# A Simulated Annealing Algorithm for Multiobjective Optimizations of Electromagnetic Devices

S. L. Ho, Shiyou Yang, H. C. Wong, and Guangzheng Ni

*Abstract*—**This paper proposes a simulated annealing algorithm for multiobjective optimizations of electromagnetic devices to find the Pareto solutions in a relatively simple manner. The algorithm is based on the successful introductions of the Pareto set as well as the parameter and objective space strings. The new rank formula and fitness-sharing functions together with the stop criterion and other improvements are investigated in the proposed method. Two numerical examples for validating the robustness of the proposed method are also reported.**

***Index Terms*—Design optimization, grid algorithm, multiobjective optimization, simulated annealing.**

## I. Introduction

THE MULTIOBJECTIVE or vector optimization is a very important research area in engineering studies because real world design problems require the optimization of a group of objectives. Thanks to the effort of scientists and engineers during the last two decades, particularly the last decade, a wealth of multiobjective optimizers have been developed, and some multiobjective optimization problems that could not be solved hitherto were successfully solved by using these optimizers. In terms of robustness and efficiency of the available vector optimizers, these optimizers are still in need of improvements and hence there are many unresolved open problems [1]. It is also observed that most of the multiobjective optimization studies have been focused on evolutionary algorithms. However, a more thorough research reveals that the performances of evolutionary algorithms are often overshadowed by local search methods, such as simulated annealing or tabu search that are generally quite complex [2]. In this paper, a simple multiobjective optimizer based on simulated annealing algorithms is proposed. Without a loss of generality, the following minimization problem is considered:

$$\min \bar{f}(\bar{x}) \quad (\bar{x} \in X) \tag{1}$$

$$X = \{\bar{x} \in E^n | \bar{g}(\bar{x}) \geq 0, \bar{h}(\bar{x}) = 0\} \tag{2}$$

where $\bar{x} = [x_1 \;\; x_2 \;\; \cdots \;\; x_n]^T$, $\bar{f}(\bar{x}) = [f_1(\bar{x}) \; f_2(\bar{x}) \cdots f_k(\bar{x})]^T$, $\bar{g}(\bar{x}) = [\bar{g}_1(\bar{x})\bar{g}_2(\bar{x}) \cdots \bar{g}_m(\bar{x})]^T$, and $\bar{h}(\bar{x}) = [\bar{h}_1(\bar{x})\bar{h}_2(\bar{x}) \cdots \bar{h}_p(\bar{x})]^T$.

Manuscript received June 18, 2002.

S. L. Ho is with the Department of Electrical Engineering, Hong Kong Polytechnic University, Hong Kong (e-mail:eeslho@polyu.edu.hk).

S. Yang and G. Ni are with the Electrical Engineering College, Zhejiang University, Hangzhou 310027, China (e-mail: shiyouyang@yahoo.com; nigz@cee.zju.edu.cn).

H. C. Wong is with the Industrial Center, Hong Kong Polytechnic University, Hong Kong (e-mail:ichcwong@polyu.edu.hk).

Digital Object Identifier 10.1109/TMAG.2003.810546

When working with multiobjective optimal problems, it is desirable to define some concepts or terminologies for the convenience of the potential users. The main terminologies used in this paper are therefore given as follows.

### A. Weakly (Strongly) Dominated and Nondominated Solutions

A solution $\bar{x}^*$ is a weakly nondominated solution if there is no $\bar{x} \in X(\bar{x} \neq \bar{x}^*)$ such that $f_i(\bar{x}) < f_i(\bar{x}^*)$ for $i = 1, 2, \ldots, k$; otherwise, solution $\bar{x}^*$ is a weakly dominated solution (a solution $\bar{x}^*$ is a strongly nondominated solution if there is no $\bar{x} \in X(\bar{x} \neq \bar{x}^*)$ such that $f_i(\bar{x}) \leq f_i(\bar{x}^*)$ for $i = 1, 2, \ldots, k$ and for at least one index of $i$ such that $f_i(\bar{x}) < f_i(\bar{x}^*)$; otherwise, the solution $\bar{x}^*$ is a strongly dominated solution).

### B. Pareto Optimal (Solution or Front)

The strongly and weakly nondominated solutions constitute the total Pareto front of a multiobjective optimization problem.

A qualitative demonstration of the Pareto front for minimizing two objectives is shown in Fig. 1.

## II. Multiobjective Simulated Annealing Algorithm

An ideal multiobjective optimal tool should have the ability to find and to sample the Pareto solutions uniformly. To achieve these two goals, different approaches for designing a robust multiobjective optimizer based on a simulated annealing algorithm of single objectives is proposed. To facilitate the understanding, an iterative procedure of the proposed algorithm is firstly given as:

Empty the Pareto set $S_{\text{Pareto}}$, set the control parameter $T_0$;
Randomly generate a feasible solution $\bar{x}$;
Repeat
 Generate randomly $\bar{y} \in X$ from $\bar{x}$, evaluate the fitness values of solution $\bar{x}$ and solution $\bar{y}$;
 Accept $\bar{y}$ with probability

$$\min\left\{1, \;\; \exp\left(\frac{(fit(\bar{y}) - fit(\bar{x}))}{T}\right)\right\};$$

 If $\bar{y}$ is not dominated by $\bar{x}$, adjust the $S_{\text{Pareto}}$;
 If the control parameter is reduced, set $T = \alpha T$;
Until the termination criterion is satisfied.

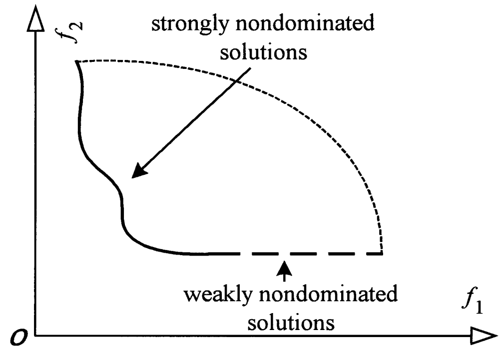


Fig. 1. Weakly and strongly nondominated solutions for minimizing a two objective case.

### A. Introduction of Pareto Set $S_{\mathrm{Pareto}}$

To report the searched Pareto solutions, a Pareto set $S_{\mathrm{Pareto}}$ is introduced in the proposed algorithm. The length of the set is finite, and its quantities are dynamically updated in the search process. To maintain the diversity of the searched Pareto solutions in both parameter and objective spaces, the solutions in the Pareto set is used in the computation of the point density around a specific point in the determination of its sharing function value.

### B. Introduction of Parameter and Objective Strings

To reduce the CPU time when computing the distances among different solutions and to evaluate the densities of a solution in the fitness sharing procedure as well as to develop an efficient multiobjective optimizer, the objective and parameter spaces are first divided into separate discrete grids according to user predefined precision parameters. Each of the discrete grids is recorded as a binary string. When a solution is identified as a Pareto solution, its location in the grids, for example in the parameter space grid, is determined by repeatedly bisecting the range of it in each direction and to identify the specific half range that contains the solution. The corresponding bit of the strings is then set to a logical 1. Since the maximum and the minimal values of an objective in optimal problems are generally unattainable before the optimization process is carried out in real problems, a reference point, giving a base value of the objective vector and a pointer to identify the position of this reference point in the objective string, is used in the evaluation of the grid location of an arbitrary Pareto solution in the objective grid. By introducing the two strings for both the parameter and objective grids, the computation of the point densities and the evaluation and reporting of the searched Pareto solutions can be done simply and efficiently. It should also be noted that the proposed strings are vital in maintaining a constant number of the searched Pareto solutions in the final stage of the search process, particularly where the Pareto solutions are found in different disconnected subregions in the parameter spaces. Moreover, these strings are very important in the development of an efficient stop criterion for the proposed algorithm.

### C. Fitness Value

To decide if a solution is to be taken as the current one, some techniques must be designed to scale the vector objectives. For this purpose, the ranking concept used in a genetic algorithm is introduced to assign a fitness value to a solution [3]. Extensive simulation results show that after the introduction of the fitness sharing function, especially in both the parameter and objective spaces for preserving the diversity of the searched Pareto solutions where the point density of a dominated solution is far smaller than that of a Pareto solution in some cases, the total fitness value of the dominated solution may be larger than that of the Pareto solution. Hence, the dominated solution will have a very high probability to be selected as the most current one to begin the next iterative cycle, giving rise to an inefficient algorithm. To overcome such drawbacks of the conventional ranking approach, a new rank formula to decide the rank of a solution $\bar{z}$ is proposed as

$$\mathrm{Rank}(\bar{z}) = \frac{1}{3} + p_i \tag{3}$$

where $p_i$ is the number of solutions in $S_{\mathrm{Pareto}}$ which dominates the solution $\bar{z}$.

To guarantee uniform distributions of the searched Pareto solutions in both the parameter and objective spaces, the fitness sharing concept is extended [4]. To address the densities of the searched Pareto solutions around a specified point $\bar{z}$, the following simple but efficient fitness sharing function is proposed:

$$f_{\mathrm{share}}(\bar{z}) = \frac{\frac{1}{d_f(\bar{z})}}{\frac{1}{d_f(\bar{y})} + \frac{1}{d_f(\bar{x})}} + \frac{\frac{1}{d_X(\bar{z})}}{\frac{1}{d_X(\bar{y})} + \frac{1}{d_X(\bar{x})}} \tag{4}$$

where $d_u(\bar{z})$ $(u = f, X; z = x, y)$ is the density of Pareto solutions found around $\bar{z}$ in the $u$ space.

The fitness value of the solution $\bar{z}$ is then given by

$$f_{\mathrm{fit}}(\bar{z}) = \frac{1}{\mathrm{Rank}(\bar{z})} + f_{\mathrm{share}}(\bar{z}). \tag{5}$$

The minimal value of the rank of a dominated solution is 4/3, and the value of the rank of a nondominated one is constant at 1/3. The total fitness value of a nondominated solution is still larger than that of the dominated solutions, even in the worst case where the maximal value of the sharing functions, which is 2, is assigned to the dominated solutions while the minimal value of the sharing functions of 0 is assigned to this nondominated one. The proposed approach for fitness assignments is thus relatively superior and efficient.

To determine the density of the Pareto optimals for a specific point, one simply needs to account for the number of logical "*1*"s in the bits of the aforementioned two strings that correspond to the neighborhood points of the specific grid point. Thus, by introducing the parameter and objective strings as proposed, the point densities evaluation is done efficiently and simply.

### D. Evaluation and Reporting the Pareto Solutions

In the optimization process, the Pareto optimal set is updated automatically in the proposed algorithm. To simplify the description, let $\bar{x}^{(l)}$ be a new solution to be considered. In the set of the Pareto solution, there is a solution $\bar{x}_j^p$ such that:

1) if $\bar{x}_j^p$ is dominated by $\bar{x}^{(l)}$, then $\bar{x}_j^p$ is substituted by $\bar{x}^{(l)}$; Set the bit of the strings for $\bar{x}_j^p$ to zero, and set the bit of the strings for $\bar{x}^{(l)}$ to 1s;
2) if $\bar{x}^{(l)}$ is dominated by $\bar{x}_j^p$, then $\bar{x}^{(l)}$ is discarded;
3) if none in the Pareto optimal set satisfies (1) or (2), then $\bar{x}^{(l)}$ becomes the current Pareto solution, and in this case:
   i) if the bit value of this solution in either the parameter or objective strings is not a logic 1, add this solution to the Pareto set; set the bits of the strings for this solution to logic 1;
   ii) otherwise, discard the solution.

By using the new parameter and objective strings, the evaluation and reporting of the new Pareto solutions become straightforward in the proposed algorithm. In the final stage of the searching process, the number of the searched Pareto solutions, which are now distributed uniformly in both the parameter and objective spaces, will also become constant.

### E. Start of New Iterations

To keep the diversity of the searched Pareto solutions after the iteration of every control parameter, the proposed algorithm will always restart from a randomly generated point which is far away from the current one in the Pareto set rather than from the last accepted point.

### F. Stop Criterion

By using a combination of the aforementioned two binary strings as well as the new evaluation and reporting schemes, the searched Pareto solutions will become constant if enough searches are executed in the searching process. Thus, a very simple stop criterion is proposed and used in this paper, i.e., the algorithm will stop the iterative process automatically if the mean square value of the total searched Pareto optimal solutions is constant for a specific number of iterative cycles.

## III. Numerical Validation

To validate the proposed algorithm and to compare it with other well known multiobjective optimizers, i.e., the genetic algorithm using uniform design (UGA) [5] and the hybrid genetic algorithm (HGA) [6], a test function as follows is selected and solved:

$$\begin{aligned} \min \quad & f_1(x,y) = \frac{1}{x^2+y^2+1} \\ \min \quad & f_2(x,y) = x^2+3y^2+1 \\ \text{s.t.} \quad & -3 \le x \le 3, 5 \le y \le 5. \end{aligned} \tag{6}$$

For this test function, the proposed algorithm is independently run ten times, and a comparison of the average performances with those of UGA and HGA is summarized in Table I. The searched Pareto solutions in the objective and parameter spaces for a typical run are, respectively, given in Figs. 2 and 3. Comparing Fig. 2 with [5, Fig. 6], one can see that: 1) although the number of the searched Pareto solutions of the proposed algorithm is less than that of UGA, the searched Pareto solutions have a more uniform distribution in the objective space and 2) the number of the searched Pareto solutions using the proposed algorithm is almost double of that using HGA. From the performance comparison of Table I, it can be seen that the proposed algorithm is the most efficient one among the three multiobjective optimizers. Although the Pareto solutions of this problem are located in different disconnected subregions in the parameter space (Fig. 3), the proposed stop criterion can still work very well and produces uniform distribution of the searched Pareto solutions in both the parameter and objective spaces, thereby illustrating the advantages of introducing the two binary strings.

TABLE I
PERFORMANCE COMPARISON OF THE PROPOSED ALGORITHM WITH AVAILABLE ONES FOR THE TEST FUNCTION

| Method | Number of found Pareto solutions | Number of function evaluations |
|---|---|---|
| Proposed algorithm | 213 | 1642 |
| UGA | 298 | 1842 |
| HGA | 112 | 11975 |

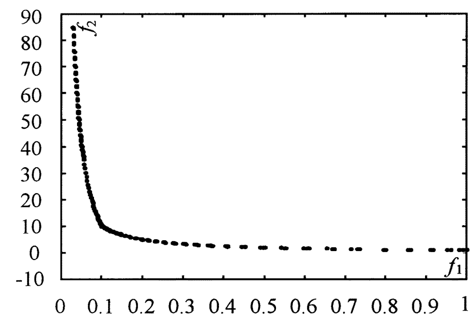


Fig. 2. The searched Pareto solution in the objective space for the test function.

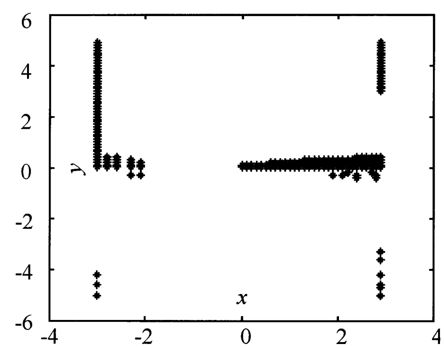


Fig. 3. The searched Pareto solution in the parameter grid for the test function.

## IV. Application

The single objective design problem of the geometrical optimization of the multisectional pole arcs of large hydrogenerators [7] is extended to include two additional objectives to demonstrate the usefulness of the proposed algorithm in solving engineering multiobjective design problems. Mathematically, the problem is formulated as

$$\begin{aligned} \max \quad & B_{f1}(X) \\ \min \quad & (e_v, THF) \\ \text{s.t} \quad & SCR - SCR_0 \ge 0 \\ & X'_d - X'_{d0} \le 0 \end{aligned} \tag{7}$$

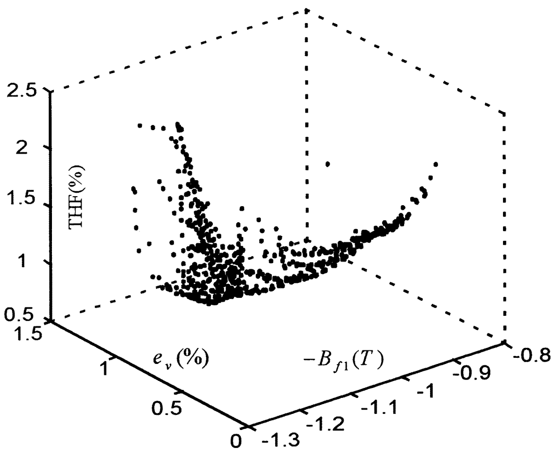


Fig. 4. The searched Pareto solutions of a 300-MW hydrogenerator in the objective space.

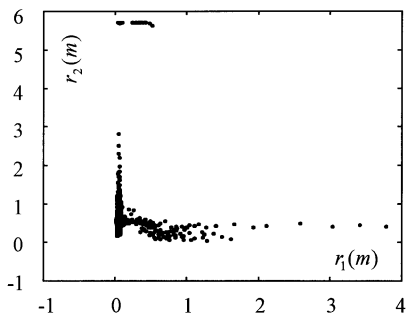


Fig. 5. The searched Pareto solutions of the 300-MW hydrogenerator in the $r_1or_2$ plane grid.

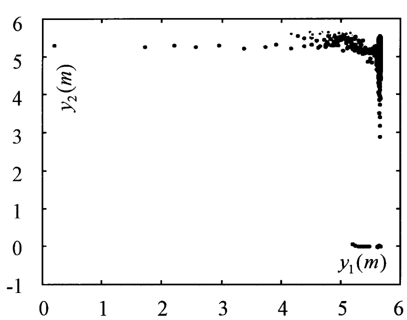


Fig. 6. The searched Pareto solutions of the 300-MW hydrogenerator in the $y_1oy_2$ plane grid.

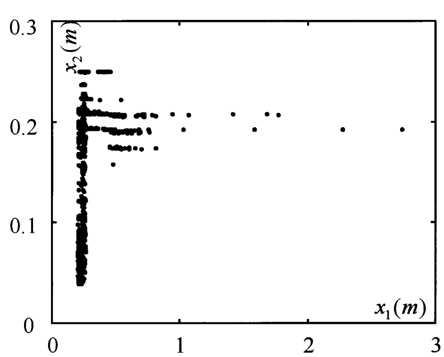


Fig. 7. The searched Pareto solutions of the 300-MW hydrogenerator in the $x_1ox_2$ plane grid.

where $B_{f1}$ is the amplitude of the fundamental component of the flux density in the air gap, $e_v$ is the distortion factor of a sinusoidal voltage of the machine running on no-load, *THF* is the abbreviation of the Telephone Harmonic Factor, $X_d'$ is the direct axis transient reactance of the motor, and *SCR* is the abbreviation of the short circuit ratio.

The corresponding geometrical parameters to be optimized are the center positions $((x_1, y_1), (x_2, y_2))$ and radii $(r_1, r_2)$ of the multisectional arcs of the pole shoes. The 628 searched Pareto solutions of a 300-MW 44-pole hydrogenerator in the objective space are given in Fig. 4. To visually demonstrate the distribution of the searched Pareto solutions in the parameter space, Figs. 5–7 show, respectively, their distributions in the $r_1or_2$, $y_1oy_2$, $x_1ox_2$ planes. From Figs. 5–7, one can see that the distribution of the Pareto solutions in the parameter space for this problem is very irregular and located in some disconnected subregions. In other words, the specific design problem is hard to solve due to the difficulties in finding and uniformly sampling the Pareto solutions. Nevertheless, the numerical results reported in Figs. 4–7 show that the proposed algorithm would not only simply find, but also can uniformly sample, the Pareto solutions in both spaces. Thus, the proposed method is very robust in solving complex multiobjective optimization problems of the real world.

## V. Conclusion

A simulated annealing-based algorithm for multiobjective optimal designs of electromagnetic devices is proposed in this paper. The numerical results reveal that the proposed algorithm can find and sample uniformly the Pareto solutions of multiobjective problems very efficiently and is thus very suitable for engineering multiobjective optimal problems, especially for problems where the Pareto solutions are located in different disconnected subregions in the parameter spaces.

## References

[1] E. Zitzler, K. Deb, and L. Thiele, "Comparison of multiobjective evolutionary algorithms: Empirical results," *Evolutionary Computation*, vol. 8, pp. 173–195, 2000.

[2] J. D. Knowles and D. W. Come, "Approximating the nondominated front using the Pareto archived evolution strategy," *Evolution Computation*, vol. 8, pp. 149–172, 2000.

[3] D. E. Goldberg, *Genetic Algorithms in Search, Optimization & Machine Learning*. Reading, MA: Addision-Wesley, 1989.

[4] C. A. Coello-Coello, "An updated survey of GA-based multiobjective optimization techniques," *ACM Computing Surveys*, vol. 32, pp. 109–143, June 2000.

[5] Y. W. Leung and Y. Wang, "Multiobjective programming using uniform design and genetic algorithm," *IEEE Trans. Systems, Man Cybern.-Part C*, vol. 30, pp. 293–304, 2000.

[6] H. Ishibuchi and T. Murata, "A multi-objective genetic local search algorithm and its application to flowshop scheduling," *IEEE Trans. Systems, Man Cybern.-Part C*, vol. 28, pp. 392–403, 1998.

[7] R. Y. Tang *et al.*, "Combined strategy of improved simulated annealing algorithm with genetic algorithm for inverse problem," *IEEE Trans. Magn.*, vol. 32, pp. 1326–1329, July 1996.