2006 IEEE Congress on Evolutionary Computation
Sheraton Vancouver Wall Centre Hotel, Vancouver, BC, Canada
July 16-21, 2006

Boolean Particle Swarm Optimization and Its Application to the Design of a Dual-Band Dual-Polarized Planar Antenna

Alireza Marandi, Farzaneh Afshinmanesh, Mahmoud Shahabadi, and Fariba Bahrami, *Member, IEEE*

Abstract—**Particle Swarm Optimization (PSO) is a powerful evolutionary computation technique for solving continuous-valued optimization problems in various fields of engineering. However, its binary version could not attract much attention such that research on the discrete form of the PSO is currently not massive. In this paper, the Boolean PSO is introduced based on the idea of using the Boolean algebra in the implementation of PSO concepts in the binary space. The proposed algorithm is applied to a hard multiobjective electromagnetic problem, namely to an antenna design problem. The appropriate behavior of the optimization process in comparison with genetic algorithm is demonstrated. The paper also presents the measurement results on a planar antenna whose layout was generated by the proposed Boolean PSO.**

I. Introduction

Particle Swarm Optimization was originally developed by Eberhart and Kennedy in 1995 [1]. This evolutionary computation technique, based on the movement of intelligent swarms, has been shown to be effective in optimizing difficult multidimensional problems [2]. Additionally, it has been demonstrated in certain instances that the PSO can outperform other methods of optimization like the genetic algorithm [3]. The PSO has a simple concept, and can implement paradigms more easily. In the PSO, each individual (*particle*) traces a trajectory in the search space, while constantly updating a *velocity* vector based on the best solutions found so far for that particle, as well as others in the population (*swarm*) [1].

There have been only few attempts for presenting an algorithm for discrete spaces based on the concepts of the PSO. In the canonical discrete PSO, the particles operate on a discrete search space, where the trajectories are defined as changes in the probability that a coordinate will take on a value from allowable discrete values [5].

The Boolean algebra has been shown to be effective in implementation of a novel PSO for binary problems [8]; based on this idea and with some modifications, here, the Boolean PSO is proposed. Theoretical comparison with the conventional binary PSO and the application of the proposed method to an antenna design problem, which is a computational electromagnetic problem, are also presented.

Computational electromagnetics is an area to which PSO has been successfully applied [9, 10, 11]. In this work by using the proposed binary PSO method, a planar antenna for use in two-way satellite Internet services in the Ku-band is designed. Some of previous designs use stacked patches with the combination of different feed mechanisms at two input ports which are mainly (coplanar microstrip line, aperture) [19, 20, 21], (via-hole, microstrip line) [22, 23, 24], and (two microstrip lines) [13, 25]. The antenna proposed in this work has a single port and one layer substrate, and it is capable of having two orthogonal polarizations at two different frequencies with the desired bandwidth.

For solving this antenna design problem, the Ordered Weighted Averaging (OWA) [6] with a Mamdani fuzzy inference system is used to implement the transformation from the three-dimensional objective space to the one-dimensional cost function. The obtained optimized design has been prototyped and tested. Agreement between theoretical and experimental results has been observed.

A short overview of the conventional PSO and the binary PSO procedures are presented in Section II and III. The proposed Boolean PSO is described in Section IV. The design methodology of the planar antenna, chosen as the application example, is described in Section V. Section VI contains the discussion of theoretical and experimental results of the fabricated antenna prototype. Finally, some conclusions are drawn in Section VII.

II. Conventional PSO

Particle swarm optimization (PSO) is a population based search algorithm initialized with a population of random solutions, called *particles*. Each particle flies through the search space with a velocity that is dynamically adjusted according to its own and its companion's previous behavior. This velocity consists of three parts, the "social", the "cognitive", and the "inertia" parts. The "social" part is the term guiding the particle to the best position achieved by the whole swarm of particles so far (g_{best}), the "cognitive" term conducts it to the best position achieved by itself so far p_{best}, and the "inertia" part is the memory of its previous velocity. In the n-*th* dimension of the search space, the updating of each particle's velocity (v_n) and position (x_n) is based on the following formulae [1, 9]:

$$v_n = \omega \cdot v_n + c_1 \cdot \varphi_1 \cdot \left(p_{best,n} - x_n\right) + c_2 \cdot \varphi_2 \cdot \left(g_{best,n} - x_n\right) \qquad (1)$$

A. Marandi, F. Afshinmanesh, M. Shahabadi, and F. Bahrami are with the School of Electrical and Computer Engineering, University of Tehran, P.O. Box 14395-515,Tehran, Iran. (email: a.marandi@ece.ut.ac.ir, f.afshinmanesh@ece.ut.ac.ir, shahabad@ut.ac.ir, fbahrami@ut.ac.ir)

$$x_n = x_n + v_n \quad (2)$$

In Equation (1), φ_1 and φ_2 are random numbers uniformly distributed between 0 and 1. Also, the algorithm has three important parameters determining the tendency of particles to the related terms: the inertia weight (ω), and the acceleration constants (c_1, c_2). Moreover, for implementing the PSO, another parameter is used for restricting the maximum velocity at which each particle can move. All these parameters directly affect the optimization behavior, for example, the inertia weight controls the exploration ability of process by balancing the local and global search ability. The acceleration constants and maximum velocity are parameters for controlling the convergence rate [17, 18].

The PSO is simple in concept, few in parameters, and easy in implementation. After its introduction, it has been widely applied to a number of optimization problems because of its ease of realization and promising optimization ability.

III. Conventional Binary PSO

Although the main attention to the PSO is concentrated on real-valued optimization problems, there are few attempts to apply the idea to the discrete and especially binary optimization ones [2]. The first discrete PSO introduced by Kennedy and Eberhart uses the same terms as in the PSO to determine the changes of probabilities of a bit being in one state or the other. Unlike real-PSO this algorithm is not used widely and there are several researches trying to hybrid it with other algorithms to achieve better performance [14]. Moreover, this method does not seem to outperform the commonly used binary optimization method, namely the genetic algorithm.

In this binary PSO, the idea of using probability of being '1' in binary space instead of velocity is resulted in the following equations for updating each bit of a particle [5]:

$$v_d = v_d + \varphi_1 \times (p_{best,d} - x_d) + \varphi_2 \times (g_{best,d} - x_d) \quad (3)$$

$$S(v_d) = \frac{1}{1 + e^{-v_d}} \quad (4)$$

$$\begin{aligned} &if\,(rand(\ ) < S(v_d)) \quad then \quad x_d = 1; \\ &else \quad x_d = 0; \end{aligned} \quad (5)$$

v_d is the probability of the *d*'th bit of a particle to become 1 while Equation (3) represents its dependence on the previous value of v_d, best positions achieved by all the particles (g_{best}) and by the particle in question (p_{best}) so far. Because the calculated v_d can be greater than 1 or less than zero, a sigmoid function (Equation (4)) is used to transform this value into the limits such that it can be used as a probability. Consequently, rand(), which is a random generator over [0,1], is used to specify the state of each bit (Equation (5)). φ_1 and φ_2 are random numbers uniformly distributed between 0 and 1. The parameter $V_{\max}$ in this algorithm is introduced as an upper and lower limit on the $S(v_d)$.

By checking all the states of x_d, p_{best}, and g_{best} it can be seen that the conventional binary PSO generally can track the main idea of the PSO, which is the tendency of particles to approach their previous best position and global best position; however, it has some considerable disadvantages because the considered velocity and distance are not of the same nature. Showing the adverse effects of this idea on the performance of the algorithm needs a detailed study of the method and comparison between the concepts of the binary and real algorithms, which is not aimed in this paper. However, a special case is examined here to seek to know why this method has not become popular like the algorithm for real-valued problems.

Although in the PSO, v_d i.e., the particle's velocity is used to update the position vector (Equation (2)), the binary PSO uses v_d to specify the probability of being "1". Note that in both methods, the velocity is computed in the same fashion. According to [5], it is obvious that velocity of the particle can be described by the number of changed bits, so this approach cannot include the inertia of particles.

The inertia is the tendency of each particle to keep its previous movement, the change of the bit in a binary space. However, in this method this tendency cannot be inferred from the first term of Equation (3). Not only is there no coefficient to perform the task of the inertia weight, which specifies the rate of this tendency, but also this term shows the tendency of the bit to become one. For example, consider a large value of v_d which results in $S(v_d) \approx 1$, so x_d with a high probability will become 1 in the next step. If the p_{best} and g_{best} have not changed in the current step, the new v_d will be unchanged; therefore, the new v_d dictates x_d to still remain at 1. However, this behavior cannot fulfill the desired function of the inertia.

In addition, as a consequence of using Equation (5), the concept of the maximum velocity $V_{\max}$ will also be incorrect. In summary, the above discrepancies have deteriorated the conventional binary PSO versus the real-valued PSO.

IV. The Boolean PSO

The binary PSO introduced in [8], which is implemented by logical operators, is shown to outperform the conventional binary PSO and GA in some well-known test functions such as De Jong's, Rastrigin's and Griewangk's functions. In addition to the similarity between the behaviors of this method and the real-valued PSO according to their main terms: "inertia," "cognitive," and "social" parts, it is simple in concept and easy in implementation in comparison with the other binary methods like GA. In this method, distance and velocity are defined as the changes in bits of a binary string, the hamming distance, and the '*and*' (·), '*or*' (+), and '*xor*'($\oplus$) operators are used to represent the particles' movement. The only parameter used to implement this method is the maximum allowed velocity, which is the allowed number of bits with the value of one in the velocity vector, implemented by the simple concept of natural selection in Artificial Immune System, setting extra bits to zero randomly [8].

In this paper based on the idea used in [8], the Boolean PSO being equipped with the major parameters like the real-valued PSO is presented. Main formulae for updating the positions of particles, Equations (6) and (7) are composed of binary variables with the mentioned logical operators. The distance between two bits is another bit whose value represents their difference and consequently the velocity bit represents the change in the next step. The second and third terms of Equation (6) calculate this distance between x_d and $p_{best,d}$, and x_d, $g_{best,d}$, respectively. The connection between all terms are established with '*or*' operators and c_1, c_2 and ω according to the relations to be presented shortly. c_1, c_2 as the acceleration coefficients and ω as the inertia coefficient are binary bits stochastically set from the system parameters C_1, C_2, Ω, the real numbers in $[0.0, 1.0]$. The probability for being '1' of c_1, c_2, ω are C_1, C_2, Ω, and in the previous representation of the method [8] are supposed to be 0.5, 0.5, 0, respectively. The procedure used to determine the velocity for each bit of a particle is also depicted in Figure (1). The calculated velocity from Equation (6) is used to determine the new *d-th* bit of the particle by an '*xor*' operator as follows

$$v_d = \omega \cdot v_d + c_1 \cdot (p_{best,d} \oplus x_d) + c_2 \cdot (g_{best,d} \oplus x_d) \tag{6}$$

$$x_d = x_d \oplus v_d \tag{7}$$

Note that the Boolean PSO (BPSO) has the same constituent terms and thus behavior as the real-valued PSO. The effects of previous velocity, distance to $p_{best,d}$, and $g_{best,d}$ depend on the three important parameters of the method, i.e., Ω, C_1, C_2. They increase or decrease the probability of change and thus can change the particle's trajectory.

Study of the parameters of the BPSO is a good way to show its superiority over the conventional binary PSO. For instance, the inertia weight, $\Omega = p(\omega = 1)$, determines the dependence of the present velocity on the previous one. If the previous velocity is zero, it will not affect the current one whereas if it is one, according to Ω, the probability of being one for the new velocity will increase. Acceleration parameters, C1 and C2, determine the self-tendency or group-tendency of the particles, respectively.

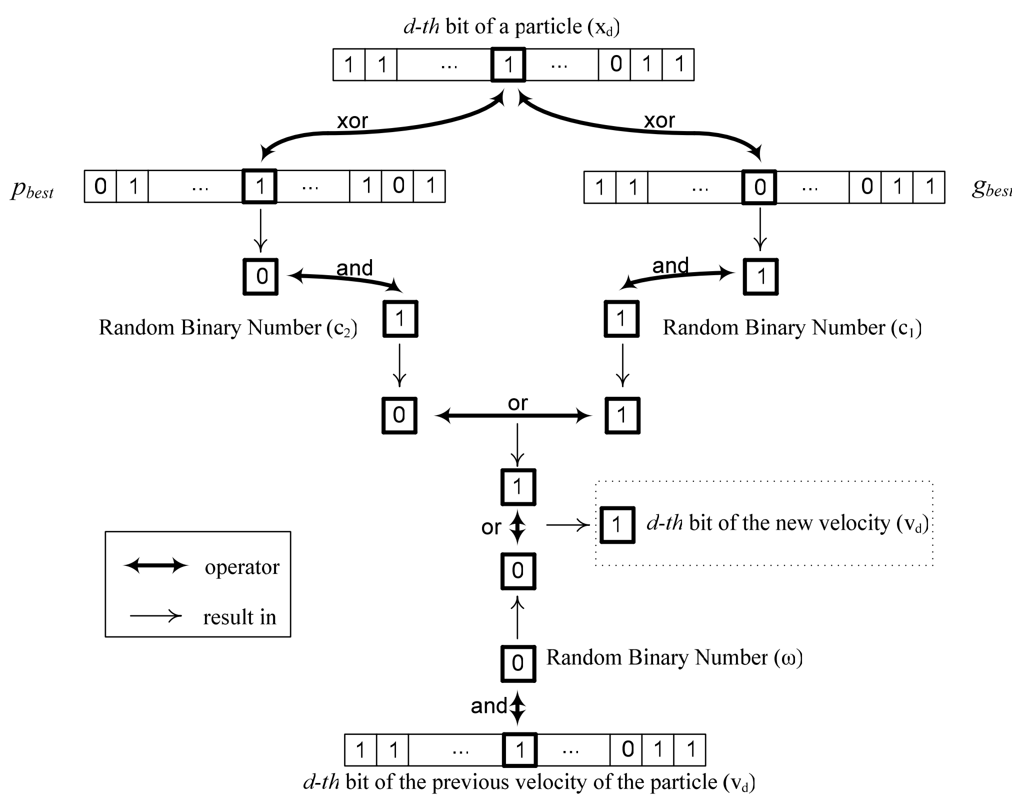

Fig. 1. Process of updating the velocity in the Boolean PSO.

V. ANTENNA DESIGN

In addition to the well-known test functions, the antenna problem in this paper is a hard shape optimization problem by which the performance of the Boolean PSO is tested. The problem is the design of a printed one-layer antenna with a single feed. The antenna should radiate two orthogonal linear polarizations at two different frequencies. The antenna is planar and thus low cost, low profile, and light weight [12]. It can be used as a dual-band and dual-polarized printed antenna in two-way satellite Internet services.

The desired operation frequencies for this antenna application are 12GHz $(\lambda = 25mm)$ and 14GHz $(\lambda \approx 21.5mm)$[13]. Figure 2 shows the 4×4 square grid along with its coding order in the coded binary array; the grid size is determined by the longer wavelength. The antenna layout will be obtained by removing some of the metallic segments of the grid shown in Figure 2. Due to the symmetry of the grid structure, there are three possible positions for placing the feed as it is shown in Figure 2 (a, b, c). Determination of the best feed position is performed by running the optimization process for each of the possible positions. Although the feed position could be added to the optimization vector, it is neglected because it adds to the complexity of the multi-dimensional multiobjective problem.

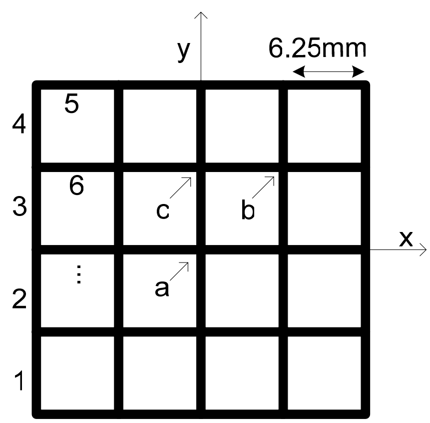

Fig. 2. Grid with possible feed positions (a,b,c) and the order of segments in the coded array.

A. Cost Functions Evaluation

The basic antenna configuration considered in this paper is comprised of 40 connected wire segments. The three design goals are: orthogonal linear polarization at two different frequencies f_1 and f_2, small return loss and high gain around $\theta = 0^\circ$ in both bands. The three objectives associated with these design goals are $obj_{pol}, obj_{S11}, obj_{Gain}$. The lesser the cost function, the better the fitness of an individual. All the radiation parameters are evaluated using a Method of Moment engine. At each frequency point and for each individual, E_θ and E_ϕ are computed in both $\Phi = 0^\circ$ and $\Phi = 90^\circ$ planes for $\theta = -10^\circ$ to $\theta = 10^\circ$. Then the axial ratio using the following formulae is computed at each point

$$\begin{cases} AR_1(\theta) = \left|\dfrac{E_\theta}{E_\phi}\right| & \Phi_1 = 0^\circ\ plane & f_1 = 12GHz & (8) \\ AR_2(\theta) = \left|\dfrac{E_\phi}{E_\theta}\right| & \Phi_2 = 90^\circ\ plane & f_1 = 12GHz & (9) \\ AR_3(\theta) = \left|\dfrac{E_\theta}{E_\phi}\right| & \Phi_1 = 0^\circ\ plane & f_2 = 14GHz & (10) \\ AR_4(\theta) = \left|\dfrac{E_\phi}{E_\theta}\right| & \Phi_2 = 90^\circ\ plane & f_2 = 14GHz & (11) \end{cases}$$

To determine the cost function specifying the amount of orthogonality between the polarizations of two frequencies, obj_{pol}, the averages of the axial ratios are calculated as:

$$Ave_n(\theta) = \frac{1}{20}\int_{\theta=-10^\circ}^{10^\circ} \left(AR_n(\theta)d\theta\right) \qquad n = 1,..,4 \tag{12}$$

Because the sense of polarization at each frequency is not important, the following process is done to specify the type of polarization which results in a better cost function

$$\begin{aligned} &if \quad Ave_1 + Ave_2 > Ave_3 + Ave_4 \\ &\qquad AR_n = (AR_n)^{-1} \qquad n = 1,2 \\ &else \quad AR_n = (AR_n)^{-1} \qquad n = 3,4 \end{aligned}$$

The $g_n(\theta)$ function is defined to assign zero to the obj_{pol} for axial ratios less than or equal to -15dB, and a positive number for other values.

$$g_n(\theta) = \left(dB(AR_n(\theta)) + 15\right)\cdot u\left(dB(AR_n(\theta)) + 15\right) \qquad n = 1,..,4 \tag{13}$$

$$obj_{pol} = \frac{1}{4}\left(\sum_{n=1}^{4}\frac{1}{20}\int_{\theta=-10^\circ}^{10^\circ} g_n(\theta)d\theta\right) \tag{14}$$

The obj_{S11} is calculated from the return loss at two desired frequencies and the threshold for the resonance is set to -15dB.

$$obj_{S11} = \frac{1}{2}\left(\sum_{n=1}^{2}\left(\left(\left|S_{11}\right|_{f_n} + 15\right)\cdot u\left(\left|S_{11}\right|_{f_n} + 15\right)\right)\right) \tag{15}$$

It is desirable that the maximum gain (*GMAX*) in all planes and frequencies located at or near $\theta = 0^\circ$. The difference between *GMAX* and the averaged gain in the angles near $\theta = 0^\circ$ is a criterion for this goal. The threshold for obj_{Gain} is set to 1. Consequently, the obj_{Gain} is calculated as follows

$$GMAX(\Phi_i, f_j) = \max\left(gain(\Phi_i, f_j, \theta)\right) \quad i,j = 1,2 \quad -90 \le \theta \le 90$$

$$h_n(\Phi_i, f_j) = GMAX(\Phi_i, f_j) - \frac{1}{20}\left(\sum_{j=1}^{2}\sum_{i=1}^{2}\int_{\theta=-10^\circ}^{10^\circ}\left(gain(\Phi_i, f_j, \theta)d\theta\right)\right)$$

$$obj_{Gain} = \left(\frac{1}{4}\sum_{j=1}^{2}\sum_{i=1}^{2} h_n(\Phi_i, f_j)\right) - 1 \tag{16}$$

B. *Fuzzy OWA Operator for Multiobjective Optimization*

The transformation from three-dimensional fitness function to a one-dimensional cost function is performed with an Ordered Weighted Averaging operator [6], a transformation of the scores to be aggregated by their respective importances with an associated weighting vector

$$W = \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix} \tag{17}$$

in which:

$$w_j \in [0,1], \ \sum_{j=1}^{3} w_j = 1 \tag{18}$$

$$F(obj_{pol}, obj_{S11}, obj_{gain}) = \sum_{j=1}^{3} w_j b_j \tag{19}$$

The $[b_1, b_2, b_3]$ is the ordered form of $[obj_{pol}, obj_{S11}, obj_{gain}]$ in which b_1 is the maximum, and b_3 is the minimum of the objectives. In addition to the mentioned characteristics, this operator should be commutative, monotonic, and idempotent. For ordering the objectives and assigning appropriate weights, the fuzzy logic is used to produce a more reliable OWA operator according to the difference between the ranges of each dimension. In this process based on the rule, *the worst objective should have the most effect on the fitness*, as the main rule of the OWA operator, a Mamdani inference system [7] calculates the appropriate weights for objectives.

The fuzzification is performed with three Gaussian membership functions applied to the normalized objectives. For the transformation of each objective to the interval [0,1], three acceptable ranges are selected. The obj_{pol} has a minimum of zero, however, when the $obj_{pol} > 15$ it means there are noticeable points at both frequencies which do not obey the assumed polarization, so this upper bound is appropriate and it is not necessary to deal with points higher than that. Based on similar hypotheses for obj_{S11}, obj_{gain}, the upper limits of 15 for obj_{S11}, obj_{pol} and 3 for obj_{gain} are used for the normalization process. The defuzzification layer of fuzzy inference system contains the same membership functions as the fuzzification layer and produce W vector with specified characteristics.

VI. Results

For implementing the optimization process, the MATLAB software (ver. 7) is used as the programming environment and the FEKO (ver. 4.0) as the antenna simulator.

Because each antenna simulation takes about 30 seconds on a Pentium IV computer with 512 MB of RAM, the whole process is very time consuming, so the complete tuning of the algorithms is not feasible, and three different sets of parameters are used for testing each algorithm and the best obtained result is presented here. The population size for all methods is 400; for the Boolean PSO, $\Omega = 0.1, C_1, C_2 = 0.5$, $V_{\max} = 5$, for GA with multipoint crossover, the mutation rate is 0.2, with 20 elites in each generation.

Figure 3 shows the behavior of the Boolean PSO in dealing with the problem with three possible feed positions in 50 iterations. The convergence comparison between the Boolean PSO and GA in this problem with the selected feed position is depicted in Figure 4.

The fabricated antenna, Figure 5, has a substrate with a thickness of 62 mils, a relative dielectric constant of 2.20, and a loss tangent of 0.0009, and the 50Ω- feeding coaxial probe has inner and outer diameters of 1.27 and 4.11mm, respectively. The simulation and measurement results are summarized in Table 1 and Figures 6, 7.

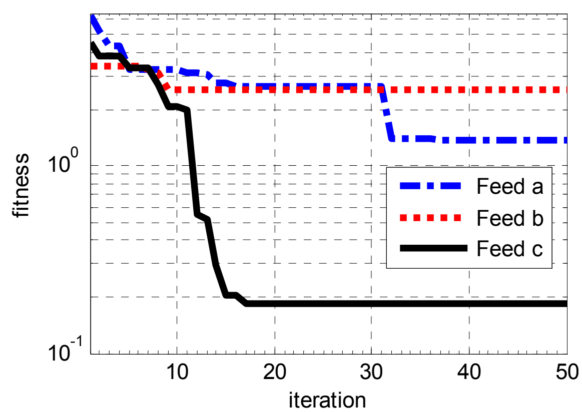

Fig. 3. Best fitness of iterations with three different feed positions.

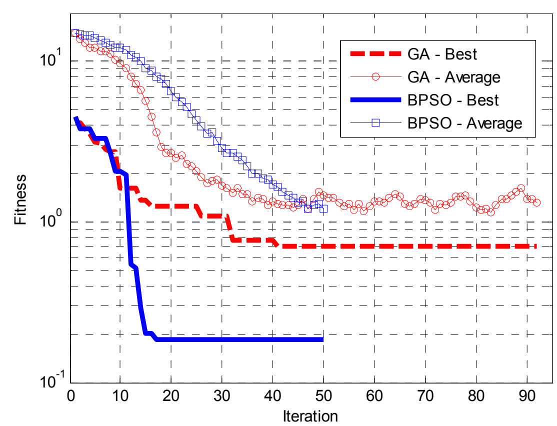

Fig. 4. Best fitness and average fitness of iterations in two methods applied to the problem.

VII. Conclusion

The BPSO, a PSO algorithm implemented by means of the Boolean algebra, is proposed for use in binary optimization problems. The behavior of particles according to three parts of PSO, i.e., "social," "cognitive," and "momentum" terms is discussed and compared with the conventional PSO. Experiments clearly indicate superior performance of the Boolean PSO algorithm in comparison with the GA. The effectiveness of the proposed algorithm is demonstrated for the optimization problem of a planar dual-band dual-polarized antenna. Based on the simulation results, an antenna prototype has been fabricated and tested. Good agreement has been observed between simulation and experimental results.

Fig. 5. Fabricated antenna on RT-Durroid 5880 substrate.

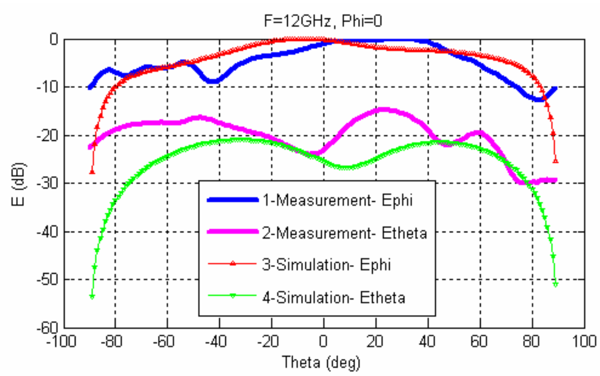

Fig. 6. Antenna pattern in 12GHz and *xz* plane.

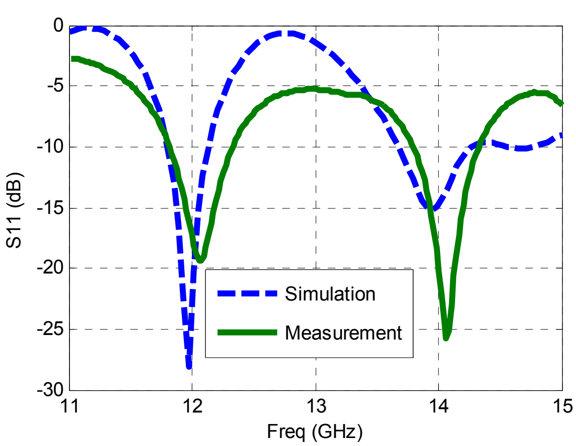

Fig. 7. Simulated and measured return loss of the antenna.

Table 1. Comparison between objectives in simulation and measurement.

	Simulation	Measurement
obj_{S11}	0	0
obj_{pol}	0.1407	0.4158
obj_{gain}	0.33702	-
Fitness	0.1844	> 0.4158

ACKNOWLEDGEMENT

The authors would like to thank the Centre of Excellence for Applied Electromagnetic Systems at the University of Tehran for their invaluable support.

REFERENCES

[1] J. Kennedy, R. Eberhart, "Particle Swarm Optimization," *Proc. IEEE Int. Conf. Neural Networks*, 1995, vol. 4, pp. 1942 - 1948.

[2] S. Mei-Ping, G. Guo-Chang, "Research on particle swarm optimization: a review," *Proc. Int. Conf. on Machine Learning and Cybernetics,* Aug. 2004, vol. 4, pp. 2236 – 2241.

[3] J. Kennedy and W. M. Spears, "Matching algorithms to problems: an experimental test of the particle swarm and some genetic algorithms on multi modal problem generator," *Proc. IEEE Int. Conf. Evolutionary Computation*, May 1998, pp. 78 – 83.

[4] B. Al-kazemi, C. K. Mohan, "Training feed forward neural networks using multi-phase particle swarm optimization," *Proc. Int. Conf. Neural Information Processing*, Nov. 2002, vol. 5, pp. 2615 – 2619.

[5] J. Kennedy, R. Eberhart, "A discrete binary version of the particle swarm algorithm," *Proc. IEEE Int. Conf Systems, Man, Cybernetics,* Oct. 1997, vol. 5, pp. 4104 - 4108.

[6] R. R. Yager, "On ordered weighted averaging aggregation operators in multicriteria decisionmaking," *IEEE Tran. Systems, Man and Cybernetics,* Jan.-Feb. 1988, vol. 18, pp. 183 – 190.

[7] T. J. Ross, *Fuzzy Logic with Engineering Applications*, 2nd ed., John Wiley & Sons, 2004.

[8] F. Afshinmanesh, A. Marandi, A. Rahimi-Kian, "A Novel Binary Particle Swarm Optimization Method Using Artificial Immune System," *Proc. IEEE Int. Conf. Computer as a Tool (EUROCON 2005), Nov. 2005, pp. 217-220.*

[9] J. Rabinson, Y. Rahmat-Samii, "Particle Swarm Optimization in Electromagnetics," *IEEE Trans. Antennas Propag.* , Feb. 2004, vol. 52, pp. 397 – 407.

[10] D. Gies, Y. Rahmat-Samii, "Particle Swarm Optimization for reconfigurable phase differentiated array design," *Microwave and Optical Technology Letters*, Aug. 2003, vol.38, pp. 168-175.

[11] J. Robinson, S. Sinton, Y. Rahmat-Samii, "Particle swarm, genetic algorithm, and their hybrids: optimization of a profiled corrugated hornantenna," *Proc. IEEE Int. Symp. Antenna and Propagation*, June 2002, vol.1, pp. 314 – 317.

[12] P. Bhartia, I. Bahl, R. Garg, A. Ittipiboon, *Microstrip Antenna Design Handbook,* 1st ed. Norwood, MA: Artech House, 2000.

[13] H. S. Noh, J. S. Yun, J. M. Kim, S. Jeon, "Microstrip patch array antenna with high gain and wideband for Tx/Rx dual operation at Ku-band," *IEEE Int. Symp. Antennas and Propagation*, June 2004, vol. 3, pp. 2480-2483.

[14] T. O. Ting, M. V. C. Rao, C. K. Loo, "A novel approach for unit commitment problem via an effective hybrid particle swarm optimization," *IEEE Trans. Power Systems,* Feb. 2006, vol. 21, pp. 411 – 418.

[15] L. Schoofs, B. Naudts, "Swarm intelligence on the binary constraint satisfaction problem*", Proc. IEEE Cong. Evolutionary Computation, May* 2002, vol. 2, pp. 12-17.

[16] W. Pang, K. Wang, C. Zhou, L. Dong, “Fuzzy discrete particle swarm optimization for solving traveling salesman problem,” *Proc. IEEE Int. Conf. Computer and Information Technology,* Sept. 2004, pp. 796 - 800

[17] Y. Zheng, L. Ma, L. Zhang, J. Qian; **“**On the convergence analysis and parameter selection in particle swarm optimization,” *Proc. IEEE Int. Conf. Machine Learning and Cybernetics*, Nov. 2003, vol. 3, pp. 1802 – 1807.

[18] M. Clerc, J. Kennedy, “The particle swarm - explosion, stability, and convergence in a multidimensional complex space,” *IEEE Trans. Evolutionary Computation,* Feb. 2002, vol.6, pp. 58 – 73.

[19] J. S. Yun, H. S. Noh, S. I. Jeon, C. J. Kim, “Design and analysis of TX/RX dual microstrip antenna using FDTD at Ku-band”, Electronics Letters, Aug. 2003, vol. 39, Issue 17, 21, Page(s):1228 – 1230,

[20] R. Shavit, Y. Tzur, D. Spirtus, “Design of a new dual-frequency and dual-polarization microstrip element,” *IEEE Tran. Antennas Propag.* , July 2003, vol. 51, pp. 1443 – 1451.

[21] J. S. Row, “Two-element dual-frequency microstrip antenna with high isolation,” *Electronics Letters*, Dec. 2003, vol. 39, pp. 1786 – 1787.

[22] Y. Kim, W. Yun, Y. Yoon “dual frequency and dual polarization wideband microsrip antenna,” in 1999, *Electron. Lett.,* 35, (17), pp. 1399-1400

[23] Y. J. Kim; W. S. Yun; Y. J. Yoon, “Dual-frequency microstrip array antenna with dual-polarization,” *Proc. IEEE Int. Conf. Microwave*, Nov. and Dec. 1999, vol. 2, pp. 375 – 378.

[24] Y. Kim; S. Kwon; E. Oh; Y. Yoon, “A study of wideband dual-frequency dual-polarization microstrip array antenna**,”** *Proc. IEEE Int. Symp. Antennas and Propagation*, July 2000, vol. 2, pp. 502 – 505.

[25] R. N. Simons, “Suspended rectangular/circular patch antennas with electromagnetically coupled inverted microstrip feed for dual polarization/frequency,” *Proc. IEEE Int. Symp. Antennas and Propagation*, July 2000, vol. 4, pp. 2204 – 2207.