Journal of Network and Innovative Computing
ISSN 2160-2174 Volume 1 (2013) pp. 203 - 213
© MIR Labs, www.mirlabs.net/jnic/index.html

Designing Yagi-Uda Antenna using Gain-Impedance Multiobjective Optimization

Satvir Singh[1] **and Etika Mittal**[2]

[1]SBS State Technical Campus,
Ferozepur, Punjab, India
satvir15@gmail.com

[2]SBS State Technical Campus,
Ferozepur, Punjab, India
etimital@gmail.com

Abstract: **Biogeography-Based Optimization (BBO) is one of the population based algorithms that has out performed most of the Evolutionary Algorithms (EAs) in various optimization applications. BBO is based on the study of geographical distribution of biological organisms over space and time. Yagi-Uda antenna design is most widely used antenna at VHF and UHF frequencies due to high gain, directivity and ease of construction. However, designing a Yagi-Uda antenna involves determination of wire-lengths and their spacings that bear highly complex and non-linear relationships with gain and impedance, etc. For example, if gain is intended to increase then imaginary part in impedance becomes significant whereas real part becomes negligible. In this paper, Non-dominated Sorting along with BBO, its variants and PSO (Particle Swarm Optimization) are investigated for multi-objective optimization of six-element Yagi-Uda antenna designs to optimize two objectives, viz. gain and impedance, simultaneously. The best results and average of multiple monte-carlo runs are presented in the ending sections of the paper for fair comparative study of convergence performance of the stochastic EAs.**

Keywords: Non-dominated Sorting, Bio-geography Based Optimization, Particle Swarm Optimization, Yagi-Uda Antenna, Multi-Objective Optimization, Antenna Gain, Antenna Impedance, BBO Migration Variants.

I. Introduction

Antenna is an electrical device that acts as an interface between free-space radiations and transmitter or receiver. The choice of an antenna depends on various factors such as requisite gain, impedance, bandwidth and frequency of operation,etc. It is simple to construct and has a high gain, typically greater than 10dB at VHF and UHF frequency range. It is a parasitic linear array of parallel dipoles, one of which is energized directly by transmission-line while the other acts as a parasitic radiators whose currents are induced by mutual coupling. The characteristics of the antenna are affected by all geometric parameters of array.

A Yagi-Uda antenna was invented in 1926 by H. Yagi and S. Uda at Tohoku University [Uda and Mushiake, 1954] in Japan, however, published in English in 1928 [Yagi, 1928]. Since its invention, continuous efforts have been put in optimizing the antenna for gain, impedance and bandwidth, etc., using different optimization techniques based on traditional mathematical approaches [Bojsen et al., 1971; Chen and Cheng, 1975; Cheng and Chen, 1973; Cheng, 1971, 1991; Reid, 1946; Shen, 1972] and Artificial Intelligence (AI) techniques [Baskar et al., 2005; Jones and Joines, 1997; Li, 2007; Singh et al., 2010, 2007; Venkatarayalu and Ray, 2004; Wang et al., 2003]. Fishenden and Wiblin proposed an approximate design of Yagi aerials for maximum gain in [Fishenden and Wiblin, 1949]. Ehrenspeck and Poehler have given a manual approach to maximize the gain of the antenna by varying various lengths and spacings of its elements [Ehrenspeck and Poehler, 1959].

Later, the availability of computer software at affordable prices made it possible to optimize antennas numerically. Bojsen *et al.* proposed another optimization technique to calculate the maximum gain of Yagi-Uda antenna arrays with equal and unequal spacings between adjoining elements [Bojsen et al., 1971]. Cheng *et al.* have used optimum spacings and lengths to maximize the gain of a Yagi-Uda antenna [Chen and Cheng, 1975; Cheng and Chen, 1973]. Cheng has proposed optimum design of Yagi-Uda antenna where antenna gain function is highly non-linear, [Cheng, 1991].

In 1975, John Holland introduced Genetic Algorithms (GAs) as a stochastic, swarm based AI technique, inspired from natural evolution of species, to optimize arbitrary system for certain cost function. Then many researchers investigated GAs to optimize Yagi-Uda antenna designs for gain, impedance and bandwidth separately [Altshuler and Linden, 1997; Correia et al., 1999; Jones and Joines, 1997] and collectively [Kuwahara, 2005; Venkatarayalu and Ray, 2003; Wang et al., 2003]. Baskar *et al.*, have optimized Yagi-Uda antenna using Comprehensive Learning Particle Swarm Optimization (CLPSO) and presented better results than other traditional optimization techniques [Baskar et al., 2005]. Li has used Differential Evolution (DE) to optimize geometrical parameters of a Yagi-Uda antenna and illustrated the capabilities of the proposed method with several Yagi-Uda antenna designs [Li, 2007]. Singh *et al.* have investigated another useful, stochastic global search and optimization technique named as Simulated Annealing (SA) for the optimal design of Yagi-Uda antenna [Singh et al., 2007].

In 2008, Dan Simon introduced yet another swarm based stochastic optimization technique based on science of biogeography where features sharing among various habitats, i.e., potential solutions, is accomplished with migration operator and exploration of new features is done with mutation operator [Simon, 2008]. Singh *et al.* have presented BBO as a better optimization technique for Yagi-Uda antenna designs, [Singh et al., 2010].

Du *et al.* proposed the concept of immigration refusal in BBO aiming at improved performance [Du et al., 2009]. In Ma and Simon introduced another migration operator named as Blended migration, to solve constrained optimization problems and make BBO convergence faster [Ma and Simon, 2011]. Pattnaik *et al.* have proposed Enhanced Biogeography Based Optimization (EBBO) in which duplicate habitats, created due to migration of features, is replaced with randomly generated habitats to increase the exploitation ability of BBO algorithm [Pattnaik et al., 2010].

The different migration and mutation variants of BBO for gain maximization optimization of the antenna design are investigated in [Singh and Sachdeva, 2012b] and [Singh and Sachdeva, 2012a], respectively. Non-dominated sorting BBO (NSBBO) algorithm was proposed and investigated to attain multiple objectives, viz. maximum gain and antenna impedance of 75Ω to optimize the antenna designs in [Singh et al., 2012b].

Most of EAs, due to their population based nature, are able to approximate whole pareto front (PF) of a Multiobjective Optimization Problems (MOP) in a single run [Stadler, 1979]. There has been a growing interest in applying EAs to deal with MOPs since Schaffer's seminal work [Schaffer, 1985], and these EAs are called Multi-Objective Evolutionary Algorithm (MOEAs). The Nondominated Sorting Genetic Algorithm (NSGA) proposed in [Srinivas and Deb, 1994] was one of the first such EAs to maintain a diverse set of solutions.

PSO has proven to be an efficient optimization method for single objective optimization and has also shown promising results for solving multiobjective optimization problems in [**?**]. What is in common among these works is the use of a basic form of PSO first introduced by Kennedy and Eberhart [Kennedy and Eberhart, 1995]. The basic form of PSO has some serious limitations in particular when dealing with multiobjective optimization problems. Lee introduced a modified PSO, Non-Dominated Sorting Particle Swarm Optimizer (NSPSO) for improved performance [Li, 2003]. The convergence performance of NSPSO and NSBBO were compared in [Singh et al., 2012a] while optimizing antenna impedance and antenna gain.

After this brief historical background survey, remaining paper is outlined as follows: Section II explains multi-objective optimization problem and non-dominated sorting algorithm. Section III is dedicated to BBO algorithm and its significant migration variants. PSO is explained in Section IV. Yagi-Uda antenna design parameters and formulation as optimization are discussed in section V. In Section VI, simulation results of multiple monte-carlo runs are presented and analyzed. Finally, conclusions and future scope have been discussed in Section VII.

II. Multi-Objective Optimization

A. Multi-Objective Problems

In single-objective optimization, optimal solution is easy to obtain as compared to multi-objective scenario where solution may not exist which could be globally optimal with respect to all objectives. Objectives under consideration may be of conflicting in nature, i.e., improvement in one objective may cause declination in other objective(s). One way to solve MOP is to scalarize the vector of objectives into one objective by averaging the objectives with a weight vector. This process allows a simpler optimization algorithm to be used, however, the obtained solution largely depends on the weight vector used in the scalarization process.

A common difficulty with MOP is the conflicting nature of objectives solution is feasible that could be globally the best for all objectives [Hans, 1988]. Thus a most favourable solution is opted which offers least objective conflict. To find such solutions all classical methods scalarize the objective vector into one objective by three commonly used methods:

1) Method of Objective Weighting

Multiple objective functions are combined into one overall objective function, Z, as given by (1):

$$Z = \sum_{i=1}^{N} w_i f_i(x), \tag{1}$$

where $x \in X$, is the feasible region. The weights w_i are fractional numbers ($0 \leq w_i \leq 1$), and all weights are summed up to one, i.e., $\sum_{i=1}^{N} w_i = 1$. In this method, the optimal solution is controlled by the weight vector w. It is clear from equation (1) that the preference of an objective can be changed by modifying the corresponding weight.

2) Method of Distance Functions

In this method, the scalarization is achieved by using a demand-level vector $\bar{y}$ which has to be specified by the decision maker. The single objective function derived from multiple objectives is as given by (2):

$$Z = [\sum_{i=1}^{N} |f_i(x) - \bar{y}_i|^r]^{1/r} \tag{2}$$

where $1 \leq r < \infty$, and $x \in X$, is the feasible region. Usually an Euclidean metric $r = 2$ is chosen, with $\bar{y}$ as individual optima of objectives. It is important to note that the solution obtained by solving equation (2) depends on the chosen demand-level vector. Arbitrary selection of a demand level may be highly undesirable. This is because a wrong demand level will lead to a non Pareto-optimal solution. As the solution is not guaranteed, the decision maker must have a thorough knowledge of individual optima of each objective prior to the selection of demand level. In a way this method works as a goal programming technique imposing a goal vector/deman level, $\bar{y}$, for given objectives. This method is similar to the method of objective weighting. The only difference is that in this method the goal for each objective function is required to be known whereas in the previous method the relative importance of each objective is required.

3) Min-Max Formulation

This method is different in principle than the above two methods. It attempts to minimize the relative derivations of the single objective functions from individual optimum, i.e., That is, it tries to minimize the objective conflicts. For a minimization problem, the corresponding min-max problem is formulated as given by (3):

$$\text{minimize } F(x) = \text{maximize } [Z_j(x)] \tag{3}$$

where $x \in X$, is the feasible region and $Z_j(x)$ is calculated for non-negative target optimal value $\bar{f}_j > 0$ as follows:

$$Z_j(x) = \frac{f_j - \bar{f}_j}{\bar{f}_j} \tag{4}$$

This method can yield, the best possible compromised solution when objectives with equal priority are required to be optimized. However, priority of each objective can be varied by introducing dimensionless weights in the formulation. This can also be modified as a goal programming technique by introducing a demand level vector in the formulation.

These above methods result in a single solution. The solutions obtained largely depend on the underlying weight-vector or demand-level.

B. Non-Dominated Sorting

To overcome these drawbacks the Pareto optimality concept, was first proposed by Edge-Worth and Pareto [Stadler, 1979]. There exists a set of solutions which are the best tradeoff solutions important for decision making and are often superior to rest of solutions when all objectives are considered, however, inferior for one or more objectives. These solutions are termed as pareto-optimal solutions or non-dominated solutions and others are dominated solutions.

MOPs result in pareto-optimal solutions instead of a single optimal solution in every run. Every solution from non-dominated set is acceptable as none of them is better than its counterpart. However, final selection of a solution is done by the designer based on nature of problem under consideration. Problem, presented in this paper, of optimizing an antenna design has two objectives, viz. (i) desired resistive antenna impedance and (ii) maximum antenna gain. Desired antenna impedance, i.e., $(Re + jIm)\Omega$, is formulated as fitness function, f_1, given as (5), that is required to be minimized.

$$f_1 = |Re - \text{desired impedance}| + |Im| \tag{5}$$

Whereas, second objective of gain maximization is also converted into minimization fitness function, f_2, given as (6)

$$f_2 = \frac{1}{Gain} \tag{6}$$

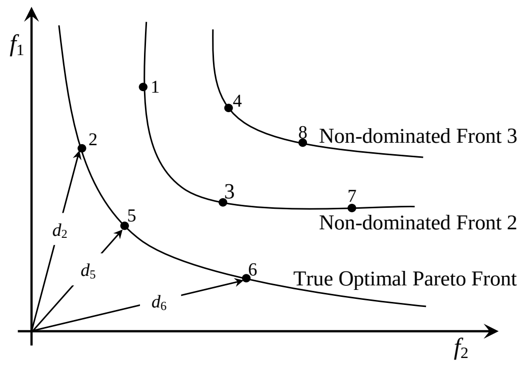


Figure. 1: Non-dominated sorting and pareto-fronts

Suppose every solution, in a swarm of NP solutions, yields f_{1_k} and f_{2_k} as fitness values (where $k = 1, 2, \ldots, NP$), using (5) and (6), that belongs to a set of either non-dominated solution set, P, or dominated solutions, D. An i-th solution in set P dominates the j-th solution in set D if it satisfies the condition of dominance, i.e., $f_{1_i} \leq f_{1_j}$ and $f_{2_i} \leq f_{2_j}$, where both objectives are to be minimized. This condition of dominance is checked for every solution in the universal set of NP solutions to assign it either P set or D set. Solution members of set P form the first non-dominated front, i.e., the pareto optimal front, and then remaining solutions, those belong to set D, are made to face same condition of dominance among themselves to determine next non-dominated front. This process continues till all solutions are classified into different non-dominated fronts, as shown in Fig. 1. Preference order of solutions is to be based on designer's choice, however, here in this paper euclidian distance is determined from origin for every member solution in a non-dominated front and are picked up in ascending order. The pseudo code of non-dominated sorting approach is depicted in Algorithm 4.

Algorithm 1 Pseudo Code for Non-dominated sorting

for $s = 1$ to NP
  $f_{1s} = |\text{Re} - \textit{desired imp}| + |\text{Im}|$ and
  $f_{2s} = \dfrac{1}{gain}$
end for
$f = 1$ % Non-dominated front f
All solutions in the swarm set $\in F$
While (No. solutions in set $F \neq 0$)
  $f = f+1$
  for $i = 1$ to NP
   for $j = 1$ to NP
    if ($i \neq j$)
     if ($f_{1i} \leq f_{1j}$ **and** $f_{2i} \leq f_{2j}$)
      j-th solution $\in D_f$
     else
      j-th solution $\in P_f$
     end if
    end if
   end for
  end for
  $F = D_f$
End while

III. Biogeography Based Optimization

As name suggests, BBO is a population based global optimization technique which got inspiration from the science of biogeography, i.e., study of distribution of animals and plants among different habitats over time and space. BBO results presented by researchers are better than other EAs such as, PSO, GAs, SA and DE, etc. [Baskar et al., 2005; Jones and Joines, 1997; Rattan et al., 2008; Venkatarayalu and Ray, 2003].

Originally, biogeography was studied by Charles Darwin [Darwin, 1995] and Alfred Wallace [A.Wallace, 2005] mainly as descriptive study. However, in 1967, the work carried out by MacAurthur and Wilson [MacArthur and Wilson, 1967] changed this view point and proposed a mathematical model for biogeography and made it feasible to predict the number of species in a habitat. Mathematical models of biogeography describe migration, speciation, and extinction of species in various islands. The term *island* is used for any habitat that is geographically isolated from other habitats. Habitats that are well suited residences for biological species are referred to have high Habitat Suitability Index (HSI) value. However, HSI is analogues to fitness in other EAs whose value depends upon many factors such as rainfall, diversity of vegetation, diversity of topographic features, land area, and temperature, etc. The factors/variables that characterize habitability are termed as Suitability Index Variables (SIVs). In other words, HSI is dependent variable whereas SIVs are independent variables.

The habitats with a high HSI tend to have a large population of its resident species, that is responsible for more probability of emigration (emigration rate, μ) and less probability of immigration (immigration rate, λ) due to natural random behavior of species. Immigration is the arrival of new species into a habitat or population, while emigration is the act of leaving one's native region. On the other hand, habitats with low HSI tend to have low emigration rate, μ, due to sparse population, however, they will have high immigration rate, λ. Suitability of habitats with low HSI is likely to increase with influx of species from other habitats having high HSI. However, if HSI does not increase and remains low, species in that habitat go extinct that leads to additional immigration. For sake of simplicity, it is safe to assume a linear relationship between HSI (or population) and immigration and emigration rates and same maximum emigration and immigration rates, i.e., $E = I$, as depicted graphically in Figure 2.

For k-th habitat values of emigration rate, μ_k, and immigration rate, λ_k, are given by (7) and (8).

$$\mu_k = E \cdot \frac{HSI_k}{HSI_{max} - HSI_{min}} \tag{7}$$

$$\lambda_k = I \cdot (1 - \frac{HSI_k}{HSI_{max} - HSI_{min}}) \tag{8}$$

The immigration of new species from high HSI to low HSI habitats may raise the HSI of poor habitats as good solutions are more resistant to change than poor solutions whereas poor solutions are more dynamic and accept a lot of new features from good solutions.

Each habitat, in a population of size NP, is represented by M-dimensional vector as $H = [SIV_1, SIV_2, \ldots, SIV_M]$

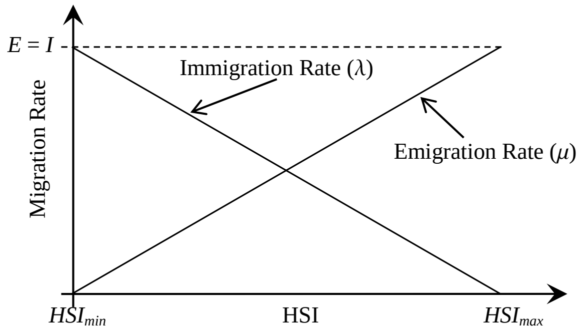


Figure. 2: Migration Curves

where M is the number of SIVs (features) to be evolved for optimal HSI. HSI is the degree of acceptability that is determined by evaluating the cost/objective function, i.e., $HSI = f(H)$. Following subsections describes the different migration variants of BBO. i.e., Standard BBO [Simon, 2008], Blended BBO[Ma and Simon, 2011], Immigration Refusal BBO [Du et al., 2009], Enhanced BBO [Pattnaik et al., 2010].

A. Standard BBO

Algorithmic flow of standard BBO involves two mechanisms, i.e., migration and mutation, these are discussed in the following subsections.

1) Migration

Migration is a probabilistic operator that improves HSI of poor habitats by sharing features from good habitats. During migration, i-th habitat, H_i (where $i = 1, 2, \ldots, NP$) use its immigration rate, λ_i given by (8), to probabilistically decide whether to immigrate or not. In case immigration is selected, then the emigrating habitat, H_j, is found probabilistically based on emigration rate, μ_j given by (7). The process of migration is completed by copying values of SIVs from H_j to H_i at random chosen sites. The pseudo code of migration operator is depicted in Algorithm 2.

Algorithm 2 Pseudo Code for Standard Migration

```
for i = 1 to NP do
    Select H_i with probability based on λ_i
    if H_i is selected then
        for j = 1 to NP do
            Select H_j with probability based on μ_j
            if H_j is selected
                Randomly select a SIV(s) from H_j
                Copy them SIV(s) in H_i
            end if
        end for
    end if
end for
```

2) Mutation

Mutation is another probabilistic operator that modifies the values of some randomly selected SIVs of some habitats that are intended for exploration of search-space for better solutions by increasing the biological diversity in the population. Here, higher mutation rates are investigated on habitats those are, probabilistically, participating less in migration process. The mutation rate, $mRate$, for k-th habitat is calculated as (9)

$$mRate_k = C \times \min(\mu_k, \lambda_k) \quad (9)$$

where μ_k and λ_k are emigration and immigration rates, respectively, given by (7) and (8) corresponding to HSI_k. To reduce fast generation of duplicate habitats,here, C, is chosen as 3 and to keep exploitation rate much higher as compared to other EAs. The pseudo code of mutation operator is depicted in Algorithm 3.

Algorithm 3 Pseudo Code for Mutation

```
mRate = C x min(μk, λk)
for n = 1 to NP do
    for j = 1 to number of SIVs do
        Select Hj(SIV) with mRate
        if Hj(SIV) is selected then
            Replace Hj(SIV) with randomly generated SIV value
        end if
    end for
end for
```

B. Blended BBO

Blended migration operator is a generalization of the standard BBO migration operator and inspired by blended crossover in GAs [McTavish and Restrepo, 2008]. In blended migration, a SIV value of immigrating habitat, *ImHbt*, is not simply replaced by a SIV value of emigrating habitat, *EmHbt*, as happened in standard BBO migration operator. Rather, a new solution feature, i.e., SIV value is comprised of two components as $ImHbt(SIV) \leftarrow \alpha \cdot ImHbt(SIV) + (1-\alpha) \cdot EmHbt(SIV)$. Here α is a random number between 0 and 1. The pseudo code of blended migration is depicted in Algorithm 4

Algorithm 4 Pseudo Code for Blended Migration

```
for i = 1 to NP do
    Select Hi with probability based on λi
    if Hi is selected then
        for j = 1 to NP do
            Select Hj with probability based on μj
            if Hj is selected
                Hi(SIV) ← α • Hi(SIV) + (1−α) • Hj(SIV)
            end if
        end for
    end if
end for
```

C. Immigration Refusal BBO

In BBO, if a habitat has high emigration rate, i.e, the probability of emigrating to other habitats is high and the probability of immigration from other habitats is low. However, the low probability does not mean that immigration will never happen. Once in a while, a highly fit solution may receive solution features from a low-fit solution that may degrade its fitness. In such cases, immigration is refused to prevent degradation of HSI values of habitats. This BBO variant with conditional migration is termed as Immigration Refusal whose performance with testbed of benchmark functions is encouraging [Du et al., 2009]. The pseudo code of Immigration Refusal migration is depicted in Algorithm 5

Algorithm 5 Pseudo Code for Immigration Refusal BBO

```
for i = 1 to NP do
    Select Hi with probability based on λi
    if Hi is selected then
        for j = 1 to NP do
            Select Hj with probability based on μj
            if Hj is selected
                if ((fitness(Hj) > (fitness(Hi))
                    apply migration
                end if
            end if
        end for
    end if
end for
```

D. Enhanced BBO

Standard BBO migration operator tends to create duplicate solutions which decreases the diversity in the population. To prevent this diversity decrease in the population, duplicate habitats are replaced with randomly generated habitats. This leads to increase exploration of new SIV values. In EBBO, clear duplicate operator is integrated in basic BBO algorithm to improve its performance. The migration pseudo code of Enhanced BBO is depicted in Algorithm 6

Algorithm 6 Pseudo Code for Enhanced BBO

```
for i = 1 to NP do
    Select Hi with probability based on λi
    if Hi is selected then
        for j = 1 to NP do
            Select Hj with probability based on μj
            if Hj is selected
                if ((fitness(Hj) == (fitness(Hi))
                    eliminate duplicates
                end if
            end if
        end for
    end if
end for
```

IV. Particle Swarm Optimization

PSO algorithm is another stochastic swarm intelligence based global search algorithm. The motivation behind PSO algorithm is social behavior of animals, e.g., flocking of birds and fish schooling. PSO has its origin in simulations created to visualize the synchronized choreography of a bird flock by incorporating certain features like nearest-neighbor velocity matching and acceleration by distance [Baskar et al., 2005; Kennedy and Eberhart, 1995; Parsopoulos and Vrahatis, 2002; Shi et al., 2001]. Later on, it was realized that the simulation could be used as an optimizer and resulted in the first simple version of PSO, the birds/particles have (1) adaptable velocities that determines their movement in the search space, (2) memory which enable them for remembering the best position in the search space ever visited and (3) the knowledge of the overall best located particle in the swarm. The position corresponding to the past best fitness is known as, $pbest$, and the overall best out of all NP the particles in the population is called global best or $gbest$. Consider that the search-space is M-dimensional and i-th particle location in the swarm can be represented by $X_i = [x_{i1}, x_{i2}, ...x_{id}..., x_{iM}]$ and its velocity can be represented by another M-dimensional vector $V_i = [v_{i1}, v_{i2}, ...v_{id}.., v_{iM}]$. Let the previously best visited location position of this particle be denoted by $P_i = [p_{i1}, p_{i2}, ...p_{id}.., p_{iM}]$, whereas, g-th particle, i.e., $P_g = [p_{g1}, p_{g2}, ...p_{gd}.., p_{gM}]$, is globally best particle location. Figure 3 depicts the vector movement of particle element from location x_{id}^n to x_{id}^{n+1} in $(n+1)$-th iteration that is being governed by past best location, p_{id}^n, global best location, p_{gd}^n, and current velocity v_{id}^n. Alternatively, the whole swarm is updated according to the equations (10) and (11) suggested by Shi & Eberhart [Shi and Eberhart, 1999].

$$v_{id}^{m+1} = \chi(\alpha v_{id}^m + \varphi_1 r_1 (p_{id}^m - x_{id}^m) + \varphi_2 r_2 (p_{gd}^m - x_{id}^m)) \quad (10)$$

$$x_{id}^{m+1} = x_{id}^m + v_{id}^{m+1} \quad (11)$$

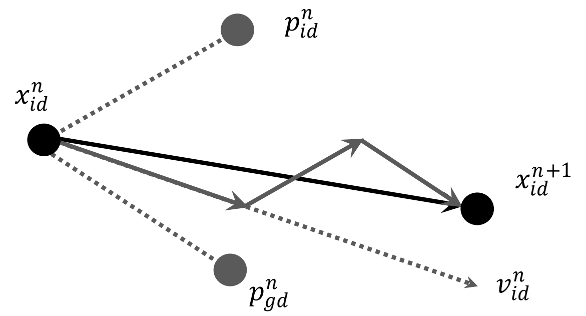


Figure. 3: Movement of i-th particle in 2-D search space

Here, inertia weight (w), cognitive learning parameter (φ_1), social learning parameter (φ_2) and constriction factor (χ), are strategy parameters of PSO algorithm, while r_1 and r_2 are random numbers uniformly distributed in the range [0,1]. Generally the inertia weight, w, is not kept fixed and is varied as the algorithm progresses. The particle movements is restricted with maximum velocity, $\pm V_{max}$, to avoid jump over the optimal location as per search space requirements.

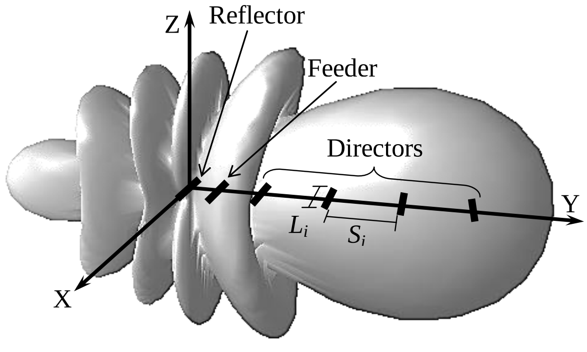


Figure. 4: Six-element Yagi-Uda Antenna

V. Antenna Design Parameters

Yagi-Uda antenna consists of three types of elements: (a) *Reflector*–biggest among all and is responsible for blocking radiations in one direction. (b) *Feeder*–which is fed with the signal from transmission line to be transmitted and (c) *Directors*–these are usually more then one in number and responsible for unidirectional radiations. Figure 4 depicts a typical six-wire Yagi-Uda antenna where all wires placed parallel to x-axis and along y-axis. Middle segment of the reflector element is placed at origin, $x = y = z = 0$, and excitation is applied to the middle segment of the feeder element.

Designing a Yagi-Uda antenna involves determination of wire-lengths and wire-spacings in between to get maximum gain and desired impedance, etc., at an arbitrary frequency of operation. An antenna with N elements requires $2N-1$ parameters, i.e., N wire lengths and $N-1$ spacings, that are to be determined. These $2N-1$ parameters, collectively, are represented as a string referred as a *habitat* in BBO given as (12).

$$H = [L_1, L_2, \ldots, L_N, S_1, S_2, \ldots, S_{N-1}] \quad (12)$$

where L_S are the lengths and S_S are the spacing of antenna elements. An incoming field sets up resonant currents on all the antenna elements which re-radiate signals. These re-radiated signals are then picked up by the feeder element, that leads to total current induced in the feeder equivalent to combination of the direct field input and the re-radiated contributions from the director and reflector elements. This makes highly non-linear and complex relationships between antenna parameters and its characteristics like gain and impedance, etc.

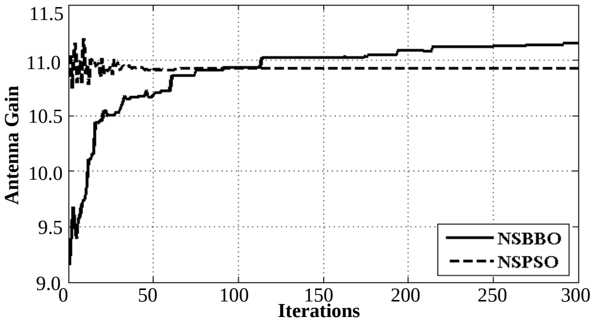
VI. Simulation Results and Discussions

To present fair analysis, a six-wire Yagi-Uda antenna design is optimized for 10 times using 300 iterations under similar evolutionary conditions. The universe of discourses to search optimal values of wire-lengths and wire-spacings are fixed as $0.40\lambda - 0.50\lambda$ and $0.10\lambda - 0.45\lambda$, respectively. However, cross-sectional radius and segment size for all wires are kept constant, i.e., 0.003397λ and 0.1λ, respectively, where λ is the wavelength corresponding to frequency of operation of 300MHz. The C++ programming platform is used for algorithm coding, whereas, method of moments based software, Numerical Electromagnetic Code (NEC2) [Burke and Poggio, 1981], is used to evaluate antenna designs. Both objectives, gain and impedance, are optimized simultaneously using two fitness functions, given by (5) and (6).

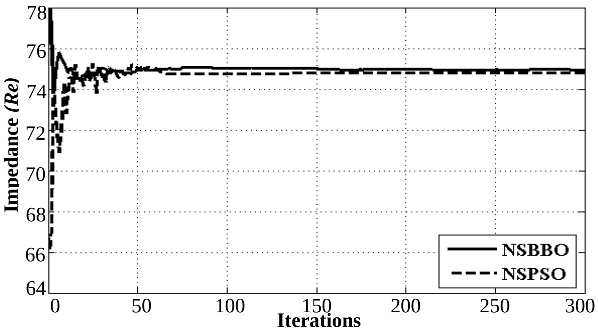
A. *Convergence flow for* 75Ω *antenna impedance and maximal gain*

Six-wire Yagi-Uda antenna designs are evolved using NSBBO and NSPSO for 75Ω resistive antenna impedance and zero reactive antenna impedance, whose fitness function is given as (5).

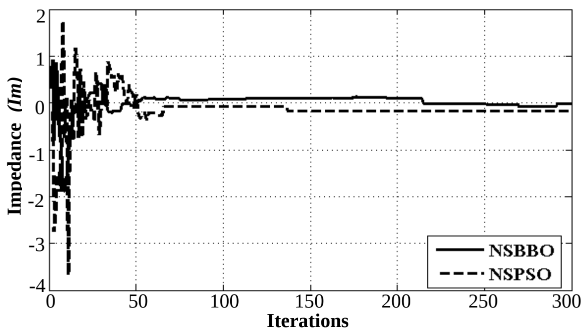
Average of 10 Monte-Carlo simulation runs for 30 habitats for each algorithm are plotted in Fig. 5 to show convergence flow while achieving (a) maximum antenna gain, (b) 75Ω resistive antenna impedance and (c) zero reactive antenna impedance.



(a) Antenna Gain Convergence



(b) Resistive Antenna Impedance



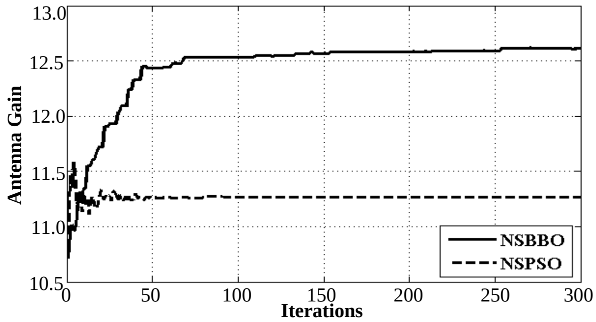
(c) Reactive Antenna Impedance

Figure. 5: NSBBO and NSPSO Convergence flow for 75Ω resistive antenna impedance

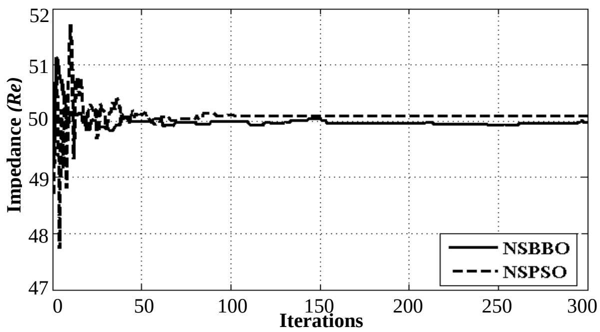
From the plots, it can be observed that best compromised solution, sometimes lead to poor solutions in terms of gain or impedance. However, with increasing iteration number best compromised solution improves in aggregate that may, improve further, if maximum iteration number is kept higher. Reasons for poor performance of PSO may include use of global best PSO model, where each particle learns from every other particle in the swarm and globally best particle, therefore, is prone to get trapped in local optima. Typically, the best antenna designs obtained during process of optimization and the average results of 10 monte-carlo runs, depicted in Fig. 5, are tabulated in Table 1.

B. *Convergence flow for* 50Ω *antenna impedance and maximal gain*

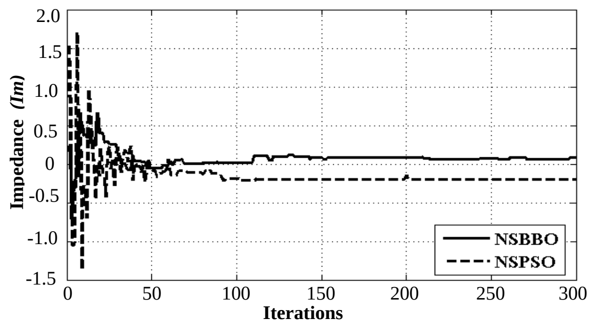
Average of 10 Monte-Carlo simulation runs for 50 habitats using NSBBO and NSPSO are plotted in Fig. 6 to show convergence flow while achieving (a) maximum antenna gain, (b) 50Ω resistive antenna impedance and (c) zero reactive antenna impedance.



(a) Antenna Gain Convergence



(b) Resistive Antenna Impedance



(c) Reactive Antenna Impedance

Figure. 6: Average NSBBO and NSPSO Convergence flow for 50Ω antenna impedance and maximal gain

From the plots, it can be observed that almost every time NSPSO gets trapped in local optimal for gain objective function and at the same time reactive impedance is capacitive and longer than that of Fig. 5. Typically, the best antenna

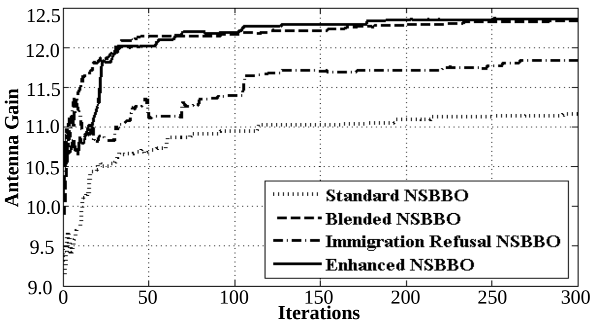
Table 1: The best antenna designs evolved for 75Ω resistive antenna impedance and maximal gain.

Element	Standard BBO		PSO	
	Length	Spacing	Length	Spacing
1(λ)	0.4732	-	0.4732	-
2(λ)	0.4780	0.1979	0.4787	0.1953
3(λ)	0.4397	0.1631	0.4396	0.2092
4(λ)	0.4316	0.2735	0.4343	0.2411
5(λ)	0.4193	0.3902	0.4167	0.4353
6(λ)	0.4307	0.3360	0.4334	0.3298
Best Gain	**12.58 dBi**		**12.28 dBi**	
Best Imp.	**74.9414 + j 0.0364 Ω**		**72.903 + j 1.490 Ω**	
Average Gain	**11.16 dBi**		**10.925 dBi**	
Best Imp.	**74.9458 - j 0.0238 Ω**		**74.8032 + j 0.178 Ω**	

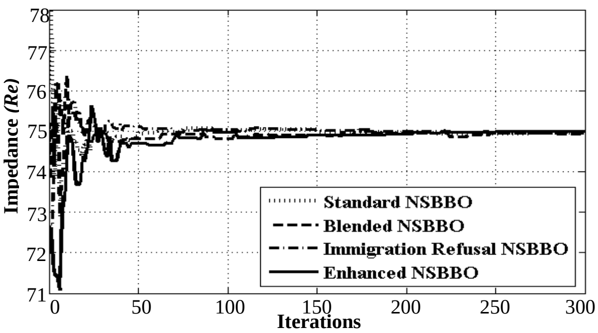
Table 2: The best antenna designs evolved for 50Ω antenna impedance and maximum gain.

Element	Standard BBO		PSO	
	Length	Spacing	Length	Spacing
1(λ)	0.4777	-	0.4744	-
2(λ)	0.4700	0.1901	0.4609	0.2025
3(λ)	0.4436	0.1826	0.4350	0.2107
4(λ)	0.4292	0.2912	0.4290	0.3042
5(λ)	0.4239	0.3553	0.4236	0.3418
6(λ)	0.4287	0.3475	0.4224	0.3529
Best Gain	**12.70 dBi**		**12.57 dBi**	
Best Imp.	**50.1265 + j 0.0124 Ω**		**50.562 - j 0.507 Ω**	
Average Gain	**12.616 dBi**		**11.263 dBi**	
Best Imp.	**49.9835 + j 0.0902 Ω**		**50.099 - j 0.131 Ω**	

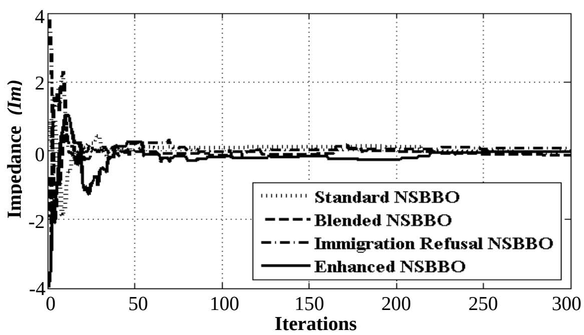
designs for maximal gain and 50Ω antenna gain obtained during process of optimization and the average results of 10 monte-carlo runs, shown in Fig. 6, are tabulated in Table 2.



(a) Antenna Gain Convergence



(b) Resistive Antenna Impedance

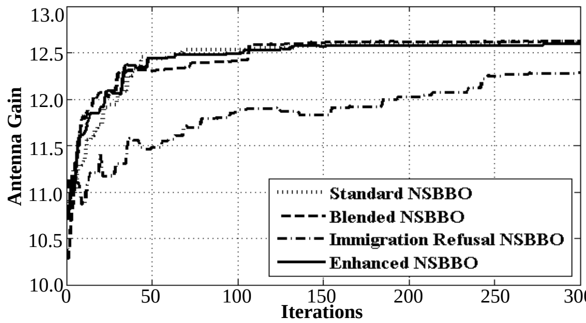


(c) Reactive Antenna Impedance

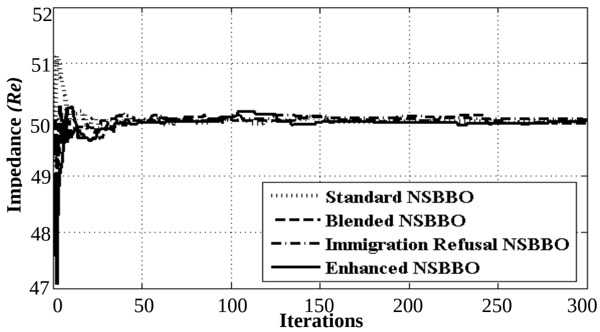
Figure. 7: Average of Convergence flow of NSBBO variant algorithms for 75Ω resistive antenna impedance and maximal gain

C. *Convergence flow of NSBBO migration variants for 75Ω antenna impedance*

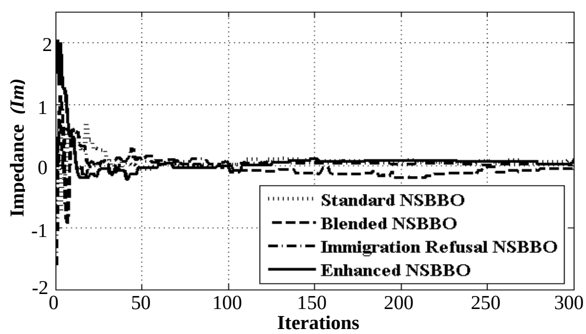
Different migration variants of BBO, discussed in Section III are experimented for gain and antenna impedance of 75Ω simultaneously.



(a) Antenna Gain Convergence



(b) Resistive Antenna Impedance



(c) Reactive Antenna Impedance

Figure. 8: Convergence flow for different NSBBO migration variants at 50Ω resistive antenna impedance

Table 3: The best antenna designs obtained during optimization and average results after 300 iterations for 75Ω impedance

Element	Standard BBO		Blended BBO		IR BBO		EBBO	
	Length	**Spacing**	**Length**	**Spacing**	**Length**	**Spacing**	**Length**	**Spacing**
$1(\lambda)$	0.4732	-	0.4738	-	0.4652	-	0.4732	-
$2(\lambda)$	0.4780	0.1979	0.4622	0.2185	0.4546	0.2308	0.4693	0.2123
$3(\lambda)$	0.4397	0.1631	0.4417	0.3929	0.4333	0.1239	0.4457	0.1741
$4(\lambda)$	0.4316	0.2735	0.4289	0.6546	0.4249	0.2295	0.4329	0.2484
$5(\lambda)$	0.4193	0.3902	0.4225	1.0289	0.4258	0.3162	0.4221	0.3644
$6(\lambda)$	0.4307	0.3360	0.4283	1.3835	0.4114	0.4423	0.4272	0.3758
Best Gain	**12.58 dBi**		**12.58 dBi**		**12.33 dBi**		**12.63 dBi**	
Best Imp.	**74.9414 + j 0.036 Ω**		**75.2441 - j 0.084 Ω**		**74.9729 + j 0.077 Ω**		**75.117 + j 0.7644 Ω**	
Average Gain	**11.16 dBi**		**12.40 dBi**		**11.843 dBi**		**12.364 dBi**	
Best Imp.	**74.946 - j 0.024 Ω**		**75.050 - j 0.073 Ω**		**74.9633 + j 0.0763 Ω**		**74.9971 - j 0.0155 Ω**	

Table 4: The best antenna designs obtained during optimization and average results after 300 iterations for 50Ω impedance

Element	Standard BBO		Blended BBO		IR BBO		EBBO	
	Length	**Spacing**	**Length**	**Spacing**	**Length**	**Spacing**	**Length**	**Spacing**
$1(\lambda)$	0.4777	-	0.4764	-	0.4754	-	0.4746	-
$2(\lambda)$	0.4700	0.1901	0.4674	0.2168	0.4652	0.2105	0.4653	0.2111
$3(\lambda)$	0.4436	0.1826	0.4428	0.1801	0.4419	0.1816	0.4407	0.1994
$4(\lambda)$	0.4292	0.2912	0.4272	0.3032	0.4286	0.3068	0.4293	0.2999
$5(\lambda)$	0.4239	0.3553	0.4235	0.3401	0.4250	0.3307	0.4233	0.3349
$6(\lambda)$	0.4287	0.3475	0.4272	0.3609	0.4280	0.3528	0.4251	0.3719
Best Gain	**12.70 dBi**		**12.68 dBi**		**12.70 dBi**		**12.66 dBi**	
Best Imp.	**50.1265 - j 0.0124 Ω**		**50.1755 - j 0.0833 Ω**		**49.9502 + j 0.0612 Ω**		**49.9784 - j 0.0599 Ω**	
Average Gain	**12.616 dBi**		**12.624 dBi**		**12.282 dBi**		**12.593 dBi**	
Best Imp.	**49.98355 + j 0.0902 Ω**		**49.95325 + j 0.00266 Ω**		**50.00034 + j 0.03253 Ω**		**49.97555 + j 0.08409 Ω**	

Average of 10 Monte-Carlo simulation runs for 30 habitats for each BBO variant algorithm are plotted in Fig. 7 to show convergence flow while achieving (a) maximum antenna gain, (b) 75Ω resistive antenna impedance and (c) zero reactive antenna impedance.
From the plots, it can be observed that EBBO performs better amongst all the migration variants, gives the maximum gain, however, the convergence performance for blended variant is fast as compared to others during initial iterations. Typically, the best antenna designs evolved and the average results of 10 monte-carlo runs, depicetd in Fig. 7, are tabulated in Table 3, respectively.

D. *Convergence flow for NSBBO variant algorithms for* 50Ω *antenna impedance and maximal gain*

Different migration variants of BBO, viz., Standard BBO, Blended BBO, IR BBO and EBBO are experimented for gain maximization and evolving antenna impedance 50Ω, simultaneously.
Average of 10 Monte-Carlo simulation runs with 50 habitats for each variant algorithm are plotted in Fig. 8 to analyse convergence flow while achieving both objectives.
The convergence performance of standard BBO, EBBO and Blended BBO are comparable, however. IRBBO resulted in poorest performance under same evolutionary conditions and 300 iterations as shown in Fig. 8, and tabulated in Table 4.

VII. Conclusions and Future Scope

In this paper, NSBBO, its variants and NSPSO algorithms are investigated for attaining multiple objectives, viz. maximum gain and antenna impedance of 75Ω and 50Ω. For fair analysis of convergence performance of stochastic global search algorithms, average of 10 monte-carlo run is plotted for every case and then tabulated along with the best results in Section VI. The maximum gain obtained for NSBBO at reactive impedance of 50Ω using 50 habitats are better as compared to the approach used in [Singh et al., 2010] i.e., 12.69 dBi. Investigation of NSBBO algorithms with different mutation variants and using different models of PSO designing is next our agenda for improved performance. Further, performance comparison study can also be conducted between NSBBO, NSGA and NSPSO, etc.

References

Altshuler, E., Linden, D., 1997. Wire-antenna Designs using Genetic Algorithms. Antennas and Propagation Magazine, IEEE 39 (2), 33–43.

A.Wallace, 2005. The Geographical Distribution of Animals. Boston, MA: Adamant Media Corporation Two, 232–237.

Baskar, S., Alphones, A., Suganthan, P., Liang, J., 2005. Design of yagi-uda antennas using comprehensive learning particle swarm optimisation 152 (5), 340–346.

Bojsen, J., Schjaer-Jacobsen, H., Nilsson, E., Bach Andersen, J., 1971. Maximum Gain of Yagi–Uda Arrays. Electronics Letters 7 (18), 531–532.

Burke, G. J., Poggio, A. J., 1981. Numerical Electromagnetics Code (NEC) method of moments. NOSC Tech. DocLawrence Livermore National Laboratory, Livermore, Calif, USA 116, 1–131.

Chen, C., Cheng, D., 1975. Optimum Element Lengths for Yagi-Uda Arrays. IEEE Transactions on Antennas and Propagation, 23 (1), 8–15.

Cheng, D., Chen, C., 1973. Optimum Element Spacings for Yagi-Uda Arrays. IEEE Transactions on Antennas and Propagation, 21 (5), 615–623.

Cheng, D. K., 1971. Optimization Techniques for Antenna Arrays. Proceedings of the IEEE 59 (12), 1664–1674.

Cheng, D. K., 1991. Gain Optimization for Yagi-Uda Arrays. Antennas and Propagation Magazine, IEEE 33 (3), 42–46.

Correia, D., Soares, A. J. M., Terada, M. A. B., 1999. Optimization of gain, impedance and bandwidth in Yagi-Uda Antennas using Genetic Algorithm. IEEE 1, 41–44.

Darwin, C., 1995. The Orign of Species. New York : gramercy Two, 398–403.

Du, D., Simon, D., Ergezer, M., 2009. Biogeography-based Optimization Combined with Evolutionary Strategy and Immigration Refusal. IEEE 1, 997–1002.

Ehrenspeck, H., Poehler, H., 1959. A New Method for Obtaining Maximum Gain from Yagi Antennas. IRE Transactions on Antennas and Propagation, 7 (4), 379–386.

Fishenden, R. M., Wiblin, E. R., 1949. Design of Yagi Aerials. Proceedings of the IEE-Part III: Radio and Communication Engineering 96 (39), 5.

Hans, A., 1988. Multicriteria optimization for highly accurate systems. Multicriteria Optimization in Engineering and Sciences 19, 309–352.

Jones, E. A., Joines, W. T., 1997. Design of Yagi-Uda Antennas using Genetic Algorithms. IEEE Transactions on Antennas and Propagation, 45 (9), 1386–1392.

Kennedy, J., Eberhart, R., 1995. Particle swarm optimization 4, 1942–1948.

Kuwahara, Y., 2005. Multiobjective optimization design of yagi-uda antenna. Antennas and Propagation, IEEE Transactions on 53 (6), 1984–1992.

Li, J. Y., 2007. Optimizing Design of Antenna using Differential Evolution. IEEE 1, 1–4.

Li, X., 2003. A non-dominated sorting particle swarm optimizer for multiobjective optimization, 198–198.

Ma, H., Simon, D., 2011. Blended Biogeography-based Optimization for Constrained Optimization. Engineering Applications of Artificial Intelligence 24 (3), 517–525.

MacArthur, R., Wilson, E., 1967. The Theory of Island Biogeography. Princeton Univ Pr.

McTavish, T., Restrepo, D., 2008. Evolving Solutions: The Genetic Algorithm and Evolution Strategies for Finding Optimal Parameters. Applications of Computational Intelligence in Biology 1, 55–78.

Parsopoulos, K. E., Vrahatis, M. N., 2002. Recent approaches to global optimization problems through particle swarm optimization. Natural computing 1 (2), 235–306.

Pattnaik, S. S., Lohokare, M. R., Devi, S., 2010. Enhanced Biogeography-Based Optimization using Modified Clear Duplicate Operator. IEEE 1, 715–720.

Rattan, M., Patterh, M. S., Sohi, B. S., 2008. Optimization of Yagi-Uda Antenna using Simulated Annealing. Journal of Electromagnetic Waves and Applications, 22 2 (3), 291–299.

Reid, D. G., 1946. The Gain of an Idealized Yagi Array. Journal of the Institution of Electrical Engineers-Part IIIA: Radiolocation, 93 (3), 564–566.

Schaffer, J. D., 1985. Multiple objective optimization with vector evaluated genetic algorithms, 93–100.

Shen, L. C., 1972. Directivity and Bandwidth of Single-band and Double-band Yagi Arrays. IEEE Transactions on Antennas and Propagation, 20 (6), 778–780.

Shi, Y., Eberhart, R. C., 1999. Empirical study of particle swarm optimization 3.

Shi, Y., et al., 2001. Particle swarm optimization: developments, applications and resources 1, 81–86.

Simon, D., 2008. Biogeography-based Optimization. IEEE Transactions on Evolutionary Computation, 12 (6), 702–713.

Singh, S., Mittal, E., Sachdeva, G., 2012a. Multi-objective gain-impedance optimization of yagi-uda antenna using nsbbo and nspso. International Journal 56.

Singh, S., Mittal, E., Sachdeva, G., 2012b. Nsbbo for gain-impedance optimization of yagi-uda antenna design, 856–860.

Singh, S., Sachdeva, G., 2012a. Mutation effects on bbo evolution in optimizing yagi-uda antenna design, 47–51.

Singh, S., Sachdeva, G., 2012b. Yagi-uda antenna design optimization for maximum gain using different bbo migration variants. International Journal 58.

Singh, U., Kumar, H., Kamal, T. S., 2010. Design of Yagi-Uda Antenna Using Biogeography Based Optimization. IEEE Transactions on Antennas and Propagation, 58 (10), 3375–3379.

Singh, U., Rattan, M., Singh, N., Patterh, M. S., 2007. Design of a Yagi-Uda Antenna by Simulated Annealing for Gain, Impedance and FBR. IEEE 1, 974–979.

Srinivas, N., Deb, K., 1994. Muiltiobjective optimization using nondominated sorting in genetic algorithms. Evolutionary computation 2 (3), 221–248.

Stadler, W., 1979. A survey of multicriteria optimization or the vector maximum problem, part i: 1776–1960. Journal of Optimization Theory and Applications 29 (1), 1–52.

Uda, S., Mushiake, Y., 1954. Yagi-Uda Antenna. Maruzen Company, Ltd.

Venkatarayalu, N., Ray, T., 2004. Optimum Design of Yagi-Uda Antennas Using Computational Intelligence. IEEE Transactions on Antennas and Propagation, 52 (7), 1811–1818.

Venkatarayalu, N. V., Ray, T., 2003. Single and Multi-Objective Design of Yagi-Uda Antennas using Computational Intelligence. IEEE 2, 1237–1242.

Wang, H. J., Man, K. F., Chan, C. H., Luk, K. M., 2003. Optimization of Yagi array by Hierarchical Genetic Algorithms. IEEE 1, 91–94.

Yagi, H., 1928. Beam Transmission of Ultra Short Waves. Proceedings of the Institute of Radio Engineers 16 (6), 715–740.

Author Biographies

Satvir Singh was born on Dec 7, 1975. He received his Bachelors degree (B.Tech.) from Dr. B. R. Ambedkar National Institute of Technology, Jalandhar, Punjab (India) with specialization in Electronics & Communication Engineering in year 1998, Masters degree (M.E.) from Delhi Technological University (Formerly, Delhi College of Engineering), Delhi (India) with distinction in Electronics & Communication Engineering in year 2000 and Doctoral degree (Ph.D.) from Maharshi Dayanand University, Rohtak, Haryana (India) in year 2011. During his 13 years of teaching experience he served as Assistant Professor and Head, Department of Electronics & Communication Engineering at BRCM College of Engineering & Technology, Bahal, (Bhiwani) Haryana, India and as Associate Professor & Head, Department of Electronics & Communication Engineering at Shaheed Bhagat Singh State Technical Campus (Formerly, SBS College of Engineering & Technology), Ferozepur Punjab, India.
His fields of special interest include Evolutionary Algorithms, High Performance Computing, Type-1 & Type-2 Fuzzy Logic Systems, Wireless Sensor Networks and Artificial Neural Networks for solving engineering problems. He is active member of an editorial board of International Journal of Electronics Engineering and published nearly 30 research papers in International Journals and Conferences. He has delivered nearly 20 Invited Talks during National and International Conferences, Seminar, Short Term Courses and Workshops. He completed two AICTE funded projects under MODROB Scheme worth 15 Lacs. He has also conducted four Faculty Development Programmes, of total duration of six weeks, around Soft Computing techniques under various schemes of AICTE and TEQIP.

Etika Mittal was born on Oct 26, 1989. She received her Bachelors degree (B.Tech.) from Adesh Institute of Engineering and Technology, Faridkot, Punjab (India) with specialization in Electronics & Communication Engineering in year 2011, and presently pursuing Master's degree from Shaheed Bhagat Singh State Technical Campus (formerly, SBS College of Engineering & Technology) Freozepur, Punjab (India) with specialization in Electronics & Communication Engineering. She has published 6 research papers in International/National Journals and Conferences. Her research interest include Evolutionary Algorithms, Antenna Design Optimization, and Wireless Sensor Networks.