Population-Variance and Explorative Power of Harmony Search: An Analysis

Arpan Mukhopadhyay[1], Anwit Roy[1], Sourav Das[1], Swagatam Das[1], and Ajith Abraham[2]

[1]*Department of Electronics and Telecommunication Engineering, Jadavpur University, Kolkata, India.*

[2]*Center of Excellence for Quantifiable Quality of Service, Norwegian University of Science and Technology, Norway ajith.abraham@ieee.org.*

Abstract

This paper presents a simple mathematical analysis of some features of the Harmony Search algorithm (HS). HS is a recently developed derivative-free optimization algorithm, which draws inspiration from the musical process of searching for a perfect state of harmony. This work analyses the evolution of the population-variance over successive generations in HS and thereby draws some important conclusions regarding the explorative power of HS. Experimental results have been provided to validate the theoretical treatment. A simple modification of the classical HS has been proposed in the light of the mathematical analysis undertaken here.

1. Introduction

In the paradigm of artificial intelligence (AI), evolutionary algorithms (EA) are randomized search and optimization techniques that mimic the processes of Darwinian evolution and natural genetics [1]. In recent past the computational cost having been reduced almost dramatically, researchers, all over the world, are coming up with new EAs on a regular basis in order to meet the demands of the complex, real-world optimization problems. In 2001, Geem *et al.* proposed Harmony Search (HS) [2], a derivative-free, meta-heuristic algorithm, mimicking the improvisation process of music players. Since its inception, HS has found several applications in a wide variety of practical optimization problems like pipe-network design [3], structural optimization [4], vehicle routing problem [5], combined heat and power economic dispatch problem [6], and Dam Scheduling [7].

A significant amount of research has already been undertaken on the application of HS for solving difficult practical optimization problems as well as to improve the performance of HS by tuning its parameters and/or blending it with other powerful optimization techniques like PSO (Particle Swarm Optimization) [8-11]. However, there has been little or no research work to explain the search-mechanism of HS analytically within a statistical-mathematical framework. This paper tries to make a humble contribution in this context. It attempts to analyze the evolution of the population-variance of HS and its influence on the explorative power of the algorithm. The efficiency of most EAs depends on how they balance between the explorative and exploitative tendencies in the course of search. Exploitation means the ability of an algorithm to use the information already collected and thus to orient the search towards the goal while exploration is the process that allows introduction of new information into the population. This paper finds an analytical expression for the population-variance of HS following the works of Beyer [12] and Zaharie [13], who did conceptually similar analysis for Evolutionary Strategy (ES) [14, 15] and Differential Evolution (DE) [16, 17] respectively. It also draws a few important conclusions, regarding the explorative power of HS by observing the change of expected population-variance over generations with and without selection.

Based on the analysis presented here, a simple modification of classical HS has been proposed. In the modified HS, the control parameter known as *distance bandwidth (bw)* has been made equal to the standard deviation of the current population. Limited experimental results (owing to the economy of space) on popular benchmarks have been presented to validate the analysis undertaken here.

2. Classical Harmony Search Algorithm

Harmony search is a new meta-heuristic optimization algorithm which imitates the music improvisation process applied by musicians. Each musician improvises the pitches of his/her instrument to obtain a better state of harmony. The goal of the process is to reach a perfect state of harmony. The different steps of the HS algorithm are described below:

Step 1:

The 1st step is to specify the problem and initialize the parameter values. The optimization problem is defined as minimize (or maximize) $f(\mathbf{x})$ such that $_L x_i \leq x_i \leq _U x_i$, where $f(\mathbf{x})$ is the objective

function, $\mathbf{x}$ is a solution vector consisting of N decision variables (x_i) and $_L x_i$ and $_U x_i$ are the lower and upper bounds of each decision variable, respectively. The parameters of the HS algorithm i.e. the harmony memory size (HMS), or the number of solution vectors in the harmony memory; harmony memory considering rate ($HMCR$); pitch adjusting rate (PAR); distance bandwidth parameter (bw); and the number of improvisations (NI) or stopping criterion are also specified in this step.

Step 2:

The 2nd step is to initialize the Harmony Memory. The initial harmony memory is generated from a uniform distribution in the ranges [$_L x_i$, $_U x_i$], where $1 \leq i \leq N$. This is done as follows:

$$x_i^j = {}_L x_i + r \times \left({}_U x_i - {}_L x_i\right)$$,

where $j = 1,2,3...,HMS$ and $r \sim U(0,1)$

Step 3:

The third step is known as the 'improvisation' step. Generating a new harmony is called 'improvisation'. The New Harmony vector $\mathbf{x}' = (x_1', x_2', x_3', x_4', ..., x_N')$ is generated using the following rules: memory consideration, pitch adjustment, and random selection. The procedure works as follows:

Pseudo-code of improvisation in HS
 for each $i \in [1, N]$ **do**
if $U(0,1) \leq HMCR$ **then**/*memory consideration*/
 begin
   $x_i' = x_i^j$, where $j \sim U(1,2,\ldots,HMS)$.
   if $U(0,1) \leq PAR$ **then** /*Pitch Adjustment*/
   begin
   $x_i' = x_i' + r \times bw$, where $r \sim U(0,1)$ and bw is the
     arbitrary distance bandwidth parameter.
else /* random selection */
 $x_i^j = {}_L x_i + r \cdot \left({}_U x_i - {}_L x_i\right)$
endif
 done

Step 4:

In this step the harmony memory is updated. The generated harmony vector $\mathbf{x}' = (x_1', x_2', x_3', x_4', ..., x_N')$ replaces the worst harmony in the HM (harmony memory), only if its fitness

(measured in terms of the objective function) is better than the worst harmony.
Step 5:

The stopping criterion (generally the number of iterations) is checked. If it is satisfied, computation is terminated. Otherwise, Steps 3 and 4 are repeated.

3. Computation of the Expected Population-Variance

The explorative power of an EA expresses its capability to explore the search-space. Expected population variance is the measure of the explorative power of the algorithm. In the original Harmony Search algorithm, we do not have to deal with any population of vectors in **step 3** (New Harmony Improvisation). Instead, a single harmony vector is created in this step. But to study the explorative power of this newly-conceptualized algorithm, we will have to consider a population of vectors in its variation step (**step 3**) i.e. the step that is responsible for exploring new variations in the search space. After the selection step (**step 4**) of the Harmony Search algorithm, population variance may increase or decrease. To avoid any premature convergence or stagnation in the successive generations and to ensure that most of the regions in the search space have been explored, the variance operation must adjust the population variance such that it has a reasonable value from one generation to another. Thus if the selection step decreases the population variance, the variance operators must increase it necessarily.

Thus in **step 3** i.e. the New Harmony Improvisation Process we consider a population of new harmonies instead of a single harmony. This is done for the sake of analysis of evolution of population variance. This new population is referred as $Y = \{Y^1, Y^2, ..., Y^{HMS}\}$, where each vector Y^i is generated following the rules described in **step 3**.

Since in Harmony Search algorithm the perturbations are made independently for each component, we can say that it will not be a loss of generality if we conduct our analysis to one-component vectors i.e. scalars. To do this we will consider an initial population of scalars $x = \{x_1, x_2, x_3, ..., x_m\}$ with elements ($x_l \in R$), where we have taken $m = HMS$, i.e. the harmony memory size.

The variance of the population x is as follows:

$$\text{var}(x) = \frac{1}{m} \sum_{l=1}^{m} (x_l - \bar{x})^2 = \overline{x^2} - \bar{x}^2 ,$$

where $\overline{x}$ = Population mean and $\overline{x^2}$ = Quadratic population mean.

If the elements of the population are affected by some random elements, the $var(x)$ will be a random variable and $E(var(x))$ will be the measure of the explorative power.

The main result of this paper is the following theorem:

Theorem 1:

Let $x = \{x_1, x_2, \ldots, x_N\}$ be the current population, $Y = \{Y_1, Y_2, \ldots, Y_N\}$ the intermediate population obtained after harmony memory consideration and pitch adjustment. If $HMCR$ be the harmony memory consideration probability, PAR the pitch-adjustment probability, bw the arbitrary distance bandwidth and if we consider the allowable range for the decision variables (x_i) to be $\{x_{\min}, x_{\max}\}$ where $x_{\max} = a$, $x_{\min} = -a$, then

$$E(var(Y)) = \frac{(m-1)}{m} \cdot [HMCR \cdot var(x) + HMCR \cdot (1 - HMCR) \cdot \overline{x}^2$$
$$+ HMCR \cdot (1 - HMCR) \cdot PAR \cdot bw \cdot \overline{x}$$
$$+ HMCR \cdot PAR \cdot bw^2 \cdot \left(\frac{1}{3} - \frac{HMCR \cdot PAR}{4}\right)$$
$$+ \frac{a^2}{3} \cdot (1 - HMCR)]$$

(1)

Proof: Here $x = \{x_1, x_2, x_3 \ldots x_m\}$ is the current population. So the population-mean is $\overline{x} = \frac{1}{m}\sum_{l=1}^{m} x_l$ and quadratic population-mean is $\overline{x^2} = \frac{1}{m}\sum_{l=1}^{m} x_l^2$.

$Y = \{Y_1, Y_2, Y_3 \ldots Y_m\}$ is the intermediate population obtained after harmony memory and pitch adjustment. Each element Y_l of the population Y is obtained as:

$$Y_l \leftarrow \begin{cases} x_r & \text{with probability } HMCR \cdot (1 - PAR) \\ x_r + bw.rand & \text{with probability } HMCR \cdot PAR \\ x_{new} & \text{with probability } (1 - HMCR) \end{cases}$$

, where r is a uniformly chosen random number from the set $\{1, 2, \ldots, m\}$, x_{new} is a new random value in the allowable range $\{x_{min}, x_{max}\}$ or $\{-a, a\}$ and $rand$ is a uniformly chosen random number between 0 and 1.

The index r being a uniformly distributed random variable with values in $\{1, 2 \ldots, m\}$, the probability $p_k = P(r = k) = \frac{1}{m}$, where k is a number within the set.

Thus x_r is a random variable and

$$E(x_r) = \sum_{k=1}^{m} p_k \cdot x_k = \sum_{k=1}^{m} P(r = k) \cdot x_k = \frac{1}{m}\sum_{k-1}^{m} x_k = \overline{x}$$

(2)

$$E(x_r^2) = \sum_{k=1}^{m} p_k \cdot x_k^2 = \sum_{k=1}^{m} P(r = k) \cdot x_k^2 = \frac{1}{m}\sum_{k-1}^{m} x_k^2 = \overline{x^2}$$

(3)

We now compute $E(Y_l)$ and $E(Y_l^2)$.

Using the lemma 4.1 from [18], we get the following expressions for $E(Y_l)$ and $E(Y_l^2)$ as follows:

$$E(Y_l) = HMCR \cdot (1 - PAR) \cdot E(x_r) + HMCR \cdot PAR \cdot E(x_r + bw \cdot rand) + (1 - HMCR) \cdot E(x_{new})$$

(4)

$$E(Y_l^2) = HMCR \cdot (1 - PAR) \cdot E(x_r^2) + HMCR \cdot PAR \cdot E(x_r + bw \cdot rand)^2 + (1 - HMCR) \cdot E(x_{new}^2)$$

(5)

So now, we need to find out $E(x_{new})$ and $E(x_{new}^2)$. x_{new} is taken from the search range in the following way:

$$x_{new} = x_{min} + rand(0,1) \cdot [x_{max} - x_{min}],$$

where $rand(0,1)$ denotes the uniformly distributed random number lying between 0 and 1. And, we consider $x_{min} = -a$, $x_{max} = a$ and $rand(0,1) = R$. So now,

$$x_{new} = -a + 2 \cdot a \cdot R \quad \text{and hence,}$$
$$x_{new}^2 = a^2 + 4 \cdot a^2 \cdot R^2 - 4 \cdot a^2 \cdot R$$

(6)

The probability distribution function for the random numbers is taken to be continuous uniform probability density function which is graphically shown in Figure 1.
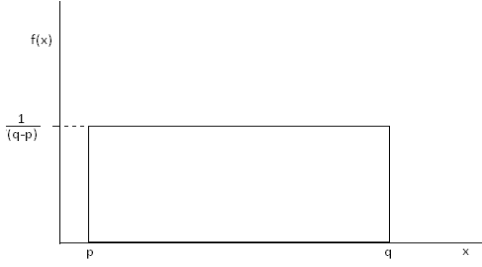
Figure 1. Continuous Uniform Probability Distribution

$f(x) =$ Continuous Uniform Probability Distribution Function. In this case, $p = 0$, $q = 1$ and $x = R$. So,

$E(x_{new}) = -a + 2 \cdot a \cdot E(R)$ [from (5)]. $E(R)$ is computed as follows:

$$E(R) = \int_0^1 R \cdot f(R) \, dR = \int_0^1 R \cdot dR = \left[\frac{R^2}{2} \right]_0^1 = \frac{1}{2} \quad (7)$$

Hence,

$$E(x_{new}) = -a + 2 \cdot E(R) \cdot a = -a + 2a.(1/2) = -a + a = 0 \quad (8)$$

Also $E(x_{new}^2) = a^2 - 4a^2 \cdot E(R) + 4a^2 \cdot E(R^2)$ [from (6)]

And $$E(R^2) = \int_0^1 R^2 f(R) dR = \int_0^1 R^2 dR = \left[\frac{R^3}{3} \right]_0^1 = \frac{1}{3} \quad (9)$$

Hence, $E(x_{new}^2) = a^2 - 4a^2 \cdot \frac{1}{2} + 4a^2 \cdot \frac{1}{3} = \frac{a^2}{3} \quad (10)$

So, from (4), we get,

$$E(Y_l) = HMCR \cdot \bar{x} + HMCR \cdot PAR \cdot \frac{bw}{2} \quad (11)$$

And from (5), we get,

$$E(Y_l^2) = HMCR \cdot \bar{x^2} + (1 - HMCR) \cdot \frac{a^2}{3} + HMCR \cdot PAR \cdot \frac{bw^2}{3}$$
$$+ HMCR \cdot PAR \cdot bw \cdot \bar{x} \quad (12)$$

We know, $E(var(Y)) = E(\overline{Y^2}) - E(\overline{Y}^2)$. So we will have to compute $E(\overline{Y^2})$ and $E(\overline{Y}^2)$ separately.

The quadratic population-mean, $\overline{Y^2} = \frac{1}{m} \sum_{k=1}^m Y_k^2$.

So,

$$E(\overline{Y^2}) = \frac{1}{m} \sum_{k=1}^m E(Y_k^2)$$

$$= \frac{1}{m} \cdot \sum_{k=1}^m \left[\begin{array}{l} HMCR \cdot \bar{x^2} + (1 - HMCR) \cdot \frac{a^2}{3} + HMCR \cdot PAR \cdot \frac{bw^2}{3} \\ + HMCR \cdot PAR \cdot bw \cdot \bar{x} \end{array} \right]$$
[By (12)]

$$= HMCR \cdot \bar{x^2} + (1 - HMCR) \cdot \frac{a^2}{3} + HMCR \cdot PAR \cdot \frac{bw^2}{3} \quad (13)$$
$$+ HMCR \cdot PAR \cdot bw \cdot \bar{x}$$

Now we need to determine $E(\overline{Y}^2)$.

We know, the population-mean, $\overline{Y} = \frac{1}{m} \sum_{k=1}^m Y_k$

So,

$$\overline{Y}^2 = \{ \frac{1}{m} \sum_{k=1}^m Y_k \}^2 = \frac{1}{m^2} [\sum_{k=1}^m Y_k^2 + \sum_{k \neq l} Y_k \cdot Y_l]$$

Also,

$$E(\overline{Y}^2) = E\{ \frac{1}{m} \sum_{k=1}^m Y_k \}^2 = \frac{1}{m^2} E[\sum_{k=1}^m Y_k^2 + \sum_{k \neq l} Y_k \cdot Y_l]$$

$$= \frac{1}{m} E[\frac{1}{m} \sum_{k=1}^m Y_k^2] + \frac{1}{m^2} E[\sum_{k \neq l} Y_k \cdot Y_l]$$

$$= \frac{1}{m} \cdot E[\overline{Y^2}] + \frac{1}{m^2} \cdot \sum_{k \neq l} E(Y_k) \cdot E(Y_l)$$

[As Y_k and Y_l are independent random variables $E(Y_k \cdot Y_l) = E(Y_k) \cdot E(Y_l)$]

$$\Rightarrow E(\overline{Y}^2) = \frac{1}{m} \cdot E(\overline{Y^2}) + \frac{1}{m^2} \cdot m \cdot (m-1) \cdot [E(Y_k)]^2$$

[Since $E(Y_k) = E(Y_l)$]

$$\Rightarrow E(\overline{Y}^2) = \frac{1}{m} \cdot E(\overline{Y^2}) + \frac{1}{m} \cdot (m-1) \cdot [E(Y_k)]^2 \quad (14)$$

So, $E(var(Y)) = E(\overline{Y^2}) - E(\overline{Y}^2)$

$$= (1 - \frac{1}{m}) \cdot E(\overline{Y^2}) + \frac{(m-1)}{m} \cdot [E(Y_k)]^2$$

$$= \frac{(m-1)}{m} \cdot [E(\overline{Y^2}) - \{E(Y_k)\}^2]$$

From (13) and (14) we get,

$$E(var(Y)) = \frac{(m-1)}{m} \cdot \left[\begin{array}{l} HMCR \cdot \bar{x^2} + (1 - HMCR) \cdot \frac{a^2}{3} + HMCR \cdot PAR \cdot \frac{bw^2}{3} \\ + HMCR \cdot PAR \cdot bw \cdot \bar{x} \end{array} \right]$$
$$- \frac{(m-1)}{m} \cdot \left[HMCR \cdot \bar{x} + HMCR \cdot PAR \cdot \frac{bw}{2} \right]^2$$

Simplifying further we have,

$$E(\text{var}(Y)) = \frac{(m-1)}{m} \cdot \begin{bmatrix} HMCR\, \text{var}(x) + HMCR\,(1-HMCR)\cdot \bar{x}^{-2} \\ + HMCR\,(1-HMCR)\cdot PAR\cdot bw\cdot \bar{x} \\ + HMCR\, PAR\cdot bw^2\cdot\left(\frac{1}{3} - \frac{HMCR\, PAR}{4}\right) \\ + \frac{a^2}{3}\cdot(1-HMCR) \end{bmatrix}$$

And the theorem is proved.

Lemma 1.1:

If *HMCR* is chosen to be very high (i.e. very near to 1) and the distance bandwidth parameter (bw) is chosen to be the standard deviation of the current population (i.e. $\sigma(x) = \sqrt{\text{var}(x)}$), then population variance (without selection) will grow almost exponentially over generations.

Proof: The expected variance of the intermediate Y population (obtained after improvisation process) is given by **Theorem 1** as follows:

$$E(\text{var}(Y)) = \frac{(m-1)}{m} \cdot \begin{bmatrix} HMCR\cdot \text{var}(x) + HMCR\cdot(1-HMCR)\cdot \bar{x}^{-2} \\ + HMCR\cdot(1-HMCR)\cdot PAR\cdot bw\cdot \bar{x} \\ + HMCR\cdot PAR\cdot bw^2\cdot\left(\frac{1}{3} - \frac{HMCR\cdot PAR}{4}\right) \\ + \frac{a^2}{3}\cdot(1-HMCR) \end{bmatrix}$$

$$(15)$$

Now, if we make $HMCR \approx 1$ then terms containing $\bar{x}$, $\bar{x}^{-2}$ and a have very little contribution to the overall expected population variance. Hence, if we choose $bw = \sigma(x) = \sqrt{\text{var}(x)}$ (i.e. the standard deviation of the current population) the expression becomes:

$$E(\text{var}(Y)) \approx \frac{(m-1)}{m} \cdot \begin{bmatrix} HMCR\cdot \text{var}(x) \\ + HMCR\cdot PAR\cdot bw^2\cdot\left(\frac{1}{3} - \frac{HMCR\cdot PAR}{4}\right) \end{bmatrix}$$

Neglecting the terms containing $(1 - HMCR)$, we may have,

$$E(\text{var}(Y)) = \frac{(m-1)}{m} \cdot \begin{bmatrix} HMCR \\ + HMCR\cdot PAR\cdot\left(\frac{1}{3} - \frac{HMCR\cdot PAR}{4}\right) \end{bmatrix}\cdot \text{var}(x)$$

$$(16)$$

From (16) it is seen that if we do not include selection in the algorithm, then the expected variance of the gth population (X_g) becomes:

$$E(\text{var}(X_g)) = \left\{ \frac{(m-1)}{m} \cdot \begin{bmatrix} HMCR. \\ + HMCR\cdot PAR\cdot\left(\frac{1}{3} - \frac{HMCR\cdot PAR}{4}\right) \end{bmatrix} \right\}^g \cdot \text{var}(X_0)$$

$$(17)$$

In (17) if we choose the values of the parameters $HMCR, PAR$ in such a way that the term within the second brackets becomes grater than unity, then we can expect an exponential growth of population variance. This growth of expected population variance over generations gives the algorithm a strong explorative power which is essential for an EA.

Hence, (17) supports the assertion made in **Lemma 1.1.**

5. Experimental Results and Observation

The aim of the experimental section is to validate the formula obtained in **Theorem 1** and to justify the claim made in **Lemma 1.1**. This section also provides a comparison between the HS algorithm and another popular optimization algorithm known as Differential Evolution (DE) on the basis of results obtained for popular numeric benchmark functions.

To validate the formula obtained in **Theorem 1**, the expected population variance plot and the theoretical variance plot [obtained from (1)] over generations are compared. The expected population variance has been calculated by averaging the sample variance for all components and for 100 independent runs. In all the runs the values of the parameters are chosen as follows: $PAR = 0.5$, $bw = \sigma(x) = \sqrt{\text{var}(x)}$ (i.e. the standard deviation of the current population x). The different curves in the plot are obtained for different values of the parameter $HMCR$. **Fig. 2(a)** and **Fig. 2(b)** show the expected variance plot and the theoretical variance plot respectively.
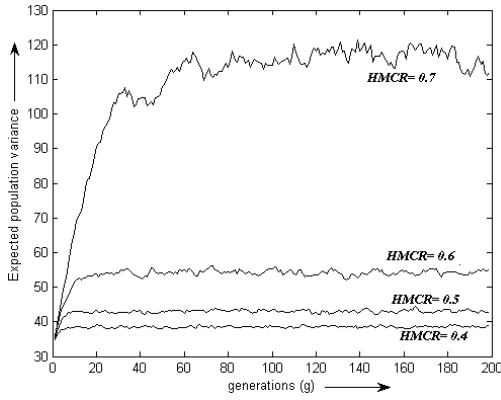
Figure 2(a). Evolution of the expected population variance after Improvisation process [Empirical results]
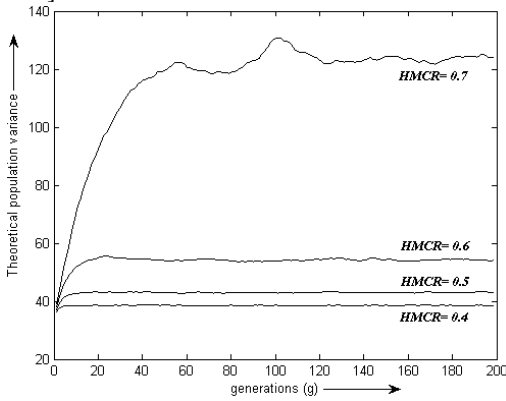

Figure 2(b). Evolution of the expected population variance after Improvisation process [Theoretical results]

To justify the claim made in **Lemma 1.1** the parameters $HMCR$, PAR are chosen in such a way that the value of the expression:
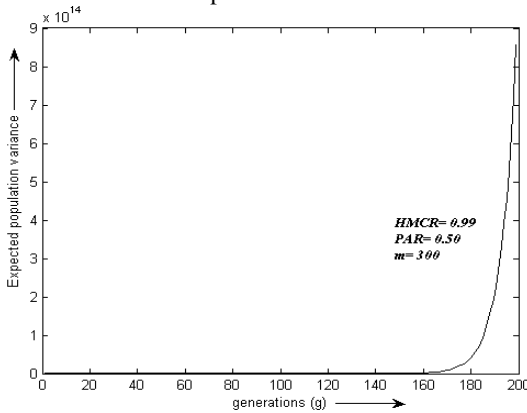

Figure 3: Evolution of expected population variance after choosing the parameter values in accordance with Lemma 1.1

$$\frac{(m-1)}{m} \cdot \left[HMCR. + HMCR \cdot PAR \cdot \left(\frac{1}{3} - \frac{HMCR \cdot PAR}{4} \right) \right]$$

becomes greater than unity. Here, the value of PAR is kept constant at 0.5. The parameter $HMCR$ is changed to meet the above criterion. While doing so it is to be considered that the value of $HMCR$ should be chosen very close to unity so that the dependence of the population variance on the population mean ($\bar{x}$) becomes negligible. The harmony memory size $HMS = m$ is chosen to be 300. Under these circumstances the value of $HMCR$ is conveniently chosen at 0.99. The resultant evolution of population variance is shown in **Fig 3.**

Fig. 3 shows an exponential growth of expected population variance over generations. This supports the assertion made in **Lemma 1.1**. For parameter values given by $HMCR = 0.99, PAR = 0.5, m = HMS = 300$ the value of the expression:

$$\frac{(m-1)}{m} \cdot \left[HMCR. + HMCR \cdot PAR \cdot \left(\frac{1}{3} - \frac{HMCR \cdot PAR}{4} \right) \right]$$

becomes 1.0901, which is greater than unity. However, for **Figure 2** the parameters are so chosen that the value of this expression is less than unity. So, instead of an exponential growth the population variance is saturated after few generations [because of the other additive terms in the original expression (1)]. This exponential growth of population variance (without selection) gives the algorithm a strong explorative power, which is essential for an EA.

The efficiency of this algorithm in finding the optimal solution for several different benchmark functions is compared with that of another popular optimization algorithm known as Differential Evolution (DE) and two recently proposed variants of the classical HS, namely the Improved HS (IHS) [8] and the Global-best Harmony Search (GHS) [9]. **Table 1** shows the comparison of the final accuracy achieved by the two algorithms over four standard numerical benchmarks in 3 dimensions. The results come as averages and standard deviations over 30 independent runs, where each run was continued to 2000 iterations.

Results in **Table 1** show that HS, if modified as suggested in this paper, can work almost as efficiently as DE (we used the DE/rand/1/bin scheme with $NP = 30$, $F = 0.8$, $Cr = 0.9$), which has already emerged as a very powerful optimization technique these days. The table also indicates the modified HS can significantly beat GHS and HIS, which are two state-of-the-art variants of HS. The result obtained in **Theorem 1** provides us the scope to further improve the efficiency of the algorithm by tuning the parameter values.

Table 1: Average and the standard deviation of the

Function	Mean best value (standard deviation)			
	Modified HS	Differential Evolution	GHS	IHS
Rosenbrock's function	1.26e-01 (9.93e-02)	2.00e-03 (3.02e-03)	3.85e-01 (5.83e-02)	2.72e-01 (6.72e-02)
Rastrigin's function	0.00e+00 (0.00e+00)	0.00e+00 (0.00e+00)	5.83e-012 (2.82e-016)	7.82e-013 (4.73e-10)
Griewank's function	1.41e-01 (2.82e-07)	0.00e+00 (0.00e+00)	8.46e-01 (3.77e-02)	4.73e-01 (6.59e-03)
Ackley's function	8.88e-016 (0.00e+00)	9.51e-016 (1.05e-09)	7.35e-010 (4.77e-012)	3.73e-011 (8.25e-013)
Schwefel's Problem 2.22	5.02e-164 (0.00e+00)	2.70e-159 (9.16e-160)	4.83e-100 (3.72e-121)	6.72e-093 (8.24e-109)

best-of-run solution for 30 independent runs tested on numerical benchmarks

6. Conclusions

This paper has presented a mathematical analysis of the evolution of population-variance for the HS meta-heuristic algorithm. The theoretical results indicate that the population-variance of HS can be made to vary exponentially by making the distance bandwidth of HS equal to the standard deviation of the current population. This intensifies the explorative power of HS and the algorithm modified in this way is able to beat the state-of-the-art variants of HS over popular benchmark functions. Even the performance of the algorithm was found to be comparable to that of another very powerful evolutionary algorithm known as Differential Evolution (DE).

References

[1] T. Bäck, D. Fogel, and Z. Michalewicz, *Handbook of Evolutionary Computation*, Oxford University Press, 1997

[2] Z.W. Geem, J.H. Kim, and G.V. Loganathan, A new heuristic optimization algorithm: harmony search, Simulation 76 (2)60–68, 2002.

[3] Z.W. Geem, J.H. Kim, and G.V. Loganathan, Harmony search optimization: application to pipe network design, Int. J. Model. Simul. 22, (2)125–133, 2002

[4] S.L. Kang, Z.W. Geem, A new structural optimization method based on the harmony search algorithm, Comput. Struct. 82 (9–10), 781–798, 2004.

[5] Z. W. Geem, K. S. Lee, and Y. Park, "Application of Harmony Search to Vehicle Routing", *American Journal of Applied Sciences*, 2005.

[6] A. Vasebi, M. Fesanghary, S. M. T. Bathaeea, "Combined heat and power economic dispatch by Harmony Search Algorithm", *International Journal of Electrical Power and Energy Systems*, 2007.

[7] Z. W. Geem, "Optimal Scheduling of Multiple Dam System using Harmony Search Algorithm", *Lecture Notes in Computer Science*, 2007.

[8] M. Mahdavi, M. Fesanghary., and E. Damangir, "An improved Harmony Search Algorithm for Solving Optimization Problems", *Applied Mathematics and Computation*, 2007.

[9] M.G. H. Omran, and M. Mahdavi, "Global-Best Harmony Search", *Applied Mathematics and Computation*, In Press.

[10] M. Fesanghary, M. Mahdavi, M. Minary-Jolandan , and Y. Alizadeh, "Hybridizing Harmony Search Algorithm with Sequential Quadratic Programming for Engineering Optimization Problems", *Computer Methods in Applied Mechanics and Engineering*, 2008.

[11] Z. W. Geem, "Improved Harmony Search from Ensemble of Music Players", *Lecture Notes in Artificial Intelligence*, 2006.

[12] H. G. Beyer, "On the explorative power of ES/EP-like algorithms", *Technical Reports, University of Dortmund*, 1998.

[13] D. Zaharie, "On the explorative power of Differential Evolution", *3rd International Workshop on Symbolic and Numerical Algorithms on Scientific Computing, SYNASC-2001*, Timişoara, Romania, October 2 – 4, 2001.

[14] T. Bäck, and H. P. Schwefel, "Evolution Strategies I. Variants and their Computational Implementation", *Genetic Algorithm in Engineering and Computer Science*, 1995.

[15] H.-P. Schwefel, "Evolution and Optimum Seeking": *New York: Wiley & Sons,* 1995.

[16] R. Storn and K. V. Price, "Differential evolution - a simple and efficient adaptive scheme for global optimization over continuous spaces", *Technical Report TR-95-012, ICSI*, http://http.icsi.berkeley.edu/~storn/litera.html, 1995.

[17] K. Price, R. Storn, and J. Lampinen, "Differential Evolution - A Practical Approach to Global Optimization", Springer, Berlin, 2005.

[18] D.Zaharie, "Recombination Operators for Evolutionary Algorithms", *Proc. of the 27th summer school, Application of mathematics in engineering and economics*, Heron Press, Sophia, 2001.