A Modified Harmony Search Threshold Accepting Hybrid Optimization Algorithm

Yeturu Maheshkumar[1,2] and Vadlamani Ravi[1,*]

[1] Institute for Development and Research in Banking Technology, Castle Hills Road #1, Masab Tank, Hyderabad – 500 057 (AP) India
[2] Department of Computer & Information Sciences, University of Hyderabad, Hyderabad – 500 046 (AP) India
ymaheshkumar527@gmail.com, rav_padma@yahoo.com

Abstract. Hybrid metaheuristics are the recent trend that caught the attention of several researchers which are more efficient than the metaheuristics in finding the global optimal solution in terms of speed and accuracy. This paper presents a novel optimization metaheuristic by hybridizing Modified Harmony Search (MHS) and Threshold Accepting (TA) algorithm. This methodology has the advantage that one metaheuristic is used to explore the entire search space to find the area near optima and then other metaheuristic is used to exploit the near optimal area to find the global optimal solution. In this approach Modified Harmony Search was employed to explore the search space whereas Threshold Accepting algorithm was used to exploit the search space to find the global optimum solution. Effectiveness of the proposed hybrid is tested on 22 benchmark problems. It is compared with the recently proposed MHS+MGDA hybrid. The results obtained demonstrate that the proposed methodology outperforms the MHS and MHS+MGDA in terms of accuracy and functional evaluations and can be an expeditious alternative to MHS and MHS+MGDA.

Keywords: Harmony Search, Threshold Accepting, Hybrid Metaheuristic, Unconstrained Optimization, Metaheuristic.

1 Introduction

Optimization is a process of attempting to find the best possible solution out of all possible solutions. When we search for an optimal solution in the given search space we will come across two types of optimal solutions: *local optimum* and *global optimum*. A *local optimal solution* is a point in a search space where all the neighboring solutions are better than the current solution and the *global optimum solution* is a point in the search space where all other points in the search space are worse than or equal to the current solution. Many search methodologies including exhaustive search, heuristics and Metaheuristics were proposed to find the optimal solutions.

* Corresponding author.

C. Sombattheera et al. (Eds.): MIWAI 2011, LNAI 7080, pp. 298–308, 2011.

Heuristics are generally referred to as trial-and-error methods. A heuristic is a method which seeks good solutions at a reasonable computation cost without being able to guarantee optimality, and possibly not feasibility and it is difficult to state how close to optimality the solution is [3]. *Metaheuristics* refers to set of concepts that can be used to define heuristic methods such that they can be applied to a wide set of different problems [4]. They can be seen as a general algorithmic framework which can be applied to solve different optimization problems with relatively few modifications to make. Metaheuristic can also be considered as a master strategy that guides and modifies other heuristics to produce solutions beyond those that are normally generated in a quest for local optimality. The heuristics guided by such a meta-strategy may be high level procedures or may embody nothing more than a description of available moves for transforming one solution into another, together with an associated evaluation rule. At present Metaheuristics is one of the important research areas in searching methodologies. In literature the term Metaheuristics refer to broad collection of relatively sophisticated heuristic methods like Particle Swarm Optimization (PSO), Harmony Search (HS) [5], Tabu Search (TS), Genetic Algorithm (GA), etc. This rapid focus on the area of metaheuristics led to development of new strategy named hybridization where more than one metaheuristics is employed such that the obtained hybrid will possess the advantage of both metaheuristics. Most commonly used hybrid strategy employs one or more metaheuristics to explore the search space and go near the optimal region where the probability of finding the global optimal solution is more than other regions and then other metaheuristic was used to exploit the near optimal region to find the global optimal solution. This strategy consists of employing both global search method and local search methods. There were many strategies adopted by researchers in developing hybrid metaheuristics.

The rest of the paper is organized as follows. Section 2 describes the literature survey of several metaheuristics. Section 3 describes the proposed strategy and how it is implemented. Section 4 presents the results and discusses the performance of the proposed hybrid by comparing with MHS and MHS+MGDA. Section 5 concludes the paper.

2 Literature Survey

The trend of hybridization of the existing Metaheuristics has started around 15 years ago. The hybrid optimization algorithms benefit from the advantages of the component metaheuristic algorithms. To start with, Ravi et al. [6] hybridized Non-Equilibrium Simulated Annealing (NESA) with a simplex like heuristic to develop a new algorithm called Improved Non-Equilibrium Simulated Annealing (INESA). This is one of the earliest hybrid algorithm proposed in the literature. In this paper, they improved Non-Equilibrium Simulated Annealing (NESA) by taking the solutions at regular intervals of the progress of NESA and then combining them with the best solutions obtained before the termination of NESA part of algorithm. At this stage they applied a simplex-like heuristic to obtain the global optimum. After that a hybrid metaheuristic in which 3 heuristics namely scatter search, GA and TS were employed in tandem was proposed [7]. In [7] the authors introduced the notion of memory to explore the solution space more extensively and also uses scatter search by combining the concepts of trajectory and clustering methods. The later stages of the algorithm combined the characteristics of TS and GA to test the status of new solutions and to direct the search towards global optimum. Later a hybrid Metaheuristic by hybridizing GA and NMSS was proposed [8]. In [8] the authors used GA to detect promising regions where we can find the optimal solution and uses NMSS for Intensification i.e., to locally search for global optimum in the promising region. A hybrid method [9] that hybridizes TA and DE was proposed in which TA is first applied to certain number of solutions of search space and the resultant set was passed to DE to move towards global optimal solution. After that a hybrid metaheuristic that hybridize DE by employing reflection property of the simplex method for fast convergence to global optima was developed [10]. Later, a hybrid metaheuristic using DE and Tabu Lists was developed for solving global optimization problems [11]. After that a hybrid metaheuristic, DETA was proposed [12]. In this model Differential Evolution (DE) is hybridized with Threshold Accepting (TA) that takes the advantage of efficient exploration of DE and exploitation of TA. They reported spectacular reduction in function evaluations when tested on test problems.

Many hybrid methodologies using harmony search have been proposed in the literature. To start with, a novel hybrid metaheuristic using harmony search and PSO was developed [13]. In this approach they induced harmony search inside the PSO such that before updating the position and velocity the solutions are passed as initial vector to harmony search such that a new solution generated every time is compared with worst solution and updated. Later a hybrid metaheuristic using harmony search and Sequential Quadratic Programming was developed [14]. In this approach after employing Harmony search, Sequential quadratic programming is employed on each solution to perform local search. The solution which provides better objective function value than other solutions is considered as the final accepted solution. Then two modified HS methods to deal with the uni-modal and multi-modal optimization problems have been proposed [15]. The first modified HS method is based on the fusion of the HS and Differential Evolution (DE) namely, HS-DE. The DE is employed here to optimize the members of the HS memory. The second modified HS method utilizes a novel HS memory management approach, and it targets at handling the multi-modal problems. Recently a heuristic particle swarm ant colony optimization (HPSACO) is presented for optimum design of trusses [16]. This algorithm is based on the particle swarm optimizer with passive congregation (PSOPC), ant colony optimization and harmony search scheme. HPSACO applies PSOPC for global optimization and the ant colony approach is used to update positions of particles to attain the feasible solution space. HPSACO handles the problem-specific constraints using a fly-back mechanism, and harmony search scheme deals with variable constraints. Later a new hybrid metaheuristics that includes the harmony search and MGDA [17] metaphors was proposed [1]. In this hybrid they first proposed a slight modification to harmony search and termed as modified harmony search (MHS). Then they employed MHS to explore the search space thoroughly and the best solution obtained is passed as initial solution to MGDA. The final solution obtained from MGDA is the global optimal solution.

3 MHSTA Hybrid

3.1 Overview of Harmony Search

Harmony search, proposed by Geem et al. [5] is one of the most recent meta-heuristic algorithms that found applications in science and engineering realms. The novelty in this algorithm lies in the fact that it is analogous to the improvisation technique of musicians. The algorithm in brief applies the idea of building an experience and then producing the best result that can be obtained from this experience. The analogy is such that a musical instrument represents a decision variable, its pitch range represents the value range, and solution vector is represented by the harmony and with thorough iterations (analogous to practice). The fitness value of the objective function is to be improved during iterations which is represented by the aesthetics. Given the random nature of the technique, it is highly likely that it will escape the local optima. An added benefit is that it performs very little operation on each prospective solution thereby substantially reducing program execution time. But, one of the major issues with Harmony Search is that for prolonged periods of time (in terms of iterations), during the execution of the program, its solution remains unchanged, especially during the final stages. As a result of which several unproductive iterations are performed with no genuine improvement to the solution. The modifications proposed by Choudhuri et al. [1] on Harmony Search are:

- -- The value of *hmcr* is kept dynamically increasing from 0 to 1 during the execution of the program.
- -- The HS algorithm is terminated when the difference between the best and the worst solution in the harmony memory is found to be less than some predefined constant.

The algorithm for Modified Harmony Search is explained in a step-by-step way as follows:

Here, we have two parameters: Harmony Memory Considering Rate (*hmcr),* which determines the percentage amount of the variable to be considered from memory and Pitch Adjusting Ratio (*par),which* is the probability with which the value is considered.

1. Generate a set of *hms* number of solutions randomly and initialize harmony memory with this set.
2. Create a new solution vector with components of the solutions selected from harmony memory with a probability of *hmcr* such that the components when selected from the harmony memory are chosen randomly from different solutions within the harmony memory. Note that as the number of iterations increase the value of *hmcr* is increased linearly.
3. Perform the pitch adjustment operation by altering the variables' value by *delta* with a probability *par* ('*delta*' value is used in case of discrete optimization problems).

4. If the objective function value of this vector is found to be better than the worst solution in the memory then replace the worst solution with this vector.
5. Repeat this procedure (Steps 2 through 4) till the difference between the best and worst solutions within the harmony memory becomes less than some predefined *diff1* (a small value), or maximum number of iterations (predefined) is reached, whichever happens earlier. This step terminates MHS and the best solution in the memory is the optimal solution by MHS. Note that the value of *diff1* is set to larger value than usual in order to facilitate early termination of MHS and begin with the next phase.

3.2 Threshold Accepting Algorithm

Threshold Accepting algorithm was proposed by Dueck and Sheur [2]. It is a point based search technique. It is a variation of Simulated Annealing (SA) algorithm while in SA, a new solution is accepted on a probabilistic basis, but in TA the new solution is accepted based on a deterministic criterion. In TA any new solution that is not much worse than the previous solution is accepted.

The pseudo code of TA is as follows:

Initialize the solution randomly and set global iteration counter itr=0, old=99999, thresh=2

```
f_i ← fitness value of initial solution
while itr<gitr   // gitr is the number of global iterations
 DO
    itr ← itr+1
    ii ← 0           // ii - inner iteration value
    while ii < limit or del1 > thresh
     DO
        ii ← ii+ 1
        Generate a candidate solution vector using the following equation
              candidate solution = old solution+(max-min)*(2*u-1)pindex
        f_j ← fitness value for the candidate solution
        del1 ← f_i – f_j
     END
 If del1 < thresh , set f_i = f_j
 If thresh < thrtol , set del2 = (new – old) / old
 Report current solution as the optimal one if abs (del2) < acc and exit if itr < gitr
 Else
     old ← new
     thresh = thresh * (1-eps)
END
```

TA is applied on a single solution. The algorithm runs for '*gitr*' number of global iterations and for every inner iteration, a candidate solution is generated. The fitness value is calculated for each candidate solution and the solutions that are not much worse than the previous one are selected for exploring. The algorithm terminates when the difference between objective function values of previous and present is very small as determined by the parameter acc which is set to 10^{-6} to obtain highly accurate solution. The parameter *thresh* is used to determine the acceptance of candidate solution and is generally set to 2 at the beginning and is gradually decreased in a geometric progression based on an epsilon value that is generally set to 0.01. *limit* is the number of inner iterations. *max*, *min* are the boundaries of the decision variables and *pindex* is generally an odd integer between 3 and 33 and is used to generate a value that is added to the old solution to generate a neighborhood solution.

3.3 MHSTA Algorithm

MHSTA is a new hybrid metaheuristic that employs both MHS and TA. The proposed method is a 2 phase process. The first phase of hybrid metaheuristic starts with employing MHS to explore the search space thoroughly such that finally a near optimal region is obtained where there is high probability to find the global optimal solution. In this phase MHS is not employed to its full extent by terminating the algorithm by choosing a small value for *diff1* (in our proposed hybrid the value of *diff1* is set to 0.0001). The best solution out of all the solutions i.e., the solution which provides better fitness value than other solutions is considered for the second phase. In second phase, TA is employed by considering the best solution from the phase1 as its initial solution. Here TA tries to exploit the near optimal region to find the global optimal solution. The algorithm of MHSTA is as follows:

Start
Consider the objective function to optimize and the search space of the objective function and initialize the harmony memory.
Phase 1
Employ MHS to find the near optimal region (the algorithm for MHS is explained previously)
The best solution obtained is considered for phase 2
Phase 2
Employ TA on the best solution obtained from phase1
The final solution obtained is considered as the global optimal solution.
End

The schematic view of the proposed approach was depicted in Figure 1. The ‘problem’ in the figure represents any optimization problem and ‘N’ represents the number of solutions to consider which is user defined. After employing MHS with N solutions the solution which gives the best optimal value is considered for the next phase which is represented as ‘B’ in the figure. Threshold Algorithm is invoked with ‘B’ as its initial solution. The optimal solution provided by the TA is the final solution obtained from the proposed hybrid model.

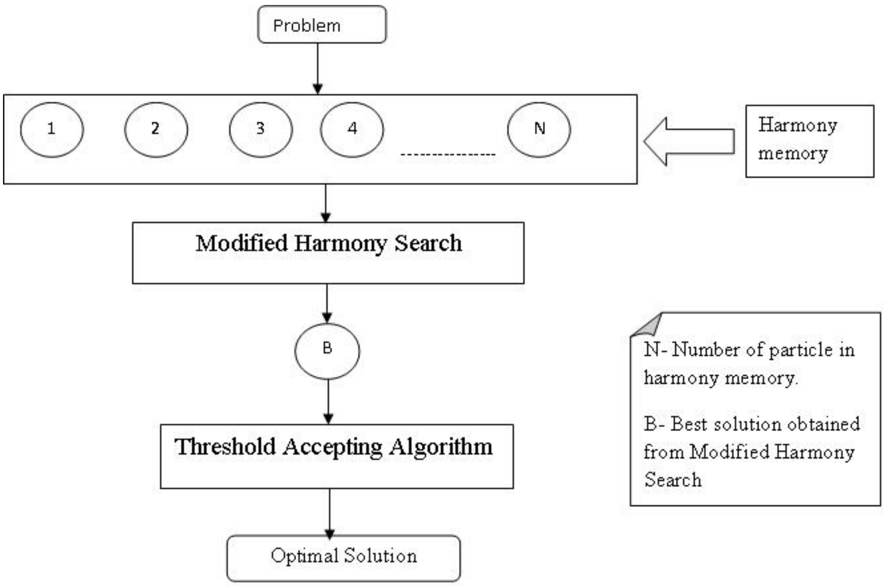

Fig. 1. Schematic view of MHSTA hybrid metaheuristic

4 Results and Discussions

The effectiveness of the proposed hybrid is analyzed by testing the proposed hybrid on 22 benchmark unconstrained optimization problems taken from [18]. The global optimal values obtained for each benchmark problem and the corresponding functional evaluations required are presented in Table 1. The results presented in Table 1 are the average results of 30 simulations with different seed values. In the Table 1 Function column contains the names of the benchmark problems and OPTIMAL VALUES represents average functional value obtained using the corresponding method (MHS, MHS+MGDA and MHSTA). The numbers 1-22 represents the serial number of the benchmark functions as per Table 1. FEval column contains the functional evaluations taken to provide the optimal value. The results clearly demonstrate that the proposed hybrid outperformed MHS and MHS+MGDA methods in terms of functional evaluations and optimal value. Out of 22 benchmark problems, in the case of 19 problems, the proposed hybrid produced better optimal value with less functional evaluations and in case of 3 problems the proposed hybrid yielded better optimal value than that of MHS and MHS+MGDA but after consuming little more functional evaluations. The proposed hybrid yielded better results on higher dimensional problems (sphere – 50 and 100 dimensions) with respect to functional evaluations and objective function values as well. Fig. 2 depicts the graphical comparison of functional evaluations consumed by MHS, MHS+MGDA, MHSTA hybrid Metaheuristics. The dark line represents MHS, the rounded dot line represents MHS+MGDA and the vertical dotted line represents MHSTA. Fig. 2 clearly demonstrates the supremacy of MHSTA over other Metaheuristics by consuming very less functional evaluations. The results clearly demonstrate the supremacy of MHSTA over other hybrid methods.

Table 1. Average optimal solutions and functional evaluations

		MHS		MHS+MGDA		MHSTA	
SNo	**FUNCTION**	**OPTIMAL VALUES**	**FEval**	**OPTIMAL VALUES**	**FEval**	**OPTIMAL VALUES**	**FEval**
1	Aluffi Penttini [19]	-0.352032	4497	-0.352226	3015	-0.352343	1408
2	Becker [20]	0.000013	6582	0.000261	2201	0.000027	1456
3	Bohachevsky1 [21]	0.009631	6619	0.0116	4162	0.000001	1883
4	Bohachevsky2 [21]	0.011379	7165	0.006072	4032	0.000001	1883
5	Camelback3 [22]	0.000263	4261	0.000317	4081	0.000050	1574
6	Camelback6 [22,23]	-1.031211	4478	-1.03113	4243	-1.031431	1336
7	Dekker's [24]	-24756.5839	3444	-24772.369	2276	-24771.979	1513
8	Easom [23]	-0.966595	5543	-0.96656	4038	-0.998774	1653
9	Goldstein [22]	3.01732	3888	3.060314	1033	3.002347	1432
10	Hartman3 [22]	-3.862748	5840	-3.86278	3661	-3.862746	1771
11	Miele [25]	0	7717	0.000018	2853	0.000025	1770
12	Mod.Rosenbrock [20]	0.020783	7353	0.0267	4570	0.005817	2104
13	Periodic [20]	0.900265	7015	0.90078	4480	0.906822	1869
14	Powell [25]	0.142005	10039	0.0134726	6687	0.022234	4952
15	Salomon 10 d [26]	0.042011	1480	0.05187	1042	0	1324
16	Schaffer2 [23]	0.710002	4234	0.0172575	3678	0.355321	1602
17	Schaffer1 [23]	0.014096	4442	0.012822	3107	0.012209	2217
18	Schwefel (10 d) [27]	-2094.040	8755	-2094.914	8726	-4187.6411	10039
19	Sphere (5d) [28]	0.005837	8577	0.001032	4771	0.000133	4037
20	Zakharov [29]	0.01826	9468	0.0015	5888	0.000215	5478
21	Sphere (50d) [28]	0.56729	13678	2.71172882	14267	0.052370	11075
22	Sphere (100d) [28]	0.58902	14478	3.11667	15167	0.37238	11075

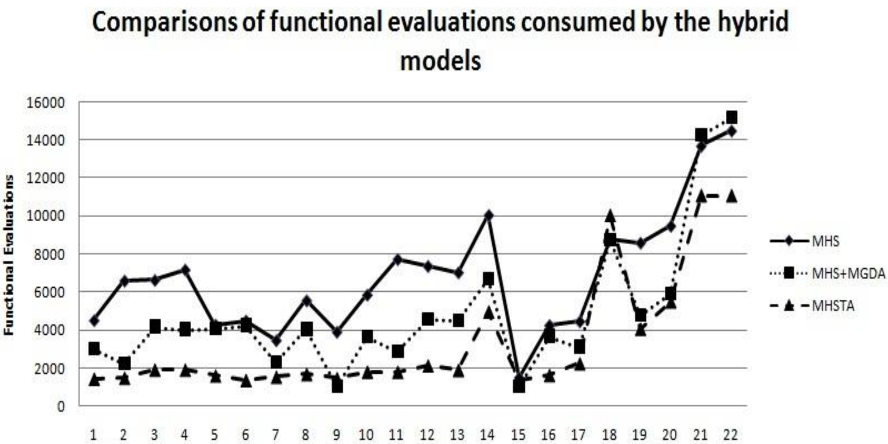

Fig. 2. Functional evaluations consumed by MHS, MHS+MGDA, MHSTA

5 Conclusions

MHSTA hybrid metaheuristic includes the advantages of efficient exploration of MHS combined with efficient exploitation of TA. The process of employing TA with best the solution produced by MHS helps to provide more accurate objective function value. The results demonstrate that the proposed hybrid metaheuristic is more expeditious than that of the exiting hybrid methodologies MHS, MHS+MGDA. The proposed hybrid produced better optimal value with almost 50% reduction in the consumption of functional evaluations which depicts its supremacy in terms of accuracy and speed.

References

1. Choudhuri, R., Ravi, V., Mahesh Kumar, Y.: A Hybrid Harmony Search and Modified Great Deluge Algorithm for Unconstrained Optimization. Int. Jo. of Comp. Intelligence Research 6(4), 755–761 (2010)
2. Dueck, G., Scheur, T.: Threshold Accepting: A General Purpose Optimization Algorithm appearing Superior to Simulated Annealing. Jo. of Comp. Physics 90, 161–175 (1990)
3. Edmund, K.B., Graham, K.: Search Methodologies: Introductory Tutorials in Optimization and Decission Support Techniques. Springer, Heidelberg (2005)
4. Glover, F.: Future Paths for Integer Programming and Links to Artificial Intelligence. Computers and Op. Research 13(5), 533–549 (1986)
5. Geem, Z., Kim, J., Loganathan, G.: A new heuristic optimization algorithm: harmony search. Simulation 76, 60–68 (2001)
6. Ravi, V., Murthy, B.S.N., Reddy, P.J.: Non-equilibrium simulated annealing-algorithm applied to reliability optimization of complex systems. IEEE Trans. on Reliability 46, 233–239 (1997)

7. Trafalis, T.B., Kasap, S.: A novel metaheuristics approach for continuous global optimization. Jo. of Global Optimization 23, 171–190 (2002)
8. Chelouah, R., Siarry, P.: Genetic and Nelder-Mead algorithms hybridized for a more accurate global optimization of continuous multi-minima functions. European Jo. of Op. Research 148, 335–348 (2003)
9. Schimdt, H., Thierauf, G.: A Combined Heuristic Optimization Technique. Advance in Engineering Software 36(1), 11–19 (2005)
10. Bhat, T.R., Venkataramani, D., Ravi, V., Murty, C.V.S.: Improved differential evolution method for efficient parameter estimation in biofilter modeling. Biochemical Eng. Jo. 28, 167–176 (2006)
11. Srinivas, M., Rangaiah, G.: Differential Evolution with Tabu list for Global Optimization and its Application to Phase Equilibrium and Parameter Estimation. Problems Ind. Engg. Chem. Res. 46, 3410–3421 (2007)
12. Chauhan, N., Ravi, V.: Differential Evolution and Threshold Accepting Hybrid Algorithm for Unconstrained Optimization. Int. Jo. of Bio-Inspired Computation 2, 169–182 (2010)
13. Li, H., Li, L.: A novel hybrid particle swarm optimization algorithm combined with harmony search for higher dimensional optimization problems. In: Int. Conference on Intelligent Pervasive Computing, Jeju Island, Korea (2007)
14. Fesanghary, M., Mahdavi, M., Joldan, M.M., Alizadeh, Y.: Hybridizing harmony search algorithm with sequential programming for engineering optimization problems. Comp. Methods Appl. Mech. Eng. 197, 3080–3091 (2008)
15. Gao, X.Z., Wang, X., Ovaska, J.: Uni-Modal and Multi Modal optimization using modified harmony search methods. IJICIC 5(10(A)), 2985–2996 (2009)
16. Kaveh, A., Talatahari, S.: PSO, ant colony strategy and harmony search scheme hybridized for optimization of truss structures. Computers and Structures 87, 267–283 (2009)
17. Ravi, V.: Optimization of Complex System Reliability by a Modified Great Deluge Algorithm. Asia-Pacific Jo. of Op. Research 21(4), 487–497 (2004)
18. Ali, M.M., Charoenchai, K., Zelda, B.Z.: A Numerical Evaluation of Several Stochastic Algorithms on Selected Continuous Global Optimization Test Problems. Jo. of Global Optimization 31, 635–672 (2005)
19. Aluffi-Pentini, F., Parisi, V., Zirilli, F.: Global optimization and stochastic differential equations. Jo. of Op. Theory and Applications 47, 1–16 (1985)
20. Price, W.L.: Global Optimization by Controlled Random Search. Computer Jo. 20, 367–370 (1977)
21. Bohachevsky, M.E., Johnson, M.E., Stein, M.L.: Generalized simulated annealing for function optimization. Techno Metrics 28, 209–217 (1986)
22. Dixon, L., Szego, G.: Towards Global Optimization 2. North Holland, New York (1978)
23. Michalewicz, Z.: Genetic Algorithms + Data Structures = Evolution Programs. Springer, Heidelberg (1996)
24. Dekkers, A., Aarts, E.: Global optimization and simulated annealing. Mathematical Programming 50, 367–393 (1991)
25. Wolfe, M.A.: Numerical Methods for Unconstrained Optimization. Van Nostrand Reinhold Company, New York (1978)
26. Salomon, R.: Reevaluating Genetic Algorithms Performance under Co-ordinate Rotation of Benchmark Functions. Bio. Systems 39(3), 263–278 (1995)
27. Muhlenbein, H., Schomisch, S., Born, J.: The parallel genetic algorithm as function optimizer. In: Belew, R., Booker, L. (eds.) Proceedings of the Fourth Int. Conference on Genetic Algorithms, pp. 271–278. Morgan Kaufmann (1991)

28. Sphere problem; global and local optima,
http://www.optima.amp.i.kyoto.ac.jp/member/student/hedar/Hedar_files/TestGO_files/Page113.html (cited on November 20, 2010)
29. Zakharov Problem Global and local optima,
http://www.optima.amp.i.kyotoc.jp/member/student/hedar/Hedar_files/TestGO_files/Page3088.htm (cited on November 20, 2010)