A New Implementation of Population Based Incremental Learning Method for Optimization Studies in Electromagnetics

S.Y. Yang[1], S. L. Ho[2], G. Z. Ni[1], José Márcio Machado[3] and K.F. Wong[2]
[1] College of Electrical Engineering, Zhejiang University, Hangzhou, 310027, China
[2] Department of Electrical Engineering, Hong Kong Polytechnic University, Hong Kong
[3] São Paulo State University (UNESP) - São José do Rio Preto, Brazil
eeslho@polyu.edu.hk

Abstract—To enhance the global search ability of Population Based Incremental Learning (PBIL) methods, it is proposed that multiple probability vectors are to be included on available PBIL algorithms. As a result, the strategy for updating those probability vectors and the negative learning and mutation operators are redefined as reported. Numerical examples are reported to demonstrate the pros and cons of the newly implemented algorithm.

I. A New Implementation of PBIL Methods

Genetic algorithm (GA) is commonly used as the standard algorithm for function optimizations. In a GA algorithm, the three key operators are selection, crossover and mutation. However, these operators are very complex in both theory and numerical implementation. It is thus preferable to design a genetic based algorithm which inherits the searching power of available GAs and excludes the use of, at least partly, the aforementioned operators. In this regard, the PBIL evolution algorithm [1], which is a combination of GA and competitive learning, is a worthy candidate deserving further attention. As similar to a GA, it works on a binary encoded representation of an optimal problem. The salient feature of the algorithm is the introduction of a real valued probability vector from which potential solutions are generated. All the values of this vector is initialized to 0.5 and sampling from this vector will produce a uniform distribution of the initial population, as there is an equal probability in generating "1" or "0" for each binary bit of the chromosome of the solution with this proposal. As the search progresses, the probability vector is expected to shift gradually towards solutions with the highest fitness values.

However, numerical experiences and theoretical analysis have revealed that the PBIL method is often trapped in a local optimum due to a lack of diversity in the population, because only one single probability vector is used to generate the entire population. To address this problem, multiple probability vectors are proposed, i.e., every individual uses different probability vectors to generate its own children. Moreover, the experiences in the neighborhood of an individual are also used to update its probability vector to speed up the convergence of the algorithm. Consequently, after an iteration of a population, the probability vector of individual j is updated using

$$p^j(i) = (1.0 - LR)\cdot p^j(i) + LR\cdot r\cdot best(i) + LR\cdot(1.0 - r)\cdot neighbor\_best(i) \quad (1)$$

where; LR is a positive learning rate; $best(i)$ and $neighbor\_best(i)$ are, respectively, the values of the i^{th} bit of the binary encoded string of the best solutions so far searched by individual j and its neighbor individuals; r is a random parameter chosen from within the interval [0,1].

To move away from the worst individual, a negative learning concept is employed in the proposed multiple probability vectors. For example, for individual j, its probability vector is updated so as to move away from the worst individual using

$$p^j(i) = \begin{cases} p^j(i) & (best(i) = worst(i)) \\ p^j(i)\cdot(1 - NLR) + best(i)\cdot NLR & (best(i) \neq worst(i)) \end{cases} \quad (2)$$

where, $worst(i)$ is the value of the i^{th} bit of the binary encoded string of the worst individual so far searched by individual j, NLR is a negative learning rate.

To further enhance diversity, two different mutation operators, one acts directly on the generated individuals and one on the probability vectors, are incorporated simply. Due to space limits in this abstract, the detailed description of these vectors is reported only in the full paper.

II. Numerical Example

Example 1: It is the mathematical function 2 of [2]. This function has 15^5 local optima.

Example 2: This is the optimal design of the end region of a power transformer [2].

Table I gives the performance comparison of 100 independent runs of the proposed and the original PBIL methods for the mathematical function, and Table II reports the results of the proposed and a tabu search methods for solving the optimal design problem in the end region of a 63 MVA 110 kV power transformer. From these numerical results, it is clear that (1) the global search ability of the proposed PBIL method for solving a very hard mathematical function having 15^5 local optima is increased from 39% to 98%, and (2) the performance of the proposed method is comparable to that of a well developed tabu algorithm in the solution of example 2.

TABLE I
Performance Comparison of Different Optimal Methods on the Mathematical Function for 100 Independent Runs

Algorithms	Averaged iterations	Chance of finding global solution
Common PBIL	2895	39/100
Proposed PBIL	3425	98/100

TABLE II
Optimized Results of a 63 MVA 110 kV Power Transformer

Algorithms	X_1(mm)	X_2(mm)	X_3(mm)	X_4(mm)	Iterations	objective
Proposed	39.9	40.3	85.8	120.5	2032	0.90
Tabu	41.1	39.6	86.1	113.1	1836	0.90

III. References

[1] S. Baluja and R. Caruana, "Removing the genetics from the standard genetic algorithm," *Proc. of ICML95*, 1995, pp. 38-46.
[2] J. M. Machado, S. Y. Yang, S.L. Ho, *Comput. Methods Appl. Mech. Engrg.*, vol.190, pp.3501-3510, 2001.

1-4244-0320-0/06/$20.00 ©2006 IEEE