ChemCon'04 Mumbai

Optimization Using Hybrid Differential Evolution Algorithms

B.V. Babu* and Rakesh Angira
Department of Chemical Engineering
Birla Institute of Technology and Science
Pilani – 333 031 (Rajasthan) India
Fax: 91 1596 244183; Tel: 91 1596 245073 Ext. 205;
E-mail: bvbabu@bits-pilani.ac.in < B V Babu >
angira@bits-pilani.ac.in < Rakesh Angira >

Abstract: In recent years, evolutionary algorithms (EAs) have been used for the solution of nonlinear multimodal problems encountered in many engineering disciplines. They differ from the traditional gradient based algorithms since, in general, only the information regarding the objective function is required. In the present study, a hybrid evolutionary algorithm is proposed. This new method is a combination of Differential Evolution (DE) and classical Quasi-Newton method and named as Hybrid Differential Evolution (HDE). HDE is used to solve the benchmark test functions and then evaluated for water pumping system. Performance of HDE is compared with DE. The results show that both DE and HDE have high reliability in locating the global minimum, and that HDE converges faster than DE thus reducing the computational time and number of function evaluations.

Key Words: Evolutionary Computation, Optimization, Differential Evolution, Quasi-Newton, Hybrid Differential Evolution.

1. Introduction

The optimization of non-linear constraint problems is relevant to chemical engineering practice [1, 2]. Non-linearities are introduced by process equipment design relations, by equilibrium relations and by combined heat and mass balances. Search space for general optimization problem can be divided into several convex regions and each region has a local optima. Classical gradient-based optimization methods show better performance than evolutionary algorithm (EA) in finding the minimum of a convex region. However, they are not good at finding the global optima of the problem with multiple local optima.

Most of the traditional optimization algorithms based on gradient methods have the possibility of getting trapped at local optimum depending upon the degree of non-linearity and initial guess. Unfortunately, none of the traditional algorithms are guaranteed to find the global optimal solution, but population based algorithms are found to have a better global perspective than the traditional methods [3]. In the recent past, non-traditional search and optimization techniques (Evolutionary Computation) based on natural phenomenon such as Genetic Algorithms (GAs) [4] and Differential Evolution (DE) [5] have been developed to overcome these problems. Among their advantages are: (1) they do not require the objective function to be continuous and /or differentiable, (2) they do not require extensive problem formulation, (3) not sensitive to starting point, (4) usually do not get stuck into so called local optima, (5) simple & easy to use, (6) significantly faster & robust at numerical optimization, and (7) they are more likely to find out a function's true global optimum. These advantages enhance their application to various fields. They have been successfully applied in many engineering design problems [4, 6, 7, 8, 9, 10, 11, 12, 13] to name a few. Recently, Onwubolu and Babu [3] compiled new techniques and their applications to various disciplines of engineering and management.

This paper proposes an accelerated hybrid evolutionary method exploiting gradient search method such as quasi-Newton. This new method is hybrid of DE and quasi-Newton (QN) method (Hybrid Differential Evolution) and is good at finding the global optima. There are ten different strategies of DE but most widely used is DE/rand/1/bin [3, 14]. DE and Hybrid Differential Evolution (HDE) are used to solve the benchmark test functions (**HIM**: Himmelblau function; **GP$_2$**: Goldstein and Price function; **ES$_2$**: Easom; **H$_3$**: Hartmann function; **R$_2$ & R$_5$**:

Rosenbrock function; $\mathbf{Z_2}$ & $\mathbf{Z_5}$: Zakharov function, subscript 2, 3, and 5 indicate dimension of the problem) and further evaluated for water pumping system (WPS) problem [15]. Performance of HDE & DE is compared.

2. DE and HDE algorithms

DE is an improved version of simple GA. The controlling parameters of DE are: *D* – Dimension, NP – Population size, F – scaling factor, CR – crossover constant. The pseudo code for DE and HDE used in the present study is as follows:

- ***Input*** the value of *D*, NP, CR, F, gen_max, and lower & upper bounds of variables ($x_j^{(lo)}, x_j^{(hi)}$).
- ***Initialize*** all the vector population randomly in the given upper & lower bound.

  count = 0; (*count* is generation counter)

  for i ≤ NP and for $j \le D$:

  $$x_{i,j,count=0} = x_j^{lo} + rand_j[0,1]*\left(x_j^{(hi)} - x_j^{(lo)}\right)$$

  End for.

  Calculate *cost*.

  End *for*.
- Find out the vector with the *lowest cost* and assign bestit$_j$ = best$_j$ =$x_{j,\ count}$ & $f_{\min}= f\left(x_{i,count}\right)$.
- ***Repeat:***

  1. **Perform *mutation & crossover*.**
     For each target vector ($x_{j,count}$), select three distinct vectors randomly from the current population (primary array) other than target vector. Randomly select *r*1, *r*2, *r*3, $\in \{1,2,3......,\mathrm{NP}\}$,
     Except: $r1 \ne r2 \ne r3 \ne i$; $j \in \{1,2,....,D\}$, randomly selected for each *i*.

  for k ≤*D*:

  $$u_{j,count+1} = x_{r3,j,count} + \mathrm{F}*\left(x_{r1,j,count} - x_{r2,j,count}\right)$$

  if (rand$_j$ [0, 1]< CR or k = *D*)

  $$u_{j,count+1} = x_{i,j,count} \quad \text{Otherwise}$$

  If bounds are violated:

  $$u_{j,count+1} = x_j^{lo} + rand_j[0,1]*\left(x_j^{(hi)} - x_j^{(lo)}\right);$$

  $$if\ \left(u_{j,count+1} < x_j^{lo} \text{ or } u_{j,count+1} > x_j^{(hi)}\right)$$

  $$u_{j,count+1} = u_{j,count+1}; \quad \text{Otherwise}$$

  End for.

  2. **Selection.**
     best$_j$ = $u_{j,count+1}$ if $f\left(u_{j,count+1}\right) \le f_{\min}$

  Where **Selection** is done as:

  $$x_{i,count+1} = \begin{cases} u_{i,count+1} & \text{if } f\left(u_{i,count+1}\right) \le f\left(x_{i,count}\right) \\ x_{i,count} & \text{otherwise} \end{cases}$$

  End For.

  bestit$_j$ = best$_j$.

  Find maximum and minimum value of '*f*'.

  count = count+1;

  Call QN to improve bestit$_j$. /* for HDE */

  Till termination criteria do not meet.
- Print results.

3. Test Functions

A program has been developed for HDE based on the pseudo code given in previous section. The reliability and efficiency of DE and HDE are tested and compared for several multimodal functions, which were used in earlier literature [16]. The selected functions are briefly described below and details of the global minimum are summarized in Table 1.

Problem 1

The objective is to minimize the function (HIM).

$$f(x_1, x_2) = (x_1^2 + x_2 - 11)^2 + (x_1 + x_2^2 - 7)^2$$

Subject to $0 \le x_1,\ x_2 \le 6$.

Problem 2

The objective is to minimize the function (GP_2).

$$f(x) = [1+ (\mathrm{A1}*\mathrm{A2})][30 + (\mathrm{A3}*\mathrm{A4})]$$

Where $\mathrm{A1} = (x_1 + x_2 +1)^2$; $\mathrm{A3} = (2x_1 - 3x_2^2)$;

$\mathrm{A2} = (19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2)$;

$\mathrm{A4} = (18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2)$;

Subject to $-2 \le x_1, x_2 \le 2$.

Problem 3

The objective is to minimize the function (ES_2).

$$f(x) = -\cos(x_1)\cos(x_2)\exp\,[-((x_1 - \pi)^2 + (x_2 - \pi)^2]$$

Subject to $-100 \le x_1, x_2 \le 100$.

Problem 4

The objective is to minimize the function (H_3).

$$\text{Min } f(x) = -\sum_{i=1}^{4} c_i \exp\left[-\sum_{j=1}^{3} \mathrm{a_{ij}}\left(x_j - p_{ij}\right)^2\right]$$

Subject to $0 \le x_j \le 1$.

Problem 5, 6

The objective is to minimize the function (R_n).

$$f(x) = \sum_{i=1}^{D-1} [100(x_i^2 - x_{i+1})^2] + [x_i - 1]^2$$

Subject to $-5 \le x_i \le 10$; $i = 1, \ldots, D$.

Problem 7, 8

The objective is to minimize the function (Z_n).

$$f(x) = \left(\sum_{i=1}^{D} x_i^2\right) + \left(\sum_{i=1}^{D} 0.5ix_i\right)^2 + \left(\sum_{i=1}^{D} 0.5ix_i\right)^4$$

Subject to $-5 \le x_i \le 10$, $i = 1, \ldots, D$.

Table 1. Details of global minimum

Function	D	Global minimum	Remarks
GP_2	2	3 at $x = \{0, -1\}$	Four local minima
ES_2	2	-1 at $x = \{\pi, \pi\}$	Several local minima
H_3	3	-3.86278 at x={0.114624, 0.555649, 0.852547}	Four local minima
R_D	2, 5	0 at $x = \{1, \ldots, 1\}$	Several local minima
Z_D	2, 5	0 at $x = \{0, \ldots, 0\}$	Several local minima

4. Water Pumping System

A water pumping system [15] consists of two parallel pumps drawing water from a lower reservoir and delivering it to another which is 40 m higher, as shown in Fig. 1. In addition to overcoming the pressure difference due to elevation, the friction in the pipe is $7.2w^2$ kPa, where w is the combined flow rate in kg/s. The pressure-flow-rate characteristics of the pumps are:

Pump 1: $\Delta p\ (\text{kPa}) = 810 - 25w_1 - 3.75w_1^2$

Pump 2: $\Delta p\ (\text{kPa}) = 900 - 65w_2 - 30w_2^2$

Where w_1 and w_2 are the flow rates through pump 1 and pump 2, respectively.

The system can be represented by four simultaneous equations. The pressure difference due to elevation and friction is:

$$\Delta p = 7.2w^2 + \frac{(40\,\text{m})(1000\,\text{kg/m}^3)(9.807\,\text{m/s}^2)}{1000\,\text{Pa/kPa}} \quad (1)$$

Pump 1: $\Delta p = 810 - 25w_1 - 3.75w_1^2$ (2)

Pump 2: $\Delta p = 900 - 65w_2 - 30w_2^2$ (3)

Mass balance: $w = w_1 + w_2$ (4)

The objective here is to minimize Δp subject to the constraints (1), (2), (3), and (4). Hence,

$$Min.\ \Delta p = 7.2w^2 + \frac{(40\,\text{m})(1000\,\text{kg/m}^3)(9.807\,\text{m/s}^2)}{1000\,\text{Pa/kPa}}$$

Subject to:

$\Delta p = 810 - 25w_1 - 3.75w_1^2$.

$\Delta p = 900 - 65w_2 - 30w_2^2$.

$w = w_1 + w_2$.

Stoecker [15] used the method of successive substitution for solving this problem.

4.1 Problem Modification

Liebman et al. [17] modified the above problem. Ryoo & Sahinidis [18] solved this problem using Branch and Reduce algorithm. They used different strategies of Branch and Reduce algorithm. Babu & Angira [19] reformulated and used DE to solve it.

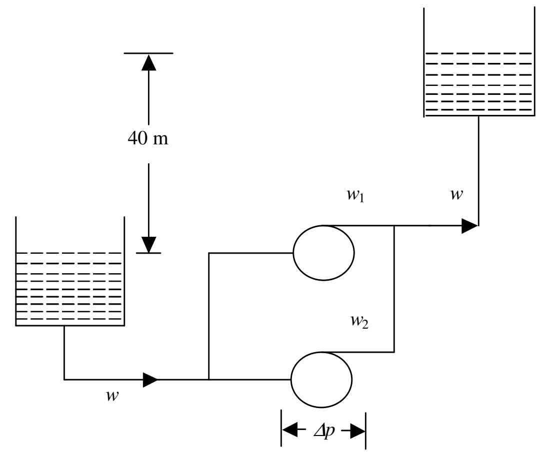

Fig 1. Water Pumping System

4.2 Problem Reformulation

In the present study modified problem as in [19] is taken as case study. Babu & Angira [19] used first equality as true objective function while the other two equalities are converted into inequalities and treated as inequality constraints. Since, QN is unconstraint optimization method. Therefore, in this paper the problem is reformulated as unconstraint optimization problem, incorporating constraints into objective function, as follows:

$$Min.\ f = x_3 = 150 + 0.5(x_1 + x_2)^2 + R(h^2 + g^2)$$

Where

$h = 6x_1^2 - 30x_1 - 250.0 + 150.0 + 0.5(x_1 + x_2)^2 = 0.0$;

$g = 12x_2^2 - 20x_2 - 300.0 + 150.0 + 0.5(x_1 + x_2)^2 = 0.0$;

and R ($=10^{10}$) is penalty on constraint violation.

$0 \le x \le (9.422, 5.903)$.

The global optimum obtained is: (x; f) = (6.293429, 3.821839; 201.159334).

5. Results and Discussion

Table 2 shows the results obtained by both DE and HDE. It may be noted that the global optimum is same as reported in literature (17, 18, 19]. It presents the comparison, in terms of CPU-time in seconds and Average Absolute Deviation (AAD). The termination criterion used is $|OF_{cal} - OF_{Anal}| < (10^{-4}\ OF_{Anal} + 10^{-6})$ where OF_{cal} is the objective function value at the best point found in each successful run and OF_{Anal} is the known global minimum. In this table, CPU-time and AAD represents the average value per execution over all the 10 executions and key parameters used are NP = 10*D, CR = 0.5, F = 0.8). The strategy used is DE/rand/1/bin in all the experiments. All the executions are done on Matlab-5.1 installed on PIII/500 MHz/128 MB RAM. Fig 2 shows the variation of Absolute Deviation (AD = $|OF_{cal} - OF_{Anal}|$) for DE and HDE for test function GP_2.

Table 2. Comparison of DE with HDE

Function	DE (CPU-time*)	HDE (CPU-time*)	DE (AAD)	HDE (AAD)
HIM	1.338	0.061	4.2E-07	7.3E-16
GP_2	1.358	0.077	1.4E-04	4.2E-14
ES_2	2.064	0.506	3.7E-05	1.7E-15
H_3	2.084	0.141	2.6E-04	2.1E-07
R_2	3.164	0.062	6.4E-07	1.4E-15
Z_2	1.312	0.058	5.5E-07	5.2E-17
R_5	78.800	2.08	0.027	1.4E-15
Z_5	41.013	0.285	6.9E-07	1.8E-15
WPS	2.992	0.088	9.2E-03	6.1E-08

* CPU-time on PC with Pentium PIII, 500 MHz/128 MB RAM/ 10 Gb HD with strategy DE/rand/1/bin

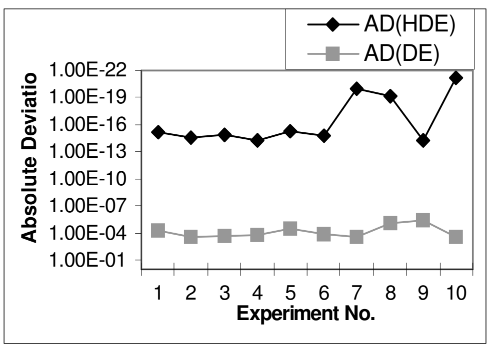

Fig 2. AD for DE and HDE (for GP_2)

In Table-2, AAD values for HDE are much less than AAD for DE. This indicates that HDE is able to locate global minimum more accurately. Also, CPU-time is much less than that for DE in each of test function and water pumping system problem. It is interesting to note that the performance of HDE is significantly better for higher dimension problems (R_5 and Z_5).

6. Conclusions

In this paper a hybrid evolutionary algorithm (HDE) is proposed. HDE is tested on several standard test functions and further evaluated for water pumping system. A comparison of HDE with DE is made. The HDE took less CPU-time as compared to DE. Also, the global optimum is located with high accuracy (AAD values very low) as compared to DE. The strategy used is DE/rand/1/bin in all the experiments. The performance of HDE is found to be better than the DE. For high dimension problems, the performance of HDE is significantly better. The results obtained by these methods (viz. DE, and HDE algorithm) are same and matches with that reported in literature.

References

[1] R. L. Salcedo, Industrial & Engineering Chemistry Research, 31 (1992), 262.
[2] C. A. Floudas, Nonlinear and mixed-integer optimization, Oxford University Press, New York, 1995.
[3] Godfrey C. Onwubolu and B. V. Babu, New Optimization techniques in Engineering, Springer Verlag, Germany, 2004.
[4] D. E. Goldberg, Genetic algorithms in search, optimization, and machine learning, Reading, MA: Addison-Wesley, 1989.
[5] K. Price and R. Storn, Dr. Dobb's Journal, 22 (1997) 18-24 & 78.
[6] I. P. Androulakis and V. Venkatasubramanian, Computers & Chemical Engineering, 15 (1991) 217-228.
[7] R. Storn, International Computer Science Institute, TR-95-018, 1995.
[8] J. P. Chiou and F. S. Wang, Computers & Chemical Engineering, 23 (1999), 1277-1291.
[9] B. V. Babu and K. K. N. Sastry, Computers & Chemical Engineering, 23 (1999) 327 – 339.
[10] J. C. Lu and F. S. Wang, Canadian Journal of Chemical Engineering, 79 (2001) 246-254.
[11] Rakesh Angira and B. V. Babu, Proc. Int. Symp. on Process Systems Engineering and Control (ISPSEC' 03) - For Productivity Enhancement through Design and Optimization, IITBombay, Mumbai, 3 – 4 January 2003, paper No. FMA2.
[12] B. V. Babu and Rakesh Angira, Proc. 4th Asia Pacific Conference on Simulated Evolution and Learning (SEAL-2002), Singapore, 18 – 22 November 2003, Vol. 2, 880-884.
[13] B. V. Babu and Rakesh Angira, Proc. Int. Symp. and 55th Annual session of IIChE (CHEMCON-2002), Hyderabad, 19 – 22 December 2002.
[14] B. V. Babu, Process Plant Simulation, Oxford University Press, New Delhi, 2004.
[15] W. F. Stoecker, Design of Thermal Systems, 3rd ed.,McGraw-Hill, Singapore, 1971, p. 117-121.
[16] Y. S. Teh and G. P. Rangaiah, Computers & Chemical Engineering, 27 (2003) 1665-1679.
[17] J. Leibman, L. Lasdon, L. Schrage and A. Waren, Modeling and optimization with GINO, The Scientific Press, Palo Alto, CA, 1986.
[18] H. S. Ryoo, and N. V. Sahinidis, Computers & Chemical Engineering, 19 (1995) 551-566.
[19] B. V. Babu and Rakesh Angira, Proc. 2nd Int. Conf. on Computational Intelligence, Robotics, and Autonomous Systems (CIRAS-2003), Singapore, 15 – 18 December 2003.