Review Article

Harmony Search Method: Theory and Applications

X. Z. Gao,[1] V. Govindasamy,[2] H. Xu,[3] X. Wang,[1] and K. Zenger[1]

[1]*Department of Electrical Engineering and Automation, Aalto University School of Electrical Engineering, 00076 Aalto, Finland*
[2]*Department of Information Technology, Pondicherry Engineering College, Pondicherry 605 014, India*
[3]*College of Mechanical and Electrical Engineering, Harbin Engineering University, Harbin 150001, China*

Correspondence should be addressed to V. Govindasamy; vgopu@pec.edu

The Harmony Search (HS) method is an emerging metaheuristic optimization algorithm, which has been employed to cope with numerous challenging tasks during the past decade. In this paper, the essential theory and applications of the HS algorithm are first described and reviewed. Several typical variants of the original HS are next briefly explained. As an example of case study, a modified HS method inspired by the idea of Pareto-dominance-based ranking is also presented. It is further applied to handle a practical wind generator optimal design problem.

1. Introduction

Firstly proposed by Geem et al. in 2001 [1], the Harmony Search (HS) method is inspired by the underlying principles of the musicians' improvisation of the harmony. The HS has the distinguishing features of algorithm simplicity and search efficiency. During the recent years, it has been successfully used in areas such as function optimization [2], mechanical structure design [3], pipe network optimization [4], and optimization of data classification systems [5]. In this paper, we first introduce the underlying inspiration and principles of the basic HS method in Section 2. Some representative modified HS algorithms are next explained in Section 3. The applications of the HS in various real-world areas are surveyed and reviewed in the following section. In Section 4, to demonstrate an illustrative example, we present and discuss a new HS method for the constrained optimization with application in the wind generator design. Finally, in Section 5, the paper is concluded with some remarks and conclusions.

2. Harmony Search Method

As we know, when musicians compose the harmony, they usually try various possible combinations of the music pitches

stored in their memory. This search for the perfect harmony is indeed analogous to the procedure of finding the optimal solutions to engineering problems. The HS method is actually inspired by the working principles of the harmony improvisation [1]. Figure 1 shows the flowchart of the basic HS method, in which there are four principal steps involved. The pseudocode of the HS is given in Algorithm 1.

Step 1. Initialize the HS Memory (HM). The initial HM consists of a certain number of randomly generated solutions to the optimization problems under consideration. For an n-dimension problem, an HM with the size of N can be represented as follows:

$$\text{HM} = \begin{bmatrix} x_1^1, x_2^1, \ldots, x_n^1 \\ x_1^2, x_2^2, \ldots, x_n^2 \\ \vdots \\ x_1^{\text{HMS}}, x_2^{\text{HMS}}, \ldots, x_n^{\text{HMS}} \end{bmatrix}, \qquad (1)$$

where $[x_1^i, x_2^i, \ldots, x_n^i]$ $(i = 1, 2, \ldots, \text{HMS})$ is a solution candidate. HMS is typically set to be between 50 and 100.

Step 2. Improvise a new solution $[x_1', x_2', \ldots, x_n']$ from the HM. Each component of this solution, x_j', is obtained based
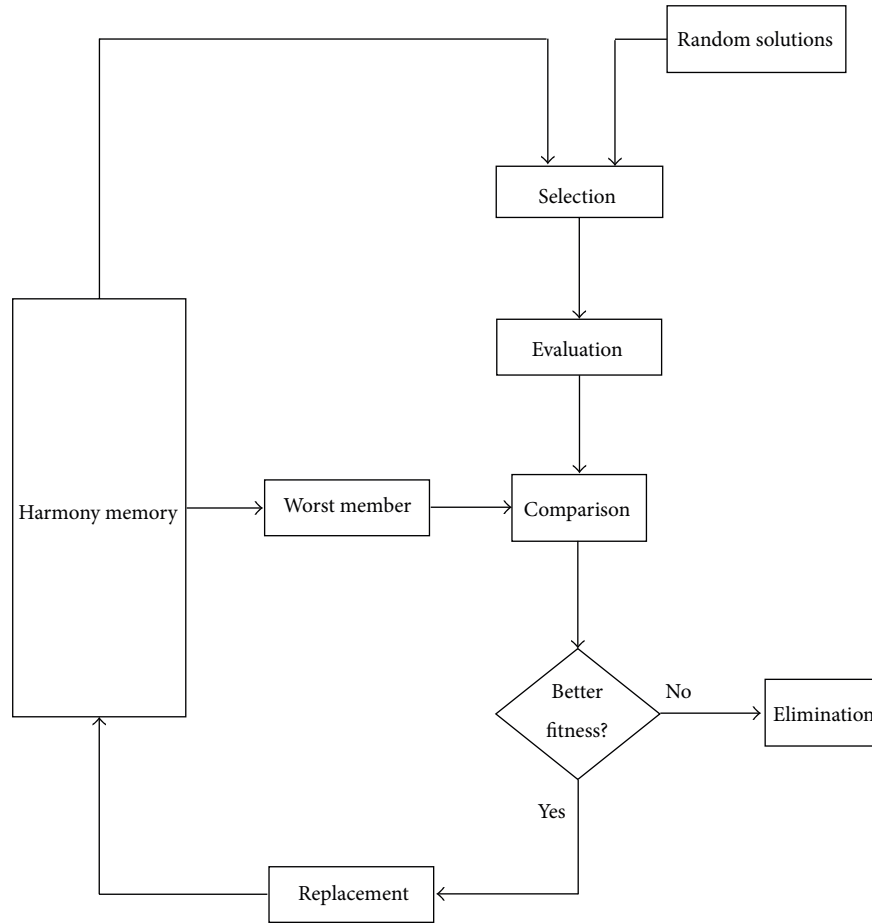
Figure 1: Harmony Search (HS) method.

on the Harmony Memory Considering Rate (HMCR). The HMCR is defined as the probability of selecting a component from the HM members, and 1-HMCR is, therefore, the probability of generating it randomly. If x'_j comes from the HM, it is chosen from the jth dimension of a random HM member and is further mutated according to the Pitching Adjust Rate (PAR). The PAR determines the probability of a candidate from the HM to be mutated. As we can see, the improvisation of $[x'_1, x'_2, \ldots, x'_n]$ is rather similar to the production of offspring in the Genetic Algorithms (GAs) [6] with the mutation and crossover operations. However, the GA creates new chromosomes using only one (mutation) or two (simple crossover) existing ones, while the generation of new solutions in the HS method makes full use of all the HM members.

Step 3. Update the HM. The new solution from Step 2 is evaluated. If it yields a better fitness than that of the worst member in the HM, it will replace that one. Otherwise, it is eliminated.

Step 4. Repeat Step 2 to Step 3 until a preset termination criterion, for example, the maximal number of iterations, is met.

Similar to the GA and swarm intelligence algorithms [7], the HS method is a random search technique. It does not require any prior domain knowledge, such as the gradient information of the objective functions. However, different from those population-based approaches, it only utilizes a single search memory to evolve. Therefore, the HS method has the distinguishing feature of computational simplicity.

3. Variants of HS Method

A lot of modified HS algorithms have been studied in the past decade so as to enhance the performances of the original version. As a matter of fact, a special discrete variation of the HS is proposed by Geem on the basis of introducing the stochastic derivatives for the discrete variables involved [8]. The stochastic derivatives give the selection probabilities of certain discrete variables during the evolution procedure of the HS. It is efficient at manipulating discrete optimization problems and has been employed in the optimal design of fluid-transport networks. Omran and Mahdavi embed the ideas borrowed from swarm intelligence into the regular HS and develop a new variant: global-best HS (GHS) [9]. In the GHS, the adjustment of new solutions improvised

```
/* HM initialization */
for (i = 1; i ≤ HMS; i++)
    for (j = 1; j ≤ n; j++)
        Randomly initialize x_j^i in HM.
    endfor
endfor
/* End of HM initialization */
Repeat
    /* Construction and evaluation of new solution candidate x */
    for (j = 1; j ≤ n; j++)
        if (rand(0, 1) < HMCR)
            Let x_j in x be the jth dimension of a randomly selected HM member.
            if (rand(0, 1) < PAR)
                Apply pitch adjustment distance bw to mutate x_j:
                x_j = x_j ± rand(0, 1) × bw.
            endif
        else
            Let x_j in x be a random value.
        endif
    endfor
    Evaluate the fitness of x: f(x).
    /* End of construction and evaluation of new solution candidate x */
    /* HM update */
    if (f(x) is better than the fitness of the worst HM member)
        Replace the worst HM member with x.
    else
        Disregard x.
    endif
    /* End of HM update */
Until a preset termination criterion is met.
```

ALGORITHM 1: Pseudocode of HS method.

is only based on the best harmony selected from the HM without the involvement of the distance bandwidth (bw). This interesting approach adds the unique social learning capability to the GHS. The investigation experiments of ten benchmark functions prove that the GHS can generally outperform the original HS. Inspired by the local versions of the Particle-Swarm Optimization (PSO) [10] and GHS, Pan et al. propose a local-best variant of the HS method with dynamic subpopulation: DLHS [11]. In the DLHS, the whole HM is divided into multiple sub-HMs, each of which can evolve independently. However, these sub-HMs will form the HM again after searching for the optimal solutions in their own regions. With this subpopulation policy and a simple local search strategy, the DLHS is capable of achieving a satisfactory compromise between the exploration and exploitation in search. It has been successfully applied to attack the difficult lot-streaming flow shop scheduling problem. Inspired by the GHS and DLHS, Geem further develops the Particle-Swarm Harmony Search (PSHS) [12], which has been validated to be better than the original HS algorithm for small-size problems but worse in case of large-scale one. A few novel HS methods have also been introduced by the authors of the present paper [13–17].

The parameters of HMCR and PAR usually play a critical role in the optimization performance of the HS method. Unfortunately, properly choosing the appropriate values for them is always a challenging topic. Mahdavi et al. study an adaptive strategy for adjusting PAR and bw in the Improved HS (IHS) algorithm [18]. The values of PAR and bw dynamically increase and decrease with the growth of HS iterations, respectively, so as to enhance the performance of the IHS. The IHS has been demonstrated to achieve comparable performances with other evolutionary and mathematical programming techniques in dealing with several test problems in terms of both the number of the fitness function evaluations required and the quality of the solutions found. Unfortunately, the lower and upper limits for the update of PAR and bw are often case dependent and are therefore difficult to determine. In [19], another adaptive HS method is proposed and explored. It takes advantage of two varying control parameters, η and ρ, to generate new harmony vectors. Both of these parameters are selected from the average values that are observed within the current harmony memory matrix using a given probability density function. This adaptive HS algorithm has found great successes in handling large steel structure optimization problems. Wang and Huang propose a new self-adaptive HS technique in [20]. Their almost parameter-free HS uses the information stored in the HM (self-consciousness), that is, the minimum and maximum of the present HM members,

to automatically control the pitch adjustment step. The low-discrepancy sequences are also utilized to initialize the HM. It has been compared with the aforementioned IHS and GHS and can offer superior performances on four optimization problems tested. Geem and Sim introduce the Parameter-Setting-Free (PSF) technique to eliminate the common difficulty of selecting suitable HS parameters [21]. The developed PSF-HS has a new operation step, namely, rehearsal, in which certain numbers of new solutions are generated with the initial HMCR and PAR. The adaptive HMCR and PAR are then calculated based on the rehearsal results evaluated. The PSF-HS has been shown to be more robust than the original HS method, although its computational complexity is moderately high. The authors of the present paper also study a fusion of the HS and Cultural Algorithm (CA), HS-CA, in which the search knowledge stored in the CA is utilized to guide the mutation direction and size of the HS. This HS-CA is further used to effectively cope with an optimal wind generator design problem [15]. In [16], a hybrid HS method inspired by the opposition-based learning is proposed by the same authors. The HS method is merged with the Population-Based Incremental Learning (PBIL) for the optimal design of electrical machines in [17].

Theoretical research on the working principles and search mechanism of the HS method has been reported in the recent literature, which can provide a useful guideline for users to design this algorithm in practice. Das et al. discuss the exploratory power of the HS by analyzing the evolution of the population variance over successive generations of the HM [22]. Based on their analysis work, they further propose a modified HS algorithm, Exploratory HS (EHS), in which bw for the pitch adjustment is set to be proportional to the standard deviation of the HM population. In the simulation study, the EHS can not only outperform three existing HS variants, IHS, GHS, and MHS [23], over all the test functions, but also yield better or at least comparable results when compared with a few state-of-the-art swarm intelligence techniques. Unfortunately, how to choose the optimal proportional gain k for bw in the EHS is still an open issue.

4. Applications of HS Method

In the real world, modern science and industry are indeed rich in the problems of optimization. Since the HS has been originally proposed by Geem and applied to solve the optimization problem of water distribution networks in 2000, the applications of the HS have covered numerous areas including industry, optimization benchmarks, power systems, medical science, control systems, construction design, and information technology [24].

4.1. Optimization Benchmarks. Optimization benchmarks for the hybridization of the HS method with other approaches are one principal application area. Different variants based on the HS have demonstrated their improvement and efficiency through various benchmark functions. Combined

with semantic genetic operators, a Geometric Selective Harmony Search (GSHS) method was proposed by Castelli et al. with three main differences from the original HS: (1) the memory consideration process involves the presence of a selection procedure, (2) the algorithm integrates a particular recombination operator that combines the information of two harmonies, and (3) the algorithm utilizes a mutation operation that uses the PAR parameter. Therefore, geometric semantic crossover produces offspring that is not worse than the worst of its parents, and geometric semantic mutation causes a perturbation on the semantics of solutions, whose magnitude is controlled by a parameter. Five different HS algorithms have been compared using 20 benchmark problems, and the GSHS outperforms the others with statistically significant enhancement in almost all the cases [25].

4.2. Industry. Industry is a prominent area full of various multimodal, constrained, nonlinear, and dynamical optimization problems. The HS algorithm proposed by Saka [26] determines the optimal steel section designations from the available British steel section table and implements the design constraints from BS5950. Recently, an Enhanced Harmony Search (EHS) in [27] is developed enabling the HS algorithm to quickly escape from local optima. The proposed EHS algorithm is utilized to solve four classical weight minimization problems of steel frames including two-bay, three-storey planar frame subject to a single-load case, one-bay, ten-storey planar frame consisting of 30 members, three-bay, twenty-four-storey planar frame, and spatial 744-member steel frame. In [28], the HS is used to select the optimal parameters in the tuned mass dampers. Fesanghary et al. propose a hybrid optimization method based on the global sensitivity analysis and HS for the optimal design of shell and tube heat exchangers [29].

4.3. Power Systems. There is a lot of work focused on the optimization issues concerning power systems, such as cost minimization. A modified HS algorithm is proposed to handle nonconvex economic load dispatch of real-world power systems. The economic load dispatch and combined economic and emission load dispatch problems can be converted into the minimization of the cost function [30]. Sinsuphan et al. combine the HS with sequential quadratic programming and GA to solve the optimal power flow problems. The objective function to be optimized is the total generator fuel costs in the entire system [31]. The chaotic self-adaptive differential HS algorithm, proposed by Arul et al., is employed to deal with the dynamic economic dispatch problem [32].

4.4. Signal and Image Processing. Li and Duan modify the HS by adding a Gaussian factor to adjust the bw. With this modified HS, they develop a pretraining process to select the weights used in the combining of feature maps to make the target more conspicuous in the saliency map [33]. In their method based on the HS, Fourie et al. design a harmony filter using the Improved HS algorithm for a robust visual tracking system [34].

4.5. Others. In addition to the aforementioned applications, the HS has also been widely employed in a large variety of fields, including transportation, manufacturing, robotics, control, and medical science [24]. Xu et al. explore the applications of the HS in the prototype optimization and selection of the reconfigurable mobile robots in the sandy terrain [35, 36]. Many traffic modeling software packages are capable of finding the optimal or near-optimal signal timings using different optimization algorithms. For example, Ceylan proposes a modified HS with embedded hill climbing algorithm for further tuning the solutions in the stochastic equilibrium network design [37]. The modified HS algorithm is also used in parameter identification of the solar cell mathematical models [38]. Miguel et al. employ the HS method in damage detection under the ambient vibration [39].

5. A Modified HS Method for Constrained Optimization: A Case Study

5.1. Constrained Optimization Problems. Most of the practical optimization problems are indeed constrained optimization problems, whose goal is to find the optimal solution that satisfies a set of given constraints. In general, a constrained optimization problem is described as follows.

Find $\vec{x} = (x_1, x_2, \ldots, x_n)$ to minimize $f(\vec{x})$, subject to $g_i(\vec{x}) \leq 0$, $i = 1, 2, \ldots, M$, and $h_j(\vec{x}) = 0$, $j = 1, 2, \ldots, N$, where $f(\vec{x})$ is the objective function and $g_i(\vec{x})$ and $h_j(\vec{x})$ are the inequality and equality constraint functions, respectively. As a matter of fact, the equality constraint functions can be transformed into the inequality constraint functions:

$$\left| h_j(\vec{x}) \right| - \varepsilon \leq 0, \tag{2}$$

where ε is a small enough tolerance parameter. Therefore, we only consider the inequality constraint functions $g_i(\vec{x}) \leq 0$, $i = 1, 2, \ldots, M$, here.

The constrained optimization problems are generally difficult to deal with, because the constraint functions can divide the whole search space into some disjoint islands. Numerous constraint-handling techniques have been proposed and investigated during the past decades [40–44]. One popular solution is to define a new fitness function $F(\vec{x})$ to be optimized [41]. $F(\vec{x})$ is the combination of the objective function $f(\vec{x})$ and weighted penalty terms $P_i(\vec{x})$, $i = 1, 2, \ldots, M$, which reflect the violation of the constraint functions:

$$F(\vec{x}) = f(\vec{x}) + \sum_{i=1}^{M} w_i P_i(\vec{x}), \tag{3}$$

where w_i $(i = 1, 2, \ldots, M)$ are the preset weights. The overall optimization performance depends on the penalty terms and their weights and may significantly deteriorate with inappropriately chosen ones. Unfortunately, there is no analytic way yet to find the best $P_i(\vec{x})$ and w_i. In this section, on the basis of the Pareto-dominance, we present a modified HS method for the direct handling of these constraints.
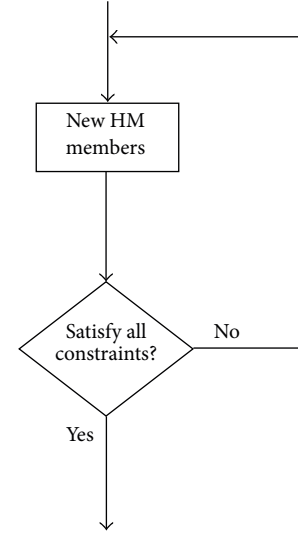

Figure 2: Generation of new HM members satisfying constraints in HS method.

5.2. A Modified HS Method for Constrained Optimization. It is well known that the regular HS method is not efficient in attacking the constrained optimization problems. As aforementioned, the HM only stores the feasible solution candidates. The new HM members are generated either from the existing HM members or in a random way. Nevertheless, they are not guaranteed to always meet all the constraints. Figure 2 shows that, in the HS method, the new HM members satisfying the constraints can be obtained based on only *trial and error*, which may lead to a time consuming procedure, especially in case of complex constraint functions.

In our modified HS method [14], we make full use of those HM members that do not even meet the constraints. The key issue is how to rank the HM members according to their objective as well as constraint function values. Here, the values of the constraint functions of all the HM members are stored together with their objective function values in the HM. The HM members are divided into two different parts: feasible members and infeasible members, as illustrated in Figure 3. The former satisfy all the constraint functions, while the latter do not. Thus, the ranking of the HM members is separated into two consecutive stages: ranking of the feasible HM members and ranking of the infeasible ones. The ranking of the feasible HM members is straightforward: they can be sorted using their objective function values. However, for the infeasible ones, the ranking is based on the Pareto-dominance of these HM members [42]. An infeasible HM member dominates another, if none of its constraint function values is larger and at least one is smaller. Formally, the Pareto-dominance is defined as follows. Suppose there are two infeasible HM members, $\vec{x}^1$ and $\vec{x}^2$. If

$$\forall i \in \{1, 2, \ldots, M\} : g_i(\vec{x}^1) \leq g_i(\vec{x}^2)$$
$$\wedge \exists i \in \{1, 2, \ldots, M\} : g_i(\vec{x}^1) < g_i(\vec{x}^2), \tag{4}$$
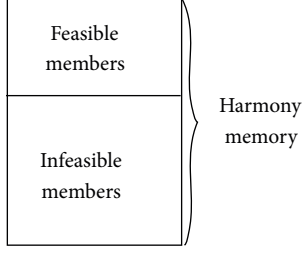
FIGURE 3: Harmony memory with feasible and infeasible members.

we conclude that $\vec{x}^1$ dominates $\vec{x}^2$. For each infeasible HM member, we can calculate the number of the others that dominate it, which implies its relative degree of violation of the constraint functions. That is, the rank of an infeasible HM member is determined by the number of other infeasible HM members by which it is dominated.

After the HM has been ranked, the worst HM member $\vec{x}^\#$ can be selected and compared with the new solution candidate $\vec{x}^*$. Note that $\vec{x}^*$ does not need to be feasible. When $\vec{x}^\#$ is compared with $\vec{x}^*$, $\vec{x}^*$ will replace $\vec{x}^\#$ only in one of the following three cases:

(1) $\vec{x}^*$ is feasible, and $\vec{x}^\#$ is infeasible,

(2) both $\vec{x}^*$ and $\vec{x}^\#$ are feasible, and $f(\vec{x}^*) < f(\vec{x}^\#)$,

(3) both $\vec{x}^*$ and $\vec{x}^\#$ are infeasible, and $\vec{x}^*$ dominates $\vec{x}^\#$.

More precisely, $\vec{x}^*$ replaces $\vec{x}^\#$ on the condition that

$$
\begin{cases}
\forall i \in \{1, 2, \ldots, M\} : g_i(\vec{x}^*) \leq 0 \\
\quad \wedge \\
\exists i \in \{1, 2, \ldots, M\} : g_i(\vec{x}^\#) > 0
\end{cases}
$$
$$
\vee
$$
$$
\begin{cases}
\forall i \in \{1, 2, \ldots, M\} : g_i(\vec{x}^*) \leq 0 \\
\quad \wedge \\
\forall i \in \{1, 2, \ldots, M\} : g_i(\vec{x}^\#) \leq 0 \\
\quad \wedge \\
f(\vec{x}^*) < f(\vec{x}^\#)
\end{cases}
\tag{5}
$$
$$
\vee
$$
$$
\begin{cases}
\exists i \in \{1, 2, \ldots, M\} : g_i(\vec{x}^*) > 0 \\
\quad \wedge \\
\exists i \in \{1, 2, \ldots, M\} : g_i(\vec{x}^\#) > 0 \\
\quad \wedge \\
\forall i \in \{1, 2, \ldots, M\} : g_i(\vec{x}^*) \leq g_i(\vec{x}^\#) \\
\quad \wedge \\
\exists i \in \{1, 2, \ldots, M\} : g_i(\vec{x}^*) < g_i(\vec{x}^\#).
\end{cases}
$$

Figure 4 illustrates this procedure of comparison between $\vec{x}^\#$ and $\vec{x}^*$, replacement of $\vec{x}^\#$ with $\vec{x}^*$, and elimination of $\vec{x}^*$.

It is observed from the above descriptions that the infeasible HM members violating the given constraints can also evolve in the modified HS method. In other words, we do not have to always search for new feasible HM members by repeatedly examining them with the constraint functions, as in Figure 2. Compared with the original HS method, our approach needs only a considerably smaller number of constraint function evaluations and, thus, has a lower computational complexity.

5.3. Wind Generator Design. In this section, we investigate the constrained optimization effectiveness of the modified HS method using a real-world design problem. Wind generator design has been an important but difficult topic in the modern electrical machinery industry. The wind generator shown in Figure 5 is a radial flux type permanent magnet generator, in which the NdFeB magnets are surface-mounted [45]. The remanence flux density of the magnets is 1.05 T, and coercivity 800 kA/m. The stator winding is a three-phase two-layer full-pitch diamond winding. The number of the slots per pole and phase is 2. The stator slot and constant dimensions of the slot are illustrated in Figure 6. The iron core consists of 55 mm long subcores, between which there are radial 6 mm wide ventilation ducts. The length of the subcore is constant, and the number of the ventilation ducts is a decimal fraction in the calculations. The stator frame, bearing shields, and rotor steel body are all 20 mm thick. In the rotor body disc, there are holes, and around 50% of the disc is iron and 50% holes. The iron loss factor is $p_{15} = 6.6$ W/kg with 50 Hz and 1.5 T, and the air-gap length is 5 mm. The rated values of this practical wind generator are given in Table 1.

The detailed design principles of the above wind generator are explained in [45]. The objective function $f(\mathbf{x})$ (in €) to be minimized is the sum of the material costs and capitalized costs of the total losses of this generator:

$$
\begin{aligned}
f(\mathbf{x}) = {} & k_{\text{Fe}} m_{\text{Fe}} + k_{\text{Cu}} m_{\text{Cu}} + k_{\text{PM}} m_{\text{PM}} \\
& + k_{\text{Fef}} m_{\text{Frame}} + k_{\text{Loss}} P_{\text{tot}},
\end{aligned}
\tag{6}
$$

where m_{Fe}, m_{Cu}, m_{PM}, and m_{Frame} are the masses of the stator iron core, stator winding, permanent magnets, and stator frame and rotor body, respectively, k_{Fe}, k_{Cu}, k_{PM}, and k_{Fef} the unit prices of the stator core, copper, permanent magnets, and stator frame and rotor body, respectively, and k_{Loss} capitalized costs of the losses (Table 2). More details of the calculation of (6) can be found in [45]. The cost of the stator core actually includes the punching, waste parts of the sheet, and assembly of the stator core. The manufacturing cost of the winding is taken into account in the copper cost. The permanent magnet cost includes the corrosion protection, assembly into bigger cassettes, and magnetization of the magnets. The stator frame and rotor body costs consist of the material cost and cost of manufacturing the frame and body.
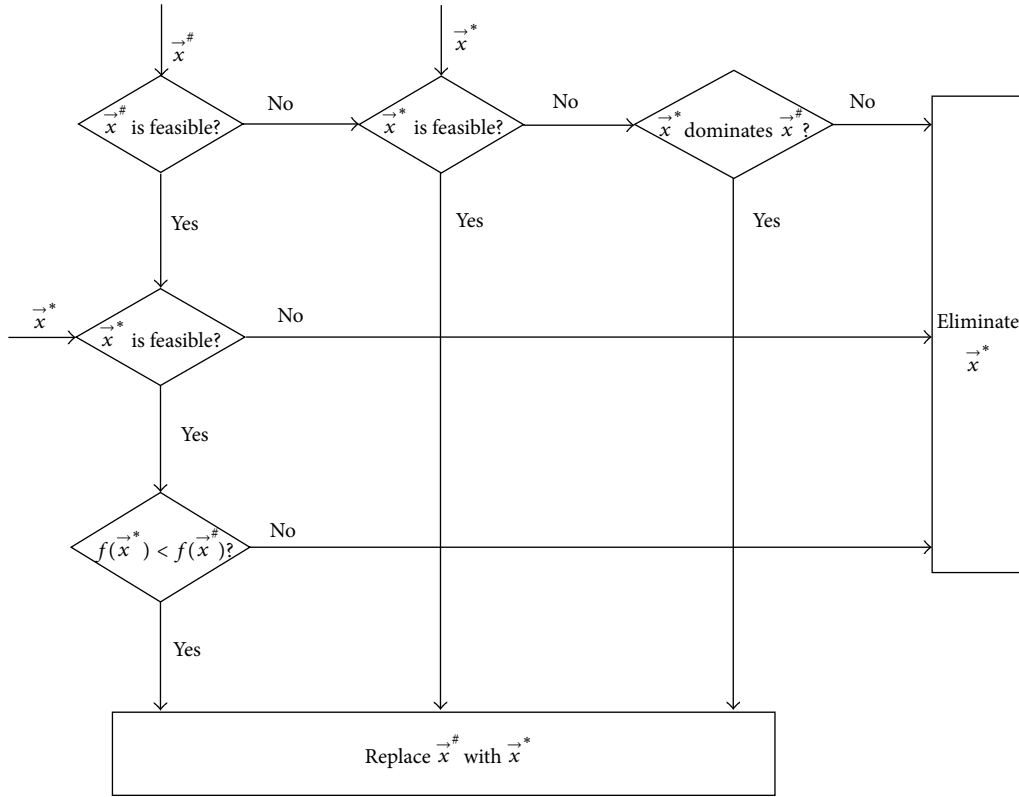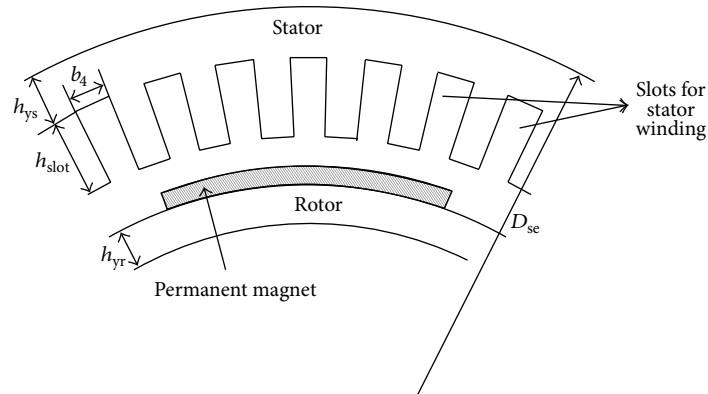
FIGURE 4: Comparison between $\vec{x}^{\#}$ and $\vec{x}^{*}$, replacement of $\vec{x}^{\#}$ with $\vec{x}^{*}$, and elimination of $\vec{x}^{*}$.

FIGURE 5: Cross section and dimensions of permanent magnet generator.

The stator resistive losses are calculated at the temperature of 100°C, and the iron losses in the stator teeth are

$$P_{\text{Fed}} = 2 \cdot p_{15} \left(\frac{B_d}{1.5\,\text{T}} \right)^2 \left(\frac{f}{50\,\text{Hz}} \right)^{1.5} m_d, \qquad (7)$$

where p_{15} is the iron loss factor, B_d the maximum flux density in the teeth, f the frequency, and m_d the mass of the stator teeth. The iron losses in the stator yoke are

$$P_{\text{Fey}} = 1.5 \cdot p_{15} \left(\frac{B_y}{1.5\,\text{T}} \right)^2 \left(\frac{f}{50\,\text{Hz}} \right)^{1.5} m_y, \qquad (8)$$

where B_y is the maximum flux density in the yoke and m_y the mass of the yoke. The losses in the permanent magnets are assumed to be 1% of the rated power, that is, 30 kW. The additional losses are assumed to be 3% of the rated power, that is, 90 kW. The friction and ventilation losses are

$$P_{\rho} = 10 \cdot D_r \left(l + 0.6 \cdot \tau_p \right) \left(\pi n D_r \right)^2 [\text{W}], \qquad (9)$$

where D_r is the outer rotor diameter, τ_p the pole pitch, and n the rotational speed of the rotor. Table 3 gives the nine design parameters to be optimized and their valid ranges.

Like most of the practical design problems, the design variables of this wind generator are also subject to constraints.
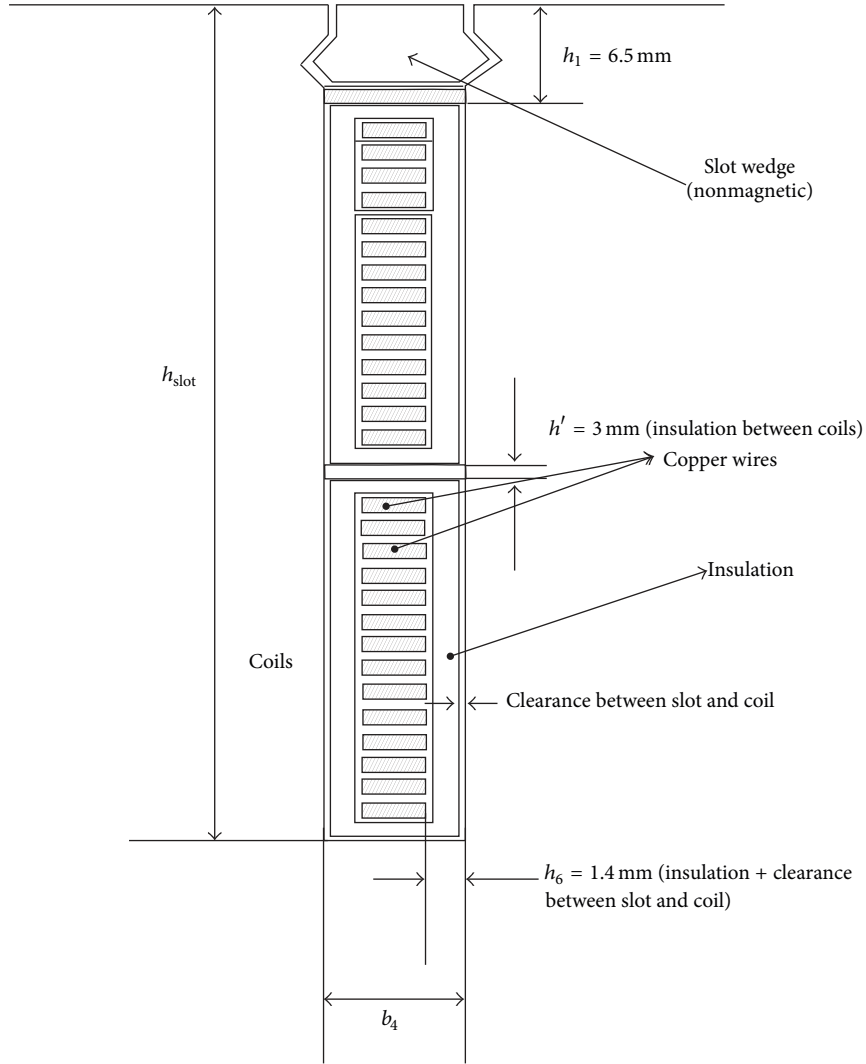
FIGURE 6: Slot form and constant dimensions of slot.

TABLE 1: Rate values of wind generator.

Power	3 MW
Voltage	690 V
Connection	Star
Speed	16.98 rpm
Number of phases	3

TABLE 2: Unit prices of materials and capitalized loss costs.

Electrical steel, k_{Fe}	4€/kg
Copper, k_{Cu}	12€/kg
NdFeB magnets, k_{PM}	60€/kg
Stator frame and rotor steel body, k_{Fef}	2€/kg
Losses, k_{Loss}	2€/W

A total of five given constraints are provided in Table 4. It can be observed that among them there are one constraint on the stator tooth width, three constraints on the flux density of stator and rotor yoke and stator tooth, and one constraint on the output power of the wind generator. Therefore, these constraints must be satisfied by the optimized design variables.

5.4. Modified HS Method-Based Optimal Design of Wind Generator. Both the original and our modified HS methods are applied to attack the above demanding wind generator

optimal design problem with constraints. The optimization coefficients used are chosen as follows: HMCR = 0.8 and PAR = 0.75. To assess the constrained optimization capabilities of these two HS methods, the Number of Constraint Evaluations (NCEs) is used as the performance criterion here. In other words, the smaller the NCEs are, the more efficient the optimization method is. The evolution procedures of the two HS methods are terminated, when a preset optimization design goal is met, and the NCEs used are compared with each other. In our simulations, there are six optimization goals, that is, 6.65×10^5, 6.625×10^5, 6.6×10^5, 6.575×10^5,

TABLE 3: Wind generator design parameters with ranges.

Parameters	Symbols	Ranges
Stator core length including ventilation ducts	l	0.3–3.0 m
Stator yoke height	h_{ys}	0.01–0.5 m
Stator outer diameter	D_{se}	3.0–8.0 m
Stator slot height	h_{slot}	0.07–0.3 m
Maximum flux density in air gap	B_{max}	0.4–0.9 T
Number of effective conductors in stator slot	z_s	8–26
Rotor yoke height	h_{yr}	0.01–0.5 m
Number of poles pairs	p	20–80
Stator slot width	b_4	0.007–0.04 m

TABLE 4: Wind generator optimization constraints.

Stator tooth width	>8 mm
Stator yoke flux density	<2.2 T
Rotor yoke flux density	<2.2 T
Stator tooth flux density	<2.2 T
Maximum output power	>4.8 MW

TABLE 5: NCEs used by two HS methods for achieving optimization goals.

Optimization goals	HS	Modified HS	Improvement percentages
6.65×10^5	1.8874×10^4	1.2193×10^4	35.40%
6.625×10^5	2.4925×10^4	1.5694×10^4	37.04%
6.6×10^5	3.2848×10^4	1.8874×10^4	42.54%
6.575×10^5	4.2013×10^4	2.3214×10^4	44.75%
6.55×10^5	6.0106×10^4	2.9916×10^4	50.23%
6.625×10^5	8.8975×10^4	4.0342×10^4	54.66%

6.55×10^5, and 6.625×10^5. After a total of 1,000 independent trials have been run, the average NCEs used by the original and modified HS methods are given in Table 5. Apparently, the modified HS method uses less NCEs than the original HS for reaching the same optimal design goals. That is to say, the former is more efficient than the latter in coping with the given constraints in this optimal wind generator design problem. It is also worth noting that the improvement of the NCEs usage grows with the increase of the optimization goal.

6. Conclusions

In this paper, we provide an overview of the theory and applications of the HS method. The fundamentals of the HS are first introduced. Next, both the basic HS algorithm and some typical variants are discussed in detail. The applications of the HS method in a few applications areas, such as optimization, power systems, signal processing, and robotics, are also presented. Furthermore, as a case study example, a modified HS proposed by the authors for coping with the constrained optimization problems is demonstrated. Based on the Pareto-dominance ranking of the HM members, the given constraints can be directly handled in this new HS method. A real-world optimal wind generator design problem has been employed to verify its effectiveness.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

References

[1] Z. W. Geem, J. H. Kim, and G. V. Loganathan, "A new heuristic optimization algorithm: harmony search," *Simulation*, vol. 76, no. 2, pp. 60–68, 2001.

[2] K. S. Lee and Z. W. Geem, "A new meta-heuristic algorithm for continuous engineering optimization: harmony search theory and practice," *Computer Methods in Applied Mechanics and Engineering*, vol. 194, no. 36–38, pp. 3902–3933, 2005.

[3] K. S. Lee and Z. W. Geem, "A new structural optimization method based on the harmony search algorithm," *Computers and Structures*, vol. 82, no. 9-10, pp. 781–798, 2004.

[4] Z. W. Geem, J. H. Kim, and G. V. Loganathan, "Harmony search optimization: application to pipe network design," *International Journal of Modelling and Simulation*, vol. 22, no. 2, pp. 125–133, 2002.

[5] X. Wang, X.-Z. Gao, and S. J. Ovaska, "Fusion of clonal selection algorithm and harmony search method in optimisation of fuzzy classification systems," *International Journal of Bio-Inspired Computation*, vol. 1, no. 1-2, pp. 80–88, 2009.

[6] R. Poli and W. B. Langdon, *Foundations of Genetic Programming*, Springer, Berlin, Germany, 2002.

[7] A. P. Engelbrecht, *Fundamentals of Computational Swarm Intelligence*, John Wiley & Sons, West Sussex, UK, 2005.

[8] Z. W. Geem, "Novel derivative of harmony search algorithm for discrete design variables," *Applied Mathematics and Computation*, vol. 199, no. 1, pp. 223–230, 2008.

[9] M. G. H. Omran and M. Mahdavi, "Global-best harmony search," *Applied Mathematics and Computation*, vol. 198, no. 2, pp. 643–656, 2008.

[10] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, pp. 1942–1948, Perth, Australia, December 1995.

[11] Q.-K. Pan, P. N. Suganthan, J. J. Liang, and M. F. Tasgetiren, "A local-best harmony search algorithm with dynamic subpopulations," *Engineering Optimization*, vol. 42, no. 2, pp. 101–117, 2010.

[12] Z. W. Geem, "Particle-swarm harmony search for water network design," *Engineering Optimization*, vol. 41, no. 4, pp. 297–311, 2009.

[13] X.-Z. Gao, X. Wang, and S. J. Ovaska, "Uni-modal and multi-modal optimization using modified harmony search methods,"

International Journal of Innovative Computing, Information and Control, vol. 5, no. 10, pp. 2985–2996, 2009.

[14] X.-Z. Gao, X. Wang, S. J. Ovaska, and H. Xu, "A modified harmony search method in constrained optimization," *International Journal of Innovative Computing, Information and Control*, vol. 6, no. 9, pp. 4235–4247, 2010.

[15] X.-Z. Gao, X. Wang, T. Jokinen, S. J. Ovaska, A. Arkkio, and K. Zenger, "A hybrid optimization method for wind generator design," *International Journal of Innovative Computing, Information and Control*, vol. 8, no. 6, pp. 4347–4373, 2012.

[16] X. Z. Gao, X. Wang, S. J. Ovaska, and K. Zenger, "A hybrid optimization method of harmony search and opposition-based learning," *Engineering Optimization*, vol. 44, no. 8, pp. 895–914, 2012.

[17] X. Z. Gao, X. Wang, T. Jokinen, S. J. Ovaska, A. Arkkio, and K. Zenger, "A hybrid PBIL-based harmony search method," *Neural Computing and Applications*, vol. 21, no. 5, pp. 1071–1083, 2012.

[18] M. Mahdavi, M. Fesanghary, and E. Damangir, "An improved harmony search algorithm for solving optimization problems," *Applied Mathematics and Computation*, vol. 188, no. 2, pp. 1567–1579, 2007.

[19] O. Hasaņebi, F. Erdal, and M. P. Saka, "Adaptive harmony search method for structural optimization," *ASCE Journal of Structural Engineering*, vol. 136, no. 4, pp. 419–431, 2010.

[20] C.-M. Wang and Y.-F. Huang, "Self-adaptive harmony search algorithm for optimization," *Expert Systems with Applications*, vol. 37, no. 4, pp. 2826–2837, 2010.

[21] Z. W. Geem and K.-B. Sim, "Parameter-setting-free harmony search algorithm," *Applied Mathematics and Computation*, vol. 217, no. 8, pp. 3881–3889, 2010.

[22] S. Das, A. Mukhopadhyay, A. Roy, A. Abraham, and B. K. Panigrahi, "Exploratory power of the harmony search algorithm: analysis and improvements for global numerical optimization," *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 41, no. 1, pp. 89–106, 2011.

[23] Y. M. Cheng, L. Li, T. Lansivaara, S. C. Chi, and Y. J. Sun, "An improved harmony search minimization algorithm using different slip surface generation methods for slope stability analysis," *Engineering Optimization*, vol. 40, no. 2, pp. 95–115, 2008.

[24] Z. W. Geem, Ed., *Geem Music-Inspired Harmony Search Algorithm*, Springer, Berlin, Germany, 2009.

[25] M. Castelli, S. Silva, L. Manzoni, and L. Vanneschi, "Geometric selective harmony search," *Information Sciences*, vol. 279, pp. 468–482, 2014.

[26] M. K. Saka, "Optimum design of steel skeleton structures," in *Music-Inspired Harmony Search Algorithm*, Z. W. Geem, Ed., Springer, Berlin, Germany, 2009.

[27] M. R. Maheri and M. M. Narimani, "An enhanced harmony search algorithm for optimum design of side sway steel frames," *Computers and Structures*, vol. 136, pp. 78–89, 2014.

[28] G. Bekdaş and S. M. Nigdeli, "Estimating optimum parameters of tuned mass dampers using harmony search," *Engineering Structures*, vol. 33, no. 9, pp. 2716–2723, 2011.

[29] M. Fesanghary, E. Damangir, and I. Soleimani, "Design optimization of shell and tube heat exchangers using global sensitivity analysis and harmony search algorithm," *Applied Thermal Engineering*, vol. 29, no. 5-6, pp. 1026–1031, 2009.

[30] B. Jeddi and V. Vahidinasab, "A modified harmony search method for environmental/economic load dispatch of real-world power systems," *Energy Conversion and Management*, vol. 78, pp. 661–675, 2014.

[31] N. Sinsuphan, U. Leeton, and T. Kulworawanichpong, "Optimal power flow solution using improved harmony search method," *Applied Soft Computing Journal*, vol. 13, no. 5, pp. 2364–2374, 2013.

[32] R. Arul, G. Ravi, and S. Velusami, "Chaotic self-adaptive differential harmony search algorithm based dynamic economic dispatch," *International Journal of Electrical Power & Energy Systems*, vol. 50, no. 1, pp. 85–96, 2013.

[33] J. Li and H. Duan, "Novel biological visual attention mechanism via Gaussian harmony search," *Optik*, vol. 125, no. 10, pp. 2313–2319, 2014.

[34] J. Fourie, S. Mills, and R. Green, "Harmony filter: a robust visual tracking system using the improved harmony search algorithm," *Image and Vision Computing*, vol. 28, no. 12, pp. 1702–1716, 2010.

[35] H. Xu, X. Z. Gao, G.-L. Peng, K. Xue, and Y. Ma, "Prototype optimization of reconfigurable mobile robots based on a modified Harmony Search method," *Transactions of the Institute of Measurement and Control*, vol. 34, no. 2-3, pp. 334–360, 2012.

[36] H. Xu, Z. Zhang, K. Alipour, K. Xue, and X. Z. Gao, "Prototypes selection by multi-objective optimal design: application to a reconfigurable robot in sandy terrain," *Industrial Robot*, vol. 38, no. 6, pp. 599–613, 2011.

[37] H. Ceylan, "A hybrid harmony search and TRANSYT hill climbing algorithm for signalized stochastic equilibrium transportation networks," *Transportation Research Part C: Emerging Technologies*, vol. 25, pp. 152–167, 2012.

[38] A. Askarzadeh and A. Rezazadeh, "Parameter identification for solar cell models using harmony search-based algorithms," *Solar Energy*, vol. 86, no. 11, pp. 3241–3249, 2012.

[39] L. F. F. Miguel, J. Kaminski Jr., and J. D. Riera, "Damage detection under ambient vibration by harmony search algorithm," *Expert Systems with Applications*, vol. 39, no. 10, pp. 9704–9714, 2012.

[40] C. A. C. Coello, "Constraint-handling using an evolutionary multiobjective optimization technique," *Civil Engineering and Environmental Systems*, vol. 17, no. 4, pp. 319–346, 2000.

[41] C. A. C. Coello, "Use of a self-adaptive penalty approach for engineering optimization problems," *Computers in Industry*, vol. 41, no. 2, pp. 113–127, 2000.

[42] C. A. C. Coello and E. M. Montes, "Constraint-handling in genetic algorithms through the use of dominance-based tournament selection," *Advanced Engineering Informatics*, vol. 16, no. 3, pp. 193–203, 2002.

[43] K. Deb, "An efficient constraint handling method for genetic algorithms," *Computer Methods in Applied Mechanics and Engineering*, vol. 186, no. 2–4, pp. 311–338, 2000.

[44] Z. Michalewicz, *Genetic Algorithms + Data Structures = Evolution Programs*, Springer, Berlin, Germany, 3rd edition, 1996.

[45] J. Pyrhönen, T. Jokinen, and V. Hrabovcová, *Design of Rotating Electrical Machines*, John Wiley & Sons, West Sussex, UK, 2008.