, the results of the slicing analysis at 10 equidistant points (different thicknesses) is depicted. The results are denoted with the use of red crosses and each single slice analysis was performed with the use of [24] with 10, 000 function calls, which sums up to 100, 000 function calls in the assumed Ω.

B. COAXIALLY LOADED CYLINDRICAL WAVEGUIDE

As a second example, the propagation coefficients of a coaxially loaded cylindrical waveguide are analyzed. The structure cross-section with its dimensions is depicted in Fig. 5, and it was previously analyzed in [33] and [34] with the use of a root tracing technique. The analysis was performed in the area of the normalized propagation coefficient as a function of frequency: $\Omega = \{z \in \mathbb{C}, t \in \mathbb{R} : -4.5 < Re(z) < 4.5 \wedge -0.25 < Im(z) < 2.25 \wedge 2 \cdot 10^{9} < \mathrm{t} < 7 \cdot 10^{9}\}$. The calculated dispersion characteristics are presented in Fig. 5 for a 3D view and Fig. 6 shows real and imaginary axis projections. The assumed $N_{max} = 100,000$ gave 1, 000 initial nodes. By comparing the results with the results obtained in [34], it can be seen that the proposed algorithm provides a more complete picture of

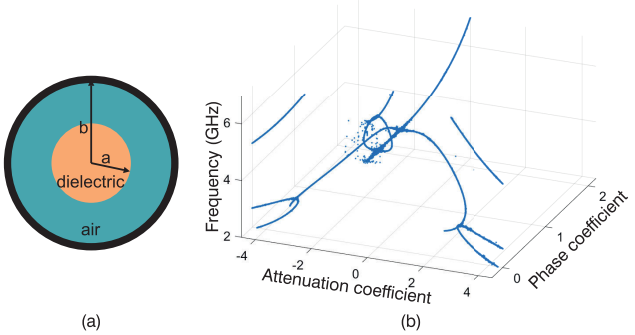

FIGURE 5. Coaxially loaded cylindrical waveguide ($\varepsilon_r = 10$ of the inner dielectric core, $a = 6.35$ mm and $b = 10$ mm) (a) cross-section (b) normalized propagation coefficient as a function of frequency.

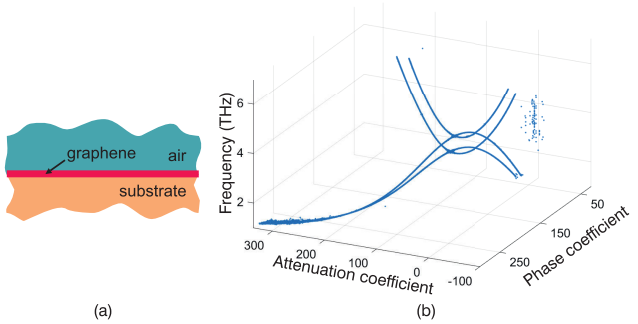

FIGURE 7. Graphene transmission line ($\varepsilon_r = 11.9$ of the substrate) (a) cross-section (b) normalized propagation coefficient as a function of frequency.

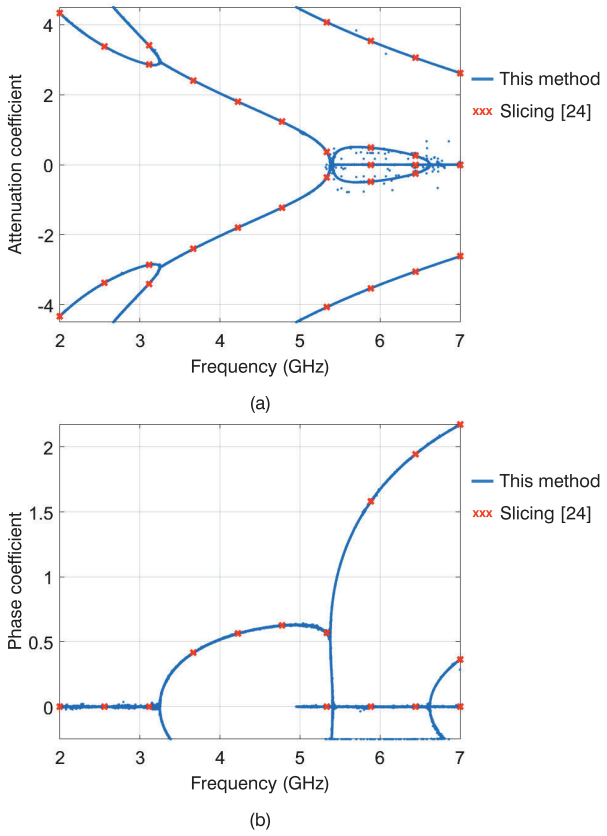

FIGURE 6. Normalized complex wavenumber of the coaxially loaded cylindrical waveguide as a function of frequency (a) normalized attenuation coefficient (b) phase coefficient.

the solutions in the analyzed region. Fig. 6 also shows the results of the slicing analysis [24] at 10 equidistant frequency points denoted with the use of red crosses (each slice analysis was performed with the use of 10,000 function calls, which sums up to 100,000 function calls in the assumed Ω). The results obtained from both algorithms are consistent, although much greater discretization in a function of frequency is required to obtain a complete picture of the solutions with the slicing method.

C. GRAPHENE TRANSMISSION LINE

The third example considers the analysis of a graphene transmission line for which the propagation coefficients were evaluated. Similar to the previous example, this structure was also analyzed in [33], and its cross-section is depicted in Fig. 7. The analysis was performed in the area of the normalized propagation coefficient as a function of frequency: $\Omega = \{z \in \mathbb{C}, t \in \mathbb{R} : -100 < Re(z) < 350 \wedge 25 < Im(z) < 300 \wedge 1 \cdot 10^{12} < t < 7 \cdot 10^{12}\}$. The calculated dispersion characteristics are presented in Fig. 7 for a 3D view and Fig. 8 shows real and imaginary axis projections. Also in this example, the assumed $N_{max} = 100,000$ gave 1,000 initial nodes.

As can be noticed, in the subregion of the picture the curves intersect, or approach each other. Further analysis in the selected subregion allows one to clearly assess the nature of the curves as shown in Fig. 9.

D. FABRY-PÉROT OPEN RESONATOR

The last example considers a Fabry-Pérot open resonator [39], which is widely used to determine the permittivity of materials. A sample under test is placed between two mirrors (see Fig. 10) and during the measurement, the shift of resonant frequencies caused by the introduced sample can be observed. The frequency shift is the difference between the resonant frequency obtained with a dielectric sample and the one for the empty resonator, i.e. without a sample. The permittivity of the material can be then determined using an electromagnetic model of the structure. The dimensions of the analyzed resonator are $R = 150$ mm, $D_{ap} = 180$ mm and $D = 100$ mm, and the considered sample is defined as follows: $h = 1003$ μm and $\varepsilon_r = 5.1217$, as in [39]. The technique utilized for the analysis in this example is a combination of the finite element method and an analytical approach described in [16]. The number of elements used in the finite element method is 5887 and was chosen based on the convergence analysis described in [16]. In this example, the point of interest is the influence of the losses on the resonant frequency. Since the structure is electrically large, the example is computationally demanding, and the analysis

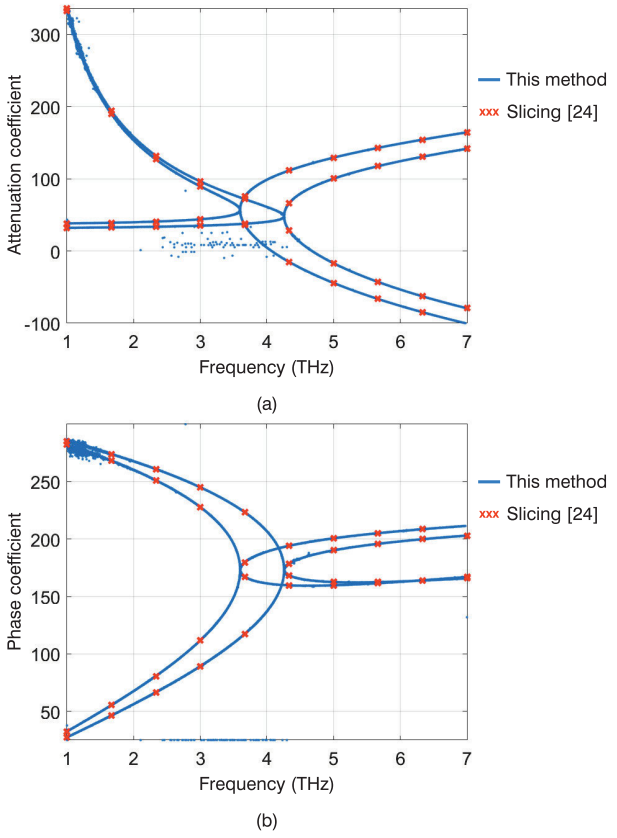

FIGURE 8. Normalized complex wavenumber of the Graphene transmission line as a function of frequency (a) normalized attenuation coefficient (b) phase coefficient.

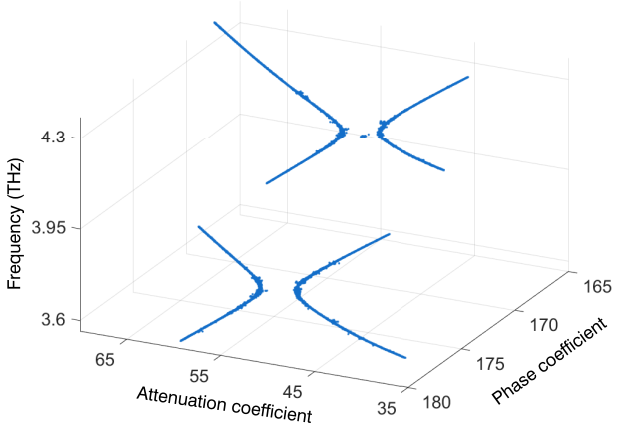

FIGURE 9. Normalized propagation coefficient as a function of frequency for the graphene transmission line example in a selected subregion of Ω.

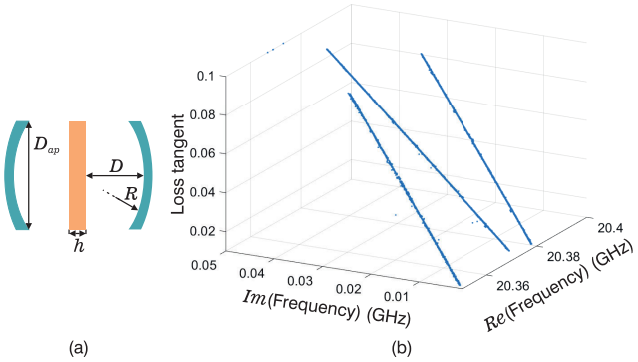

FIGURE 10. Fabry-Pérot Open Resonator (a) schematic view (b) complex resonant frequency as a function of the loss tangent.

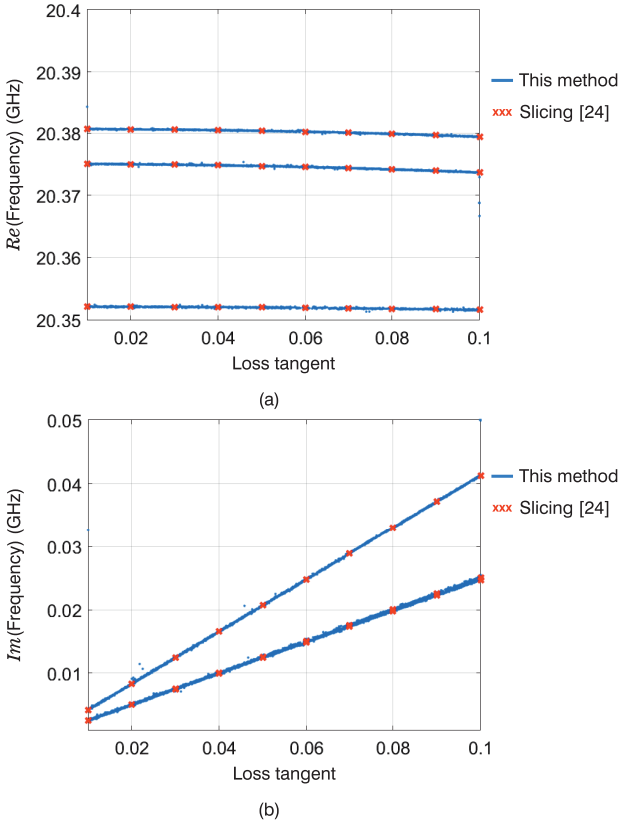

FIGURE 11. Resonant frequency as a function of loss tangent for the Fabry-Pérot Open Resonator (a) real part (b) imaginary part.)

was limited to relatively narrow ranges of frequency and loss tangent (additional parameter): $\Omega = \{z \in \mathbb{C}, t \in \mathbb{R} : 20.35 \cdot 10^9 < Re(z) < 20.40 \cdot 10^9 \wedge 10^6 < Im(z) < 50 \cdot 10^6 \wedge 0.01 < \mathrm{t} < 0.1\}$. Due to the limited ranges, the analysis was performed with $N_{max} = 10,000$, which was sufficient to obtain a clear picture of the characteristics. The calculated characteristics are presented in Fig. 10 for a 3D view and Fig. 11 shows real and imaginary axis projections. As it can be noticed, the increase in losses causes a linear increase in the imaginary part of the frequency, while the real part remains practically unchanged, which coincides with the measurements made in [16], where $Re(f) = 20.372$ GHz for the resonance $TEM_{0,0,27}$.

V. CONCLUSION

A global algorithm for finding zeros/poles was proposed, which gives a full picture of solutions in a fixed region $\Omega \subset \mathbb{C} \times \mathbb{R}$. The effectiveness of the algorithm was tested by

analyzing four example structures. For the first three examples, whose models are based on analytic expressions, the analysis took less than a minute. However, the last structure was much more numerically demanding (a single function call took 0.5 s) which gave a total analysis time of ca. 1.5h on an Intel(R) Xeon(R) X5680 3.33GHz CPU, 12GB RAM. As it can be seen from the presented examples, with the use of the same number of function calls (similar computational cost), the slicing method [24] does not give a full picture from which the nature of the tested curves can be assessed. In order to obtain a similar image as in the proposed method, an increased density of $t = \text{const}$ slices or their appropriate selection is required in the slicing method, which is the responsibility of the user. Depending on the example and the considered area, this density could be several or several dozen times higher, which is difficult to predict initially. The conducted analysis showed that the proposed method is an efficient alternative to the complex plane zero search algorithms, and to tracking algorithms and does not suffer from the disadvantages of these techniques.

.

APPENDIX A ABOUT THE ALGORITHM

Here are a few notes that might help the reader better understand the nuances of the described algorithm.

- If the calculated value of the function in a given node is a numerical zero (below machine precision) or infinity, then the argument of the function cannot be determined. In this case, all edges attached to that node are always treated as candidate edges.
- The algorithm assumes the possibility of increasing N_{max} after the analysis. If the user decides to increase this number, a part of the nodes (one percent) in the first step is added evenly throughout the domain, as in the case of the initial mesh. This is done by adding nodes at the centers of the currently longest edges (not only at the candidate edges).
- Variation of the additional parameter t, but also of $Re(z)$ and $Im(z)$, can be separately normalized to the desired range so that all variables are of a similar order of magnitude. In the algorithm, the Ω domain is by default normalized to a cube.

APPENDIX B SOURCE CODE

The source code in the Matlab environment for the Global Complex Roots and Poles Finding Algorithm in the $\mathbb{C} \times \mathbb{R}$ domain can be found at: https://github.com/PioKow/GRPF3D, and is licensed under the MIT License.

REFERENCES

[1] M. K. Schmidt, M. C. O'Brien, M. J. Steel, and C. G. Poulton, "ARRAW: Anti-resonant reflecting acoustic waveguides," *New J. Phys.*, vol. 22, no. 5, May 2020, Art. no. 053011, doi: 10.1088/1367-2630/ab7d79.

[2] D. Trofimowicz and T. P. Stefański, "Testing stability of digital filters using optimization methods with phase analysis," *Energies*, vol. 14, no. 5, p. 1488, Mar. 2021.

[3] L. Y. T. Nguyen, Y. H. Lee, Y. F. Chang, C. C. Hsu, J. Y. Lin, and H. C. Kan, "Subwavelength-resolution imaging of surface plasmon polaritons with up-conversion fluorescence microscopy," *Opt. Exp.*, vol. 30, no. 2, pp. 3113–3124, 2022.

[4] H. Wang, C. Wu, T. Jia, and D. Li, "A phase unwrapping method based on sparse depth for depth perception," *Opt. Lasers Eng.*, vol. 109, pp. 60–67, Oct. 2018.

[5] P. Y. Chen and Y. Sivan, "Robust location of optical fiber modes via the argument principle method," *Comput. Phys. Commun.*, vol. 214, pp. 105–116, May 2017.

[6] A. A. Albero, "Study of the lineshape of the chi (c1)(3872) state," *Phys. Rev. D, Part. Fields*, vol. 102, no. 9, 2020, Art. no. 092005.

[7] I. Chekhovskoy, S. B. Medvedev, I. A. Vaseva, E. V. Sedov, and M. P. Fedoruk, "Introducing phase jump tracking—A fast method for eigenvalue evaluation of the direct Zakharov–Shabat problem," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 96, May 2021, Art. no. 105718.

[8] F. Mesa, G. Valerio, R. Rodríguez-Berral, and O. Quevedo-Teruel, "Simulation-assisted efficient computation of the dispersion diagram of periodic structures: A comprehensive overview with applications to filters, leaky-wave antennas and metasurfaces," *IEEE Antennas Propag. Mag.*, vol. 63, no. 5, pp. 33–45, Oct. 2021.

[9] F. Gasdia and R. A. Marshall, "A new longwave mode propagator for the Earth–ionosphere waveguide," *IEEE Trans. Antennas Propag.*, vol. 69, no. 12, pp. 8675–8688, Dec. 2021.

[10] M. Warecka, R. Lech, and P. Kowalczyk, "Hybrid method analysis of unshielded guiding structures," in *Proc. 23rd Int. Microw. Radar Conf. (MIKON)*, Oct. 2020, pp. 398–401.

[11] F. Mesa, R. Rodríguez-Berral, and F. Medina, "On the computation of the dispersion diagram of symmetric one-dimensionally periodic structures," *Symmetry*, vol. 10, no. 8, p. 307, Aug. 2018.

[12] M. Warecka, R. Lech, and P. Kowalczyk, "Propagation in the open cylindrical guide of arbitrary cross section with the use of field matching method," *IEEE Trans. Antennas Propag.*, vol. 66, no. 6, pp. 3237–3240, Jun. 2018.

[13] E. L. Tan and D. Y. Heh, "Application of belevitch theorem for pole-zero analysis of microwave filters with transmission lines and lumped elements," *IEEE Trans. Microw. Theory Techn.*, vol. 66, no. 11, pp. 4669–4676, Nov. 2018.

[14] R. Lech, "Calculation of resonance in planar and cylindrical microstrip structures using a hybrid technique," *IEEE Trans. Antennas Propag.*, vol. 66, no. 1, pp. 497–500, Jan. 2018.

[15] R. Lech, W. Marynowski, and A. Kusiek, "An analysis of elliptical-rectangular multipatch structure on dielectric-coated confocal and nonconfocal elliptic cylinders," *IEEE Trans. Antennas Propag.*, vol. 63, no. 1, pp. 97–105, Jan. 2015.

[16] B. Salski, T. Karpisz, M. Warecka, P. Kowalczyk, P. Czekala, and P. Kopyt, "Microwave characterization of dielectric sheets in a plano-concave Fabry–Pérot open resonator," *IEEE Trans. Microw. Theory Techn.*, vol. 70, no. 5, pp. 2732–2742, May 2022.

[17] M. Warecka, S. Dziedziewicz, P. Kowalczyk, R. Lech, P. Czekala, B. Salski, and P. Kopyt, "Efficient Fabry–Pérot open resonator analysis by the use of a scattering matrix method," in *Proc. 24th Int. Microw. Radar Conf. (MIKON)*, Sep. 2022, pp. 1–4.

[18] E. C. G. Sudarshan, C. B. Chiu, and V. Gorini, "Decaying states as complex energy eigenvectors in generalized quantum mechanics," *Phys. Rev. D, Part. Fields*, vol. 18, no. 8, pp. 2914–2929, Oct. 1978.

[19] W. Press, S. Teukolsky, W. Vetterling, and B. Flannery, *Numerical Recipes in Fortran 77: The Art of Scientific Computing*, vol. 1. Cambridge, U.K.: Cambridge Univ. Press., 1992.

[20] M. Abramowitz and I. A. Stegun, *Handbook of Mathematical Functions With Formulas, Graphs, and Mathematical Tables. National Bureau of Standards Applied Mathematics Series 55. Tenth Printing*. New York, NY, USA: U.S. Government Printing Office, 1972.

[21] A. P. Austin, P. Kravanja, and L. N. Trefethen, "Numerical algorithms based on analytic function values at roots of unity," *SIAM J. Numer. Anal.*, vol. 52, no. 4, pp. 1795–1821, Jan. 2014.

[22] Y. Nakatsukasa, O. Sète, and L. N. Trefethen, "The AAA algorithm for rational approximation," *SIAM J. Scientific Comput.*, vol. 40, no. 3, pp. A1494–A1522, Jan. 2018.

[23] P. Kowalczyk, "Global complex roots and poles finding algorithm based on phase analysis for propagation and radiation problems," *IEEE Trans. Antennas Propag.*, vol. 66, no. 12, pp. 7198–7205, Dec. 2018.

[24] S. Dziedziewicz, M. Warecka, R. Lech, and P. Kowalczyk, "Self-adaptive mesh generator for global complex roots and poles finding algorithm," *IEEE Trans. Microw. Theory Techn.*, vol. 71, no. 7, pp. 2854–2863, Jul. 2023.
[25] L. M. Delves and J. N. Lyness, "A numerical method for locating the zeros of an analytic function," *Math. Comput.*, vol. 21, no. 100, pp. 543–560, 1967.
[26] M. H. Meylan and L. Gross, "A parallel algorithm to find the zeros of a complex analytic function," *ANZIAM J.*, vol. 44, pp. E236–E254, Jan. 2002.
[27] P. Kowalczyk, "Complex root finding algorithm based on Delaunay triangulation," *ACM Trans. Math. Softw.*, vol. 41, no. 3, pp. 1–13, Jun. 2015.
[28] G. P. Zouros, "CCOMP: An efficient algorithm for complex roots computation of determinantal equations," *Comput. Phys. Commun.*, vol. 222, pp. 339–350, Jan. 2018.
[29] J. J. Michalski, "Complex border tracking algorithm for determining of complex zeros and poles and its applications," *IEEE Trans. Microw. Theory Techn.*, vol. 66, no. 12, pp. 5383–5390, Dec. 2018.
[30] M. Jasinski, S. Dziedziewicz, M. Jozwicka, and P. Kowalczyk, "An improvement of global complex roots and poles finding algorithm for propagation and radiation problems," in *Proc. 13th Eur. Conf. Antennas Propag. (EuCAP)*, Mar. 2019, pp. 1–5.
[31] J. J. Michalski and P. Kowalczyk, "Efficient and systematic solution of real and complex eigenvalue problems employing simplex chain vertices searching procedure," *IEEE Trans. Microw. Theory Techn.*, vol. 59, no. 9, pp. 2197–2205, Sep. 2011.
[32] P. Kowalczyk and W. Marynowski, "Efficient complex root tracing algorithm for propagation and radiation problems," *IEEE Trans. Antennas Propag.*, vol. 65, no. 5, pp. 2540–2546, May 2017.
[33] S. Dziedziewicz, R. Lech, and P. Kowalczyk, "A self-adaptive complex root tracing algorithm for the analysis of propagation and radiation problem," *IEEE Trans. Antennas Propag.*, vol. 69, no. 8, pp. 5171–5174, Aug. 2021.
[34] S. Dziedziewicz, M. Warecka, R. Lech, and P. Kowalczyk, "Multipath complex root tracing," in *Proc. 24th Int. Microw. Radar Conf. (MIKON)*, Sep. 2022, pp. 1–4.
[35] J. W. Brown and R. V. Churchill, *Complex Variables and Applications*, 6th ed. New York, NY, USA: McGraw-Hill, 1996.
[36] X. Ying and I. N. Katz, "A reliable argument principle algorithm to find the number of zeros of an analytic function in a bounded domain," *Numerische Math.*, vol. 53, nos. 1–2, pp. 143–163, Jan. 1988.
[37] P. Kravanja and M. Van Barel, "A derivative-free algorithm for computing zeros of analytic functions," *Computing*, vol. 63, no. 1, pp. 69–91, Jul. 1999.
[38] P. Kowalczyk, "On root finding algorithms for complex functions with branch cuts," *J. Comput. Appl. Math.*, vol. 314, pp. 1–9, Apr. 2017.
[39] B. Salski, T. Karpisz, P. Kopyt, and J. Krupka, "Rigorous scattering matrix analysis of a Fabry–Pérot open resonator," *IEEE Trans. Microw. Theory Techn.*, vol. 68, no. 12, pp. 5093–5102, Dec. 2020.

SEBASTIAN DZIEDZIEWICZ was born in Elblag, Poland, in 1995. He received the M.Sc.E.E. degree in microwave engineering from the Gdańsk University of Technology, in 2019, where he is currently pursuing the Ph.D. degree. He is also with the Department of Microwave and Antenna Engineering, Faculty of Electronics, Telecommunications and Informatics, Gdańsk University of Technology. His current research interests include scattering and propagation of electromagnetic wave problems, algorithms, and numerical methods.

MALGORZATA WARECKA (Student Member, IEEE) received the M.Sc.E.E. degree from the Gdańsk University of Technology, Gdańsk, Poland, in 2018, where she is currently pursuing the Ph.D. degree. She is also with the Department of Microwave and Antenna Engineering, Faculty of Electronics, Telecommunications and Informatics, Gdańsk University of Technology. Her current research interests include scattering and propagation of electromagnetic wave problems, algorithms, and numerical methods.

RAFAL LECH (Senior Member, IEEE) was born in Elblag, Poland, in 1977. He received the M.Sc.E.E., Ph.D. (Hons.), and D.Sc. degrees from the Gdańsk University of Technology, Gdańsk, Poland, in 2001, 2007, and 2018, respectively. He is currently with the Department of Microwave and Antenna Engineering, Faculty of Electronics, Telecommunications and Informatics, Gdańsk University of Technology. His main research interests include electromagnetic wave scattering, hybrid methods, filter design, complex materials, metamaterial applications at microwave frequencies, electromagnetic analysis of periodic structures, and antenna designs.

PIOTR KOWALCZYK (Member, IEEE) was born in Wejherowo, Poland, in 1977. He received the M.Sc.E.E. degree in applied physics and mathematics and the Ph.D. (Hons.) and D.Sc. degrees in electrical engineering from the Gdańsk University of Technology, Gdańsk, Poland, in 2001, 2008, and 2018, respectively. He is currently with the Department of Microwave and Antenna Engineering, Faculty of Electronics, Telecommunications and Informatics, Gdańsk University of Technology. His current research interests include scattering and propagation of electromagnetic wave problems, algorithms, and numerical methods.