

Three Proposed Hybrid Genetic Algorithms

Ban A. Mitras

dr.banah.mitras@gmail.com

Nada F. Hassan

nada.fajer10@gmail.com

College of Computer Sciences and Mathematics

University of Mosul, Mosul, Iraq

Received on: 26/09/2012

Accepted on: 30/01/2013

ABSTRACT

Genetic Algorithm has been hybridized with classical optimization methods. Hybridization has been done in three approaches, by using conjugate gradient algorithm for Fletcher and Reeves, second by using steepest descent method and lastly by creation of initial population for genetic algorithm from one of conjugate gradient method, the numerical results were encouraging.

Keywords: genetic algorithm, conjugate gradient method, and steepest descent method.

ثلاث خوارزميات جينية مهجنة مقترحة

ندى فالح حسن

بان احمد متراس

كلية علوم الحاسوب والرياضيات، جامعة الموصل

تاريخ قبول البحث: 2013/01/30

تاريخ استلام البحث: 2012/09/26

المخلص

تم تهجين الخوارزمية الجينية مع طرائق الأمثلية التقليدية لحل المسائل التصغيرية للدوال ذات الأبعاد الكبيرة في الأمثلية اللاخطية غير المقيدة وتم التهجين بثلاث أساليب: أولاً باستخدام إحدى طرائق الاتجاه المترافق للعالمان Fletcher و Reeves، ثانياً باستخدام طريقة الانحدار السلبي، ثالثاً باستخدام مجتمع ابتدائي من إحدى طرائق الاتجاه المترافق وتم تطبيقها على ثلاث دوال اختبارية والنتائج كانت مشجعة. الكلمات المفتاحية: الخوارزمية الجينية، طريقة الاتجاه المترافق، طريقة الإنحدار الحاد.

1. Introduction:

Optimization is an important tool to any problem involving decision making in many disciplines, such as engineering, mathematics, statistics, economics, and computer science. Now, more than ever, it is increasingly vital to have a firm grasp of the topic due to the rapid progress in computer technology, including the development and availability of, high-speed and parallel processors, and networks.

Mathematically speaking, optimization is the minimization or maximization of a function either subject to no constraints, so it is called unconstrained optimization which has the mathematical formulation:

$$\min_x f(x), \quad \dots(1)$$

where $x \in R^n$ is a real vector with an $n \geq 1$ and an $f : R^n \rightarrow R$ is a smooth function, or subject to constraints on its variables, so it is called constrained optimization [9].

2. Genetic Algorithms

Genetic algorithms are searches and optimization tools inspired by evolutionary processes. Their strength lies in their ability to evolve near-optimal solution to complex problems.

Genetic algorithms which are pioneered by John Holland about 40 years ago combine selection, crossover, and mutation operators with the goal of finding the best solution to a problem. Genetic algorithms search for this optimal solution until a

specified termination criterion is met. A genetic algorithm creates an initial population (a collection of chromosomes), evaluates this population, and then evolves the population through multiple generations in the search for a good solution for the problem at hand, genetic algorithms follow the process which can be seen in Figure (1) [4] and [6].

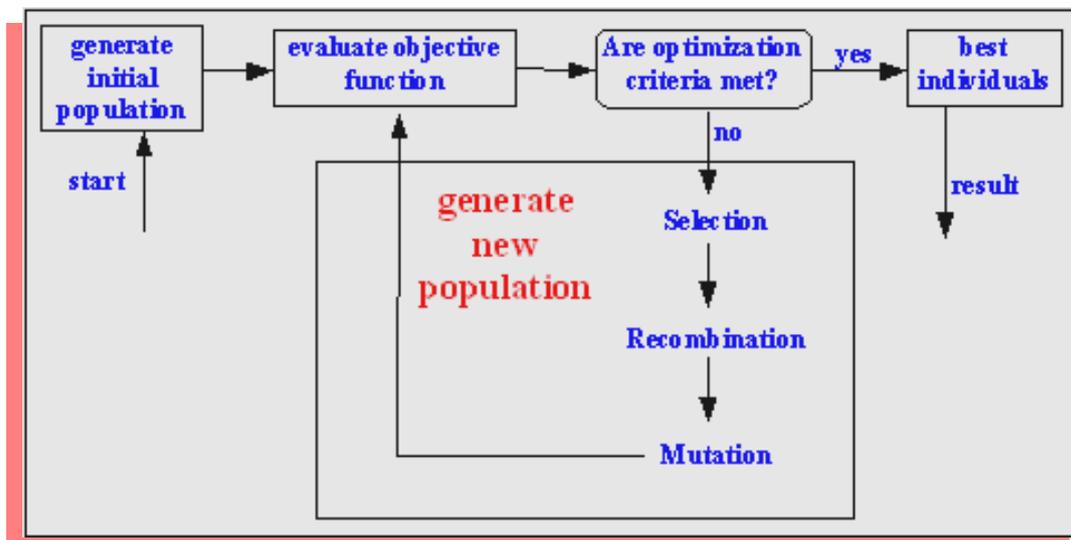


Figure (1). Structure of a Simple Genetic Algorithm

Genetic algorithms which have been recently used in solving complicated scientific problems, some hybrid genetic algorithms are developed by combining genetic algorithms with heuristic search strategy based on gradient such as conjugate gradient [10].

Previous Works:

In 1998 Okamoto et al. proposed a hybrid genetic algorithm with incorporating the modified Powell method for non-linear numerical optimization [8].

In 2003 Chelouah and Patrick proposed a continuous new hybrid algorithm combining genetic algorithm and Nelder-Mead simplex algorithm for continuous multimodal optimization problems [3].

In 2006 Kumar and Debroy used the conjugate gradient method and a hybrid optimization scheme involving conjugate gradient method and genetic algorithm to calculate the weights in the neural network model. The hybrid optimization scheme helped in finding optimal weights through a global search [5].

In 2007 Tantar, et al. proposed a parallel hybrid genetic algorithm (GA) for solving the structure of the predictive problem. Conjugated gradient-based Hill Climbing local search is combined with the GA, in order to efficiently deal with the problem by using the computational grid [12].

In 2007 Tahk, et al. proposed a hybrid optimization algorithm which combines evolutionary algorithm and the gradient search technique for optimization with continuous parameters[11].

In 2008 Zhou-Shun et al. proposed an algorithm which combines the local searching ability of conjugate gradient method with global search ability of GA organically [14].

In 2009 Li, et al. proposed a hybrid genetic algorithm which combines the conjugate gradient method with genetic algorithm to improve the performance of genetic algorithm for cable forces optimization [7].

In 2010 Xie and Liu proposed a hybrid genetic algorithm for geo-physical inversion. According to the properties of the genetic algorithm and the conjugate gradient algorithm, the method has the attributes of the global-convergence of the genetic algorithm and the fast convergence of the conjugate gradient [13].

3. Classical Optimization Methods:

3.1 Conjugate Gradient Algorithm [1]:

Conjugate gradient (CG) methods represent an important class of unconstrained optimization algorithm. The main advantages of the CG methods are its low memory requirements, its convergence speed and its poses a quadratic termination property in which the method is able to locate the minimum of quadratic function in a known finite number of iterations.

A non-linear conjugate gradient method generates a sequence $\{x_k\}$, k is an integer number, $k \geq 0$. Starting from an initial point x_0 , the value of x_k is calculated by the following equation:

$$x_{k+1} = x_k + \lambda_k d_k, \quad \dots(2)$$

where, the positive step size $\lambda_k > 0$ is obtained by a line search, and the directions d_k are generated as:

$$d_{k+1} = -g_{k+1} + \beta_k d_k, \quad \dots(3)$$

where, $d_0 = -g_0$, the value of β_k is determined according to the algorithm of Conjugate Gradient (CG), and its known as a conjugate gradient parameter, $s_k = x_{k+1} - x_k$ and $g_k = \nabla f(x_k) = f'(x_k)$, consider $\|\cdot\|$ which is the Euclidean norm and $y_k = g_{k+1} - g_k$. The termination conditions for the conjugate gradient line search are often based on some version of the Wolfe conditions. The standard Wolfe conditions are as follows:

$$f(x_k + \lambda_k d_k) - f(x_k) \leq \rho \lambda_k g_k^T d_k, \quad \dots(4)$$

$$g(x_k + \lambda_k d_k)^T d_k \geq \sigma g_k^T d_k, \quad \dots(5)$$

where d_k is a descent search direction and $0 < \rho \leq \sigma < 1$, where β_k is defined by one of the following formulas:

$$\beta_k^{(HS)} = \frac{y_k^T g_{k+1}}{y_k^T d_k} \text{ (Hestenes and Stiefel)} \quad \dots(6)$$

$$\beta_k^{(FR)} = \frac{g_{k+1}^T g_{k+1}}{g_k^T g_k} \text{ (Fletcher and Reves)} \quad \dots(7)$$

$$\beta_k^{(PRP)} = \frac{y_k^T g_{k+1}}{g_k^T g_k} \text{ (Polak - Ribiere ;and Polyak)} \quad \dots(8)$$

$$\beta_k^{(CD)} = -\frac{g_{k+1}^T g_{k+1}}{g_k^T d_k} \text{ (Conjugatedescent)} \quad \dots(9)$$

$$\beta_k^{(LS)} = -\frac{y_k^T g_{k+1}}{g_k^T d_k} \text{ (Liu and Stoery)} \quad \dots(10)$$

$$\beta_k^{(DY)} = \frac{g_{k+1}^T g_{k+1}}{y_k^T s_k} \text{ (Dai and Yuan)} \quad \dots(11)$$

Outlines of the CG methods:

Step1: Given $x_0 \in R^n$, compute $g_0 = \nabla f(x_0)$. If $\|g_0\| \leq \epsilon = 10^{-6}$ then stop, otherwise, set $k=1$ and continue

Step 2: Set $d_k = -g_k$.

Step 3: Compute $x_{k+1} = x_k + \alpha_k d_k$.

Step 4: If $\|g_{k+1}\| < \epsilon$ then stop else continue.

Step5: Compute the new search direction defined by: $d_{k+1} = -g_{k+1} + \beta_k d_k$. Where, β_k is the conjugacy coefficient.

Step6: Check for convergence if $\|g_{k+1}\| < \epsilon$ then stop, otherwise set $k=k+1$ and go back to step 3.

3.2 Steepest Descent (SD) Method:

The Cauchy's steepest descent method is one of the oldest and most widely known simplest method for solving unconstrained minimization problems

The method of steepest descent is a fundamental first order method. One advantage of the steepest descent method is that it requires calculation of the first gradient, but not second derivatives. However, it can be excruciatingly slow on difficult problems.

The SD method is the simplest method for solving non-linear optimization problems. The SD method is a line search method that moves along $d_k = -g_k$ at every step.

4. Three Proposed Hybrid Genetic Algorithms:

This section contains three new hybrid genetic algorithms with conjugate gradient methods :

4.1 First Proposed Hybrid Genetic Algorithm:

1-Create Initial Population:

The population consists of a number of individuals detected by the algorithm designer and according to the nature of the problem. In the first hybrid algorithm, the population has been detected by 12 individuals which are generated randomly. The chromosome consists of 12 number of values which are decimals.

2- Fitness Evaluation:

The fitness of a solution is a measure that can be used to compare solutions to determine which is better. The fitness functions in this work are the test functions that are defined in section 6.

3- Selection:

In this phase, we have to create a new population from the current generation. The selection operation determines which parent chromosomes participate in producing

offspring for the next generation. The most common way is to set the selection probability equals to:

$$Pro(x) = \frac{f(x_i(t))}{\sum_{i=1}^{pop_size} f(x_i(t))} \quad \dots(12)$$

$f(x)$: fitness value of chromosome.

i : chromosome number; t : population number.

4- Crossover Operator:

Crossover operator is usually applied to select pairs of parents. single-point crossover is the most basic crossover operator, where a crossover point is selected randomly, and two parent chromosomes are interchanged at this point.

5- Mutation:

The most common way of implementing mutation is the order changing mutation (which is used in this work), two positions are chosen randomly in the chromosome and exchange their values with a probability equals to a very low (which is equal to 0.08 in this work).

After mutation, a new population had been created. Each chromosome in the new population is an initial point for conjugate gradient method, then the hybridization with conjugate gradient start. The steps of conjugate gradient eq. (2) and (3) are executed to generate a new population to start genetic algorithm again until the stopping criterion is satisfied.

6- Stop Criterion:

The stopping criterion in the three hybrid genetic algorithms decides whether the algorithm continues in searching or stops. The stop criterion here depends on two approaches: generations number or minimum function value.

• Outlines of the First Proposed Hybrid Algorithm

Step1: Create initial population randomly, $k=0, 1, 2, \dots, n$.

Step2: Calculate fitness function for each chromosome in the population.

Step3: [New population] Create a new population by repeating the following steps until a new population is complete.

- **(Selection)** Select two parent chromosomes from a population according to their fitness.
- **(Crossover) (recombination)**] Crossover the parents to form a new offspring (children).
- **(Mutation)** Mutate new offspring by using order change at random position.

Step4: Let x_k =new population, Set $k=k+1$.

Step5: If $k=1$ then $d_1=-g_1$, where g is the gradient vector else continue.

Step6: $x_{k+1}=x_k+\alpha d_k$ where α is a step size which is equal to 0.001.

Step7: Calculate fitness function for each chromosome in the population, then find minimum function value.

Step8: If $k=n$ or minimum function value is very small then stop; print Minimum function value, otherwise continue.

Step9: $d_{k+1}=-g_{k+1}+\beta_k*d_k$, where β_k is FR formula as defined in eq.(7) go to step 3.

4.2 Second Proposed Hybrid Genetic Algorithm:

The steepest descent method is used in the second proposed hybrid algorithm as follows:

• Outlines of the Second Proposed Hybrid Algorithm

Step1: Create initial population randomly, $k=0,1,2,\dots n$.

Step2: Calculate fitness function for each chromosome in the population.

Step3: [New population] Create a new population by repeating the following steps until a new population is complete.

- **(Selection)** Select two parent chromosomes from a population according to their fitness.
- **(Crossover) (recombination)]** Crossover the parents to form a new offspring (children).
- **(Mutation)** Mutate new offspring using order change at random position.

Step4: Let x_k =new population, Set $k=k+1$.

Step5: If $k=1$ then $d_1=-g_1$, where g is the gradient vector else continue.

Step6: $x_{k+1}=x_k+\alpha d_k$ where α is a step size which is equal to 0.001.

Step7: Calculate fitness function for each chromosome in the population, then find minimum function value.

Step8: If $k=n$ or minimum function value is very small, then stop; print Minimum function value, otherwise set $d_{k+1}=-g_{k+1}$, go to step 3.

4.3 Third Proposed Hybrid Genetic Algorithm:

It is the same as the first hybrid algorithm except that the initial population has been created by execution classical conjugate gradient method for each test function several times with different dimensions. At each exestuation the minimum point vector represents a chromosome in the initial population and so on, so the algorithm and the flow chart for the third hybrid are the same as the first proposed hybrid except the creation of the initial population.

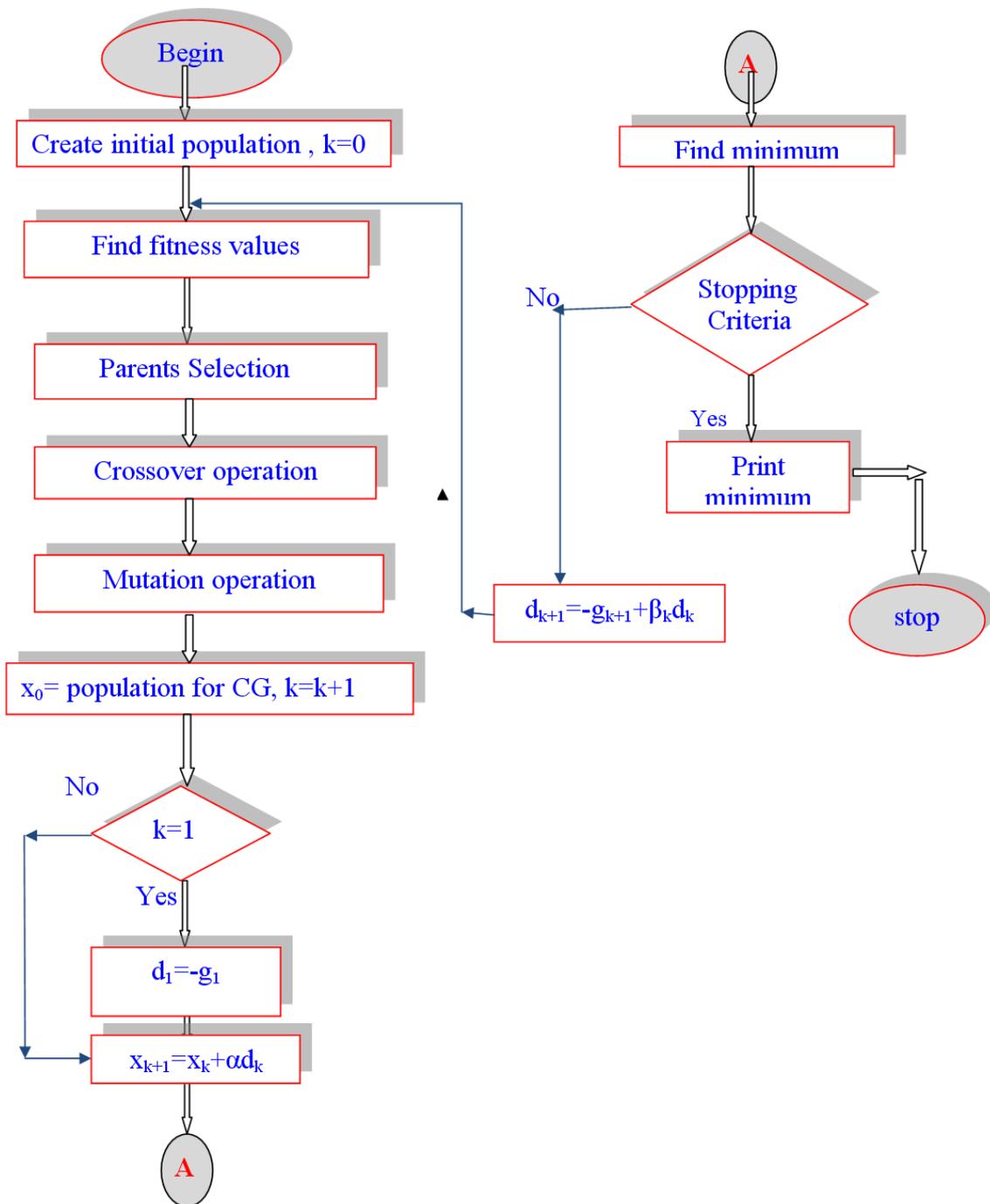


Figure (2). Flow Chart for the First Hybrid Algorithm

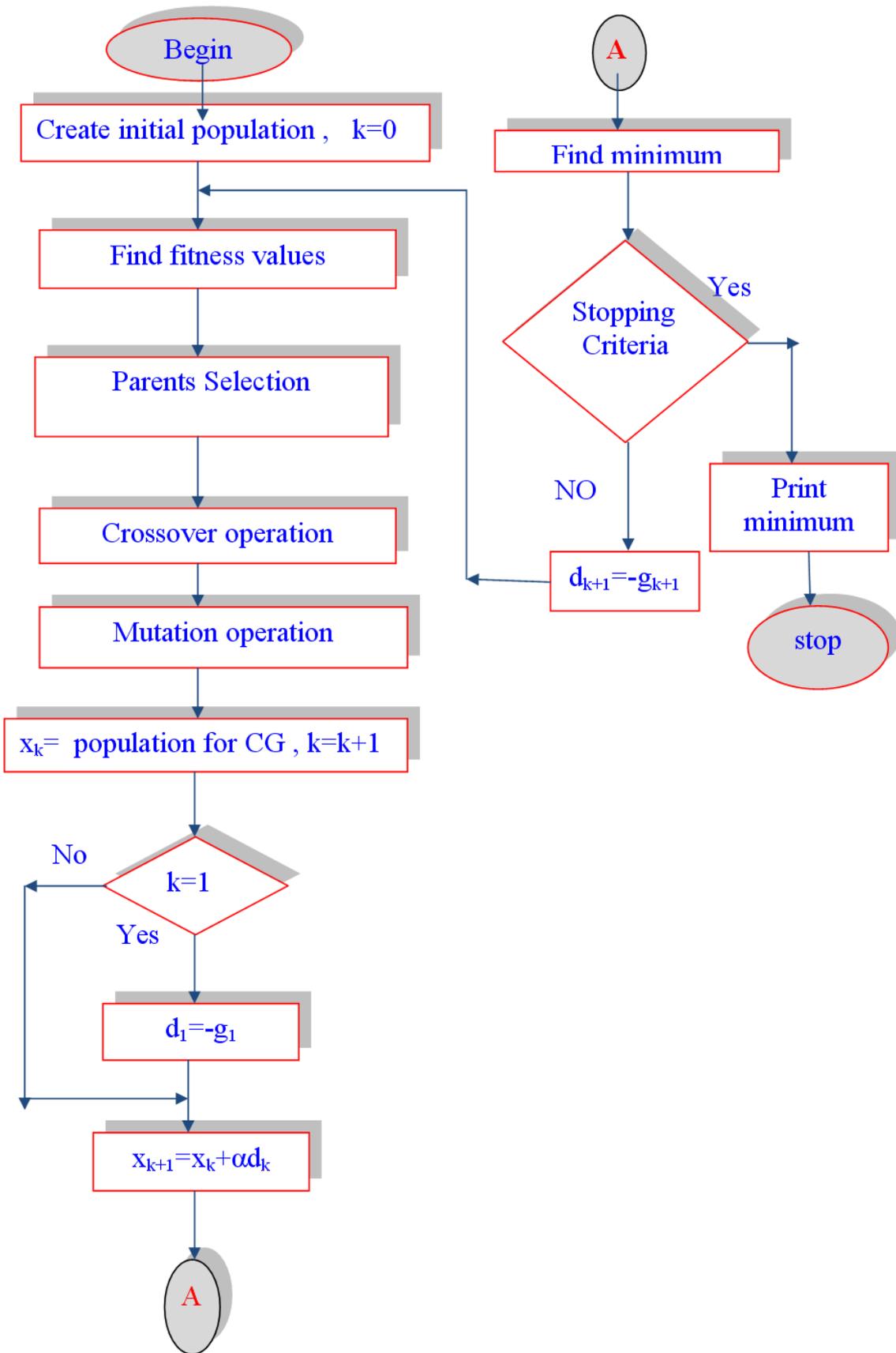


Figure (3). Flow Chart for the Second Hybrid Algorithm

5. Numerical Results and Discussion:

In order to assess the performance of the new three hybrid algorithms, some generalized selected well-known test functions, Powell function, Osp (Oren and Spdicato) function, and Diagonal4 function are used.

All programs are written in MATLAB version R2007a. the comparative performances for all these hybrid algorithms are evaluated by considering minimum function value or number of generations.

The initial population is randomly chosen in the first and second hybrid algorithms, so the first stopping criterion which is the number of generations is used to determine the best minimum for each test function. After determining the best minimum for each function, the second stopping criterion which is the best minimum is used to determine the number of generations to reach the best minimum.

Fifty trials are carried out for each test function in each hybrid algorithm and the following numerical results for each hybrid algorithm and for each test function are concluded out of these trials.

Comparison between three new hybrid genetic algorithm and genetic algorithm is shown in Table (1) with respect to the minimum value. In Table (1) HGA1, HGA2, HGA3 refers to first, second and third proposed hybrid genetic algorithm.

5.1 Numerical Results for the First Proposed Hybrid Algorithm:

1- Powell Function: The first stop criterion is a number of generations which is equal to 5000 generation, the minimum values for this function out of 50 trials are:

$$1E - 14 \leq \min \leq 1E - 11$$

The second stop criterion is, when the minimum value the for this function is:

$$\min \leq 1 \times E - 14$$

using β_k as defined in eq.(7), the results to reach the minimum ($\min \leq 1 \times E - 14$) out of 50 trials are:

Six trials out of 50 trials are failed. Two trials reach $1 \times E - 13$ minimum in 5000 generation, the rest trials (42 trials) reach the minimum ($1 \times E - 14$) in the generations: 3884, 3872, 4080, 3831, 4228, 4310, 4003, 3727, **3718**, 4181, 4097, 4137, 3897, 4421, 4003, 4339, 4999, 4298, 4228, 3976, 3955, 4845, 4170, 4069, 4374, 4431, 4486, 3746, 4255, 3855, 3966, 4257, 4464, 4167, 3878, 4427, 4610, 3983, 4139, 4250, 3483 and 4018.

2- Osp (Oren and Spdicato) Function: The first stop criterion is a number of generations which is equal to 5000 generation, the minimum

$$1E - 6 \leq \min \leq 1E - 4$$

The second stop criterion is, when the minimum value for this function is:

$$\min \leq 1 \times E - 4$$

Using β_k as it is defined in eq.(7), the results to reach the minimum ($\min \leq 1 \times E - 4$) out of 50 trials are:

Sixteen trials reach the minimum value ($1 \times E - 03$) in 5000 generation, the rest trials (34 trials) reach the minimum which ($1 \times E - 4$) in the generations:

83, 183, 74, 123, 113, 106, 155, 3699, 89, 379, 2936, 147, 4459, 101, 633, 163, 97, **67**, 70, 132, 162, 115, 77, 487, 139, 75, 107, 130, 267, 156, 3122, 102, 4294 and 2467.

3- Diagonal4 Function: The first stop criterion is a number of generations which is equal to 5000 generation, the minimum values for this function out of 50 trials are:

$$1E-12 \leq \min \leq 1E-10$$

The second stop criterion is, when the minimum value for this function is:

$$\min \leq 1 \times E - 12$$

Using β_k as defined in eq.(7), the results to reach the minimum

($\min = 1 \times E - 12$) out of 50 trials are:

Six trials reach the minimum value ($1 \times E - 011$), six trials reach the minimum value ($1 \times E - 010$), two trials reach the minimum value ($1 \times E - 09$) and one trial reaches the minimum value ($1 \times E - 08$). They all reached these minimum value in 5000 generation. The results of the rest trials (35 trials) reached the minimum ($1 \times E - 12$) in the generations:

4520, 4370, 4151, 4076, **3409**, 4320, 4462, 4661, 4737, 4517, 4173, 4956, 4895, 4701, 4987, 4536, 4624, 3920, 4987, 4462, 4631 and 4998.

5.2 Numerical Results for the Second Proposed Hybrid Algorithm:

1- Powell Function: The minimum values for this function in 5000 generations are $\min \leq 1E-08$, in 50 trials fourteen trials failed to reach any minimum, the rest of trials reached this minimum in the following generations:

2214, 2339, 2275, 2139, 2275, 1851, 2629, 2325, 2773, 2754, 2123, 2250, 2396, 2039, 2215, 1991, **1770**, 2466, 2164, 2299, 2168, 2150, 2518, 2188, 2858, 2330, 2790, 2877, 2186, 2233, 2722, 2212, 2380, 2029, 2707 and 2321.

2- Osp (Oren and Spdicato) Function: The minimum values for this function in 5000 generations are the same minimum values as the first hybrid algorithm, in 50 trials, twenty-six trials reach the minimum $1 \times E - 03$, one trial reaches the minimum ($1 \times E - 07$) in (137) generations, and another trial reaches the minimum $1 \times E - 08$ in (185) generations which is the best minimum, the rest of trials reach the minimum ($1 \times E - 04$) in the following generations:

120, 143, 81, 106, 4809, 2086, 91, 367, **73**, 2299, 129, 77, 149, 122, 95, 112, 596, 95, 320, 84, 130 and 107.

3- Diagonal4 Function: The minimum values for this function in 5000 generations are the same minimums as the first hybrid algorithm.

In 50 trials, one trial reaches the minimum $1 \times E - 13$ in (1175) generation, which is the best minimum, the rest of trials reach the minimum ($1 \times E - 012$) in the following generations:

1305, 1192, 1272, 1258, 1378, 1146, 1157, 1369, 1236, 996, 1328, 1302, 1389, **1104**, 1251, 1283, 1352, 1236, 1191, 1211, 1307, 1209, 1278, 1221, 1282, 1304, 1443, 1113, 1127, 1225, 1276, 1174, 1197, 1336, 1445, 1323, 1418, 1180, 1312, 1247, 1325, 1114, 1278, 1451, 1234, 1569, 1214, 1393 and 1190.

5.3 Numerical Results for the Third Proposed Hybrid Algorithm:

1- Powell Function: Conjugate gradient method with this test function had been executed in 11 different dimensions. The minimized values for this function, in all these dimensions, are about ($1.0E-06$) - ($1.0E-07$).

The minimized values vectors in these different dimensions plus the initial point for this test function (3,-1.0,1,...) are used to create the initial population (12 chromosome) for the third hybrid method.

After two iteration, the third hybrid method gave the minimum (7.540316246596e-009) for this test function.

2- Osp (Oren and Spdicato) Ffunction: Conjugate gradient method with this test function had been executed in different dimensions. The minimized value for this function in all these dimensions is about (E-06).

The minimized values vectors in these different dimensions plus the initial point for this test function (1, ...) are used to create the initial population (12 chromosome) for the third hybrid method.

After two iteration, the third hybrid method gave the minimum (4.836750314048e-024) for this test function

3- Diagonal4 Function: Conjugate gradient method with this test function had been executed in different dimensions. The minimized value for this function in all these dimensions is about (1.0E-07-1.0E-09).

The minimized values vectors in these different dimensions plus the initial point for this test function (1, 1, ...) are used to create the initial population (12 chromosome) for the third hybrid method.

After one iteration, the third hybrid method gave the minimum (1.780114834221e-011) for this test function.

Table (1). Comparison Between Three Proposed Hybrid Genetic Algorithms and Original Genetic Algorithm.

function	HGA1 minimum	HGA2 minimum	HGA3 minimum	Original GA minimum
1- Powell	(1E - 11) - (1E - 14)	1E - 08	7.540316246596e-09	0.667248567724685
2- Osp.	(1E - 4) - (1E - 6)	1E - 04	4.836750314048e-24	0.146460647330478
3- Diagonal4	(1E - 10) - (1E - 12)	1E - 012	1.780114834221e-11	0.853013845034343

6. Appendix: These three test functions from [2]:

1. *Generalized powell function:*

$$f(x) = \sum_{i=1}^{n/4} \left((x_{4i-3} - 10x_{4i-2})^2 + 5(x_{4i-1} - x_{4i})^2 + (x_{4i-2} - 2x_{4i-1})^2 + 10(x_{4i-3} - x_{4i})^4 \right)$$

Starting point: (3,-1,0,1,.....) ^T

2. *Generalized OSP function:*

$$f(x) = \left(\sum_{i=1}^n ix_i^2 \right)^2$$

Starting point: (1,.....) ^T

3. *Diagonal4 function:*

$$f(x) = \sum_{i=1}^{n/2} \frac{(x_{2i-1}^2 + c * x_{2i}^2)}{2.0}, c = 100$$

Starting point: (1,1,.....) ^T

REFERENCES

- [1] Andrei, N., (2007), "Scaled conjugate gradient algorithms for unconstrained optimization" *Comput. Optim. Appl.* 38, pp.401-416.
- [2] Andrei, N., (2008), "An Unconstrained Optimization Test Functions Collection", *Adv. Model. Optimiz.* Vol. 10, pp. 147-161.
- [3] Chelouah, R. and Partick, S., (2003), "Genetic and Nelder-Mead Algorithms Hybridized for More Accurate Global Optimization of Continues Multi-Minima Functions" *European Journal of Operational Research*, Vol. 148, pp.335-345.
- [4] Koner, A., (2000), "Artificial Intelligence and Soft Computing", CRC Press, Inc.
- [5] Kumer, A., and Deproy, T., (2006), "Neural Network Model and Fluid Flow in Gas Metal arc Fillet Welding Based on GA and CG Optimization", *Science and Technology of Welding and Joining*, Vol. 11, No.1.
- [6] Lavender, B.E., (2010), "Implementation of Genetic Algorithms into a Network Intrusion Detection System (netga) and integration into Nprobe", M.Sc. Thesis, California State University, Sacramento.
- [7] Li, J.; Li, S. and Zhang, B., (2009), "A Hybrid Genetic Algorithm for Cable Forces Optimization of CFST Arch Bridge", *Computational and Software Engineering*, CiSE, pp.1-4.
- [8] Okamoto, M.; Nonaka, T.; Ochiai, S., and Tominaga, T., (1998), "Nonlinear Numerical Optimization with use of a Hybrid GA Incorporating the Modified Powell Method", *Applied Math. and Computation J.* Vol. 91 pp.63-72.
- [9] Patel, R.V., (2000), "System Optimization", Dept. of Electrical and Computer Engineering University of Western Ontario London, Ontario, Canada.
- [10] Sun, L. and Zhang, W., (2006), "An Acceleration Micro Genetic Algorithm for Numerical Optimization", *Lecture Notes in Computer Science*, Vol. 4247, pp 277-283.
- [11] Tahk, M.; Woo, H.; and Park, M., (2007), "A Hybrid Optimization Algorithm of Evolutionary Algorithm and Gradient Search", *Engineering Optimization*, Vol. 39, No.1, pp87-104.
- [12] Tanter, A., Melab, N., Talib, E.G., Parent, B and Hoevath, D., (2007), "A Parallel Hybrid Genetic Algorithm for Protein Structure Prediction on the Computational Grid", *Future eneration Computer System*, Vol. 23, pp.398-409.
- [13] Xie, W. and Liu, J., (2010), " Integrated Programming and Application of Genetic Algorithm and Conjugate Gradient Method" *Progress in Electromagnetic Research Symposium Proceeding*, Xian, China, pp. 22-26.
- [14] Zhou-Shun, Z.; Xiao-Hui, Y. and Guanghui, H., (2008), "GA with Conjugate Gradient Operator" *J. of Shangrao Normal*, No. 03.