A Computationally Efficient Vector Optimizer Using Ant Colony Optimizations Algorithm for Multobjective Designs

Siu-Lau Ho[1] and Shiyou Yang[2]

[1]Department of Electrical Engineering, the Hong Kong Polytechnic University, Hong Kong
[2]College of Electrical Engineering, Zhejiang University, Hangzhou 310027, China

An efficient vector optimizer is proposed based on the hybridization of an ant colony optimization method and a novel exploiting search mechanism. To inherit the learning and searching power of an ant colony algorithm while excluding the usage of a tedious and awkward pheromone updating scheme, it is proposed that an algorithm that models the foraging strategy of *pachycodyla apicalis* ants is employed and modified. In order to yield better Pareto solutions, the gradient balance concept is used to design the exploitation search process in which some *a priori* information about the characteristics of the objective functions is used in the selection of nests for subsequent intensifying searches. Numerical experiments are reported to validate the merits and advantages of the proposed vector optimizer for solving practical engineering design problems.

***Index Terms*—Ant colony optimization (ACO), evolutionary computation, multiobjective optimization, optimal design.**

I. Introduction

In MANY scientific and engineering optimization studies, there is a common need for the designer to satisfy several seemingly conflicting criteria/objectives simultaneously. Such problems often require the finding of the best possible designs to satisfy a set of objectives with different tradeoffs. Consequently, the multiobjective optimization study has become a very challenging research topic and has attracted a lot of attention from both academicians and engineers alike. Even though significant progress has been made in multiobjective optimal studies in the last couple of decades, there are still many open issues which must be addressed both qualitatively and quantitatively.

It is well known that the presence of a set of conflicting multiple objectives in design problems will give rise to a set of optimal solutions called the Pareto optimal solutions. Designers are often required to find as many Pareto optimal solutions as possible. Therefore, the search for Pareto optimal solutions for a multiobjective problem involves two opposite objectives, i.e., the minimization of the distance between the searched solutions to the true Pareto front, and the maximization of the diversity among the generated solutions in both objective and parameter spaces. However, these two ultimate goals cannot be realized readily using traditional evolutionary algorithms (EA) and most of these methods have difficulties when dealing with the finding of the best trade-off between uniform distribution and computational burden in the search for near-complete and near-optimal Pareto solutions. Under such situations, some *a priori* information about the characteristics of the objective functions would be helpful to enhance the computational efficiency of a multiobjective optimizer, and such information should preferably be embedded into the algorithm without sacrificing diversity and computation resources. Based on these arguments, an improved ant colony optimization (ACO) algorithm is proposed; an intensification searching phases using the gradient balance concept of [1] is deployed to guide the search procedure to favor exploitations around the neighborhoods of Pareto solutions; and a computationally efficient multiobjective optimizer based on the hybridization of the proposed ACO and a novel search algorithm is finally proposed.

Digital Object Identifier 10.1109/TMAG.2007.914864

II. ACO-Based Vector Optimizer

ACO is an EA to model the behavior of almost blind ants in establishing the shortest path from their colony to their feeding sources and back [2]. As an ant moves, it lays varying amount of pheromones, which are detectable by other ants, along its path, thereby marking the path by a trail of such substances. As more ants pass by, more pheromones are deposited on the path. As the ants chase after pheromones, the richer the trail of pheromones in a path, the more likely it would be followed by other ants. Hence, ants can establish the shortest way from their colony to the feeding sources and back. Moreover, the collective behavior of ant colonies exhibits the so-called "autocatalytic" characteristics [3]. By virtue of the learning and searching power of an autocatalytic self-organization such as an ant colony, ACO-based algorithms are receiving increasing attentions and have enjoyed great success in solving traditionally difficult optimization problems [2], [3].

However, even the available ACO algorithms could search the best solutions in the simulated cases, their search procedures are rather complex in terms of conceptualization and numerical implementations. For example, the tedious pheromone updating scheme is explicitly included in conventional ACO algorithms. Moreover, most of these applications are confined to optimal problems having a single objective only. In this regard, a model based on the foraging strategy of the *pachycondyla APIcalis* (API) ant [4], which inherits the learning and searching power of an ant colony while excluding the usage of tedious and awkward pheromone updating scheme, is employed and modified for the multiobjective design study of electromagnetic devices.

A. Brief Introduction of API Algorithm

Pachycondyla apicalis ants are found in the Mexican tropical forest near the Guatemalan boarder. Colonies of these ants comprise of around 20 to 100 ants. The foraging strategy of such ants can be characterized as follows. First, these ants create their own hunting sites which are distributed relatively uniformly around their nest within a radius of approximately 10 m. In this way, using a small mosaic of areas, the ants cover a rather large region around their nest. Second, the ants will intensify their searches for prey around some selected sites. In this foraging process, the ants communicate with each other using visual landmarks rather than pheromone trails. After capturing their prey, the ants will move to a new nest based on a recruitment mechanism called tandem running to begin a new cycle of foraging. Generally, these ants use relatively simple principles, both locally and globally, in their search for prey. Based on such a metaphor, Monmarche *et al.* proposed an API algorithm for the solution of optimization problems as reported in [4].

B. Fitness Assignment

To assess a solution in a Pareto optimal sense, the ranking concept used in a genetic algorithm is introduced to assign the fitness value to a solution [5] in this paper. Based on our extensive simulation analysis, it is found that, after the introduction of the fitness sharing functions, in both parameter and objective spaces which are used for preserving the diversity of the searched Pareto solutions, especially at which the point density of a dominated solution is far smaller than that of a Pareto solution, the total fitness value of the dominated solution may become larger than that of the Pareto solution. In such cases, the dominated solution will have a very high probability of being selected as the new best solution to begin the next iterative cycle, thereby giving rise to an inefficient algorithm. To overcome such drawbacks of conventional ranking approach, a new rank formula to determine the rank of a solution x^i is proposed as

$$\text{Rank}(x^i) = 1/3 + p_i \tag{1}$$

where p_i is the number of solutions which dominates solution x^i.

C. Exploitation Search

To find high-quality Pareto solutions efficiently, an exploitation search is designed to intensify the searches around some promising solutions which could yield better Pareto solutions. To overcome the shortcomings of an API algorithm that has poor usage of memory information, which is a common drawback characterizing ant colony systems, every ant in the proposed algorithm memorizes and uses its best and second best solutions, i.e., using *a posteriori* knowledge about the explored space and individuals that have been searched so far in the current exploiting phase to update its position around a specific nest in this search process. For example, if the nest selected by ant i in the current exploiting search is denoted by x_N^i, the position updating mechanism for ant i is proposed as

$$v_j^i = r_1 \cdot \left(x_j^i\right)_{\text{Best}} + (1 - r_1) \cdot \left(x_j^i\right)_{\text{Second-best}} \tag{2}$$

$$x_j^i = r_2 \cdot \left(x_N^i\right)_j + (1 - r_2) \cdot v_j^i \tag{3}$$

where $\left(x_j^i\right)_{\text{Best}}$ and $\left(x_j^i\right)_{\text{Second-best}}$ are, respectively, the jth component of the best and second best solutions so far searched by ant i in the current exploitation search; r_1 and r_2 are two random parameters within the interval of [0, 1].

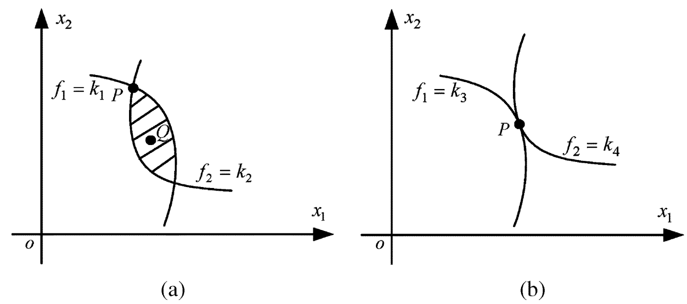

Fig. 1. Illustration of an intersection point for a design problem comprising of two decision parameters and two objective functions.

D. Selection of Nests for Starting Exploiting Search

In most common multiobjective optimizers, only the information gathered from the so far searched solutions, the *a posteriori* information, is used to guide the search toward more promising solutions. However, if some *a priori* knowledge about the unexplored space and individuals are available and could be used, the search efficiency and quality of the solutions of the optimizer could be improved. In this regard, some *a priori* knowledge about the search space and individuals, which are based on the concept of the gradient balance [1], are employed to identify a nest that has the potential to yield promising solutions in the exploitation search.

To give an illustrative explanation about the gradient balance concept, one considers the minimization of an optimal problem involving two decision parameters $x = (x_1, x_2)$ and two objective functions $(f_1(x), f_2(x))$. Mathematically, if the contour lines of f_1 and f_2 in the parameter space having different convexities are intersecting at a feasible point P as shown in Fig. 1(a), there will be at least one point Q in the feasible space which is better than point P for both objective functions. By repeatedly and symmetrically reducing the values of both functions f_1 and f_2, it is possible to locate a Pareto optimal point at the tangential point between the contour lines of the two objectives, as shown in Fig. 1(b). In order to characterize the feature of an intersection point of the contour lines of the two objectives in the feasible parameter space, one defines the preference function as

$$p(\overline{x}) = \frac{\nabla f_1(x)}{\|\nabla f_1(x)\|} \bullet \frac{\nabla f_2(x)}{\|\nabla f_2(x)\|} + 1. \tag{4}$$

Geometrically, the function $p(x)$ reaches its minimum value of zero at all Pareto optimal points. In other words, this function provides a "gauging factor" that measures the "closeness" of a feasible point to the Pareto optimal solutions of the optimal problem. Therefore, if one selects the starting nests using a Roulette wheel selection scheme with probabilities which are inversely proportional to the function values as defined in (4), the subsequent exploitation search will intensify searches around points which are close to the exact Pareto solutions, equipping the algorithm with an ability to find better Pareto solutions.

For the general case of a multiobjective optimal problem, in which the number of objective functions is more than 2, the preference functions are defined in a "pairwise" sense for every pair of objectives. For example, for the ith objective function f_1 and the jth objective function f_j, their corresponding pairwise defined preference function is formulated as

$$p_{ij}(x) = \frac{\nabla f_1(x)}{\|\nabla f_1(x)\|} \bullet \frac{\nabla f_1(x)}{\|\nabla f_1(x)\|} + 1. \quad (5)$$

Consequently, the probability for a solution x^m to be selected as the starting nest for exploitation searches is generally given as

$$\text{Probability}(x^m) = \frac{\frac{1}{\sum\limits_{i\neq j}(p_{ij}(x^m)+\alpha)^\beta}}{\sum\limits_{k=1}^{N_{\text{nest}}} \frac{1}{\sum\limits_{i\neq j}(p_{ij}(x^k)+\alpha)^\beta}} \quad (6)$$

where N_{nest} is the number of current nests; α and β are two positive constants.

E. Iterative Procedures of the Proposed Algorithm

The iterative procedures of the proposed algorithm can be explained step by step as follows.

Step 1) Initialization: Set up the algorithm parameters, initializes the nests and the external Pareto set.

Step 2) Select a nest according to the probability of (6), and start the exploitation search.

2.1) Intensifying search: For each ant, intensify the searches around the selected nest using (2) and (3).

2.2) Information sharing: for every ant, update its best and second best solutions, i.e., $(x^i)_{\text{best}}$ and $(x^i)_{\text{Second-best}}$ according to the solutions of its own and those of its neighbors.

2.3) Nest movement: If the condition for nest movements is satisfied, go to Step 3); otherwise, go to Step 2.1).

Step 3) Generation of new nest (exploration): use a mechanism which is similar to that which is used in a tabu search method to create some new nests N_{new}, and then update the nests;

Step 4) Terminate test. If the test is passed, stop; Otherwise, go to Step 2).

It should be pointed out that the neighborhood as used in Step 2.2) is defined in a topological sense.

III. Numerical Examples

To explore the applicability and compare the performances of the proposed algorithm with other multiobjective optimizers for engineering design problems, the optimal design of the multisectional pole arcs of a large hydrogenerator in [6] is selected and solved. This problem is formulated as

$$\begin{aligned} \max \quad & B_{f1}(X) \\ \min \quad & (e_v, \text{THF}) \\ \text{s.t.} \quad & \text{SCR} - \text{SCR}_0 \geq 0 \\ & x_d' - X_d' \leq 0 \end{aligned} \quad (7)$$

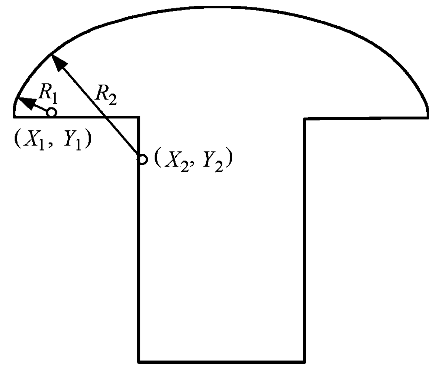

Fig. 2. Schematic diagram of the decision parameters of the multisectional pole shoes being studied using the proposed algorithm.

TABLE I
PERFORMANCE COMPARISON OF DIFFERENT ALGORITHMS

Algorithm	Algorithm A	Algorithm B	Algorithm C	SA	PSO
No. of average iterations	1535	1824	1789	1786	1678
No. of average Pareto solutions	415	402	399	445	410

where B_{f1} is the amplitude of the fundamental component of the flux density in the air gap at no-load condition; e_v is the distortion factor of a sinusoidal voltage of the machine on no-load; THF is the abbreviations of the Telephone Harmonic Factor; X_d' is the direct axis transient reactance of the generator; SCR is the abbreviations of the short-circuit ratio.

The decision parameters of this case study are the center positions and radii of the multisectional arcs of the pole shoes as depicted in Fig. 2. For the purpose of performance comparison, this case study is solved by using, respectively, the proposed method, a simulated annealing (SA) vector algorithm [6], and a particle swarm optimization (PSO)-based vector optimizer [7]. In the numerical experiments, the performance parameters required in (7) are determined by using finite element analysis. To investigate extensively the performances of the proposed algorithm, it is implemented in three different ways: Algorithm A—in which all of the improvements made in this paper are included, Algorithm B—in which the nests for the exploiting searches are selected randomly, and Algorithm C—in which the information about the best and second best solutions so far searched by an ant in the current exploitation search is not employed to update its position around the selected nest. Under such conditions, the performance comparisons of the aforementioned five algorithms on solving the multisectional pole arcs design of a 300-MW hydrogenerator are given in Table I. For each algorithm, the iteration number given in this table is the averaged value of five independent runs. The searched Pareto solutions in the objective space for a typical run by using, respectively, the proposed, the SA, and the PSO algorithms are shown in Figs. 3–5. From these primary numerical results, one observes the following.

1) The proposed, the SA- and the PSO-based vector optimizers have all yielded smooth and uniform distributions of the searched Pareto solutions.

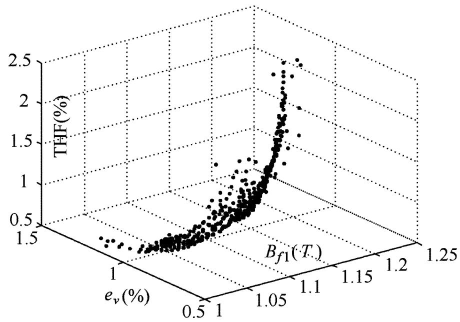

Fig. 3. Searched Pareto solutions by using the proposed algorithm.

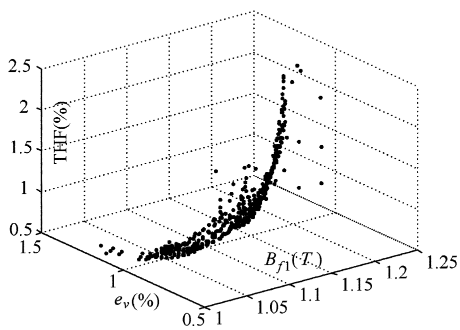

Fig. 4. Searched Pareto solutions by using the SA-based method.

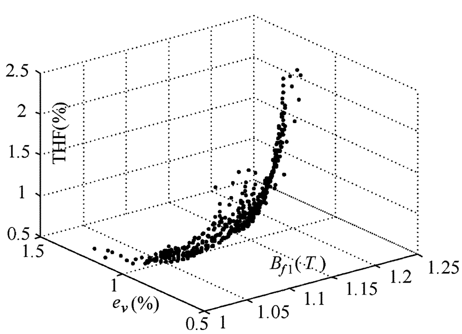

Fig. 5. Searched Pareto solutions by using the PSO-based method.

2) The proposed algorithm is the most efficient one among the five vector optimizers being studied.
3) Excluding the usage of either the *a posteriori* information (Algorithm C) or the *a prior* knowledge (Algorithm B) will slow down the convergence speed, and at the same time degrading slightly the quality of the searched Pareto solutions.

IV. Conclusion

To provide a robust and efficient multiobjective optimal tool for electromagnetic design problems, this paper introduces an ant colony-based vector optimizer in which both the *a posteriori* information about the searched solutions and the *a priori* knowledge about the unexplored parameter space are used to guide the search toward more and better Pareto solutions. The performances of the proposed algorithm are extensively studied and compared with other well developed vector optimizers such as simulated annealing and particle swarm optimization-based algorithms on a practical engineering design problem. The numerical results have shown that the proposed ACO-based vector algorithm is comparable to the SA- and PSO-based methods in terms of solution quality, and it outperforms marginally the SA and PSO algorithms in terms of computational efficiency. For further work along this direction, the authors will strive to develop a surrogate model to approximate the searched Pareto front and the efficient approaches to determine the gradient information as required in the proposed algorithm. Also, further efforts shall be devoted to the optimal tuning of parameters to help promoting the proposed algorithm to become a band vector optimizer.

Acknowledgment

This work was supported by the Polytechnic University, under Project 1-BBZ7 and by the Research Grant Council of the Hong Kong Special Administrative Region, China, under Project PolyU 5242/04E.

References

[1] P. Di Barba, "Multiobjective design optimization: A microeconomics-inspired strategy applied to electromagnetcis," *Int. J. Appl. Electromagn. Mechan.*, vol. 21, pp. 101–117, 2005.
[2] M. Dorigo, V. Maniezzo, and A. Colorni, "Ant system: Optimization by a colony of cooperative agents," *IEEE Trans. Syst., Man, Cybern., B Cybern.*, vol. 26, no. 1, pp. 29–41, 1996.
[3] G. Bilchev and I. C. Parmee, "The ant colony metaphor for searching continuous design spaces," in *Lecture Notes Computer Science*. Berlin, Germany: Springer-Verlag, 1995, vol. 993, pp. 25–39.
[4] N. Monmarche, G. Venturini, and M. Slimane, "On how pachcondyla apicalis ants suggest a new search algorithm," *Future Generation Comput. Syst.*, vol. 16, pp. 937–946, 2000.
[5] C. A. Coello-Coello, "An updated survey of GA-based multiobjective optimization techniques," *ACM Comput. Surv.*, vol. 32, pp. 109–143, 2000.
[6] S. L. Ho, S. Y. Yang, H. C. Wong, and G. Z. Ni, "A simulated annealing algorithm for multiobjective optimizations of electromagnetic devices," *IEEE Trans. Magn.*, vol. 39, pp. 1285–1288, 2003.
[7] S. L. Ho, S. Y. Yang, G. Z. Ni, E. W. C. Lo, and H. C. Wong, "A particle swarm optimization based method for multiobjective design optimizations," *IEEE Trans. Magn.*, vol. 41, pp. 1756–1759, 2005.

Manuscript received June 24, 2007. Corresponding author: S. L. Ho (e-mail: eeslho@polyu.edu.hk).