Progress In Electromagnetics Research, PIER 77, 181–192, 2007

AN EFFICIENT HYBRID SWARM INTELLIGENCE-GRADIENT OPTIMIZATION METHOD FOR COMPLEX TIME GREEN'S FUNCTIONS OF MULTILAYER MEDIA

M. Ghaffari-Miab, A. Farmahini-Farahani, R. Faraji-Dana and C. Lucas

School of Electrical and Computer Engineering
University of Tehran
P. O. Box: 14395-515, Tehran, Iran

Abstract—A new hybrid technique for optimization of a multivariable function is proposed. This method is applied to the problem of complex time Green's function of multilayer media. This technique combines Particle Swarm search algorithm with the gradient based quasi-Newton method. Superiority of the method is demonstrated by comparing its results with other optimization techniques.

1. INTRODUCTION

A group of popular and powerful approaches for search and optimization problems are Evolutionary Algorithms (EAs) [1, 2]. During the past decades, EAs have been widely used in many engineering applications [3, 4], and recently in electromagnetic problems [5–9]. Particle Swarm Optimization (PSO) is one of the evolutionary computation techniques. It was developed to simulate a simplified social system [10]. PSO is a powerful and promising optimization method which has a wide range of applications in engineering optimization including electromagnetic and antenna design problems [11, 12]. Like the other evolutionary computation techniques, PSO is a population-based search algorithm which is initialized with a population of random solutions, called particles. Unlike other evolutionary computation techniques, each particle in PSO has its own velocity. Particles fly through the search space with velocities which are dynamically adjusted according to their historical behaviors. Therefore, the particles have a tendency to fly toward a better region of search space over the course of search process. It is also shown that PSO converges faster than genetic algorithm (GA) [13].

In this paper, a hybrid PSO-Gradient technique is used for optimization of complex time Green's functions expansion in multilayer media.

The remaining sections of this paper are organized as follows: Section 2 reviews the Green's function of a grounded dielectric slab. Section 3 describes the PSO, gradient based algorithms and our hybrid methodology. The achieved results are explained in Section 4. Finally, section 5 concludes the paper.

2. GREEN'S FUNCTION OF A GROUNDED DIELECTRIC SLAB

The fast and efficient time domain analysis of microstrip structures is one of the most challenging problems in microwave planar circuit analysis. A closed-form wideband representation of the spatial Green's functions for a horizontal electric dipole (HED) over a grounded dielectric slab was first developed in [14, 15]. In this representation the Green's function is given in the form of a series expansion valid for a wide range of frequencies in contrast to the conventional spatial complex images which is only valid at a single frequency.

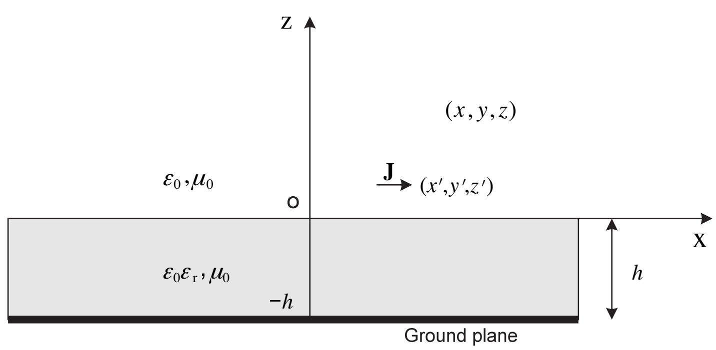

Figure 1. A HED over a grounded dielectric slab.

The geometry of the problem is depicted in Fig. 1. Vector and scalar Green's functions in the air region is needed for analyzing this structure using conventional Mixed Potential Integral Equation (MPIE) technique. The spectral domain Green's functions for magnetic vector potential, $\tilde{G}_A^{xx}$, and electrical scalar potential, $\tilde{G}_q$, can be represented as [16]

$$\tilde{G}_A^{xx} = \frac{\mu_0}{4\pi}\frac{1}{j2k_{z0}}\left[e^{-jk_{z0}(z-z')} + R_{TE}e^{-jk_{z0}(z+z')}\right] \quad (1a)$$

$$\tilde{G}_q = \frac{1}{4\pi\varepsilon_0}\frac{1}{j2k_{z0}}\left[e^{-jk_{z0}(z-z')} + (R_{TE}+R_q)e^{-jk_{z0}(z+z')}\right] \quad (1b)$$

where,

$$R_{TE} = -\frac{r_{10}^{TE} + e^{-j2k_{z1}h}}{1 + r_{10}^{TE}e^{-j2k_{z1}h}} \quad (2a)$$

$$R_q = \frac{2k_{z0}^2(1-\varepsilon_r)(1-e^{-j4k_{z1}h})}{(k_{z1}+k_{z0})(k_{z1}+\varepsilon_r k_{z0})(1+r_{10}^{TE}e^{-j2k_{z1}h})(1-r_{10}^{TM}e^{-j2k_{z1}h})} \quad (2b)$$

$$r_{10}^{TE} = \frac{k_{z1}-k_{z0}}{k_{z1}+k_{z0}}, \quad r_{10}^{TM} = \frac{k_{z1}-\varepsilon_r k_{z0}}{k_{z1}+\varepsilon_r k_{z0}} \quad (3)$$

$$k_{z1} = \sqrt{(\varepsilon_r-1)k_0^2 + k_{z0}^2}, \quad \mathrm{Im}ag\{k_{z1}\} \leq 0 \quad (4a)$$

$$k_{z0} = \sqrt{k_0^2 - k_\rho^2}, \quad \mathrm{Im}ag\{k_{z0}\} \leq 0 \quad (4b)$$

The corresponding spatial domain Green's functions are the inverse Hankel transform of (1). These inverse transforms are in the form of Sommerfeld's integrals. They are as follows:

$$G_A^{xx} = \frac{\mu_0}{4\pi}\left(\frac{e^{-jk_0r_0}}{r_0} + \int_{-\infty}^{\infty}\frac{1}{j2k_{z0}}R_{TE}e^{-jk_{z0}(z+z')}H_0^{(2)}(k_\rho\rho)k_\rho dk_\rho\right) \quad (5a)$$

$$G_q = \frac{1}{4\pi\varepsilon_0}\left(\frac{e^{-jk_0r_0}}{r_0} + \int_{-\infty}^{\infty}\frac{1}{j2k_{z0}}(R_{TE}+R_q)e^{-jk_{z0}(z+z')}H_0^{(2)}(k_\rho\rho)k_\rho dk_\rho\right) \quad (5b)$$

where, $r_0 = \sqrt{\rho^2 + (z-z_0)^2}$.

Because of oscillatory behavior of these types of integrals, their numerical computation is time consuming. It can be shown that the following series approximation of R_{TE} and R_q could obviate the need of numerical calculation of these integrals. These series are also valid in a wide range of frequencies from 0 to f_0. This representation has the interesting meaning of complex time [14, 17]:

$$R_{TE} - R_{TE0} \cong \sum_{n=1}^{N} a_n e^{b_n k_{z0}} e^{c_n k_0}, \ R_{TE0} = \lim_{k_0 \to 0} R_{TE} = -e^{-j2hk_{z0}} \quad (6a)$$

$$R_q - R_{q0} \cong \sum_{n=1}^{N} a'_n e^{b'_n k_{z0}} e^{c'_n k_0}, \ R_{q0} = \lim_{k_0 \to 0} R_q = \frac{K(1 - e^{-j4k_{z0}h})}{1 - Ke^{-j2k_{z0}h}} \quad (6b)$$

where,

$$k_0 = \frac{2\pi f}{c_0}, \ c_0 = 3 \times 10^8, \ K = \frac{1 - \varepsilon_r}{1 + \varepsilon_r} \quad (7)$$

$$k_{z0} = k_0 \left[-ju + (1 - \frac{u}{u_0}) \right] \ 0 \leq u \leq u_0 \quad (8)$$

The a_n, b_n, c_n coefficients must be chosen in such a way that the two dimensional exponential expansion of (6) has a small error over the rectangular area of $0 \leq u \leq u_0$, $0 < f \leq f_0$. Therefore, the fitness function could be represented as follows.

$$\sum_{p=1}^{20} \sum_{q=0}^{10} f \cdot \left| (R_{TE} - R_{TE0}) - \sum_{n=1}^{N} a_n e^{b_n k_{z0}} e^{c_n k_0} \right|_{\substack{f = \frac{f_0}{20} p \\ u = \frac{u_0}{10} q}} \quad (9a)$$

$$\sum_{p=1}^{20} \sum_{q=0}^{10} f \cdot \left| (R_q - R_{q0}) - \sum_{n=1}^{N} a'_n e^{b'_n k_{z0}} e^{c'_n k_0} \right|_{\substack{f = \frac{f_0}{20} p \\ u = \frac{u_0}{10} q}} \quad (9b)$$

With substituting (6) into (5) and using Sommerfeld's identity one can simply attain the following representations for the spatial domain Green's functions valid for $0 < f \leq f_0$.

$$G_A^{xx} \cong \frac{\mu_0}{4\pi} \left(\frac{e^{-jk_0 r_0}}{r_0} - \frac{e^{-jk_0 r'_0}}{r'_0} + \sum_{n=1}^{N} a_n e^{c_n k_0} \frac{e^{-jk_0 R_n}}{R_n} \right) \quad (10a)$$

$$\begin{aligned} G_q \cong \frac{1}{4\pi\varepsilon_0} \Bigg(& \frac{e^{-jk_0 r_0}}{r_0} + K \frac{e^{-jk_0 r''_0}}{r''_0} + K^2 \frac{e^{-jk_0 r_1}}{r_1} - K \frac{e^{-jk_0 r_2}}{r_2} \\ & - K^2 \frac{e^{-jk_0 r_3}}{r_3} + \sum_{n=1}^{N} a_n e^{c_n k_0} \frac{e^{-jk_0 R_n}}{R_n} + \sum_{n=1}^{N} a'_n e^{c'_n k_0} \frac{e^{-jk_0 R'_n}}{R'_n} \Bigg) \end{aligned} \quad (10b)$$

where, $R_n = \sqrt{\rho^2 + (z + z' + jb_n)^2}$, $R'_n = \sqrt{\rho^2 + (z + z' + jb'_n)^2}$.
$r'_0 = \sqrt{\rho^2 + (z + z' + 2h)^2}$, $r''_0 = \sqrt{\rho^2 + (z + z')^2}$,
$r_n = \sqrt{\rho^2 + (z + z' + 2nh)^2}$; $n = 1, 2, 3, \ldots$.

3. HYBRID SWARM INTELLIGENCE-GRADIENT OPTIMIZATION TECHNIQUE

3.1. Particle Swarm Algorithm

The Particle Swarm Optimizer is a swarm intelligence algorithm that emulates a flock searching over the solution landscape. This algorithm uses sampling points in search space and the swarm converges to the most promising regions. A particle moves through the solution space along a trajectory defined by its velocity [18]. The basic structure of a particle in PSO is significantly more complex than a member of a GA population. A particle consists of five components:

- $\boldsymbol{x}$, a vector containing the current location in the solution space. The size of $\boldsymbol{x}$ is dictated by the number of variables used by the problem being solved.
- $\boldsymbol{fitness}$, the quality of the solution represented by the vector $\boldsymbol{x}$, as computed by a problem-specific evaluation function.
- $\boldsymbol{v}$, a vector containing the velocity for each particle. The velocity of a particle indicates the changes of the corresponding $\boldsymbol{x}$ vector (particle location) in the next iteration. Altering the $\boldsymbol{v}$ vector values assigns the direction that the particle will move through the search space.
- ***gbest***, is the fitness value of the best solution attained by a particular particle. Each particle keeps track of its coordinates in the problem space, which are associated with the best solution (fitness) it has achieved so far.
- ***pbest***, is a copy of the $\boldsymbol{x}$ for the location of the best solution achieved by a particular particle.

Each particle is also aware of *gbest*, the particle reporting the current best fitness in the neighborhood for any given iteration. A neighborhood may consist of some small group of particles. Alternately, the entire swarm may be considered a single neighborhood, and *gbest* applies globally (global PSO). Also, $fitness_g$ is the fitness value of the *gbest*.

The PSO begins with a random population and searches for best fitness just like the genetic algorithm (GA), but in the PSO algorithm, particles will evolve by cooperation and competition among the individuals through generations instead of using genetic operators [19].

The heart of the PSO algorithm is the process by which $\boldsymbol{v}$ is modified, forcing the particles to search through the most promising areas of the solution space again and again. On each iteration, the previous values of $\boldsymbol{v}$ constitute the particle's momentum. This momentum is essential, as it is the feature of PSO that allows particles to escape local extremums.

At each time step, the velocity of each particle changes toward the *pbest* and *gbest* locations. Moving to these locations is weighted by a random term, with separate random numbers being generated for acceleration toward *pbest* and *gbest* locations. This randomness insures that the step size will be varying to avoid aliasing and insure the particle doesn't "fall into a rut", where the particle endlessly follows the exact same path. The modified velocity and location of each individual particle can be calculated using the current velocity and the distance from $pbest_i$ to *gbest*, as shown in the following formula:

$$v_i^{k+1} = wv_i^k + c_1 r_1 (pbest_i - x_i^k) + c_2 r_2 (gbest - x_i^k) \tag{11}$$

$$x_i^{k+1} = x_i^k + v_i^{k+1} \tag{12}$$

where x_i^k is the current location of individual i at iteration k, which has v_i^k as velocity. This velocity satisfies $V^{\min} \leq v_i^k \leq V^{\max}$. *pbest* is the historical best location of x_i^k and *gbest* is the global best location in the population's history. Besides, there are five parameters that should be defined, w is the inertia weighting factor, c_1 and c_2 are acceleration constants and r_1 and r_2 are random number between 0 and 1 with uniform distribution. The evolution process generally begins with a random distribution of particles and evolutes as the formulation (11), (12).

In the above procedures the parameter $V_{\max}$ determines the resolution, or fitness, with which regions between the present location and target location are searched. If $V_{\max}$ is too high, particles may fly over the good solutions. If $V_{\max}$ is too small, particles may not sufficiently explore beyond local solutions. In previous experience with PSO, $V_{\max}$ was often set at 10–20% of the dynamic range of the variable on each dimension.

The constants c_1 and c_2 represent the weighting of the stochastic acceleration terms that pull each particle toward *pbest* and *gbest* locations. Low values allow particles to roam far from target regions before being tugged back. On the other hand, high values result in abrupt movement toward, or past, the target regions. Hence, the acceleration constants c_1 and c_2 were often set to be 2.0 according to past experiences.

The complete algorithm for the particle swarm optimizer is shown in Fig. 2.

```
Initialize the swarm:
  for each particle i,
    Set x_i to a random value
    Set pbest to x_i
    Set v_i to a random value on the range [V_min , V_max]
  end for

Perform the search:
  until a terminating condition is met
    for each iteration k,
      for each particle i,
        compute fitness_i
      update gbest and fitness_g if fitness_i is better than fitness_g
      update pbest and fitness_p if fitness_i is better than fitness_p
      compute v_i (Equation (11))
      compute x_i (Equation (12))
      end for
    end for
  end while
Report results
```

Figure 2. The basic PSO algorithm.

3.2. BFGS Quasi-Newton Method

An efficient gradient based method for calculating the minimum of a multivariable function is quasi-Newton method in which we start from an arbitrary initial point and approximate the function near that point with a quadratic model as follows:

$$\frac{1}{2}x^T Hx + c^T x + b \tag{13}$$

where H is the Hessian matrix, c is a constant vector and b is a constant scalar. The minimum of this approximated function is:

$$x_o = -H^{-1}c \tag{14}$$

x_0 is the starting point of the next iteration. This procedure continues until the gradient of the function reaches zero and hence the local optimum of the function corresponding to the first starting point is achieved.

Since calculating the exact Hessian matrix on each iteration is time-consuming, the BFGS method approximates this matrix by the following update equation [20–23]:

$$H_{k+1} = H_k + \frac{q_k q_k^T}{q_k^T s_k} - \frac{H_k^T s_k^T s_k H_k}{s_k^T H_k s_k} \tag{15}$$

where,

$$s_k = x_{k+1} - x_k,\ q_k = \nabla f(x_{k+1}) - \nabla f(x_k) \tag{16}$$

This method efficiently minimizes the computation costs.

3.3. Hybrid Approach

The gradient based methods suffer from sensitivity to starting point of the algorithm. Different starting points may lead to different local optima. To avoid this problem, we run the BFGS Quasi-Newton method with different starting points given by the PSO method. The Block diagram of this hybrid approach is depicted in Fig. 3. In other words, we apply the PSO algorithm to a new function which has the information of all local optima of the original function. So this method searches among the local optima.

4. NUMERICAL RESULTS

The microstrip structure of Fig. 1 is examined with $h = 1\,\text{mm}$, $\varepsilon_r = 12.6$. For many applications $f_0 = 20\,\text{GHz}$ and $u_0 = 10$ are good choices, so they are chosen for the simulation. As we expect, with increasing the number of images N, the proposed algorithm leads to better approximations but simulation time increases. Numerical results shows that $N = 4$ which leads to 24 independent variables, gives good results. The proposed Hybrid PSO-Gradient method is applied to the fitness functions of the structure which are represented by Equations (9a) and (9b). Also, simple genetic algorithm, invasive weed optimization (IWO) [24] and simple PSO algorithm are applied to this problem for comparison. As could be seen from Table 1, the proposed method can find better solutions in comparison with other algorithms. Also the proposed method finds considerably better solutions in comparison with the previous results [14, 15].

Figure 4 shows the magnitude of the magnetic vector potential (10a) versus horizontal distance at four different frequencies for $z = z' = 0$ using the coefficients given by the proposed method (dashed lines). The direct numerical computation of Sommerfeld's integral (5a) is also shown in this figure (solid lines). It can be seen that the difference between the closed-form Green's function and numerical integration is nearly unobservable.

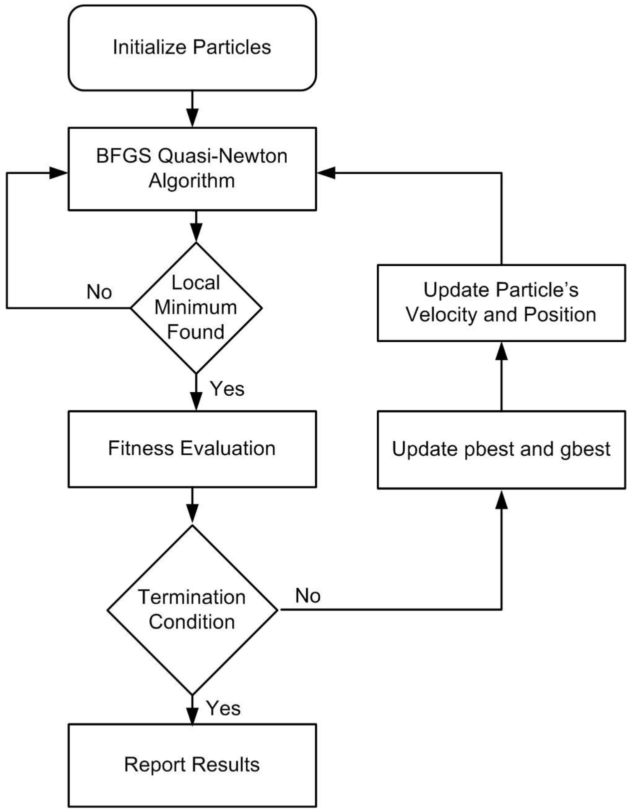

Figure 3. Block diagram of the proposed hybrid PSO-gradient method.

Table 1. Best fitness value achieved from different methods.

Method	Best fitness value R_{TE}	R_q
GA	91.7	124
IWO	86.2	115
PSO	64.9	93.9
References [14, 15]	11.8	40.8
PSO-Gradient	7.78	22.9

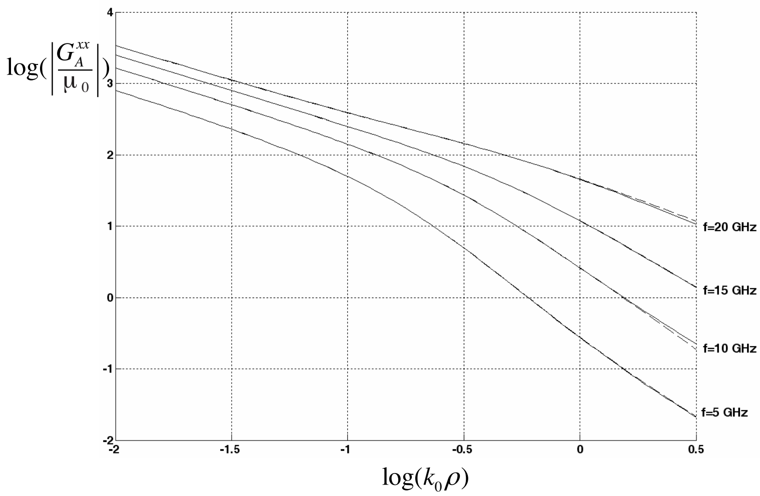

Figure 4. The magnitude of the magnetic vector potential.

5. CONCLUSION

A combination of PSO and BFGS gradient based method is used for the complex time representation of Green's functions in a grounded dielectric slab. Numerical results show that the proposed method have the advantages of both methods. This hybrid method gives considerably better solutions to the problem compared with other optimization methods.

REFERENCES

1. Eiben, A. E. and J. E. Smith, *Introduction to Evolutionary Computing*, Springer-Verlag, Berlin Heildberg, 2003.
2. Mitchell, T., *Machine Learning*, McGraw-Hill, New York, 1997.
3. Fogel, D. B., "System identification through simulated evolution: A machine learning approach to modeling," *Needham Heights*, Ginn Press, MA, 1991.
4. Back, T., U. Harnmel, and H. P. Schwefel, "Evolutionary computation: Comments on the history and current state," *IEEE Trans. Evol. Comput.*, Vol. 1, 3–17, Apr. 1997.
5. Tu, T. C. and C. C. Chiu, "Path loss reduction in an urban area by genetic algorithms," *Journal of Electromagnetic Waves and Applications*, Vol. 20, No. 3, 319–330, 2006.

6. Chen, X., D. Liang, and K. Huang, "Microwave imaging 3-D buried objects using parallel genetic algorithm combined with FDTD technique," *Journal of Electromagnetic Waves and Applications*, Vol. 20, No. 13, 1761–1774, 2006.
7. Tian, Y. B., "Ultraconveniently finding multiple solutions of complex transcendental equations based on genetic algorithm," *Journal of Electromagnetic Waves and Applications*, Vol. 20, No. 4, 475–488, 2006.
8. Sijher, T. S. and A. A. Kishk, "Antenna modeling by infinitesimal dipoles using genetic algorithms," *Progress In Electromagnetics Research*, PIER 52, 225–254, 2005.
9. Chen, X., K. Huang, and X.-B. Xu, "Microwave imaging of buried inhomogeneous objects using parallel genetic algorithm combined with FDTD method," *Progress In Electromagnetics Research*, PIER 53, 283–298, 2005.
10. Kennedy, J. and R. Eberhart, "Particle swarm optimization," *Proc. IEEE Intl. Conf. Neural Networks*, Vol. 4, 1942–1948, 1995.
11. Lee, K. C. and J. Y. Jhang, "Application of particle swarm algorithm to the optimization of unequally spaced antenna arrays," *Journal of Electromagnetic Waves and Applications*, Vol. 20, No. 14, 2001–2012, 2006.
12. Mahmoud, K. R., M. El-Adawy, S. M. M. Ibrahem, R. Bansal, K. R. Mahmoud, and S. H. Zainud-Deen, "A comparison between circular and hexagonal array geometries for smart antenna systems using particle swarm optimization algorithm," *Progress In Electromagnetics Research*, PIER 72, 75–90, 2007.
13. Song, M. P. and G. C. Gu, "Research on particle swarm optimization: a review," *Proc. Intl. Conf. Machine Learning and Cybernetics*, Vol. 4, 2236–2241, Aug. 2004.
14. Aminian, M., "Wide band analysis of Green's functions of multilayer media and its application in accurate and fast analysis in time domain," M.Sc. Thesis, University of Tehran, Summer 2002.
15. Aminian, A., R. Faraji-Dana, and N. Hojjat, "A new wideband closed-form Green's function for a HED over microstrip structure," *IEEE. Intl. Symp. Antennas and Propagation Society*, Vol. 4, 3940–3943, June 2004.
16. Chow, Y. L., J. J. Yang, D. G. Fang, and G. E. Howard, "A closed form spatial Green's function for the thick microstrip substrate," *IEEE Trans. Microwave Theory Tech.*, Vol. 39, 588–592, Mar. 1991.

17. Haddad, H., "A new time-domain analysis of microwave circuits using complex time Green's function," M.Sc. Thesis, University of Tehran, Summer 2005.
18. Mikki, S. M. and A. A. Kishk, "Physical theory for particle swarm optimization," *Progress In Electromagnetics Research*, PIER 75, 171–207, 2007.
19. Shi, Y. and R. Eberhart, "A modified particle swarm optimizer," *Proc. IEEE World Cong. Computational Intelligence*, 96–73, 1998.
20. Broyden, C. G., "The convergence of a class of double-rank minimization algorithms," *J. Inst. Maths. Applics.*, Vol. 6, 76–90, 1970.
21. Fletcher, R., "A new approach to variable metric algorithms," *Computer Journal*, Vol. 13, 317–322, 1970.
22. Goldfarb, D., "A family of variable metric updates derived by variational means," *Mathematics of Computing*, Vol. 24, 23–26, 1970.
23. Shanno, D. F., "Conditioning of quasi-newton methods for function minimization," *Mathematics of Computing*, Vol. 24, 647–656, 1970.
24. Mehrabian, A. and C. Lucas, "A novel numerical optimization algorithm inspired from weed colonization," *Ecological Informatics*, Vol. 1, No. 4, 355–366, Dec. 2006.