Chapter 5

Design of Chemo-GA for Constrained Optimization and its Application

In the earlier chapters the proposed CGA is applied in unconstrained optimization problem. In this Chapter, the code is developed to solve constrained optimization problems. The newly developed algorithm is introduced in the name of Chemo-GA for constrained optimization (CGAC). The better performance of CGAC is realized over some recent techniques reported in the literature through a test bed of 15 benchmark functions. The algorithm is compared with LXPMC and HLXPMC. Here, LXPM represents a real coded GA that uses Laplace crossover (LX) and power mutation (PM) in each of its iteration. The hybridization of LXPM with Quadratic Approximation (QA) operator is called HLXPMC. More over it has also been compared with some popular algorithms over a set of 11 benchmarking functions for comparison of time and the number of function evaluations. CGAC has been compared with Teaching-learning-based optimization (TLBO), Multi membered Evolutionary Strategy (M-ES), Particle Evolutionary Swarm Optimization (PESO), Cultural Differential Evolution (CDE), Co-evolutionary Differential Evolution (CoDE) and Artificial Bee Colony (ABC). The better performance of CGAC is concluded in solving constrained optimization problems.

Further, 6 typical engineering problems are solved by CGAC and the numerical results are compared with, (i) Differential Evolution with Level Comparison (DELC), (ii) Differential Evolution with Dynamic Stochastic Selection (DEDS), (iii) Hybrid Evolutionary Algorithm and Adaptive constraint-handling technique (HEAA), (iv) Genetic Algorithm (GA), (v) Social Behavior inspired Optimization technique (SBO), (vi) Rank Selection based PSO (RSPSO), (vii) Co-evolutionary Differential Evolution (CDE), and (viii) Hybrid PSO with a feasibility based rule (HPSO) and (ix) Co-evolutionary PSO (CPSO) and many other algorithms. The computational result confirms the out perperformance of CGAC over others.

5.1 INTRODUCTION

Most of the optimization problems arise in the real world involve constraints and are non-linear. It is rather more difficult to solve a constrained optimization problem than an unconstrained one. Again, the difficulty of solving a constrained problem increases if it involves some equality constraints. Now-a-days, it is a challenge among the researchers to develop techniques for solving such non-linear constrained optimization problems especially when the complexity is high. In recent past, Evolutionary Algorithm (EAs) gained its popularity over the traditional methods for solving such problems (detailed below), due to its simple random concept and easy implementation. Hybrid mechanisms become more effective in finding the near optimal solution, if not the optimal one.

Montes and Coello (2004) reported an improved version of simple evolution strategy (SES) that uses an improved diversity mechanism where the infeasible solutions are allowed to remain in the population close to the feasible region. Takahama et al. (2005) presented the hybridization of $\in -$ constrained PSO and $\in -$constrained GA. $\in -$ PSO is the combination of the $\in -$constrained method and PSO and $\in -$GA is the combination of the $\in -$ constrained method and GA. The $\in -$ constrained method is used to convert the constrained optimization problem into unconstrained one and the algorithm has been tested on some well known benchmark problems. Wang et al. (2009) proposed a hybrid EA and an adaptive constraint handling technique. The hybrid EA incorporates simplex crossover and two mutation operators (diversity mutation & improved breeder GA (BGA)) to generate the offspring population. Araujo et al. (2009) devised quadratic and linear approximation methodology as a local search operator decoupled with GA for constrained optimization. Elsayad et al. (2011) designed an improved Genetic Algorithm (GA) with a new multi-parent crossover and local search technique. The algorithm uses a diversity operator instead of mutation and it has been tested over 13 benchmark functions. Deb and Srivastava (2012) proposed a novel GA based on Augmented Lagrangian method where the critical parameters are updated in an adaptive manner based on population statistics. Lin (2013) introduced a novel hybridization of GA with rough set theory, called the Rough penalty Genetic algorithm. Based on the constraint violation, the infeasible solutions are subjected to rough penalties and the algorithm is evaluated on 11 test problems. Recently, Deep and Das (2013) proposed LXPMC for constrained optimization. It is the real coded GA that uses Laplace crossover and Power mutation. The hybridization of Quadratic Approximation (QA) with LXPMC is later named as HLXPMC.

In solving optimization problems, GA-BFO hybridization is one of the popular hybrid algorithms. However, it is observed that the mechanism of few of the operators seem to be repeated in the process of hybridization. But the major operator in BFO is its chemo tactic step. Therefore, in order to save the computational time and to reduce the complexity, only the chemo tactic step of BFO has been picked up to be hybridized with GA (namely CGA). In our earlier Chapter (2&3), the efficiency of CGA algorithm has been verified for solving unconstrained optimization problems. However, realizing the necessity of robust constrained handling algorithm, an attempt has been made in this Chapter.

A novel hybrid GA named Chemo-inspired GA for handling constrained optimization problem (CGAC) is proposed. In short it is named as Chemo-GA for Constrained Optimization (CGAC). Here the working priniciple of CGA is follwed. In evaluating the fitness function value, initially the consrained problem is converted into the unconstrained one by using the Penalty function approach. Bracket operator penalty has been used to construct the penalty function.

The rest of this chapter is organized as follows. Section 5.2 contains the existing constrained handling mechanisms. The proposed CGAC method is explained in Section 5.3. The lists of benchmark functions, experimental set up and result discussions of benchmark functions are given in Section 5.4. Section 5.5 describes six constrained engineering problems and the corresponding result analysis for them. The final conclusion of this Chapter is drawn in Section 5.6.

5.2 HANDLING CONSTRAINTS

5.2.1 Literature Review

It has been realized that Evolutionary Algorithms (EAs) are effective for unconstrained optimization problems, but existence of constraints may greatly affect the performances of EAs. Thus, it is necessary to use constraint handling techniques to guide the search towards the feasible space and then to optima. Researchers attempted to handle constraints in many possible ways. Schoenauer and Xanthakis (1993) started with a random population of individuals and optimize the objective function rejecting the infeasible individuals. Powell and Skolnick (1993) inserted a heuristic rule in order to distinguish between feasible and infeasible individuals. They mapped the feasible solutions into $(-\infty, 1)$ and infeasible solution into the interval $(1, \infty)$. Homaifer et al. (1994) proposed a constraint handling technique where for each constraint, several levels of violation are created and for each level of violation and for each constraint, a penalty coefficient is assumed. Joines and Houck (1994) assumed dynamic penalties where the individuals are evaluated based on the generation number. Authors claimed that it requires only a few parameters for execution. Michalwicz and Attia (1994) proposed a method that maintains feasibility of all linear constraints where a feasible solution is converted into another feasible solution using a set of closed operators. Deb and Agarwal (1999) proposed a Niche penalty approach for constraints handling in GAs which does not require any penalty parameter. Coello and Montes (2002) proposed a constraint handling technique where dominance based selection scheme is used and it does not require fine tuning of the parameter to maintain the diversity. Aktar et al. (2002) introduced a socio-behavioral simulation approach where the constraints are handled by ranking the solutions based on non-dominance checking insidetheir corresponding society. Chootinan and Chen (2006) suggested for a systematic repair of infeasible solutions by using the gradient information derived from the constraint set. The proposed repair procedure is embedded into a simple GA as a special operator. Rahman et al. (2013) proposed a modified dynamic penalty function and used in Binary-real coded Genetic Algorithm. In spite of several constraint-handling methods available in the literature, use of penalty function (Deb, 1998) is the most popular approach. Deep and Das (2013) recently used the dynamic penalty parameter proposed by Crossley and Williams (1997) and bracket operator penalty function to handle constrained optimization problem. The proposed Chemo-GA for Constrained Optimization (CGAC) makes use of this recent mechanism used in (Deep and Das, 2013). A discussion is briefed in the next section.

5.2.2 Strategies for handling constraints

The following strategies (shown in steps) are being adopted for handling constrained optimization problems.

Step 1: Evaluation of Fitness

The fitness of an individual in the population is calculated by

$$Fit\_Fun = \begin{cases} F\left(x^{(t)}, R^{(t)}\right), & for\ Minimization\ fuction \\ 1/\left(1+F\left(x^{(t)}, R^{(t)}\right)\right), & for\ Maximization\ function \end{cases} \tag{5.1}$$

Where $F\left(x^{(t)}, R^{(t)}\right)$ called the penalty function defined in equation (5.2).

$$F\left(x^{(t)}, R^{(t)}\right) = f\left(x^{(t)}\right) + \Omega\left(R^{(t)}, g^{(t)}, h^{(t)}\right) \tag{5.2}$$

Here f stands for the objective function, t for generation number, g for inequality constraints, h for equality constraints and Ω for penalty term called **Bracket operator penalty**, which is given by

$$\Omega = R\langle(g(x))\rangle^2, \qquad \text{where} \tag{5.3}$$

$$\langle (g(x)) \rangle = \begin{cases} 0, if\,(g(x)) \geq 0 \\ (g(x)), if\,(g(x)) < 0 \end{cases} \tag{5.4}$$

and R is the penalty parameter as defined in (5.5). It is worth here to note that each equality constraint in (5.2) is being equivalently expressed in two inequality terms with $\leq$ and $\geq$.

Step 2: Penalty parameter

The penalty parameter $R^{(t)}$ for a particular generation't' is proposed as follows.

$$R^{(t)} = 1.2^{\left\lfloor \frac{t}{22} \right\rfloor} \tag{5.5}$$

Where $\lfloor \cdot \rfloor$ indicates the floor function.

For the constraint handling technique, an exterior penalty term has been used, where the bracket operator assigns a positive value to the infeasible points. The algorithm starts with a small initial value of $R^{(t)}$ and it increases step-wise with t. This way, the quality of the solution improves gradually by forcing the infeasible points towards the feasibility and finally approaches to a near optimal solution. The value of R in (5.5) is so fixed because it is suitable for the entire constrained problem considered, which is experimentally verified.

Step 3: Selection of Individuals

The CGAC uses the tournament selection according to Deb's rule (Deb, 2000), where two candidates in the population are compared by applying the following criteria.

i. Feasible solution is preferred over an infeasible solution.
ii. Between two feasible solutions, the one having better objective function value is preferred.
iii. Between two infeasible solutions, the one having smaller constraint violation is preferred.

The idea of tournament selection is illustrated in Fig.5.1, through three different instances.

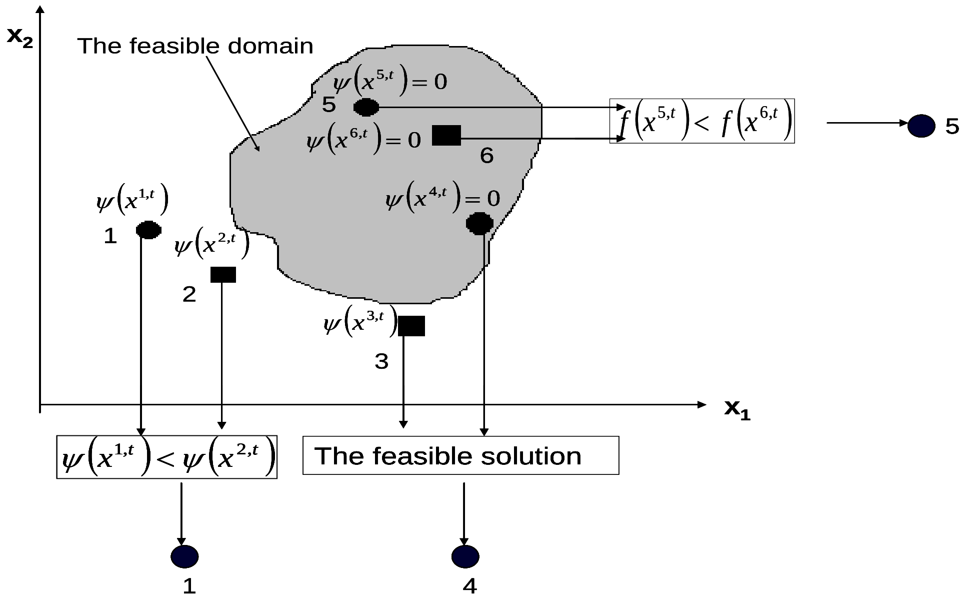

Fig.5.1. The idea of the constraint tournament method

5.3 PROPOSED CHEMO-GA FOR CONSTRAINED OPTIMIZATION (CGAC)

The detailed mechanisms of both GA and BFO are presented in the previous Chapter (2&3). Collectively, the objectives of both the mechanism are the same as to explore the search space and coming up with a near optimal solution. Each of them has some pitfalls in their inherent mechanism. In previous Chapters(2&3), the design of CGA has been discussed, where the Chemo tactic step of BFO has been hybridized with GA and the efficiency of CGA over the GA and GA-BF (a hybridization of GA and BFO) and Quadratic Approximation hybridized Genetic Algorithm(QGA) is well verified for solving unconstrained optimization problems. In this Chapter, CGA is made capable of handling constraints. The algorithm thus proposed in the name of '*Chemo-GA for constrained optimization (CGAC)*'. It has been tested over a set of benchmark problems and applied to solve some engineering design problems also.

The detailed mechanism of CGAC is as follows. Clearly, CGAC has 5 major steps viz. Selection, Crossover, Mutation, Elitism and Chemo taxis. The initial parameters considerd for CGAC is same as CGA which is already explained and can be reffered from (Subsection 2.4.1(Section 2.4), Chapter 2)

Pseudo code for CGAC:

The stepwise proposed algorithm CGAC is given below.

[Step 1] **Begin**: Initialize the population. Calculate the fitness function by (5.1).

While *(termination criterion is not satisfied)* ***do***

[Step 2] Apply binary tournament selection

[Step 3] Apply crossover operator

[Step 4] Apply mutation operator

[Step 5] Use Elitism Operator

[Step 6] Chemo taxis loop:

[Sub step a] **Re initialization of the search space:**

For $j = 1 : N_c$

For $i = 1 : S$

Compute the fitness function $fit(X^i(j))$.

Update F_{best} and X_{best}

Squeeze the search space as Eq. 2.5 and Eq. 2.6 (Chapter 2).

[Sub step b] **Tumble and Move:**

Calculate the new position in $j+1^{th}$ Chemo tactic step as per Eq. (5.6)

$$X^i(j+1) = X^i(j) + C_{step} \frac{\Delta(i)}{\sqrt{\Delta(i)\Delta^T(i)}} \tag{5.6}$$

Where $\Delta(i) \in R^n$ with each element $\Delta_m(i),\ m = 1,2\ldots,n,$

a random number on [-1, 1].

This results in a step size C_{step} in the direction of tumble for bacterium i

Calculate the C_{step} according to adaptive or modified chemo tactic step size defined as per (Eq. (2.4) or Eq. (2.7), Chapter 2)

[Sub step c] **Swim Step:**

Let $m = 0$ (counter for swim length).

While $m < N_S$ (If have not climbed down too long) **do**

Let $m = m + 1$

If $fit\left(X^i(j+1)\right) < F_{best}$

Update F_{best} and X_{best}

Evaluate the modification of new position in $j+1^{th}$

Chemo tactic step as per Eq. (5.7) as follows.

$$X^i(j+1) = X^i(j+1) + C_{step}\frac{\Delta(i)}{\sqrt{\Delta(i)\Delta^T(i)}} \tag{5.7}$$

Then compute the $fit\left(X^i(j+1)\right)$ again.

Else keep the bacterium stay still.

$X^i(j+1) = X^i(j)$.

Calculate the fitness value of $X^i(j+1)$,

Let $\mathrm{m} = \mathrm{N}_S$.

End **While**

End for$(i = 1:S)$

End for$(j = 1:N_c)$

End **While**

End **begin**

5.4 LIST OF BENCHMARK FUNCTIONS

A set of 15 well known benchmark functions (listed below) have been picked up for numerical simulation to testify the performances of CGAC (Deep and Das, 2013).

Problem 1

$$\underset{x}{Max}\ f(x) = 3x_1 + x_2 + 2x_3 + x_4 - x_5$$

Subject to

$$25x_1 - 40x_2 + 16x_3 + 21x_4 + x_5 \le 300$$
$$x_1 + 20x_2 - 50x_3 + x_4 - x_5 \le 200$$
$$60x_1 + x_2 - x_3 + 2x_4 + x_5 \le 600$$
$$-7x_1 + 4x_2 + 15x_3 - x_4 + 65x_5 \le 700$$
$$1 \le x_1 \le 4; 80 \le x_2 88; 30 \le x_3 \le 35$$
$$145 \le x_4 \le 150; 0 \le x_5 \le 2$$

solution: $x = (4, 88, 35, 150, 0)^T$, $f^*(x) = 320$

Problem 2

$$\underset{x}{Max}\ f(x) = 25(x_1 - 2)^2 + (x_2 - 2)^2 + (x_3 - 1)^2 + (x_4 - 4)^2 + (x_5 - 1)^2 + (x_6 - 4)^2$$

Subject to

$$x_1 + x_2 - 2 \ge 0, \quad 6 - x_1 + x_2 \ge 0$$
$$2 + x_1 - x_2 \ge 0, \quad 2 - x_1 + 3x_2 \ge 0$$
$$(x_3 - 3)^2 + x_4 - 4 \ge 0$$
$$(x_5 - 3)^2 + x_6 - 4 \ge 0$$
$$0 \le x_1 \le 5 \quad ; 0 \le x_2 \le 1$$
$$1 \le x_3 \le 5; \quad 0 \le x_4 \le 6$$
$$0 \le x_5 \le 5; \quad 0 \le x_6 \le 10$$

Solution: $x = (5, 1, 5, 0, 5)^T$, $f^*(x) = 310$

Problem 3

$$\underset{x}{Min}\ f(x) = (x_1^2 + x_2 - 11)^2 + (x_1 + x_2^2 - 7)^2$$

Subject to

$$4.84-(x_1-0.05)^2-(x_2-2.5)^2 \geq 0$$
$$x_1^2+(x_2-2.5)^2-4.84 \geq 0$$
$$0 \leq x_1 \leq 6 \quad ; \quad 0 \leq x_2 \leq 6$$

Solution: $x=(2.246826, 2.381865)^T$, $f^*(x)=13.59085$

Problem 4

$$\underset{x}{Min}\ f(x)=(x_1-10)^2+5(x_2-12)^2+x_3^4+3(x_4-11)^2$$
$$+10x_5^6+7x_6^2+x_7^4-4x_6x_7-10x_6-8x_7$$

Subject to

$$282-7x_1-3x_2-10x_3^2-x_4+x_5 \geq 0$$
$$127-2x_1^2-3x_2^4-x_3-4x_4^2-5x_5 \geq 0$$
$$196-23x_1-x_2^2-6x_6^2+8x_7 \geq 0$$
$$-4x_1^2-x_2^2+3x_1x_2-2x_3^2-5x_6+11x_7 \geq 0$$
$$-10 \leq x_i \leq 10, i=1,2,\ldots,7$$

Solution:

$$x=\begin{pmatrix}2.330499, 1.951372, -0.4775414, \\ 4.365726, -0.6244870, 1.038131, 1.594227\end{pmatrix}^T,$$
$$f^*(x)=680.6300573$$

Problem 5

$$\underset{x}{Min}\ f(x)=x_1^{0.6}+x_2^{0.6}+x_3^{0.4}+2x_4+5x_5-4x_3-x_6$$

Subject to

$$x_2 - 3x_1 - 3x_4 = 0, \quad x_3 - 2x_2 - 3x_5 = 0$$

$$4x_4 - x_6 = 0, \quad x_1 + 2x_4 - 4 \le 0$$

$$x_2 + x_5 - 4 \le 0$$

$$x_3 + x_6 - 6 \le 0$$
$$0 \le x_1 \le 3; 0 \le x_2 \le 4$$
$$1 \le x_3 \le 4; 0 \le x_4 \le 2$$
$$0 \le x_5 \le 2; 0 \le x_6 \le 6$$

Solution: $x = (0.67, 2, 4, 0, 0, 0)^T$, $f^*(x) = -11.96$

Problem 6

$$\underset{x}{Min}\ f(x) = -x_1 - x_2$$

Subject to

$$x_2 \le 2 + 2x_1^4 - 8x_1^3 + 8x_1^2$$
$$x_2 \le 4x_1^4 - 32x_1^3 + 88x_1^2 - 96x_1 + 36$$

$$0 \le x_1 \le 3; 0 \le x_2 \le 4$$

solution: $x = (2.3295, 3.1783)^T$, $f^*(x) =$ -5.5078

Problem 7

$$\underset{x}{Min}\ f(x) = -12x_1 - 7x_2 + x_2^2$$

Subject to

$$-2x_1^4 - x_2 + 2 = 0$$

$$0 \le x_1 \le 2; 0 \le x_2 \le 3$$

Solution: $x = (0.7175, 1.47)^T$, $f^*(x) = -16.7391$

Problem 8

$$\underset{x}{Min}\ f(x) = 3x_1 + 0.000001x_1^3 + 2x_2 + (0.000002/3)x_2^3$$

Subject to

$$x_4 - x_3 + 0.55 \geq 0$$
$$x_3 - x_4 + 0.55 \geq 0$$
$$1000\sin(-x_3 - 0.25) + 1000\sin(-x_4 - 0.25) + 894.8 - x_1 = 0$$
$$1000\sin(x_3 - 0.25) + 1000\sin(x_3 - x_4 - 0.25) + 894.8 - x_2 = 0$$
$$1000\sin(x_4 - 0.25) + 1000\sin(x_4 - x_3 - 0.25) + 1294.8 = 0$$
$$0 \leq x_i \leq 1200, \quad i = 1,2; \ -0.55 \leq x_i \leq 0.55, \quad i = 3,4$$

Solution:

$$x = (679.9453, 1,026.067, 0.1188764, -0.3962336)^T, \ f^*(x) = 5126.4981$$

Problem 9, 10, 11

$$\underset{x}{Min}\ f(x) = \begin{Bmatrix} f_1 = x_2 + 10^{-5}(x_2 - x_1)^2 - 1, & if\ 0 \leq x_1 \leq 2 \\ f_2 = \dfrac{1}{27\sqrt{3}}\left((x_1 - 3)^2 - 9\right)\ x_2^3, if\ 2 \leq x_2 \leq 4 \\ f_3 = \dfrac{1}{3}(x_1 - 2)^3 + x_2 - \dfrac{11}{3}, & if\ 4 \leq x_3 \leq 6 \end{Bmatrix}$$

Subject to

$$\frac{x_1}{\sqrt{3}} - x_2 \geq 0, -x_1 - \sqrt{3}x_2 + 6 \geq 0, 0 \leq x_2 \leq 5,$$

Solution: $x = (0,0)^T, \left(3, \sqrt{3}\right)^T, (4,0)^T, \ f^*(x) = -1$

Problem 12

$$\underset{x}{Min}\ f(x) = 100\left(x_2 - x_1^2\right)^2 + (1 - x_1)^2$$

Subject to

$$-x_1 - x_2^2 \le 0$$
$$-x_1^2 - x_2 \le 0$$
$$-0.5 \le x_1 \le 0.5$$
$$0 \le x_2 \le 1$$

Solution: $x = (0.5, 0.25)^T$, $f^*(x) = 0.25$

Problem 13

$$\underset{x}{Min}\ f(x) = 0.01 \times (x_1^2 + x_2^2)$$

Subject to

$$-x_1 x_2 + 25 \le 0$$
$$-x_1^2 - x_2^2 + 25 \le 0$$
$$2 \le x_1 \le 50;\ 0 \le x_2 \le 50$$

Solution: $x = \left(\sqrt{250}, \sqrt{250}\right)^T$, $f^*(x) = 0.5$

Problem 14

$$\underset{x}{Max}\ f(x) = \frac{\sin^3(2\pi x_1)\sin(2\pi x_2)}{x_1^3(x_1 + x_2)}$$

$$x_1^2 - x_2 + 1 \le 0$$
$$1 - x_1 + (x_2 - 4)^2 \le 0$$
$$0 \le x_1 \le 10; 0 \le x_2 \le 10$$

solution: $x = (1.2279713, 4.2453733)^T$,
$f^*(x) = 0.095825$

Problem 15

$$\underset{x}{Min}\ f(x) = (x_1 - 10)^3 + (x_2 - 20)^3$$

Subject to

$$(x_1 - 5)^2 + (x_2 - 5) - 100 \geq 0$$
$$-(x_1 - 6)^2 - (x_2 - 5) + 82.82 \geq 0$$
$$13 \leq x_1 \leq 100; \quad 0 \leq x_2 \leq 100$$

solution: $x = (14.095, 0.84296)^T$, $f^*(x) = -6961.81381$

5.4.1 Experimental Setup

At first, a set of 15 well known benchmark functions (listed in Section 5.4, above) have been picked up for numerical simulation to testify the performances of CGAC. The comparison is made with LXPMC and HLXPMC (Deep and Das, 2013). LXPMC is a real coded GA that uses Laplace crossover and Power Mutation and its hybridization with Quadratic Approximation (QA) Operator is named as HLXPMC. Here ‘C’ stands for it is being capable of handling constraints. Again 11 benchmarking functions have been considered for comparison of time and function evaluation with some popular and continuous algorithms (Tang et al., 2011; Rao et al., 2011). The time comparison has been done with Improved Genetic Algorithm based on a novel selection strategy (IGA), Pareto strength evolutionary algorithm (ZW), Stochastic ranking algorithm (RY) and Homomorphous mapping method (KM) (Tang et al., 2011). The function evaluations have been compared with Teaching-learning-based optimization (TLBO), Multi membered Evolutionary Strategy (M-ES), Particle Evolutionary Swarm Optimization (PESO), Cultural Differential Evolution (CDE), Co-evolutionary Differential Evolution (CoDE) and Artificial Bee Colony (ABC) (Rao et al., 2011).

Further, the CGAC is applied to solve some engineering design problems widely used in literature. The comparison of CGAC is done with Differential Evolution (DE) that uses level comparison (DELC) and other 5 existing recent approaches (Wang and Li, 2010).

The proposed CGAC is designed in C++ and the experiment is carried out on a P-IV, 2.8 GHz machine with 512 MB RAM under WINXP platform. After a series of hand tuning experiments, the recommended values of the parameters are as follows. Probability of crossover $(P_c = 0.9)$, probability of mutation $(P_m = 0.05)$, maximum step size $(C_{\max} = 0.1)$, minimum step size $(C_{\min} = 0.008)$, Bit length $(l = 20)$. For problem 5 and problem 7, the minimum and maximum step sizes are set to be 0.001 and 10^{-7} respectively. A total of 40 Chemo tactic steps are considered for each problem. However, for problem 5 chemo tactic steps is considered to be 90. The population size is fixed at 40. After completion of GA cycle only 4 individuals/bacteria are re-initialized within search space and allowed to go for the completion of chemo tactic loop. The chemo tactic loop is initiated after 10 generations for problem 5 & problem 7. But for the rest, the chemo tactic loop is initiated after 100 generations. The stopping criteria are either a maximum of 2000 generations is attained or no improvement is found in the best objective function value achieved so far in a consecutive 100 generations. A run is said to be success if the objective function value obtained by the algorithm is within 1% accuracy of the known optimal solution.

5.4.2. Result and Discussion

With the above experimental set up, at the outset each of 15 benchmark problems undergoes a total of 100 independent runs. The success rates are reported in Table 5.1. For the successful runs, the average function evaluations are reported in Table 5.2. The computational time is reported in Table 5.3. The mean function value and the standard deviation (SD) are presented in Table 5.4. In Tables 5.1-5.4, the best values are highlighted with bold faced letters. Moreover to achieve the best reported function value the time and function evaluation required for some popular algorithms (Tang et al., 2011; Rao et al., 2011) are reported in Table 5.5(time comparison) and Table 5.6(function value) for 11 benchmarking functions.

Looking at Table 5.1, CGAC nowhere achieves less success rate than LXPMC and HLXPMC (Deep and Das, 2013) in all 15 problems. From Table 5.4, it is also seen that CGAC can achieve the optimal solution in most of the cases (better in 8 cases and equal in 4 cases and worst in 3 cases). Of course CGAC needs more number of function evaluations and time (Table 5.2 and 5.3), but in return it leaves better objective function values. Again, the last part of Table 5.4 bears the S. D. of all the methods. It is clear that the S. D. for CGAC is better in 9 cases, equal in 5 cases and worst in one case. From Table 5.5 and Table 5.6 it can be concluded CGAC takes less time and function evaluation in comparison to Improved Genetic algorithm, Teaching–Learning based optimization and other popular algorithms to achieve the best reported result. Therefore CGAC is more accurate and more stable as compared with LXPMC and HLXPMC and other popular algorithms mentioned.

Table 5.1: Percentage of success for 100 runs

Pb. No.	**CGAC**	**HLXPMC**	**LXPMC**	**Pb. No.**	**CGAC**	**HLXPMC**	**LXPMC**
1	**100**	**100**	**100**	**9**	**100**	**100**	**100**
2	**100**	98	99	**10**	**100**	2	0
3	**100**	5	3	**11**	**100**	**100**	**100**
4	**100**	**100**	98	**12**	**100**	**100**	52
5	**13**	5	5	**13**	**100**	38	37
6	**100**	12	11	**14**	**100**	29	13
7	**46**	22	15	**15**	**100**	4	2
8	**58**	52	34	**--**	--	--	--

Table 5.2: Average no of function evaluations

Pb.No.	CGAC	HLXPMC	LXPMC	Pb. No.	CGAC	HLXPMC	LXPMC
1	**9671**	25,439	25,649	**9**	88704	**10,038**	10,055
2	72893	**30,472**	30,768	**10**	51572	**10,800**	*
3	438459	**10,204**	10,586	**11**	22645	**10,060**	10,190
4	708050	**49,209**	52,504	**12**	11981	10,295	**10,105**
5	62844	**31,428**	32,520	**13**	109494	10,344	**10,269**
6	708050	**10,273**	10,405	**14**	48432	**12,524**	13,766
7	115471	**10,153**	10,246	**15**	976010	**10,120**	10,560
8	**12,611**	31,259	30,829	**--**	--	--	--

Table 5.3: Average computational time in (sec)

Pb. No.	CGAC	HLXPMC	LXPMC	Pb. No.	CGAC	HLXPMC	LXPMC
1	**0.062**	0.5116	0.1203	**9**	1.49895	0.3574	**0.2845**
2	0.31543	0.7176	**0.2195**	**10**	**0.18466**	0.4060	*
3	1.87665	0.1904	**0.0620**	**11**	**0.07661**	0.3611	0.2900
4	8.23954	1.4472	**0.7026**	**12**	0.07628	0.2222	**0.0706**
5	0.5903	0.7908	**0.3188**	**13**	0.6991	0.2057	**0.0740**
6	3.723	0.2163	**0.0712**	**14**	0.58992	0.3571	**0.0961**
7	0.414717	0.1945	**0.0627**	**15**	5.77489	0.2110	**0.0748**
8	**0.1205**	0.6169	0.2983	**--**	--	--	--

Table 5.4: Mean function value and S.D

Pb. No.	Mean function values			S.D.		
	CGAC	HLXPMC	LXPMC	CGAC	HLXPMC	LXPMC
1	**-320.000**	**--320.000**	**-320.000**	**0.0000**	**0.0000**	**0.0000**
2	-310.000	**-313.000**	**-313.000**	**0.0000**	**0.0000**	**0.00000**
3	**13.5909**	13.641	13.644	**1.49e-05**	0.0451	0.0223
4	**680.732**	682.282	682.433	**0.088591**	0.8681	1.3023
5	**-12.5672**	-11.951	-11.976	0.453764	**0.0693**	0.0726
6	**-5.50801**	-5.475	-5.475	**8.36e-14**	0.0125	0.0125
7	-16.7069	**-16.745**	-16.742	**0.038897**	0.1072	0.0999
8	5136.99	**5134.106**	5139.564	**10.7796**	18.7828	16.0353
9	**-1.000**	**-1.000**	**-1.000**	**0.0000**	**0.0000**	**0.0000**
10	**-1.000**	-0.993	*	**0.0000**	0.0006	*
11	**-1.000**	**-1.000**	**-1.000**	**0.000**	**0.0000**	**0.0000**
12	**0.25**	**0.250**	**0.250**	**0.0000**	**0.0000**	0.0004
13	**0.5**	0.501	0.502	**1.81e-07**	0.0014	0.0015
14	**-0.09583**	-0.096	-0.096	**5.45e-17**	0.0003	0.0006
15	**-6974.1**	-6937.021	-6955.844	**3.95e-06**	19.2784	35.4933

Table 5.5: Time comparison (in Sec.) (Tang et al., 2011)

Methods	P1	P2	P3	P4	P5	P6
CGAC	**3.73284**	**25.23**	**31.1781**	**2.30697**	**3.86718**	47.3953
IGA	40.62	31.04	43.35	35.26	50.89	**45.34**
ZW	45.35	29.26	40.51	39.48	56.24	50.32
RY	60.78	40.65	50.63	41.62	64.61	57.36
KM	---	35.27	45.36	40.14	60.78	58.21

Table 5.6: Comparison of number of Function Evaluations (Rao et al., 2011)

Methods	**P1**	**P2**	**P3**	**P4**	**P5**
M-ES	2,40,000	2,40,000	240000	2,40,000	240000
PESO	3,50,000	3,50,000	350000	3,50,000	350000
CDE	1,00,100	1,00,100	100100	1,00,100	100100
CoDE	2,48,000	---	248000	---	248000
ABC	2,40,000	2,40,000	240000	2,40,000	240000
TLBO	25000	**1,00,000**	100000	1,00,000	50000
CGAC	**20280**	1,04,047	**97210**	**31629**	**42000**

5.5 APPLICATION ON ENGINEERING PROBLEMS

5.5.1 Engineering Problems (EPs)

The robustness of the proposed algorithm CGAC is also verified on 6 well-known typical engineering design problems (explained below). These problems contain both linear and non-linear constraints.

(EP1) Welded Beam Design (case 1) (Deb, 2000)

A welded beam is designed for the minimum cost subject to constraints on shear stress (τ), bending stress in the beam (σ), buckling load on the bar (P_C), end deflection of the beam (δ) and side constraints. This problem has 4 design variables: thickness of weld $h(x_1)$, length of weld $l(x_2)$, width of the beam $t(x_3)$, and the thickness of the beam $b(x_4)$. The problem can be stated as follows.

$$Minimize\ C(h,l,t,b) = 1.10471h^2 l + 0.04811tb(14.0 + l)$$

Subject to

$$g_1(\tau)=13{,}600-\tau\geq 0,\ g_2(\sigma)=30{,}000-\sigma\geq 0$$
$$g_3(b,h)=b-h\geq 0,\ g_4(P_c)=P_c-6{,}000\geq 0$$
$$g_5(\delta)=0.25-\delta\geq 0$$

The expressions for $\tau, \sigma, \mathrm{P_C}$ and δ are given by :

$$\tau=\sqrt{(\tau')^2+(\tau'')^2+\frac{l\tau\tau'}{\alpha}}\quad,\quad \alpha=\sqrt{0.25\left(l^2+(h+t)^2\right)}\quad,\quad \sigma=504000/\left(t^2b\right)\quad,$$
$$P_c=64746.022(1-0.0282346)\,tb^3,\ \delta=2.1952/\left(t^3b\right),\ \tau'=6000/\left(\sqrt{2}hl\right)$$

(EP2) Welded beam design (case 2) (He & Wang, 2007a)

$$Min\ f(x)=1.1047x_1^2x_2+0.04811x_3x_4(14+x_2)$$

$$g_1(x)=\tau(x)-13600\leq 0,$$
$$g_2(x)=\rho(x)-30{,}000\leq 0$$
$$g_3(x)=x_1-x_4\leq 0$$
$$g_4(x)=0.10471x_1^2+0.04811x_3x_4(14+x_2)-5.0\leq 0$$
$$g_5(x)=0.125-x_1\leq 0,\ g_6(x)=\delta(x)-0.25\leq 0$$
$$g_7(x)=6000-P_c(x)\leq 0$$

Where, $\tau(x)=\sqrt{(\tau')^2+2\tau'\tau''\frac{x_2}{2R}+(\tau'')^2},\ \tau'=\frac{6000}{\sqrt{2}x_1x_2},\ \tau''=\frac{MR}{J}$

$$M=6000\left(14+(x_2/2)\right),\ R=\sqrt{\frac{x_2^2}{4}+\left(\frac{x_1+x_3}{2}\right)^2}$$
$$J=2\left(x_1x_2\sqrt{2}\left[\frac{x_2^2}{12}+\left(\frac{x_1+x_3}{2}\right)^2\right]\right)$$
$$\sigma(x)=\frac{504000}{x_4x_3^2}$$
$$\delta(x)=\frac{65856000}{\left(30\times 10^6\right)x_4x_3^3}$$

$$P_c = \frac{4.013\left(30\times10^6\right)\sqrt{\frac{x_3^2 x_4^6}{36}}}{196}\left(1-\sqrt{\frac{0.79056x_3}{28}}\right)$$

$0.1 \le x_1, x_4 \le 2$ *and* $0.1 \le x_2, x_3 \le 10.0$

(EP3) Pressure vessel design problem (He & Wang, 2007b)

A pressure vessel is a cylindrical vessel that is capped at both ends by the hemispherical heads. The vessel is designed to minimize total cost including the cost of the material, forming and welding. There are 4 design variables: thickness of the shell $T_s(x_1)$, thickness of the head $T_h(x_2)$, the inner radius $R(x_3)$, and the length of the cylindrical section of the vessel not including the head $L(x_4)$. T_s and T_h are integer multiples of 0.0625 inches, which are the available thickness of rolled steel plates, where as R and L are continuous. The design variables are given in inches and the weight is written as:

$$Min\, W(T_s, T_h, R, L) = 0.6224 T_s T_h R + 1.7781 T_h R^2 + 3.1661 T_s^2 L + 19.84 T_s^2 R$$

Subject to

$$g_2(T_h, R) = T_h - 0.00954R \ge 0$$

$$g_3(R, L) = \pi R^2 L + 4/3\pi R^3 - 1,296,000 \ge 0$$

$$g_4(L) = -L + 240 \ge 0$$

Where,

$$0.0625 \le T_s, T_h \le 5, \quad 10 \le R, \text{ and } \quad L \le 200$$

(EP4) The Tension/Compression Spring (He & Wang, 2007b)

This problem aims to minimize the volume V of a tension/compression spring subject to constraints on minimum deflection, shear stress, surge frequency, limits on outside diameter D and on design variables. The design variables are the mean coil diameter, the wire diameter d, and the number of active coils N. The problem has 4 inequality constraints. The volume of the coil to be minimized is:

$$Minimize \quad V(N,D,d)=(N+2)Dd^2$$

Subject to

$$g_1(x)=1-D^3N/(71785d^4)\le 0$$
$$g_2(x)=\frac{4D^2-4D}{12566(Dd^3-d^4)}+\frac{1}{5108d^2}-1\le 0$$
$$g_3(x)=1-\frac{140,45d}{D^2N}\le 0$$
$$g_4(x)=(D+d)/1.5-1\le 0$$

Where, $0.05\le d\le 2,\quad 0.25\le D\le 1.3,\qquad 2\le N\le 15$

(EP5) Three bar truss design problem (Ray &Liew, 2003)

$$Minimize \quad f(x)=\left(2\sqrt{2}x_1+x_2\right)\times l$$

Subject to

$$g_1(x)=\frac{\sqrt{2}x_1+x_2}{\sqrt{2}x_1^2+2x_1x_2}P-\sigma\le 0$$
$$g_2(x)=\frac{x_2}{\sqrt{2}x_1^2+2x_12x_2}P-\sigma\le 0$$
$$g_3(x)=\frac{1}{x_1+\sqrt{2}x_2}P-\sigma\le 0$$
$$0\le x_1\le 1, 0\le x_2\le 1$$

$$l=100cm,\ P=2KN/cm^2,\ \sigma=2KN/cm^2$$

(EP6) Speed reducer design optimization problem (Ray and Liew, 2003)

This problem represents the design of a simple gear box such as might be used in a light airplane between the engine and propeller to allow each to rotate at its most efficient speed. The design of the speed reducer has 7 design variables. The face width x_1, module of teeth x_2, number of teeth on pinion x_3, length of the first shaft between bearings x_4, length of the second shaft between bearings x_5, diameter of the first shaft x_6 and diameter of the second shaft x_7. The mathematical model of the problem is as follows:

$$\begin{aligned} Minimize \quad f(x) = {} & 0.7854 x_1 x_2^2 \left(3.3333 x_3^2 + 14.9334 x_3 - 43.0934\right) \\ & -1.508 x_1 \left(x_6^2 + x_7^2\right) + 7.4777\left(x_6^3 + x_7^3\right) + 0.78054\left(x_4 x_6^2 + x_5 x_7^2\right) \end{aligned}$$

Subject to

$$c_1(x) = \frac{27}{x_1 x_2^2 x_3} - 1 \le 0, c_2(x) = \frac{397.5}{x_1 x_2^2 x_3^2} - 1 \le 0$$

$$c_3(x) = \frac{1.93 x_5^3}{x_2 x_3 x_6^4} - 1 \le 0, c_4(x) = \frac{1.93 x_5^3}{x_2 x_3 x_7^4} - 1 \le 0$$

$$c_5(x) = \frac{1.0}{110 x_6^3} \sqrt{\left(\frac{750.0 x_4}{x_2 x_3}\right)^2 + 16.9 \times 10^6} - 1 \le 0$$

$$c_6(x) = \frac{1.0}{85 x_7^3} \sqrt{\left(\frac{750.0 x_5}{x_2 x_3}\right)^2 + 157.5 \times 10^6} - 1 \le 0$$

$$c_7(x) = \frac{x_2 x_3}{40} - 1 \le 0, \; c_8(x) = \frac{5 x_2}{x_1} - 1 \le 0$$

$$c_9(x) = \frac{x_1}{12 x_2} - 1 \le 0, c_{10}(x) = \frac{1.5 x_6 + 1.9}{x_4} - 1 \le 0$$

$$c_{11}(x) = \frac{1.1 x_7 + 1.9}{x_5} - 1 \le 0$$

$$2.6 \le x_1 \le 3.6, 0.7 \le x_2 \le 0.8$$

$$17 \le x_3 \le 28, 7.3 \le x_4 \le 8.3$$

$$7.3 \le x_5 \le 8.3, 2.9 \le x_6 \le 3.9 \; and \; 5.0 \le x_7 \le 5.5$$

5.5.2 Experimental Setup

All the above 6 engineering problems are solved by CGAC. The result obtained by CGAC is compared with Differential Evolution with level comparison (DELC), Differential Evolution with dynamic stochastic Selection(DEDS), Hybrid evolutionary algorithm and adaptive constraint-handling technique (HEAA), GA, Social behavior inspired optimization Technique (SBO) and Rank Selection based PSO (RSPSO), an effective Co-evolutionary differential evolution for constrained optimization(CDE), a hybrid PSO with a feasibility based rule for constrained optimization (HPSO), An effective Co-evolutionary PSO for constrained engineering design problem(CPSO); taken from (Wang and Li, 2010). To handle all the problems together, the fine tuned parameter setting are reported in Table 5.7.

For each problem, 30 independent runs are performed by keeping the following basic parameters fixed.

i. Population size = 40
ii. The Probability of cross over (P_c)=0.9
iii. Probability of mutation (P_m)=0.05
iv. Bit length (l)=12
v. Chemo tactic loop is initiated after 70 generations
vi. The stopping criteria is either a maximum of 4000 generations is attained or no improvement is observed in the best objective function value in consecutive 2000 generations.

The best, median, mean, worst objective function values along with the total number of function evaluations(TNFE) and standard deviation(S.D) are reported in Table 5.8, 5.10, 5.12, 5.14, 5.16,5.18 for EP1toEP6 respectively. The best solutions obtained by all the methods are compared in Table 5.9, 5.11, 5.13, 5.15, 5.17, 5.19 respectively for EP1 to EP6. In all the tables, the bold face letters indicate the best result obtained by the corresponding mechanism/algorithm.

Table 5.7: Fine tuned parameter setting for engineering problems

Pb. No.	**Real life problems**	**No of bacteria initialized in chemotactic loop**	**Chemo tactic step**	**Max stepsize (Cmax)**	**Min step size (Cmin)**
1	**EP1**	80	22	0.1	0.008
2	**EP2**	80	40	0.1	0.007
3	**EP3**	4	20	0.1	0.006
4	**EP4**	20	22	0.001	0.0000001
5	**EP5**	6	80	0.001	0.0000001
6	**EP6**	2	25	0.1	0.006

Table 5.8: Comparison of objective function values (best/median/mean/worst/SD) for EP1

Algorithm	**Best**	**Median**	**Mean**	**Worst**	**Std**	**TNFE**
CGAC	**2.38095657684507**	2.380959152	2.380964352	2.3809926722288	1.05 E-05	739680
DELC	2.38095658	**2.38095658**	**2.38095658**	**2.38095658**	**2.6E-12**	**20000**
DEDS	2.38095658	**2.38095658**	**2.38095658**	**2.38095658**	3.2E-10	24000
HEAA	2.38095723	2.38096560	2.3809656	2.38102095	1.3E-05	30000
GA	2.38119	2.38289	NA	2.64583	NA	40080
SBO	2.381065	3.0025883	3.2551371	6.3996785	9.6E-01	33095
RSPSO	2.38095658	NA	2.380959	2.3810190	1.1E-05	30000

Table 5.9: Comparison of best solution for EP1

	CGAC	**DELC**	**DEDS**	**HEAA**	**GA**	**SBO**	**RPSO**
x_1	**0.244368974824662**	0.2443689758	0.2443689758	0.2443688943	NA	0.24443822760	0.24436898
x_2	**6.21751973411622**	6.2175197152	6.2175197152	6.2175179741	NA	6.2379672340	6.2175971
x_3	**8.29147140908299**	8.2914713905	8.2914713905	8.2914773014	NA	8.2885761430	8.29147140
x_4	**0.244368974845275**	0.2443689758	0.2443689758	0.2443689510	NA	0.2445661820	0.24456898
$f(x)$	**2.38095657684507**	2.38095658	2.38095658	2.38095723	2.38119	2.3854347	2.3809565817

Table 5.10: Comparison of objective function values (best/median/mean/worst/SD) forEP2

Algorithm	**Best**	**Median**	**Mean**	**Worst**	**Std**	**TNFE**
CGAC	**1.7248523052983**	**1.724852305**	**1.724852323**	**1.72485254197767**	4.83079E-08	120,01940
DELC	**1.724852**	**1.724852**	**1.724852**	**1.724852**	**4.1E-13**	**20000**
FGA	1.728226	NA	1.792654	1.993408	7.4E-02	80000
CEA	**1.724852**	NA	1.971809	3.179709	4.4E-01	50020
CPSO2	1.728024	NA	1.748831	1.782143	1.3E-02	200,000
HPSO	**1.724852**	NA	1.749040	1.814295	4.0E-02	81000
CDE	1.733461	NA	1.768158	1.824105	2.2E-02	240,000

Table 5.11: Comparison of best solution for EP2

	CGAC	**DELC**	**FGA**	**CEA**	**CPSO**	**HPSO**	**CDE**
x_1	**0.205729638868118**	**0.20572964**	0.205986	**0.205700**	0.202369	**0.205730**	0.203137
x_2	**3.47048867762648**	**3.47048867**	3.471328	**3.470500**	3.544214	**3.470489**	3.542998
x_3	**9.03662393051431**	**9.03662391**	9.020224	**9.036600**	9.048210	**9.036624**	9.033498
x_4	**0.205729638868333**	**0.20572964**	0.206480	**0.205700**	0.205723	**0.205730**	0.206179
$f(x)$	**1.7248523052983**	**1.724852**	1.728226	**1.724852**	1.728024	**1.724852**	1.733461

Table 5.12: Comparison of objective function values (best/median/mean/worst/SD) for EP3

Algorithm	Best	Median	Mean	Worst	Std	TNFE
CGAC	**5885.35875**	**5971.3004507**	**5958.31**	**6001.157**	41.107	12,59687
DELC	6059.7143	6059.7143	6059.7143	6059.7143	**2.1e-11**	**30,000**
FGA	6059.9463	NA	6177.2533	6469.3220	130.9	50,020
CPSO	6061.0777	NA	6147.1332	6363.8041	86.5	200,000
HPSO	6059.7143	NA	6099.9323	6288.6770	86.2	81,000
CDE	6061.0777	NA	6085.2303	6371.0455	43.0	240,000

Table 5.13: Comparison of best solution for EP3

	CGAC	DELC	FGA	CPSO	HPSO	CDE
x_1	**0.778174093211299**	0.8125	0.8125	0.8125	0.8125	0.8125
x_2	**0.384651857473358**	0.4375	0.4375	0.4375	0.4375	0.4375
x_3	**40.319901202658**	42.0984455959	42.097398	42.091266	42.0984	42.098411
x_4	**199.996067701114**	176.6365958424	176.654047	176.746500	176.6366	176.637690
$f(x)$	**5885.35875**	6059.7143	6059.9463	6061.0777	6059.7143	6059.7340

Table 5.14: Comparison of objective function values (best/median/mean/worst/SD) EP4

Algorithm	Best	Median	Mean	Worst	Std	TNFE
CGAC	**0.0126652329519018**	0.012666561	0.0126707229	0.012688879695068	7.46E-06	32,85140
DELC	0.012665233	**0.012665233**	0.012665267	0.012665575	1.3E-07	**20,000**
DEDS	0.012665233	0.012665304	0.012669366	0.012738262	1.3E-05	24,000
HEAA	0.012665233	0.012665234	**0.012665234**	**0.012665240**	**1.4E-09**	24,000
CEA	0.0127210	NA	0.0135681	0.015116	8.4E-04	50,020
HPSO	0.0126652	NA	0.0127072	0.0127191	1.6E-05	81,000
CDE	0.0126702	NA	0.012703	0.012790	2.7E-05	240,000

Table 5.15: Comparison of best solution for EP4

	CGAC	DELC	DEDS	HEAA	CEA	HPSO	CDE
x_1	**0.35664569954802**	0.3567177413	0.3567177469	0.3567292035	0.317395	0.357126	0.354714
x_2	**0.0516860663861192**	0.0516890611	0.0516890614	0.0516895376	0.050000	0.051706	0.051609
x_3	**11.2931904972553**	11.2889656626	11.2889653382	11.2882937035	14.031795	11.265083	11.410831
$f(x)$	**0.0126652329519018**	0.012665233	0.012665233	0.012665233	0.0127210	0.0126652	0.0126702

Table 5.16: Comparison of objective function values (best/median/mean/worst/SD) for EP5

Algorithm	Best	Median	Mean	Worst	Std	TNFE
CGAC	**263.895843376641**	**263.8958434**	**263.8958434**	263.895843940103	1.19E-07	2831177
DELC	263.8958434	**263.8958434**	**263.8958434**	**263.8958434**	**4.3E-14**	**10,000**
DEDS	263.8958434	**263.8958434**	263.8958436	263.8958498	9.7E-07	15,000
HEAA	263.895843	263.895848	263.895865	263.896099	4.9E-05	15,000
SBO	263.8958466	263.8989	263.9033	263.96975	1.3E-02	17,610

Table 5.17: Comparison of best solution for EP5

CGAC	CGAC	DELC	DEDS	HEAA	SBO
x_1	**0.788674649234619**	0.7886751287	0.7886751359	0.3567292035	0.7886210370
x_2	**0.408249663271532**	0.4082483070	0.4082482868	0.0516895376	0.4084013340
$f(x)$	**263.895843376641**	263.8958434	263.8958434	263.8958434	263.8958466

Table 5.18: Comparison of objective function values (best/median/mean/worst/SD) for EP6

Algorithm	Best	Median	Mean	Worst	Std	TNFE
CGAC	**2902.03044938979**	**2924.521829**	**2925.87**	**2959.59909616779**	15.7711	6,98187
DELC	2994.471066	2994.471066	2994.471066	2994.471066	**1.9E-12**	**30,000**
DEDS	2994.471066	2994.471066	2994.471066	2994.471066	3.6E-12	**30,000**
HEAA	2994.499107	2994.599748	2994.613368	2994.752311	7.0E-02	40,000

Table 5.19: Comparison of best solution for EP6

Var.	CGAC	DELC	DEDS	HEAA	SBO	MDE
x_1	**3.51123035556909**	3.5	3.5	3.5000228993	3.5000681	3.500010
x_2	**0.7**	0.7	0.7	0.7000003924	0.70000001	0.70000
x_3	**17**	17	17	17.0000128592	17	17
x_4	**7.577220959287**	7.3	7.3	7.3004277414	7.32760205	7.300156
x_5	**7.73362180545155**	7.7153199115	7.7153199115	7.7153774494	7.71532175	7.800027
x_6	**2.9**	3.3502146661	3.3502146661	3.3502309666	3.35026702	3.350221
x_7	**5.28816024668864**	5.2866544650	5.2866544650	5.2866636970	5.28665450	5.286685
$f(x)$	**2902.03044938979**	2994.471066	2994.471066	2994.499107	2994.744241	2996.356689

5.5.3 Results and Discussions

From Tables 5.8, 5.10, 5.12, 5.14, 5.16, 5.18 it is observed that the best objective function values achieved by CGAC for all 6 functions are better comparable with its next competitor whereas it supersedes many of the rest methods under consideration. Especially, in case of EP3 and EP6, CGAC yield much improved result than all rest. Of course, CGAC takes much computational time and number of function calls, but in return it reaches the global optimal solution more closely for all the engineering design problems.

5.6 CONCLUSIONS

In this Chapter CGA is further developed to handle constraints by using bracket penalty parameter. The algorithm thus generated for constrained optimization problems is abbreviated as CGAC. The validation of better performance of CGAC is investigated through a set of 15 constrained benchmark problems of different taste. From the results and discussion it is concluded that though CGAC takes more number of function evaluations and computational time in comparison to LXPMC and HLXPMC, still it provides more success rate, yields better objective function values and standard deviation (S.D) in most of the cases as compared to LXPMC and HLXPMC (real coded GA that uses Laplace crossover and Power mutation). Secondly, it is also observed that CGAC is taking less time and function evaluation in comparison to Improved Genetic Algorithm; Teaching-Learning based optimization and other popular algorithms over a set of 11 benchmarking functions. Further, 6 engineering design problems have been solved in which the parallel result is achieved with a conclusion that CGAC performs similar or better than 7 different recent methods available in literature. Therefore, the proposed hybrid algorithm CGAC is more reliable, more efficient and more accurate as compared to recent state of the art constrained evolutionary algorithms for solving non-linear constrained optimization problems.