The Iraqi Journal For Mechanical And Material Engineering, Vol.15, No4, 2015

# OPTIMIZATION of Al-BASE COMPOSITE USING GENETIC ALGORITHM

**Shahad Ali Hammood*, Haydar A.H.Al-Ethari**, Haydar A.H. al- Jubouri*****

***** (Msc. Student at Dept. of Metallurgical Eng., College of Material's Eng./University of Babylon-Hilla-IRAQ) (shahad_ali100@yahoo.com)
****** (Prof. at Dept. of Metallurgical Eng., College of Material's Eng./University of Babylon-Hilla-IRAQ) (draletharihah@yahoo.com)
*** (Ass. Prof. at Dept. of Metallurgical Eng., College of Material's Eng./University of Babylon-Hilla-IRAQ) (drhayderalgibory@yahoo.com)

**ABSTRACT**

This paper provides six tests were carried out for the base alloy (BA) (Al 2%Mg) and the three composite samples (($A_1$ (Al- 2%Mg-2%CKD), $A_2$ (Al-2%Mg-8%CKD) & $A_3$ (Al-2%Mg-16% CKD))) which were prepared by using powder metallurgy technique. As a results, it was found an optimum composite material using the hybrid method represented by genetic algorithms by using through carry out two ways of crossover (1X, 2X), basing on statistical data obtained from experimental results. The basic data were built, depending on their properties, to describe the composite. Then, the evolution algorithm is to make procedure for the genetic clustering process and provides a number of required clusters; to avoid the overlapping between clusters with the other. One of the clustering validity measures called "Davies-Bouldin index" as fitness function of that algorithm that used. Then, the two types of properties for each cluster: mechanical properties (hardness, thermal conductivity, wear rate, friction coefficient) and machining properties (surface roughness, tool life) were extracted. This paper concludes that composite (43&33) represented optimum composite material by using one point and two point crossover operators (1X,2X) respectively.

**Key words: genetic algorithm, optimum, hardness, thermal conductivity, wear rate, friction coefficient, surface roughness, tool life.**

# تحقيق الامثليه للمواد المركبه ذات اساس المنيوم باستخدام الوراثه الجينيه

**شهد علي حمود، د. حيدر عبد الحسن العذاري، د. حيدر عبد حسن الجبوري**

**الخلاصه :-**

هذا البحث يبين ست اختبارات للسبيكة الاساس وثلاث نماذج من المواد المركبه A1,A2,A3 المحضرة باستخدام تقنيه الباودر ميتالورجي. ومن النتائج تم الحصول على السبيكة المثلى باستخدام طرق الوراثة الجينية بالاعتماد على البيانات الإحصائية الناتجة من النتائج التجريبية المتمثله بخواص المواد المركبه. استخدمت داله الصلاحيه لقياس تداخل المجاميع. يوجد نوعين من الخواص لكل مجموعه (الخواص الميكانيكيه وتشمل الصلادة ، الموصليه الحراريه ، معدل البلى ومعامل الاحتكاك) و(الخواص التشغيليه التي تشمل خشونه السطح وعمر العده) . نستنتج من هذا البحث بان السبائك (43 و33) تمثل السبائك المثلى.

## 1. INTRODUCTION :-

A genetic algorithm (GA) is a search and optimization method which works by mimicking the evolutionary principles and chromosomal processing in natural genetics. A GA begins its search with a random set of solutions usually coded in binary strings. Every solution is assigned a fitness which is directly related to the objective function of the search and optimization problem. Therefore, the population of solutions is modified to a new population by applying three operators similar to natural genetic operators-reproduction, crossover, and mutation. It works iteratively by successively applying these three operators in each generation till a termination criterion is satisfied. Over the past decade and more, GAs have been successfully applied to a wide variety of problems, due to their simplicity, global perspective, and inherent parallel processing [Sedighizadeh, 2008].

## 2. LITERATURE REVIEW :-

### 2.1. Overview of the Genetic Algorithms and Operators:

Genetic Algorithm ,GA, is a heuristic used to find approximate solutions for difficult to solve problems through application of the principles of evolutionary biology to computer science. Genetic algorithms use biologically-derived techniques such as inheritance, mutation, natural selection, and recombination (or crossover) [Samaher, 2005].

A basic genetic algorithm that can produce acceptable results in many practical problems is composed of five operators:

1) Reproduction process goal is to allow the genetic information, stored in the good fitness artificial strings, survive the next generation. The typical case is where the population's string has assigned a value according to its aptitude in the object function. This value has the probability of being chosen as the parent in the reproduction process of a new generation [Hussein ,2013].

2) Fitness function plays the most important role in genetic search. This function has to evaluate the goodness of each chromosome in a population [Sedighizadeh M., 2008 & Mansouri, 2012]. Thus, the input of the fitness function is a chromosome and it returns a numerical evaluation representing the goodness of the feature subset. The fitnesss of a chromosome is calculated by using the Davies- Bouldin index. This index is a function of the ratio of the sum of within-cluster scatter to between-cluster separation [Kumar,2014 & Mr.Goual, 2009]. The scatter within the ith cluster, is computed as:

$$\mathbf{Si,q} = \left(\frac{1}{|\mathbf{Ci}|}\sum_{\mathbf{x}\in\mathbf{Ci}}\{\|\mathbf{X}-\mathbf{Zi}\|_2^q\}\right)^{1/q} \qquad (1\text{-}1)$$

and the distance between cluster Ci and Cj is defined as:

$$\mathbf{d\ ij,t} = ||\mathbf{Zi}-\mathbf{Zj}||\mathbf{t} \qquad (1\text{-}2)$$

Si,q is the qth root of the qth moment of the |Ci| points in cluster Ci with respect to their mean zi, and is a measure of the dispersion of the points in the cluster. Specifically Si,q used in this article, is the average Euclidean distance of the vectors in class i to the centroid of class i,dij,t is the Minkowski distance of order t between the centroids zi and zj that characterize clusters Ci and Cj. Subsequently, the study will compute.

$$Ri,qt = \max_{j,j \neq i} \left\{ \frac{Si,q+Sj,q}{dij,t} \right\} \quad (1\text{-}3)$$

The Davies–Bouldin DB index is then defined as:

$$DB = \frac{1}{K} \sum_{i=1}^{K} Ri,qt \quad (1\text{-}4)$$

The objective is to minimize the DB index for achieving proper clustering. The fitness function for chromosome j is defined as 1/DBj, where DBj is the Cavies-Bouldin index computed for this chromosome, where the maximization of the fitness function will ensure minimization of the DB index [Bandyopadhyay, 2001].

3) Crossover operator plays an important role in producing a new generation. The crossover operator is a genetic operator that combines (mates) two chromosomes (parents) to produce a new chromosome (offspring). The idea behind crossover is that the new chromosome may be better than both of the parents if it takes the best characteristics from each of the parents. Crossover occurs during evolution according to a user definable crossover probability. There is number of cross over operators such as: One point crossover, two point crossover and uniform crossover in this research. The study has as a results, used one and two point crossover operators will be used [Abuiziah, 2013].

4) Mutation involves the modification of the value of each 'gene' of a solution with some probability pm, (the mutation probability). The role of mutation in genetic algorithm has been that of restoring lost or unexplored genetic material into the population to prevent premature convergence of the GA to suboptimal solution [Hussein, 2013 & Furdu, 2006].

5) Elitism when creating new population by genetic algorithm processes, we might lose the best chromosome since the selection of chromosomes (or candidate solutions) is more or less done at random. Elitism is the name of method, which first copies the best chromosome (or a few best chromosomes) to new population for further evolution. Elitism can very rapidly increase performance of GA because it prevents losing the best found solution. We have implemented elitism at each generation by preserving the best string seen up to that generation in a location outside the population [Chakraborty, 2003].

## 2.2. Representation of Solution

The chromosomes are makeup of real values (representing the values of the alloy properties that obtain by the laboratory tests as shown in Tables (1 & 2)) by using visual basic language. The length of a chromosome equal N gene while the length of gene is dynamic length the first gene equals one that represent the hardness properties as explained in Fig.(1), the second gene length equals one that represent the thermal conductivity properties, the third gene length equals eight that represent the wear rate properties, the fourth gene length equals eight that represent the friction coefficient properties, the fifth gene length equals eight that represent the surface roughness, the sixth gene length equals eight that represent the tool life properties.

### 2.3. Implementation

The genetic operators are used in the genetic algorithm optimization procedure according to the flowchat given in Fig. 2. It is not necessary to employ all of these operators in a genetic algorithm because each operates independently of the other, the choice or design of operators depends on the problem and the representation scheme employed. For instance operators designed for binary strings cannot be directly used on strings coded with integers or real number.

## 3. RESULTS AND DISCUSSION

This study provides method to reach to the optimum sample using the hybrid method, that represented by statistical parameters and genetic algorithms, where the use of data obtained from experiments to determine the optimum properties of alloys (i.e. in this research have been identified six of the properties of alloys). Accordingly, the database was built describe alloys depending on their properties. Results showed optimization algorithm represented genetic algorithm the chromosome (43) is the optimal alloy for 1X-crossover operator that gives the best properties according to results shown in Table(5) and the chromosome (33) is the optimal alloy for 2X-crossover operator that gives the best properties according to results shown in Table (6).

**Step1** load the alloys database that contain the (50) alloys and (34) feature (represented mechanical properties such as hardness test, thermal conductivity test, wear rate test and friction coefficient test and machining tests such as surface roughness test and tool life test with (F1,F2,F3-F10,F11-F18,F19-F26 and F27-F34) respectively .

**Step2** convert the values of above database to the values in the range [0,1].

**Step3** in this work, we apply the genetic algorithm to find the optimal sample by using two ways of crossover (1X,2X) as shown in Table (3&4) respectively. Genetic algorithm is applied to find the best values of the final results of alloys features. Before this, we need to determine some of parameters related to GA such as (population size= 50 individuals, probability of crossover= 90%, probability of mutation= 10% and number of generation= 100).

## 4. CONCLUSIONS :-

As a results which have presented the work. The study can be the following concluded:

1. The optimal alloy by using one point crossover operator is alloy (43) which means it gives the best properties similar to ($A_3$) alloy properties.

2. The optimal alloy by using two point crossover operator is alloy (33) which means it gives the best properties similar to ($A_3$) alloy properties.

3. As compared to the laboratory results, which need longer time and more cost. The results were obtained by using genetic algorithm in a shorter time and less cost.

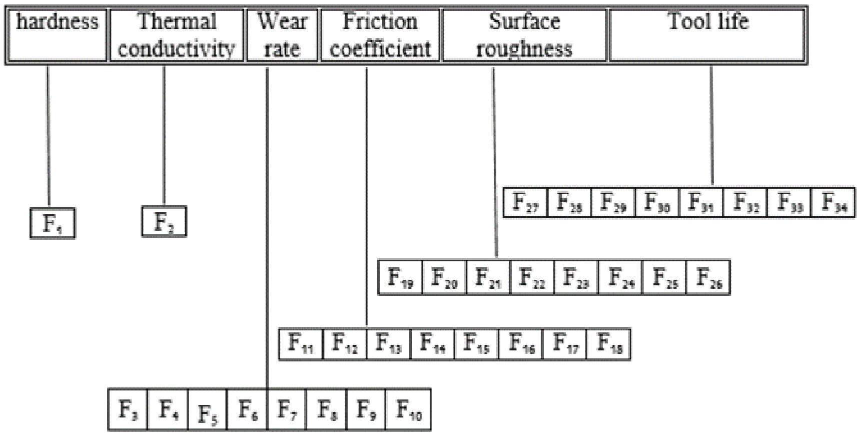


**Fig.(1):** The representation of chromosome.

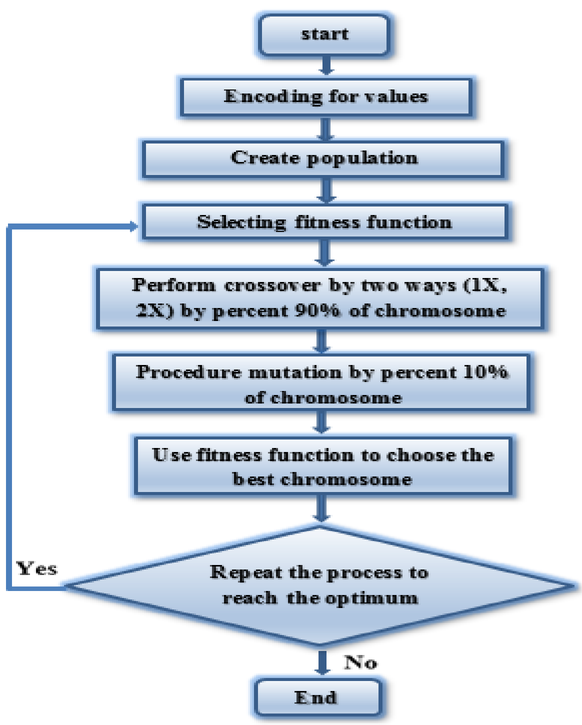


**Fig.2.** Flowchart of a simple genetic algorithm.

**Table 1.** Features of tests results.

| Features | Meaning of features | Features | Meaning of features |
|---|---|---|---|
| $F_1$ | Vickers hardness value at load 200g. | $F_2$ | Thermal conductivity value at temperature 100 °c. |
| $F_3$ | Wear rate at load 4N for 5min. | $F_4$ | Wear rate at load 4N for 10 min. |
| $F_5$ | Wear rate at load 4N for 15 min. | $F_6$ | Wear rate at load 4N for 20 min. |
| $F_7$ | Wear rate at load 10 N for 5 min. | $F_8$ | Wear rate at load 10 N for 10 min. |
| $F_9$ | Wear rate at load 10 N for 15 min. | $F_{10}$ | Wear rate at load 10 N for 20 min. |
| $F_{11}$ | Friction coefficient at 4N for 5 min. | $F_{12}$ | Friction coefficient at 4N for 10 min. |
| $F_{13}$ | Friction coefficient at 4N for 15min | $F_{14}$ | Friction coefficient at 4N for 20min. |
| $F_{15}$ | Friction coefficient at 10N for 5min | $F_{16}$ | Friction coefficient at 10N for 10 min |
| $F_{17}$ | Friction coefficient at 10N for 15 min. | $F_{18}$ | Friction coefficient at 10N for 20 min |
| $F_{19}$ | Surface roughness at speed 80 rpm, feed rate 0.05 m/rev. | $F_{20}$ | Surface roughness at speed 160 rpm, feed rate 0.05 m/rev. |
| $F_{21}$ | Surface roughness at speed 315 rpm, feed rate 0.05 m/rev. | $F_{22}$ | Surface roughness at speed 500 rpm, feed rate 0.05 m/rev. |
| $F_{23}$ | Surface roughness at speed 80 rpm, feed rate 0.1 m/rev. | $F_{24}$ | Surface roughness at speed 160 rpm, feed rate 0.1 m/rev. |
| $F_{25}$ | Surface roughness at speed 315rpm, feed rate 0.1 m/rev. | $F_{26}$ | Surface roughness at speed 500 rpm, feed rate 0.1 m/rev. |
| $F_{27}$ | Tool life at speed 80 rpm, feed rate 0.05 mm/rev. | $F_{28}$ | Tool life at speed 160 rpm, feed rate 0.05 mm/rev. |
| $F_{29}$ | Tool life at speed 315 rpm, feed rate 0.05 mm/rev. | $F_{30}$ | Tool life at speed 500 rpm, feed rate 0.05 mm/rev. |
| $F_{31}$ | Tool life at speed 80 rpm, feed rate 0.1 mm/rev. | $F_{32}$ | Tool life at speed 160 rpm, feed rate 0.1 mm/rev. |
| $F_{33}$ | Tool life at speed 315 rpm, feed rate0.1 mm/rev. | $F_{34}$ | Tool life at speed 500 rpm, feed rate 0.1 mm/rev. |

**Table 2.** Results of laboratory tests.

| Material code | Hardness test (HV) | Thermal conductivity test (w/m.k) | Load (N) | Time (min) | Wear rate (g/cm) | Friction coefficient | Spindle speed (rpm) | Feed rate (mm/rev) | Surface roughness (micron) | Tool life (sec) |
|---|---|---|---|---|---|---|---|---|---|---|
| BA (base alloy) | 50 | 245 | 4 | 5 | 0.009 | 0.77 | 80 | 0.05 | 4 | 177 |
| | | | | 10 | 0.0085 | 0.7 | 160 | | 2.95 | 111 |
| | | | | 15 | 0.0069 | 0.54 | 315 | | 1.902 | 102 |
| | | | | 20 | 0.0051 | 0.48 | 500 | | 1.623 | 51 |
| | | | 10 | 5 | 0.0099 | 0.66 | 80 | 0.1 | 4.412 | 147 |
| | | | | 10 | 0.009 | 0.37 | 160 | | 2.967 | 108 |
| | | | | 15 | 0.0074 | 0.29 | 315 | | 2.4 | 99 |
| | | | | 20 | 0.006 | 0.27 | 500 | | 2.1 | 48 |
| $A_1$ (Al-2%Mg-2%CKD) composite | 54 | 240 | 4 | 5 | 0.007 | 0.61 | 80 | 0.05 | 3.5 | 174 |
| | | | | 10 | 0.0058 | 0.58 | 160 | | 2 | 102 |
| | | | | 15 | 0.003 | 0.44 | 315 | | 1.51 | 84 |
| | | | | 20 | 0.002 | 0.32 | 500 | | 1.356 | 48 |
| | | | 10 | 5 | 0.008 | 0.5 | 80 | 0.1 | 3.621 | 90 |
| | | | | 10 | 0.0073 | 0.32 | 160 | | 3 | 78 |
| | | | | 15 | 0.005 | 0.27 | 315 | | 2.136 | 48 |
| | | | | 20 | 0.003 | 0.22 | 500 | | 1.765 | 46.8 |
| $A_2$ (Al-2%Mg-8%CKD) composite | 63 | 180 | 4 | 5 | 0.004 | 0.49 | 80 | 0.05 | 2.812 | 168 |
| | | | | 10 | 0.0029 | 0.4 | 160 | | 1.7 | 99 |
| | | | | 15 | 0.0018 | 0.35 | 315 | | 1.2 | 48 |
| | | | | 20 | 0.0015 | 0.28 | 500 | | 1 | 45 |
| | | | 10 | 5 | 0.0045 | 0.4 | 80 | 0.1 | 3 | 84 |
| | | | | 10 | 0.0043 | 0.3 | 160 | | 2 | 72 |
| | | | | 15 | 0.003 | 0.2 | 315 | | 1.7 | 47.4 |
| | | | | 20 | 0.002 | 0.18 | 500 | | 1.156 | 42 |
| $A_3$ (Al-2%Mg-16% CKD) composite | 74 | 101 | 4 | 5 | 0.0025 | 0.3 | 80 | 0.05 | 2.123 | 117 |
| | | | | 10 | 0.002 | 0.25 | 160 | | 1.4 | 51 |
| | | | | 15 | 0.0015 | 0.1 | 315 | | 0.937 | 47.4 |
| | | | | 20 | 0.0011 | 0.09 | 500 | | 0.456 | 42 |
| | | | 10 | 5 | 0.0028 | 0.2 | 80 | 0.1 | 2.5 | 63 |
| | | | | 10 | 0.0024 | 0.14 | 160 | | 1.9 | 48 |
| | | | | 15 | 0.002 | 0.08 | 315 | | 1.2 | 45 |
| | | | | 20 | 0.0013 | 0.07 | 500 | | 0.7 | 41.4 |

**Table (3):** Results of genetic algorithm for 1X way of crossover.

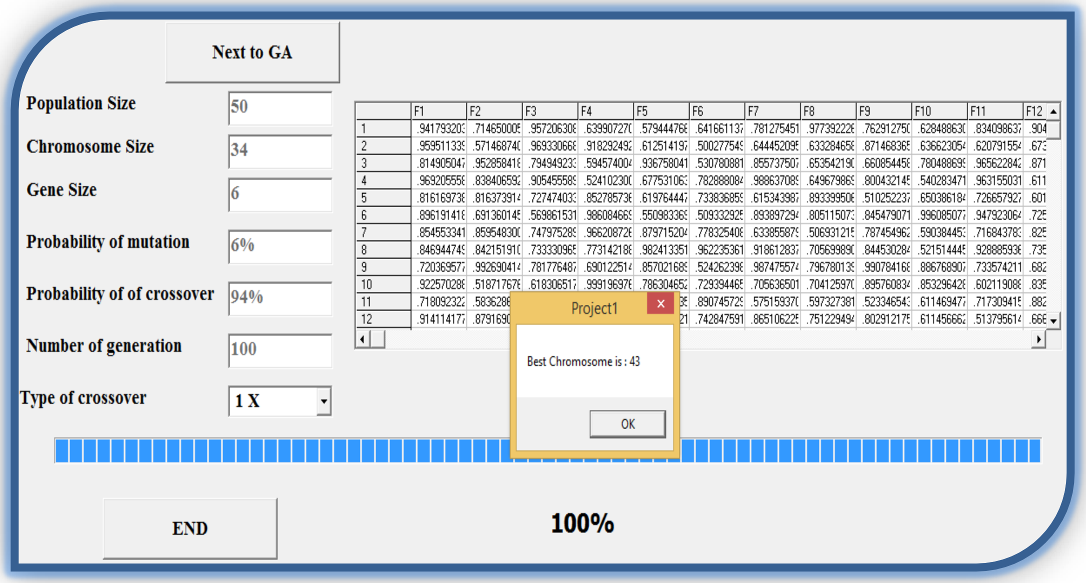


**Table (4):** Results of genetic algorithm for 2X way of crossover.

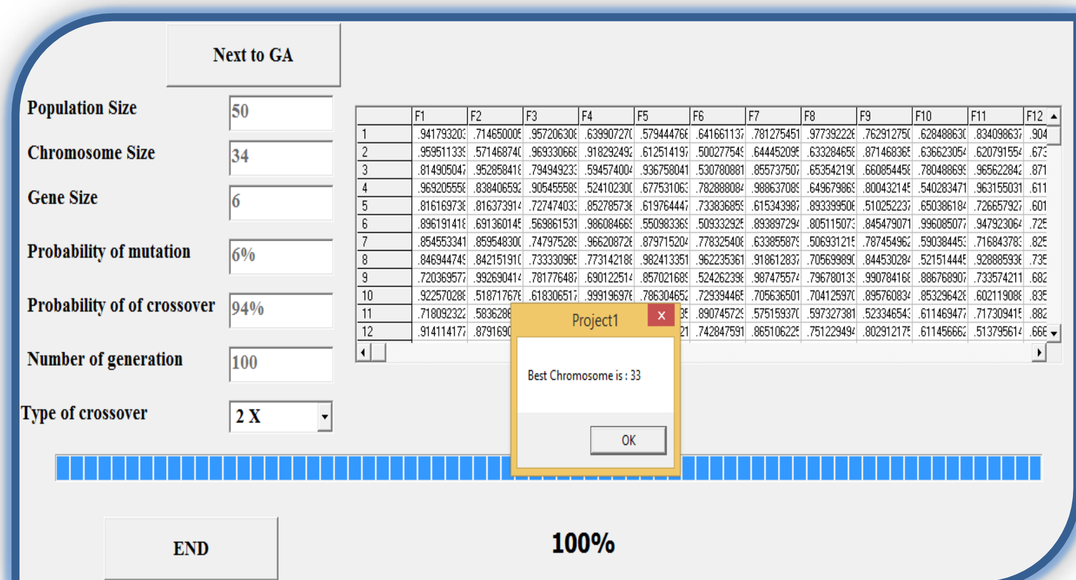


Table 5. Results of the optimization algorithm represented genetic algorithm for 1X-

crossover operator compared with the experimental results.

| Features | Experimental results | Optimization Algorithm |
|---|---|---|
| $F_1$ | 74 | 71 |
| $F_2$ | 101 | 115 |
| $F_3$ | 0.0025 | 0.0030 |
| $F_4$ | 0.002 | 0.004 |
| $F_5$ | 0.0015 | 0.0016 |
| $F_6$ | 0.0011 | 0.0018 |
| $F_7$ | 0.0028 | 0.0025 |
| $F_8$ | 0.0024 | 0.002 |
| $F_9$ | 0.002 | 0.0019 |
| $F_{10}$ | 0.0013 | 0.007 |
| $F_{11}$ | 0.3 | 0.46 |
| $F_{12}$ | 0.25 | 0.5 |
| $F_{13}$ | 0.1 | 0.08 |
| $F_{14}$ | 0.09 | 0.1 |
| $F_{15}$ | 0.2 | 0.28 |
| $F_{16}$ | 0.14 | 0.15 |
| $F_{17}$ | 0.08 | 0.1 |
| $F_{18}$ | 0.07 | 0.15 |
| $F_{19}$ | 2.123 | 2.235 |
| $F_{20}$ | 1.4 | 1.4 |
| $F_{21}$ | 0.937 | 1.1 |
| $F_{22}$ | 0.456 | 0.552 |
| $F_{23}$ | 2.5 | 2.542 |
| $F_{24}$ | 1.9 | 2.243 |
| $F_{25}$ | 1.2 | 1.647 |
| $F_{26}$ | 0.7 | 0.742 |
| $F_{27}$ | 117 | 115 |
| $F_{28}$ | 51 | 49 |
| $F_{29}$ | 47.4 | 46 |
| $F_{30}$ | 42 | 40 |
| $F_{31}$ | 63 | 58 |
| $F_{32}$ | 48 | 39 |
| $F_{33}$ | 45 | 30 |
| $F_{34}$ | 41.4 | 39 |

**Table 6.** Results of the optimization algorithm represented genetic algorithm for 2X-crossover operator compared with the experimental results.

| Features | Experimental results | Optimization Algorithm |
|---|---|---|
| $F_1$ | 74 | 60 |
| $F_2$ | 101 | 141 |
| $F_3$ | 0.0025 | 0.0032 |
| $F_4$ | 0.002 | 0.004 |
| $F_5$ | 0.0015 | 0.002 |
| $F_6$ | 0.0011 | 0.001 |
| $F_7$ | 0.0028 | 0.002 |
| $F_8$ | 0.0024 | 0.0022 |
| $F_9$ | 0.002 | 0.002 |
| $F_{10}$ | 0.0013 | 0.009 |
| $F_{11}$ | 0.3 | 0.43 |
| $F_{12}$ | 0.25 | 0.45 |
| $F_{13}$ | 0.1 | 0.15 |
| $F_{14}$ | 0.09 | 0.1 |
| $F_{15}$ | 0.2 | 0.3 |
| $F_{16}$ | 0.14 | 0.16 |
| $F_{17}$ | 0.08 | 0.06 |
| $F_{18}$ | 0.07 | 0.03 |
| $F_{19}$ | 2.123 | 2.47 |
| $F_{20}$ | 1.4 | 1.77 |
| $F_{21}$ | 0.937 | 1.151 |
| $F_{22}$ | 0.456 | 0.9 |
| $F_{23}$ | 2.5 | 2.614 |
| $F_{24}$ | 1.9 | 2.438 |
| $F_{25}$ | 1.2 | 1.519 |
| $F_{26}$ | 0.7 | 1.022 |
| $F_{27}$ | 117 | 111 |
| $F_{28}$ | 51 | 44 |
| $F_{29}$ | 47.4 | 42 |
| $F_{30}$ | 42 | 38 |
| $F_{31}$ | 63 | 50 |
| $F_{32}$ | 48 | 38 |
| $F_{33}$ | 45 | 28 |
| $F_{34}$ | 41.4 | 37 |

## REFERENCES :

Abuiziah I. & Shakarneh N., 2013**, A Review of Genetic Algorithm Optimization: Operations and Applications to Water Pipeline Systems,** International Journal of Mathematical, Computational, Physical and Quantum Engineering, Vol: 7 No:12.

Bandyopadhyay Sanghamitra & Maulik Ujjwal, 2001**, Nonparametric Genetic Clustering: Comparison of Validity Indices,** Vol. 31, No. 1, February.

Chakraborty Biman & Chaudhuri Probal, 2003**, On The Use of Genetic Algorithm with Elitism in Robust and Nonparametric Multivariate Analysis,** Austrian Journal of Statistics, Volume 32, Number 1&2, 13–27.

Furdu Iulian & Patrut Bogdan, 2006**, Genetic Algorithm for Ordered Decision Diagrams Optimization**, Proceedings of ICMI 45, Bacau, Sept.18-20, pp. 437-444.

Hussein Ali Alwan, Haidar A.H. al-Jubouri & Nabil L. Al Saffar, 2013**, Optimization to improve the Physical and Mechanical Properties of the Electric Power Transmission Wires made from waste using a Genetic Algorithm**, Volume 3 No. 12, December.

Kumar Hemant Bansal & Mahor Devesh, 2014**, Behavioral Modeling of Genetic Algorithm Processor using Model Sim,** Volume 4, Issue 1, January.

Mansouri Leila & Rafeh Reza, 2012**, Proposing a Model for Predicting Flow Stress of Aluminium Alloy in Tensile Test,** International Journal of Engineering Research and Applications, Vol. 2, Issue 6, pp.039-042.

Mr. Goyal Sachin, Dr Gupta Roopam & Mr. Bansal Ashish, 2009**, Application of Genetic Algorithm to Optimize Robustness and Fidelity of Watermarked Images (A conceptual approach**), International Journal on Computer Science and Engineering Vol.1(3), 239-242.

Samaher Hussein Ali & Muhannad M. Al-Yasiry, 2005**, Soft Computing Techniques to Extraction Knowledge of Cardiac SPECT Diagnosis**, Department of Computer Science, University of Babylon, Iraq.

Sedighizadeh M. & Rezazadeh A., 2008**, Using Genetic Algorithm for Distributed Generation Allocation to Reduce Losses and Improve Voltage Profile,** World Academy of Science, Engineering and Technology 37.