Post-print of: D. Trofimowicz and T. P. Stefański, "Multimodal Particle Swarm Optimization with Phase Analysis to Solve Complex Equations of Electromagnetic Analysis," 2020 23rd International Microwave and Radar Conference (MIKON), Warsaw, Poland, 2020, pp. 44-48, doi: 10.23919/MIKON48703.2020.9253937.

Multimodal Particle Swarm Optimization with Phase Analysis to Solve Complex Equations of Electromagnetic Analysis

Damian Trofimowicz
SpaceForest Ltd.
Gdynia, Poland
d.trofimowicz@gmail.com

Tomasz P. Stefański
Faculty of Electronics, Telecommunications, and Informatics
Gdansk University of Technology
Gdansk, Poland
tomasz.stefanski@pg.edu.pl

***Abstract*—In this paper, a new meta-heuristic method of finding roots and poles of a complex function of a complex variable is presented. The algorithm combines an efficient space exploration provided by the particle swarm optimization (PSO) and the classification of root and pole occurrences based on the phase analysis of the complex function. The method initially generates two uniformly distributed populations of particles on the complex plane and extracts the function phase in a position of each particle. By collecting phase samples, the candidate regions of root and pole occurrences are selected. Then, the second population, by iteratively converging towards candidate regions, thoroughly explores an area outside candidate regions and reduces the possibility of root or pole omission. The subsequent swarms are generated locally to explore candidate regions and decrease their size. The algorithm is verified in electromagnetic benchmark that solves the equation determining surface waves on a microstrip antenna. The numerical results show that the algorithm is able to solve multimodal problems quickly even with a small initial population and a small number of generated swarms.**

***Index Terms*—Particle swarm optimization, optimization methods, mathematical programming, electromagnetic analysis**

I. Introduction

The electromagnetic analysis very often requires the study of function properties on the complex plane. Fundamental problems related to electromagnetic field, e.g., the wave propagation in transmission lines or the leaky-wave radiation, are described by parameters that are complex numbers. For dielectric materials, electric susceptibility and permittivity are complex numbers with imaginary parts characterizing material losses. The loss tangent widely used to characterize material losses and implemented in most commercially available electromagnetic simulators is defined as the ratio of imaginary and real parts of the complex permittivity. Thus, solving electromagnetic problems is inseparably related to the complex-domain analysis. However, such an analysis in the complex domain is much more difficult than in the real one. The key to the solution of any complex electromagnetic problem is its transformation into a single equation $f(z) = 0$ and consideration of a root-finding problem for the function $f(z)$. This paper introduces the particle swarm optimization (PSO) method for finding roots and poles in electromagnetic analysis. The paper is organized as follows: In Section II, the actual state of the art in PSO and root finding techniques is presented. In Section III, the multimodal PSO with phase analysis (MPSO-WPA) algorithm is presented and analysed. Numerical benchmarks involving complex functions applied in electromagnetic analysis are presented in Section IV. Finally, the conclusion is drawn in Section V.

II. State of the Art

With the advent of fast computational machines and rapid development of artificial intelligence technologies, many new optimization techniques have arisen to solve complex electromagnetic problems. In this paper, we investigate the PSO algorithm [1], [2], which employs the swarm intelligence [3], [4] observed in social and cooperative behavior of various intelligent colonies in nature. PSO alike genetic algorithms is initialized with a random population of swarm particles. It tracks coordinates of particles in space which are associated with best known solutions as well as simultaneously samples the space while moving populations towards them. Two very important concepts result from this approach. First one is the exploration, which is the ability of the algorithm to search for solutions in regions that are far away from the actual solution. Second one is the exploitation, which is the ability to search for solutions in regions that are close to the actual solution. The trade-off between the global space exploration and the local exploitation is the inherent issue of the PSO technique. Moreover, the original algorithm is dedicated to the detection of a single global minimum and has to be extended to explore multimodal functions. The term "multimodal" should not be confused with the term "multiobjective" which describes several fitness functions evaluated at the same time. The multimodal optimization is focused on exploration of a single fitness function to find multiple (most accurate) solutions, usually in different distant space regions. Since the PSO introduction in 1995, many variants of the original method have been proposed. Mostly known variant [5] introduces the inertia to the original formula as follows:

$$v_i = wv_i' + c_1u_1(p_i - x_i) + c_2u_2(p_g - x_i) \quad (1)$$

$$x_i = x_i' + v_i. \quad (2)$$

In (1)–(2), v_i and v_i' are respectively current and previous velocities of the i-th particle. The variables p_i and p_g are respectively the best so far coordinates found by the individual particle and the best so far coordinates found by the whole swarm. The scaling factors c_1 and c_2 represent respectively attractions towards the best position of the i-th particle and the best position of the whole swarm. Parameters u_1 and u_2 are randomly generated numbers between 0 and 1. The inertia weight w determines how much the previous velocity is preserved. The most common and intuitive practice is to initialize the inertia with a large value, giving priority to the global exploration, and decrease it in subsequent iterations to obtain more accurate local exploitation [6], [7]. The particle coordinates x_i are updated by adding velocity v_i to the previous particle coordinates x_i'. The original PSO algorithm is very efficient when is applied to basic functions. Its exceptional properties stem from dynamic particle interactions. However, its efficiency deteriorates when solving multimodal functions because information is gathered from global and local sources, which may vary between them. This could lead to delays, poor convergence, oscillations or two steps forward one step back phenomenon [8]. Moreover, the original method suffers from the premature convergence to the global solution, because all particle velocities are updated in each iteration by p_g. It has led to the uprise of new learning techniques that determine the information distribution between particles [9], [10]. The swarm learning methods employ not only movement direction and velocity, but can also define how the swarms are divided and distributed in the search region. It is extremely important, especially in the case of multimodal search, where regions of interest are distant.

The swarm algorithm proposed in this work is developed to solve equations of electromagnetic analysis. In many cases, these equations can be transformed into complex equations on the complex plane, which can be solved by finding complex roots. The Newton-Raphson technique is the classic method to find quickly a local root [11]. The drawback of this technique is that it requires function derivatives and an initial starting point. Davidenko's method [12], being a reduction of Newton's method into n-coupled first-order differential equations of a dummy variable for the numerical solution of n-coupled nonlinear algebraic equations, has been widely used to solve complex eigenvalue equations [13], [14]. The drawback of this method is that it requires the analytical expression for the first derivative of the considered complex function. Kuhn's algorithm [15] overcomes this limitation, hence, it is successfully used to find surface waves in microstrip antenna problems [16]. The algorithm based on the simplex chain vertices search (SCVS) [17], [18] and supplemented with the complex boarder tracking (CBT) method [19] is applied to solve dispersion equations in the complex domain, demonstrating high efficiency and low numerical costs. Unfortunately, the majority of root finding techniques is not suitable to search for roots globally. Most of the methods is efficient when exploring small regions with a few roots and favorable initial starting point. However, an efficient and fast global complex roots and poles finding algorithm (GRPF) based on the Delaunay triangulation and phase analysis in the complex domain has been recently proposed [20], [21]. This technique is based on sampling of the function phase in nodes of a triangular mesh. Relying on phase changes between nodes, candidate regions with possible root/pole occurrence are selected. Subsequently, discretized Cauchy's argument principle (DCAP) is applied over the candidate region to evaluate if any root or pole is indeed inside the candidate region. This algorithm can be applied to a wide class of analytic functions. However, it assumes a uniform meshing in the first iteration. Hence, it can be inefficient if an initial mesh is very wide, especially when large regions are analysed on the complex plane. Moreover, the initial mesh does not contain any random nodes. Hence, a procedure is needed to validate if the initial mesh is fine enough to find all roots and poles. Therefore, we decided to propose the technique that employs the stochastic distribution of nodes and executes differently in each run. Statistical probability of omitting root/pole decreases with the number of runs. The proposed algorithm also employs the phase analysis but the mesh is obtained with the use of the stochastic distribution of swarm particles. Hence, the PSO effectiveness of global exploration and local exploitation is merged with a precise selection of candidate regions obtained from the analysis of the function phase.

III. The MPSO-WPA Algorithm

The flowchart of the MPSO-WPA algorithm is presented in Fig. 1. This algorithm is executed in the following steps:

A. Algorithm Initialization and Generation of Initial Swarm

Initialize algorithm parameters such as the algorithm accuracy ε, the search area size, the number of particles in the swarm population n, the inertia w, the scaling factor c, the initial velocity of particles v_0, the number of iterations to create subsequent swarm a. Then, generate the uniformly distributed random swarm on the complex plane

$$Z_{i,j} = \{z_{i,j,1}, z_{i,j,2}, ..., z_{i,j,n}\}. \quad (3)$$

where i is the swarm number and j is the iteration number. The indices i and j are equal to zero for the initial swarm $Z_{0,0}$, which is not updated in subsequent iterations.

B. Generation of the Second Global Swarm

Increase the index i and generate the second uniformly distributed swarm on the complex plane. The second swarm is generated to perform global exploration. Proceed to the step D.

C. Generation of Subsequent Local Swarm

Increase the index i and generate the subsequent swarm on the complex plane. The swarm is split and distributed in candidate regions. Subsequent local swarms are generated to perform local exploitation of candidate regions. Proceed to the step D.

D. Search for Regions Around Potential Root or Pole

Evaluate the function argument for each particle coordinates and compute the phase quadrant in which the corresponding function value is placed

$$Q_i = \left\{ \begin{matrix} 1, & 0 \le \arg f(z_i) < \frac{\pi}{2} \\ 2, & \frac{\pi}{2} \le \arg f(z_i) < \pi \\ 3, & \pi \le \arg f(z_i) < \frac{3\pi}{2} \\ 4, & \frac{3\pi}{2} \le \arg f(z_i) < 2\pi \end{matrix} \right\}. \tag{4}$$

Then, apply the Delaunay triangulation to particle coordinates, i.e., generate triangular connections (i.e., edges) between particles. Compute the quadrant distance along each of the connections, i.e.,

$$\Delta Q_p = Q_{p2} - Q_{p1}. \tag{5}$$

Search for connections such as $|\Delta Q_p| = 2$. Particles connected in this way must have different signs of the real and imaginary parts of function values. These connections (called candidates) are considered as a potential vicinity of either root or pole. If candidate connections are not detected then the algorithm execution is terminated, i.e., there is no root nor pole on the complex plane. When any candidate connection is detected, compute the coordinates of the centre of the candidate connection and its length. If the longest length of the collected candidate connections is less than the assumed accuracy, then proceed to the step F. If the current iteration number j is equal to the number of iterations to create subsequent swarm a, then increase the swarm number i, set $j = 0$ and execute the loop to the step C. In other cases, increase the iteration number j and proceed to the next step E.

E. Exploitation of Candidate Regions

Compute distances between particle coordinates and coordinates of candidate centres. For each particle, select the shortest distance and store the corresponding coordinates of candidate centre in the set

$$P_{i,j} = \{p_{i,j,1}, p_{i,j,2}, ..., p_{i,j,n}\} \tag{6}$$

where i is the swarm number and j is the iteration number. Accelerate the particles in i-th swarm towards the closest p by updating the velocity of each particle

$$V_{i,j+1} = wV_{i,j} + cu(P_{i,j} - Z_{i,j}). \tag{7}$$

In (7), $Z_{i,j}$ denotes the particle coordinates in the i-th swarm, $P_{i,j}$ denotes the coordinates of the closest candidate centres relative to the particle coordinates of the i-th swarm, $V_{i,j}$ denotes the particle velocities of i-th swarm, c denotes the scaling factor representing the attraction, w denotes inertia weight which determines how much the particles remain along their original direction, u denotes a random number between 0 and 1. The mathematical operations (i.e., addition, subtraction, multiplication) are executed on corresponding array elements. Then, update the coordinates of the population by

$$Z_{i,j+1} = Z_{i,j} + V_{i,j+1}. \tag{8}$$

Finally, go to the searching for candidates step D.

F. Verification of Root or Pole Occurrence

Confirm the existence of root or pole applying DCAP by integrating quadrant differences along the path between particles around the candidate region

$$q = \frac{1}{4} \sum_{p=1}^{P} \Delta Q_p. \tag{9}$$

The parameter q is a positive integer when a root is found, a negative integer when a pole is found and zero when there is neither root nor pole in the selected region.

IV. Numerical results

The analysis of surface waves on a microstrip antenna [16], [21], [22] is a typical problem in microwave engineering requiring the searching for roots. The eigenvalue equation $f(z) = 0$ for this problem is formulated based on the following function:

$$f(z) = z^2 tan^2 z + \varepsilon_r^2 z^2 + \varepsilon_r^2 (k_0 h)^2 (1 - \varepsilon_r \mu_r). \tag{10}$$

In (10), h is the thickness of dielectric, ε_r, μ_r are respectively the relative permittivity and permeability, and $k_0 = 2\pi f/c$ is the wave number. Typical values used in benchmarks are as follows: $\varepsilon_r = 5 - 2i$, $\mu_r = 1 - 2i$, $f = 1$ GHz and $h = 1$ cm. The number of roots and poles found depends on the size of analysed region. Most benchmarks are limited to a small searching region with a few poles, e.g., $|z| \le 1$ with fixed starting point. MPSO-WPA is very fast and efficient in the case of only a few distant roots that are located in the search region. Then, uniformly distributed particles converge very fast. Each algorithm run is unique due to the stochastic base of swarm distribution. Therefore, the computation time, the number of iterations and the number of particles necessary to obtain the assumed accuracy vary for each run. Exemplary run results with particle movements in subsequent iterations are presented in Fig. 2. The algorithm is executed on a personal computer equipped with Intel i7-4700MQ processor. The algorithm finds two roots $\pm(0.51511 - 0.50711i)$ in the constrained area $|z| \le 1$. The MPSO-WPA algorithm tested in five runs always returned the same root values. The obtained results of computations for varying accuracy settings, to check the computing time, the number of particles and iterations necessary to converge, are presented in Table I. The average execution time from five runs for the accuracy $\varepsilon = 1E{-}15$ is equal to 0.5 sec. On average, it involves 36 iterations with the final population of 1970 particles. In comparison to the results obtained by the multimodal genetic algorithm with phase analysis (MGA-WPA) [23] and the GRPF algorithm

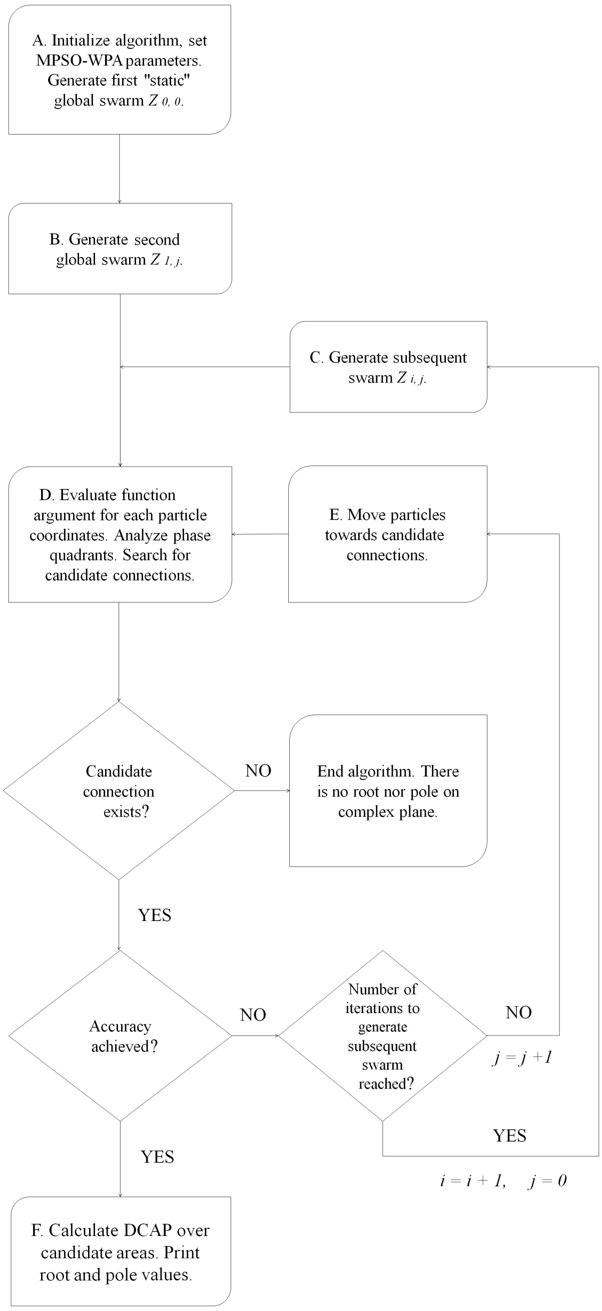

Fig. 1. Flowchart of the MPSO-WPA algorithm.

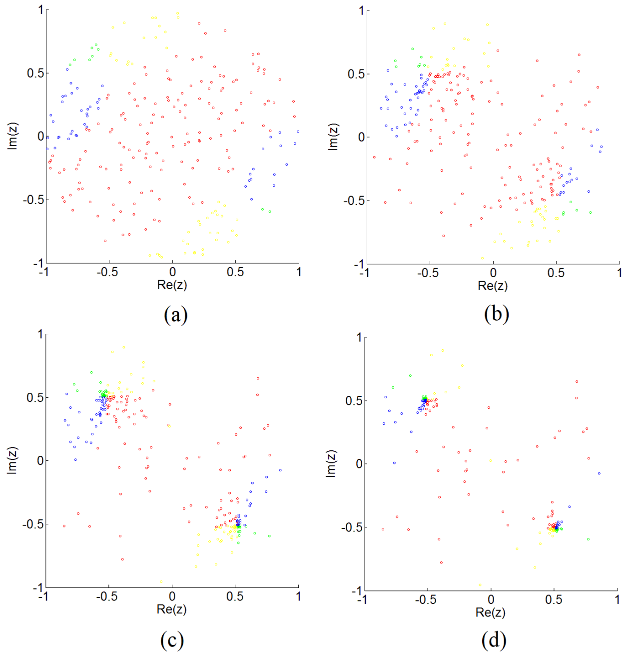

Fig. 2. Distribution of swarm particles for analysis of surface waves on microstrip antenna in region $|z| \leq 1$. (a) Initial particle distribution. (b) Particle distribution after first iteration. (c) Particle distribution after second iteration. (d) Final particle distribution. Phase quadrants of particles on the z-plane: • Q=1, • Q=2, • Q=3, • Q=4.

TABLE I
COMPUTATION TIME, NUMBER OF PARTICLES AND ITERATIONS NECESSARY TO CONVERGE FOR VARYING ACCURACY

Accuracy	Time (sec)	Particles no.	Iterations
1E-05	0.10	930	15
1E-06	0.12	1000	16
1E-07	0.13	1010	18
1E-08	0.15	1170	20
1E-09	0.17	1250	21
1E-10	0.21	1370	24
1E-11	0.26	1450	27
1E-12	0.28	1530	28
1E-13	0.30	1650	30
1E-14	0.38	1770	33
1E-15	0.50	1970	36

[20], MPSO-WPA is slightly slower than both methods. The MGA-WPA method requires about 0.44 sec whereas GRPF requires 0.35 sec to converge in this case. It stems from the global exploration performed by the second swarm in regions that are far away from detected candidates. However, the wide space exploration is an invaluable advantage of the MPSO-WPA method, which in consequence does not allow for the root/pole omission if the initial detection of candidate regions fails. In the case of the problem with two roots, the solution is found very fast. More difficult task is to explore the space where multiple roots are close to each other. It is benchmarked by expanding the search region of the surface-wave function to $|z| \leq 2$. Exemplary run results, including particle distributions as well as the Delaunay triangulation applied to the final swarm distribution, are presented in Fig. 3. All roots and poles are found in the search region. It is worth noticing that the Delaunay triangulation creates irregular mesh of particle connections concentrated around roots and poles. The subsequent swarms converge effectively towards these points. The results of the analysis of surface waves on the microstrip antenna with listed roots and poles found are presented in Table II.

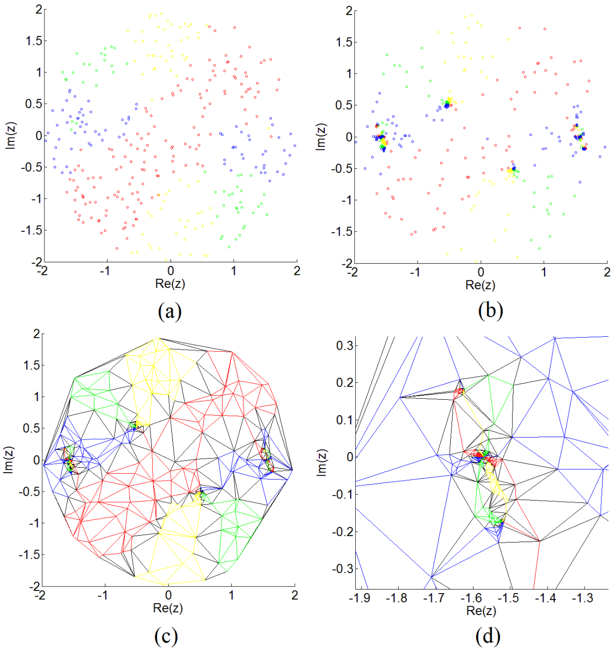

Fig. 3. Distribution of swarm particles for analysis of surface waves on microstrip antenna in region $|z| \leq 2$. (a) Initial particle distribution. (b) Final particle distribution. (c) Delaunay triangulation applied to final particle distribution. (d) Zoomed mesh around single pole $-1.57 + 0i$ and two roots $-1.62{+}0.18i$ and $-1.52{-}0.17i$. Phase quadrants of particles on the z-plane: • Q=1, • Q=2, • Q=3, • Q=4.

TABLE II
ROOTS AND POLES FOUND IN SEARCH REGION $|z| \leq 2$

Region	Root/Pole	Qualification
1	$-1.6247 + 0.1821i$	root
2	$-1.5708 - (1.6567E - 6)i$	pole
3	$1.5708 - (1.4158E - 6)i$	pole
4	$-0.5151 + 0.50711i$	root
5	$1.6247 - 0.1821i$	root
6	$-1.5202 - 0.17367i$	root
7	$1.5202 + 0.17367i$	root
8	$0.5151 - 0.50711i$	root

V. Conclusion

This paper proposes the efficient MPSO-WPA algorithm to solve complex equations occurring in microwave engineering and electromagnetic theory. The algorithm merge stochastic, population-based optimization technique with phase quadrant analysis on the complex plane. Hence, the algorithm prevents from omitting roots/poles due to the stochastic space exploration by subsequent swarms. The results obtained with the use of the MPSO-WPA algorithm are demonstrated and discussed.

References

[1] R. C. Eberhart and J. Kennedy, “A new optimizer using particle swarm theory,” in *Proceedings of the 6th International Symposium on Micro Machine and Human Science*, vol. 1, 1995, pp. 39–43.

[2] K. L. Du and M. Swamy, *Search and Optimization by Metaheuristics*. Springer, 2016.

[3] G. Beni and J. Wang, “Swarm intelligence in cellular robotics systems,” in *Proceeding of NATO Advanced Workshop on Robots and Biological System*, 1989.

[4] J. Kennedy and R. Eberhart, “Particle swarm optimization,” in *Proceedings of IEEE Int. Conf. Neural Network*, 1995, pp. 1942–1948.

[5] Y. Shi and R. Eberhart, “A modified particle swarm optimizer,” in *Proceedings of IEEE Congr. Evol. Comput.*, 1998, pp. 69–73.

[6] ——, *Parameter selection in particle swarm optimization*. Springer, 1998.

[7] ——, “A modified particle swarm optimizer,” in *Proceedings of IEEE Congr. Evol. Comput.*, 1998, pp. 69–73.

[8] K. E. Parsopoulos and M. N. Vrahatis, “On the computation of all global minimizers through particle swarm optimization,” *IEEE Trans. Evol. Comput.*, vol. 8, no. 3, pp. 211–224, 2004.

[9] J. J. Liang, A. K. Qin, P. N. Suganthan, and S. Baskar, “Comprehensive learning particle swarm optimizer for global optimization of multimodal functions,” *IEEE Trans. Evol. Comput.*, vol. 10, no. 3, pp. 281–295, 2006.

[10] Z. Zhan, J. Zhang, Y. Li, and Y. Shi, “Orthogonal learning particle swarm optimization,” *IEEE Trans. Evol. Comput.*, vol. 15, no. 6, pp. 832–847, 2011.

[11] M. Abramowitz and I. A. Stegun, *Handbook of Mathematical Functions With Formulas, Graphs, and Mathematical Tables*. Dover Publications, 1972.

[12] D. F. Davidenko, “On a new method of numerical solution of systems of nonlinear equations,” *Doklady Akad. Nauk S.S.S.R.(N.S.)*, vol. 88, pp. 601–602, 1953.

[13] H. A. N. Hejase, “On the use of Davidenko’s method in complex root search,” *IEEE Trans. Microw. Theory Tech.*, vol. 41, no. 1, pp. 141–143, 1993.

[14] S. H. Talisa, “Application of Davidenko’s method to the solution of dispersion relations in lossy waveguiding systems,” *IEEE Trans. Microw. Theory Tech.*, vol. 33, no. 10, pp. 967–971, 1985.

[15] H. W. Kuhn, *Fixed points*. Academic Press, 1977.

[16] Y. L. H. Jiang, “Numerical analysis of surface wave on microstrip antenna with lossy substrate,” *Electron. Lett.*, vol. 33, no. 19, pp. 1592–1593, 1997.

[17] 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 Tech.*, vol. 59, no. 9, pp. 2197–2205, 2011.

[18] J. J. Michalski, “Systematic solution of nonlinear equations with close or double roots in complex domain: application of SCVS algorithm for crosswise complex dispersion characteristics,” in *Proceedings of 2012 19th International Conference on Microwaves, Radar Wireless Communications*, vol. 1, May 2012, pp. 323–326.

[19] ——, “Complex border tracking algorithm for determining of complex zeros and poles and its applications,” *IEEE Trans. Microw. Theory Tech.*, vol. 66, no. 12, pp. 5383–5390, 2018.

[20] 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, 2018.

[21] ——, “Complex root finding algorithm based on Delaunay triangulation,” *ACM Trans. Math. Softw.*, vol. 41, no. 3, pp. 1–13, 2015.

[22] T. Yu-Bo, “Solving complex transcendental equations based on swarm intelligence,” *IEEJ Trans. Electr. Electron. Eng.*, vol. 4, no. 6, pp. 755–762, 2009.

[23] D. Trofimowicz and T. P. Stefański, “Multimodal genetic algorithm with phase analysis to solve complex equations of electromagnetic analysis,” in *Proceedings of 2020 23rd International Conference on Microwaves, Radar Wireless Communications*, 2020.