

# Multimodal Genetic Algorithm 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 contribution, a new genetic-algorithm-based method of finding roots and poles of a complex function of a complex variable is presented. The algorithm employs the phase analysis of the function to explore the complex plane with the use of the genetic algorithm. Hence, the candidate regions of root and pole occurrences are selected and verified with the use of discrete Cauchy's argument principle. The algorithm is evaluated in an electromagnetic benchmark that successfully solves the eigenvalue problem determining the propagation of surface waves along a spatially dispersive graphene sheet. The numerical results show that the possibility to find all roots and poles of the function may be limited by the initial population size, especially when the search region is large and roots and poles are located close to each other.

**Index Terms**—Genetic algorithms, optimization methods, mathematical programming, electromagnetic analysis

## I. INTRODUCTION

Nowadays, the electromagnetic field is present in almost all areas of human life, i.e., from cell-phone communication, Wi-Fi transmission, microwave and induction cooking up to space and military technologies. The wave propagation is the fundamental problem related to the electromagnetic field. Frequently, the analysis of the electromagnetic field requires the investigation of properties of a complex-valued function on the complex plane. The determination of complex propagation coefficients of surface waves is a typical example of such an analysis. The most intuitive and direct method of finding a solution to this problem requires its conversion into the eigenvalue equation  $f(z) = 0$  and complex root computations. The paper is organized as follows: In Section II, the actual state of the art in root finding techniques and multimodal genetic algorithms (GAs) is presented. In Section III, the multimodal GA with phase analysis (MGA-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

In this section, root finding techniques and GAs dedicated to multimodal optimization problems are presented. Let us

start considerations from Newton's method which is one of the oldest root finding techniques [1]. This method requires that a function and its derivatives are continuous in a search space. Moreover, the starting point for searching must be located close to a solution. Bisection, secant [2] and Muller's [3] methods do not require function derivatives and converge by iteratively sampling a function in two or three points and consequently approximating the next sampling point. These methods are efficient when the initial starting point is located close to a root. Moreover, these methods are able to locate only a single root for the assumed starting point.

Finding multiple roots on the complex plane is a far more difficult task. Recently, a very efficient global complex roots and poles finding (GPRF) algorithm based on the Delaunay triangulation and the function phase analysis has been proposed [4], [5]. This algorithm generates a regular triangular mesh of nodes and the complex function argument (i.e., phase quadrant) is computed in each node position. Hence, a phase quadrant distance between adjacent nodes can be computed in the next step. If its absolute value is equal to two, i.e., signs of real and imaginary parts of adjacent nodes are opposite, a region around these nodes is considered as a vicinity of possible root or pole. Such a connection between nodes is referred to as a candidate edge. Then, discretized Cauchy's argument principle (DCAP) is applied over candidate regions to evaluate if any root/pole is indeed inside. The complex border tracking (CBT) method, that also employs the tracking of complex function sign changes, is proposed in [6]. This method has a very low numerical cost but it cannot track entire regions on the complex plane due to the application of selective sampling. The novel stochastic population based concept merging the particle swarm optimization (PSO) with benefits of the function phase analysis is proposed in [7]. The stochastic nature of this method prevents from root or pole omitting in subsequent iterations. In this contribution, we compare mentioned above GPRF, CBT, MPSO-WPA methods with proposed GA demonstrating their advantages and disadvantages.

GA was originally proposed by Holland in 1975 [8]. In

this approach, the mathematical model of biological evolution is employed to solve an optimization problem. The algorithm is based on binary operations on chromosomes of individuals (i.e., population members). The chromosomes are moved to a new population by genetic-based operators of selection, crossover and mutation. The selection operator chooses the chromosomes in the population that are reproduced, the crossover operator exchanges parts of chromosomes between individuals and the mutation operator changes alleles in chromosomes. GA which does not require changing of a search space into its binary representation and operates in the continuous domain is referred to as Breeder GA (BGA) [9]. The crossover and mutation operators are main genetic operators for continuous functions. The mutation scheme in two-dimensional search space generates new population members in a square area (hypercube in a multidimensional search space) defined by coordinates of a mutated individual and a mutation range. The crossover scheme generates new population members in a rectangular area between candidates with most promising chromosomes. Benchmarks indicate that BGA outperforms classical GA in continuous search problems [10]. Moreover, the algorithm is very intuitive, easy to implement and applicable to many optimization problems. The original algorithm aims at finding a single best solution of fitness function and cannot deal with multimodal searching with multiple good solutions. Therefore, the original algorithm must be extended by a set of rules allowing for the population diversity and precise exploration of distant regions on the complex plane. Various approaches to diversify population can be found in [11]. Some algorithms diversify by generating multiple, independent populations that do not interact with each other, whereas some of algorithms are focused on a single population clustering and restriction of interactions between individuals to closest neighbours only. For instance, the island model [12] runs multiple subpopulations in parallel. On the other hand, the cellular GA [13] concentrates on population clustering.

In this paper, BGA is mixed with the selection of individual candidates based on the analysis of function phase. Therefore, the population diversity stems from the spatial segregation around candidate regions. Information obtained from the candidate selection is binary (i.e., an individual is either a candidate or not), thus, a new population is easily diversified and generated only in regions of root or pole occurrences. The mutated or crossed individuals are not replaced by new individuals, thus, the population size increases in each iteration step.

### III. THE MGA-WPA ALGORITHM

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

#### A. Algorithm Initialization and Generation of Initial Population Members

Initialize algorithm parameters such as the algorithm accuracy  $\epsilon$ , the size of search region, the size of initial population,

the mutation range  $R_M$ , the mutation operator  $K_M$ . Then, generate the uniformly distributed random population on the complex plane

$$Z = \{z_1, z_2, \dots, z_n\} \quad (1)$$

where  $n$  is the population size. Initially,  $n$  is set to the size of initial population.

#### B. Search for Regions with Potential Root or Pole Occurrence

For each individual population member, whose coordinates on the complex plane are denoted as  $z_i$  ( $i = 1, 2, \dots, n$ ), evaluate the fitness function argument and compute the phase quadrant in which the corresponding function value is located

$$Q_i = \left\{ \begin{array}{l} 1, \quad 0 \leq \arg f(z_i) < \frac{\pi}{2} \\ 2, \quad \frac{\pi}{2} \leq \arg f(z_i) < \pi \\ 3, \quad \pi \leq \arg f(z_i) < \frac{3\pi}{2} \\ 4, \quad \frac{3\pi}{2} \leq \arg f(z_i) < 2\pi \end{array} \right\}. \quad (2)$$

Then, apply the Delaunay triangulation to coordinates  $z_i$ , i.e., generate triangular connections between population members. Compute the phase quadrant distance along each of the connections

$$\Delta Q_p = Q_{p2} - Q_{p1}. \quad (3)$$

Search for connections such as  $|\Delta Q_p| = 2$ . Individuals connected in this way must have different signs of the real and imaginary parts of the function values. These connections called candidates are considered as a potential vicinity of either root or pole. If there is no root nor pole on the complex plane, candidate connections are not found and the algorithm execution is terminated. If at least a single candidate connection is found, then the candidate-connection length is computed. If the highest length of the collected candidate connections is less than the assumed accuracy, then the algorithm proceeds to the step D. In other cases, the algorithm proceeds to the next step C.

#### C. Crossover and Mutation of Population

New members of population are generated by the genetic operations of crossover and mutation. The genetic operations are executed on coordinates of candidate connections. The crossover is executed on two individuals that form a candidate connection. Then, the coordinates of a new individual resulting from the crossover are given by

$$z_{n+i} = Re(z_{i,1} + a_i(z_{i,2} - z_{i,1})) + jIm(z_{i,1} + b_i(z_{i,2} - z_{i,1})) \quad (4)$$

where  $i = 1, 2, \dots, k$ ,  $z_{n+i}$  denotes the new individual coordinates,  $z_{i,1}$  denotes the coordinates of the first individual within the  $i$ -th candidate connection,  $z_{i,2}$  denotes the coordinates of the second individual within the  $i$ -th candidate connection,  $a_i$  and  $b_i$  are random numbers between 0 and 1,  $n$  is the number of individuals in the current population and  $k$  is the number of candidate connections detected in the current iteration. The mutation operation is executed by the mutation of coordinates of candidate-connection centres. Hence, the

subsequent individual coordinates resulting from the mutation are given by

$$z_{n+k+i} = Re(s_i + 0.5|z_{i,1} - z_{i,2}|R_M c_i m_i) + jIm(s_i + 0.5|z_{i,1} - z_{i,2}|R_M d_i m_i) \quad (5)$$

$$s_i = 0.5(z_{i,1} + z_{i,2}) \quad (6)$$

$$m_i = 2^{-u_i K_M} \quad (7)$$

where  $i = 1, 2, \dots, k$ ,  $z_{n+k+i}$  denotes the new individual coordinates,  $z_{i,1}$  denotes the coordinates of the first individual within the  $i$ -th candidate connection,  $z_{i,2}$  denotes the coordinates of the second individual within the  $i$ -th candidate connection,  $s_i$  denotes the coordinates of the  $i$ -th candidate-connection center,  $c_i$  and  $d_i$  are random numbers between -1 and 1,  $R_M$  denotes the mutation range,  $u_i$  is a random number between 0 and 1,  $K_M$  denotes the mutation precision and  $k$  is the number of candidate connections. As a result of the crossover and mutation operations, new population members are generated. The number of generated new individuals depends on the number of detected candidate connections. For each candidate connection, two individuals (i.e., one crossed and one mutated) are generated, hence, the population size  $n$  is increased by  $2k$  in each iteration. After the generation of new population members, the algorithm is looped to the searching for candidates step B.

#### D. Verification of root or pole occurrence

Confirm the existence of root/pole applying DCAP by integrating quadrant differences along the path between population members around the candidate region

$$q = \frac{1}{4} \sum_{p=1}^P \Delta Q_p. \quad (8)$$

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. If multiple roots/poles are located inside the region of integration, then the parameter  $q$  is multiple of the number of roots/poles found.

## IV. NUMERICAL RESULTS

The propagation of surface waves in spatially dispersive graphene sheet decomposed on a dielectric silicon substrate is the electromagnetic problem under analysis [14]. The dispersion coefficients for TM modes propagating along the graphene sheet are obtained by solving the equation  $f(z) = 0$  for the following function:

$$f(z) = Y_1^{TM} + Y_2^{TM} + Y_\sigma^{TM} \quad (9)$$

$$Y_1^{TM} = \frac{\omega \varepsilon_{r1} \varepsilon_0}{k_{z1}} \quad (10)$$

$$Y_2^{TM} = \frac{\omega \varepsilon_{r1} \varepsilon_0}{k_{z2}} \quad (11)$$

$$Y_\sigma^{TM} = \sigma_{lo} + k_p^2 \alpha_{sd} - k_p^2 \beta_{sd} \quad (12)$$

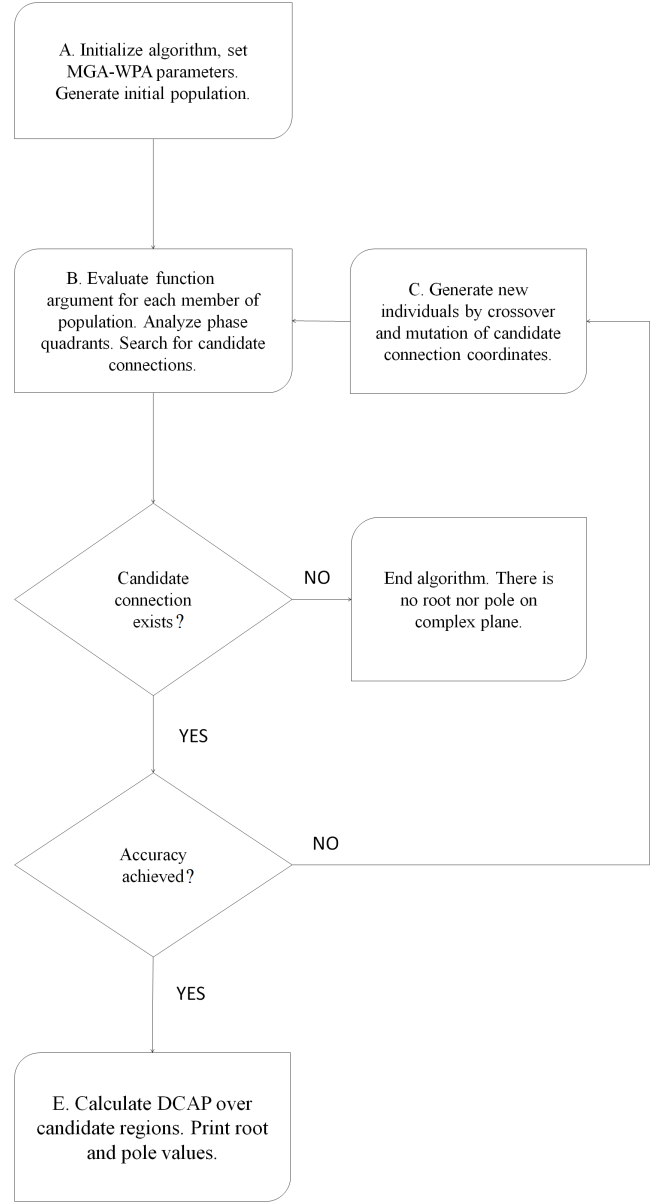


Fig. 1. Flowchart of the MGA-WPA algorithm.

$$k_{z1} = \pm \sqrt{k_1^2 - k_p^2} \quad (13)$$

$$k_{z2} = \pm \sqrt{k_2^2 - k_p^2} \quad (14)$$

$$k_1 = \sqrt{\varepsilon_{r1}} k_0 \quad (15)$$

$$k_2 = \sqrt{\varepsilon_{r2}} k_0 \quad (16)$$

$$k_p = -izk_0 \quad (17)$$

$$\sigma_{lo} = \frac{-iq_e^2 k_B T}{\pi \hbar (\omega - i\tau^{-1})^2} \ln \left( 2 + 2 \cosh \left( \frac{\mu_C}{k_B T} \right) \right) \quad (18)$$

$$\alpha_{sd} = \frac{-3v_F^2 \sigma_{lo}}{4(\omega - i\tau^{-1})^2} \quad (19)$$

$$\beta_{sd} = \frac{\alpha_{sd}}{3}. \quad (20)$$

In (9)–(20),  $k_0 = 2\pi f/c$  is the free space wave number,  $\varepsilon_0$  is the permittivity of free space,  $\varepsilon_{r1}$  and  $\varepsilon_{r2}$  are the relative permittivities of the surrounding media,  $q_e$  is the electron charge,  $v_F$  is the Fermi velocity,  $k_B$  is the Boltzmann constant and  $\hbar$  is the reduced Planck constant. The graphene parameters are the chemical potential  $\mu_c = 0.05q_e$  and the electron relaxation time  $\tau = 0.135$  psec. The relative permittivity of silicon substrate is equal to 11.9. The temperature is equal to  $T = 300$  K. Finding roots and poles is executed for the frequency equal to  $f = 1$  THz. The search region is given by  $-100 \leq \text{real}(z) \leq 500$  and  $-100 \leq \text{imag}(z) \leq 500$ . The considered complex function  $f(z) = 0$  is a pointwise product of four Riemann sheets [15], hence, function discontinuities are eliminated and roots remain the same values.

The MGA-WPA code is executed on a personal computer equipped with Intel i7-4700MQ processor. The computation time, the number of iterations and the number of individuals necessary for the convergence vary for each run of the code. It is caused by the stochastic nature of implemented GA. Initial and final population distributions for the exemplary run of the MGA-WPA algorithm are respectively presented in Figs. 2 and 3. The initial population size is set to 1037 individuals.

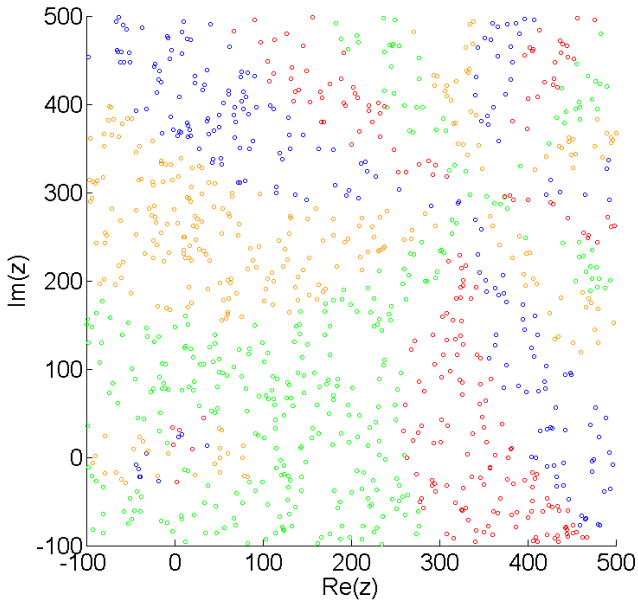


Fig. 2. Initial population distribution (1037 individuals) generated by MGA-WPA for analysis of propagation in spatially dispersive graphene in bounded region  $-100 \leq \text{real}(z) \leq 500$  and  $-100 \leq \text{imag}(z) \leq 500$ . Phase quadrants of population members on the  $z$ -plane:  $\bullet$  Q=1,  $\bullet$  Q=2,  $\bullet$  Q=3,  $\bullet$  Q=4.

The total number of generated individuals to converge with the accuracy  $\varepsilon = 1E-6$  is 2813 individuals. The code is executed 49 times in the loop to achieve required convergence. The execution time is 0.66 sec. The algorithm finds eight single roots as follows:

$$\begin{aligned} z(1) &= -38.17772 - 32.52952i \\ z(2) &= -32.10196 - 27.43086i \\ z(3) &= 332.7449 + 282.2431i \end{aligned}$$

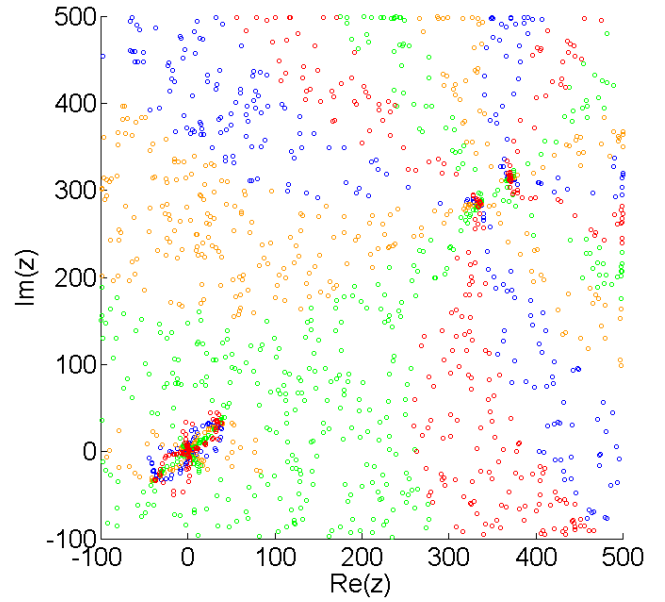


Fig. 3. Final population distribution (2813 individuals) generated by MGA-WPA for analysis of propagation in spatially dispersive graphene in bounded region  $-100 \leq \text{real}(z) \leq 500$  and  $-100 \leq \text{imag}(z) \leq 500$ . Phase quadrants of population members on the  $z$ -plane:  $\bullet$  Q=1,  $\bullet$  Q=2,  $\bullet$  Q=3,  $\bullet$  Q=4.

$$\begin{aligned} z(4) &= 336.2203 + 285.1911i \\ z(5) &= 371.0076 + 314.7004i \\ z(6) &= 368.4395 + 312.5221i \\ z(7) &= 38.17772 + 32.52952i \\ z(8) &= 32.10196 + 27.43086i. \end{aligned}$$

In the analysed region, the algorithm is able to identify the following two second-order poles:

$$\begin{aligned} z(9) &= (-3.2867E-08) - 3.4496i \\ z(10) &= (-8.2787E-08) + 3.4496i. \end{aligned}$$

For the sake of comparison, the GPRF method is evaluated which returns the same root and pole values as the MGA-WPA method. The region analysed in this example is computationally difficult because the roots and poles are concentrated close to each other. Moreover, two clusters of roots and poles are located far away from each other and almost in opposite corners of the search region. In this case, the initial population size must be large enough to avoid overlapping of candidate connections. It is worth noticing that the mesh obtained from the Delaunay triangulation is very irregular and fine in the vicinity of any root or pole. The mesh for the final population and the zoomed area around roots and poles are respectively presented in Figs. 4 and 5. The proposed MGA-WPA method is based on a stochastic generator, hence, each code run is different. Therefore, to obtain a reasonable evaluation results, mean values have to be computed from several code runs. In Table I, mean values from 5 runs for various algorithm accuracies are presented for the final population size, the algorithm execution time and the number of algorithm iterations. In Table II, the computational results obtained for the same problem

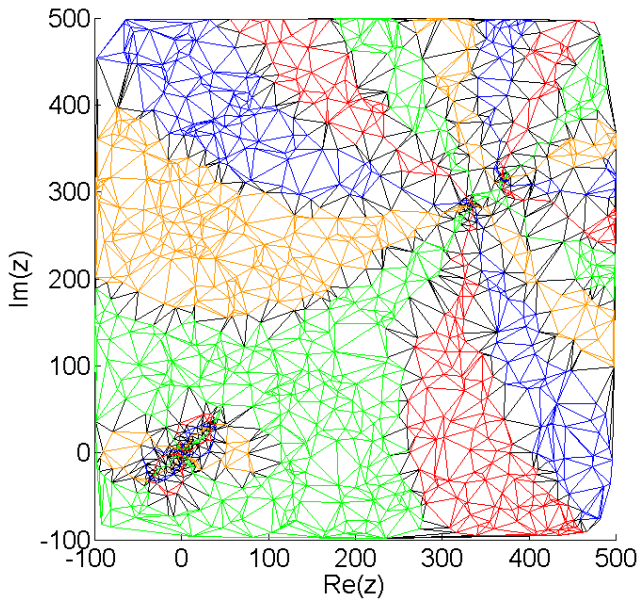


Fig. 4. Delaunay triangulation applied to final population members in bounded region  $-100 \leq \text{real}(z) \leq 500$  and  $-100 \leq \text{imag}(z) \leq 500$ .

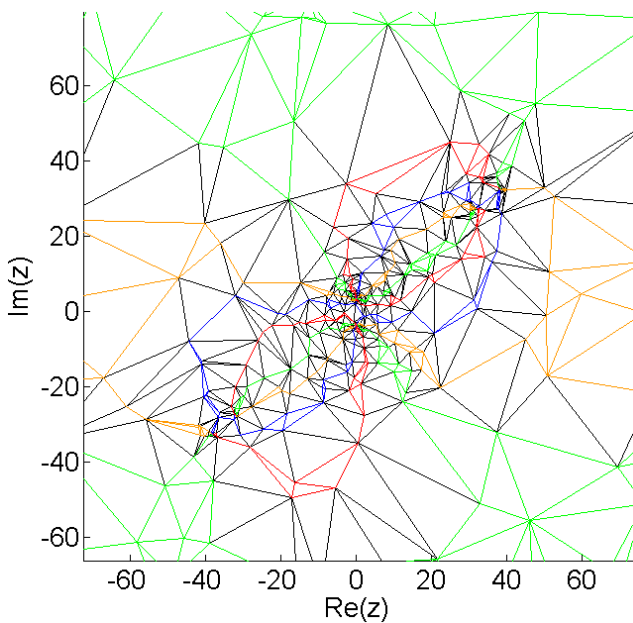


Fig. 5. Zoomed irregular mesh generated by MGA-WPA around four roots and two second-order poles.

with the use of the GPRF method are presented. The number of individuals necessary for the convergence of the MGA-WPA algorithm is much lower than for the GPRF method, especially when a high accuracy is necessary. However, MGA-WPA is slower than GPRF, because the MGA-WPA algorithm loop is iterated more times. Moreover, the stochastic nature of MGA-WPA requires additional control if individuals are not duplicated. The comparison of recently published root/pole finding methods that employ the phase analysis of complex

TABLE I  
POPULATION SIZE, EXECUTION TIME AND NUMBER OF LOOP ITERATIONS FOR VARYING ACCURACY OF MGA-WPA

Accuracy	Population size	Ex. time (sec)	Number of iterations
$1E-03$	2258	0.38	33
$1E-06$	2904	0.71	55
$1E-09$	3542	1.01	74
$1E-12$	4157	1.64	97

TABLE II  
NUMBER OF NODES, EXECUTION TIME AND NUMBER OF LOOP ITERATIONS FOR VARYING ACCURACY OF GPRF

Accuracy	Number of nodes	Ex. time (sec)	Number of iterations
$1E-03$	2545	0.23	16
$1E-06$	3418	0.4	26
$1E-09$	4305	0.68	36
$1E-12$	5215	0.97	46

TABLE III  
COMPARISON OF METHODS THAT EMPLOY PHASE ANALYSIS OF COMPLEX FUNCTION TO FIND ROOTS AND POLES

Method	Advantages	Disadvantages
CBT [6]	- very low numerical cost	- algorithm is not able to track roots/poles located on "island" like regions
GPRF [4]	- very fast and efficient	- requires initial number of nodes to be large enough to detect all candidate edges
MPSO-WPA [7]	- stochastic function exploration in subsequent iterations, prevents from omitting roots or poles	- the final number of particles and algorithm execution time may be large, especially when high accuracy is necessary
MGA-WPA	- a small number of individuals is sufficient to converge	- requires initial population size to be large enough to detect all candidate connections

function is presented in Table III.

## V. CONCLUSION

The paper proposes the fast and efficient MGA-WPA algorithm to solve complex equations of the electromagnetic analysis. The method merges GA with the phase quadrant analysis on the complex plane. A small number of individuals is sufficient for the convergence of the method, limiting final population size. The results obtained with the use of the MGA-WPA algorithm are satisfactory. The method may not find root or pole if the initial population members are separated by a large distance. However, the stochastic nature of MGA-WPA increases the chance of finding all roots and poles, when the algorithm is executed multiple times.

## REFERENCES

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

- [2] M. B. Allen and E. L. Isaacson, *Numerical Analysis for Applied Science*. John Wiley and Sons, 1972.
- [3] W. H. Press, B. P. Flannery, and S. A. Teukolsky, *Numerical Recipes in Fortran 77: The Art of Scientific Computing*. Cambridge University Press, 1992.
- [4] 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, p. 71987205, 2018.
- [5] —, "Complex root finding algorithm based on Delaunay triangulation," *ACM Trans. Math. Softw.*, vol. 41, no. 3, pp. 1–13, 2015.
- [6] J. J. Michalski, "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.
- [7] D. Trofimowicz and T. P. Stefański, "Multimodal particle swarm optimization with phase analysis to solve complex equations of electromagnetic analysis," in *Proceedings of 2020 23rd International Conference on Microwaves, Radar Wireless Communications*, 2020.
- [8] J. H. Holland, *Adaptation in Natural and Artificial Systems*. MIT Press, 1975.
- [9] H. Mhlenbein and D. Schlierkamp-Voosen, "Predictive models for the breeder genetic algorithm i. continuous parameter optimization," *Evol. Comput.*, vol. 1, no. 1, pp. 25–49, 1993.
- [10] I. D. Falco, R. D. Balio, A. D. Cioppa, and E. Tarantino, "A comparative analysis of evolutionary algorithms for function optimisation," in *Proceedings of 2nd Workshop on Evolutionary Computing (WEC2)*, 1996, pp. 29–32.
- [11] N. Casas, "Genetic algorithms for multimodal optimization: a review," *arXiv:1508.05342*, 2015.
- [12] D. Izzo, M. Ruciski, and F. Biscani, "The generalized island model," *Studies in Computational Intelligence*, vol. 415, pp. 151–169, 2012.
- [13] E. Alba and B. Dorronsoro, "Cellular genetic algorithms," *Operations Research/ Computer Science Interfaces Series*, vol. 42, pp. 1–242, 2008.
- [14] J. S. Gomez-Diaz, J. R. Mosig, and J. Perruisseau-Carrier, "Effect of spatial dispersion on surface waves propagating along graphene sheets," *IEEE Trans. Antennas Propag.*, vol. 61, no. 7, pp. 3589–3596, 2013.
- [15] P. Kowalczyk, "On root finding algorithms for complex functions with branch cuts," *J. Comput. Appl. Math.*, vol. 314, 2016.