*Engineering Optimization*
Vol. 00, No. 00, November 2006, 1–33

# Multiple offspring in differential evolution for engineering design

Efrén Mezura-Montes[1], Carlos A. Coello Coello[2]
Jesús Velázquez-Reyes[3] and Lucía Muñoz-Dávila[4]
[1] **(corresponding author)**
Laboratorio Nacional de Informática Avanzada (LANIA A.C.)
Rébsamen 80, Centro, Xalapa, Veracruz, 91000 MEXICO
Tel. +52 228 841 6100 x 5006
Fax +52 228 841 6101
email: `emezura@lania.mx`
[2] CINVESTAV-IPN (Evolutionary Computation Group)
Departamento de Computación
Av. Instituto Politécnico Nacional No. 2508
Col. San Pedro Zacatenco
México D.F. 07300, MEXICO
email: `ccoello@cs.cinvestav.mx`
[3] Intel Tecnología de México
Guadalajara Design Center, MEXICO
email: `jesus.velazquez@intel.com`
[4] Instituto Tecnológico de Apizaco
División de Estudios de Posgrado e Investigación
Av, Instituto Tecnológico S/N, Centro, Apizaco, Tlaxcala, 90300, MEXICO
email: `lucy@itapizaco.edu.mx`
*(November 2006)*

We propose a modified version of the differential evolution approach to solve engineering design problems. The aim is to allow each parent in the population to generate more than one offspring at each generation and therefore, to increase its probability of generating a better offspring. To deal with constraints, we use some criteria based on feasibility and a diversity mechanism to maintain infeasible solutions in the population. The approach is tested on a set of well-known benchmark problems. After that, it is used to solve engineering design problems and its performance is compared with those provided by typical penalty function approaches and also against state-of-the-art techniques.

**Keywords**: Evolutionary Algorithms, Differential Evolution, Constrained Optimization, Engineering Design.

## 1 Introduction

Many engineering design problems can be stated like the general nonlinear programming problem in which we want to:

$$\text{Find } \vec{x} \text{ which optimizes } f(\vec{x})$$

Subject to:

$$g_i(\vec{x}) \leq 0, i = 1, \ldots, m$$
$$h_j(\vec{x}) = 0, j = 1, \ldots, p$$

where $\vec{x} \in \mathbb{R}^n$ is the vector of solutions $\vec{x} = [x_1, x_2, \ldots, x_n]^T$, where each $x_i, \quad i = 1, \ldots, n$ is bounded by lower and upper limits $L_i \leq x_i \leq U_i$; $m$ is the number of inequality constraints and $p$ is the number of equality constraints (in both cases, constraints could be linear or nonlinear).

Differential Evolution (DE) is a novel evolutionary algorithm proposed by Storn and Price (1999). The approach works with a mutation operator which is based on the current distribution of solutions in the population, instead of being based on a fixed (usually Gaussian) distribution such as other Evolutionary Algorithms (EAs) like Evolution Strategies (Bäck 1996). At each generation in DE, each parent will generate one offspring. If this child is better than its parent, it will replace him in the population; if not, the parent will remain and the child is eliminated.

The motivation of this paper is to introduce an evolutionary optimizer which fulfills two main goals: (1) is able to increase the probability of a parent to generate a fitter offspring and (2) it does not use a penalty function to deal with the constraints of the problem. The first objective is reached by allowing each parent to generate more than one offspring at each generation. In this way, a pre-selection mechanism is incorporated to select, by using a deterministic process, the best solution among the offspring of one parent, and only this best solution will compete against its parent in order to remain in the population. The second objective refers to handling the objective function value and the constraints of the problem separately and to use a mechanism to keep solutions with a good value of the objective function, regardless of their feasibility, in the population. This issue is important because maintaining promising infeasible solutions will increase the exploration of new regions of the search space and the feasible region and it will decrease the chances of getting trapped in local optima.

The paper is organized as follows: In Section 2, we present the previous related work. Section 3 provides the description of our approach. Afterwards, in Section 4 we detail the experimental design and we present and discuss the results obtained. Finally, in Section 5 some conclusions are drawn and the future work is established.

## 2 Related Work

EAs have been widely used to solve engineering design problems. Storn (1999) proposed a constraint-relaxation mechanism coupled with the aging concept to solve optimization problems using DE. He explored the idea of allowing a solution to generate more than one offspring, but in his approach, once a child is better than its parent, the multiple offspring generation ends. Furthermore, the comparison between solutions is always deterministic and the constraint-handling mechanism is based on relaxing the constraint at the beginning in order to consider all the feasible solutions. Among the state-of-the-art approaches available to solve engineering design problems, we present the following: He et al. (2004) proposed a PSO-based approach to solve engineering design problems. The main advantage of the approach is its low computational cost measured in terms of the number of objective function evaluations. However, He's approach only works with feasible points; therefore, there is no diversity mechanism at all and an initial population of feasible solutions is always required. This is obviously the main disadvantage of the technique. Ray & Liew (2003) used a social model to solve engineering optimization problems. In their model, the population of solutions is seen as a civilization and it is divided into sub-populations known as societies. There are leaders in each society and the leaders are also grouped in a leaders' society. Then, solutions can follow its corresponding leader, a leader of another society or follow the leader of leaders. In this way, the approach aims to maintain diversity. Constraints are handled by using dominance as a selection criterion. The main advantage of the approach is that it requires a low number of evaluations of the objective function to obtain competitive results. However, it requires a ranking process at each generation besides a clustering algorithm which is used to initialize the societies. Regarding constraint-handling in EAs, several techniques have been proposed. The most common approach is the use of a penalty function (Coello 2002), whose aim is to punish infeasible solutions in order to favor feasible individuals in the selection process. As a result, the search will be biased to the feasible region of the search space. However, the main drawback of this approach is the careful fine-tuning required by the penalty factors, which determine the severity of the penalty and it has been shown that their values are problem dependent (Smith and Coit 1997). To tackle this problem, several alternatives have been proposed (Coello 2002). One of the most recent approaches are the self-adaptive penalty function proposed by Tessema and Yen (2006), where a new fitness value, called distance value (in the normalized fitness-constraint violation space) and two penalties are applied to those infeasible solutions so that the algorithm is able to identify the best infeasible individuals in the current population. Mezura and Coello (2005) proposed an evolution strategy with a set of feasibility rules (Deb 2000), a combined discrete-intermediate recombination operator and a smooth stepsize for the mutation operator to solve constrained optimization problems.

## 3 Our proposed approach

'[Insert Figure 1 about here]'

The aim of the approach proposed in this paper is to add simple modifications to the DE algorithm in order to adapt it to deal with constrained search spaces and also to improve its performance by generating more offspring per parent. At each generation, each parent will be able to generate $n_o$ offspring. Among these newly generated solutions, the best of them will be selected to compete against its parent. Then, each parent will increase its chances to generate a fitter offspring. The selection of the best child is completely deterministic based on the three following criteria (Deb 2000):

(i) Between 2 feasible solutions, the one with the highest fitness value wins.
(ii) If one solution is feasible and the other one is infeasible, the feasible solution wins.
(iii) If both solutions are infeasible, the one with the lowest sum of constraint violation is preferred
($\sum_{i=1}^{m} max(0, g_i(\vec{x}))$).

After the best offspring is selected, it will compete against its corresponding parent and the best of them will survive for the next generation. However, in this case the comparison between parent and best offspring is performed with a stochastic element. Based on a parameter called selection ratio $S_r$ the best solution is chosen based only on the value of the objective function value. In the remaining $1 - S_r$ selections, the best solution is chosen based on the three criteria mentioned before. In this way, the best feasible solutions will remain in the population, besides those solutions with a promising value of the objective function, regardless of feasibility. This mechanism, coupled with the DE's way of finding new search directions (based on the distribution of solutions in the current population) will allow the algorithm to explore the search space and its feasible region in a better way as to obtain better solutions. The detailed pseudocode of our approach is presented in Figure 1.

## 4 Experiments and Discussion

Our experimental design has three parts: (1) to test our approach in a set of well-known benchmark problems, (2) to compare our approach against different types of penalty function approaches and (3) to compare our results against state-of-the-art approaches and against a traditional DE approach which generates only one offspring per parent.

For the first experiment, we selected 13 test problems commonly used to test evolutionary algorithms dealing with constrained search spaces (Mezura-Montes and Coello Coello 2005). The main features of the problems are presented in Table 1 an the details of each function are presented in an Appendix at the end of the paper.

'[Insert Table 1 around here]'

We performed 30 independent runs with the same set of parameters as follows: $NP = 90$, $MAX\_GENERATIONS = 500$ (225,000 evaluations of the objective function), $CR = 0.9$, $n_o = 5$ and $S_r = 0.45$, F randomly generated within $[0.3, 0.9]$. Equality constraints were transformed into inequality constraints ($|h(\vec{x}) - \epsilon| \leq 0$) with a tolerance value of $\epsilon = 0.0001$.

We will measure the quality of results ("better" best solution found) and also the robustness of the approach ("better" mean and standard deviation values. These statistical results are summarized in Table 2.

'[Insert Table 2 around here]'

The results obtained by our approach are also compared against a traditional DE approach which uses the same set of initial parameters adopted by our approach as well as the same constraint handling mechanism. The only difference between traditional DE and our approach is that in the first, only one offspring per parent is generated and in our approach we generate 5 offspring per parent. The parameter values used in our approach are exactly the same used in the previous experiment. Furthermore, we decided to compare the performance obtained by our modified DE against two state-of-the-art approaches: The self-adaptive penalty function (Tessema and Yen 2006) and the Simple Multimembered Evolution Strategy (Mezura-Montes and Coello Coello 2005). The results of all approaches are presented and compared also in Table 2.

As it can be noted, our approach clearly outperformed the traditional DE approach, based on quality and robustness. In fact, the traditional DE was unable to generate feasible solutions for some test functions. With respect to the self-adaptive penalty approach (Tessema and Yen 2006), our approach provided "better" best results in 7 problems (g02, g04, g05, g06, g07, g09, g10), and "similar" best results in 5 problems (g01, g03, g08, g11 and g12). The self-adaptive technique obtained a "better" best result in problem g13. With respect to robustness, our approach provided "better" mean and standard deviation in all 13 test problems. Compared to the SMES (Mezura-Montes and Coello Coello 2005), our approach provided a "better" best result in five problems g05, g07, g09, g10 and g13; and a "similar" best result in the remaining seven problems: g01, g02, g03, g04, g06, g08, g11 and g12. Finally, our approach provided "better" mean and standard deviation values in seven problems: g02, g05, g06, g07, g09, g10 and g13 and "similar" mean and standard deviation values in the remaining six: g01, g03, g04, g08, g11 and g12.

To get a statistical support to our findings regarding the robustness of the proposed approach, we decided to calculate the 95%-confidence intervals for the mean statistic. First, we used a Kolmogorov-Smirnov one-sample test to verify the normality of the distribution. However, we found that none of the distributions was close to a normal. Therefore, we chose the bootstrapping method (which does not require to assume normality of the distribution) to calculate the intervals. We used 1000 re-samples per original sample. The results are presented in Table 3. The results showed that the proposed approach is able to accurately approximate, in the mean case, the best known solution on each problem.

The overall results suggest that our approach provided a competitive performance when solving this set of benchmark problems, where different features are included (linear and nonlinear objective function, equality and/or inequality constraints, either linear or nonlinear, different dimensionality, different size of the feasible region with respect to the whole search space, etc). Finally, it is important to remark that our approach required $225,000$ evaluations to obtain the reported results, compared to $240,000$ used by SMES (Mezura-Montes and Coello Coello 2005) and $500,000$ by the self-adaptive penalty approach (Tessema and Yen 2006).

'[Insert Table 3 around here]'

Based on the promising performance provided by our approach, we decided to apply it in some engineering design problems. For this second set of experiments, we selected four well known engineering design problems. The details of each problem are presented in the Appendix at the end of this document, but their main features are summarized in Table 4.

'[Insert Table 4 around here]'

To compare the performance of our approach, we decided first, to compare it against typical penalty-based evolutionary algorithms. Thus, we implemented four typical penalty approaches: (1) Assign a zero fitness value to infeasible solutions e.g. death penalty (Schwefel 1981), (2) static penalty, i.e. fixed penalty factor during all the process (Hoffmeister and Sprave 1996), (3) dynamic penalty i.e. the penalty factor has a low value at the beginning and a high value at the end of the process (Joines and Houck 1994) and finally, (4) an adaptive penalty i.e. the penalty factor is updated based on the behavior of the population (Hadj-Alouane and Bean 1997).

30 independent runs per technique per problem were performed. The number of evaluations of the objective function was fixed to $24,000$ for the four penalty-based approaches and also for our approach. For the penalty-based approaches we used a gray-coded genetic algorithm with roulette wheel selection, one point crossover and uniform mutation. The population size was 100 individuals and the number of generations 240. The rate of crossover was 0.6 and the mutation rate was 0.01. The parameters for the static, dynamic and adaptive approaches were defined after a trial-and-error process. The reported parameters were those which provided the best results and they are the following: Static approach: fixed penalty factor $= 1000$. Dynamic approach: $\alpha = 2$, $\beta = 2$, $C = 0.5$. Adaptive approach: $\beta_1 = 2.0$, $\beta_2 = 4.0$, $k = 50$, $\delta_{initial} = 5000$ Our DE-based approach used the following parameters: $NP = 60$, $MAX\_GENERATIONS = 80$ (24,000 evaluations of the objective function), $CR = 0.9$, $n_o = 5$ and $S_r = 0.45$, F was randomly generated within the range $[0.3, 0.9]$. Discrete variables were handled by just truncating the real value to its closest integer value. The statistical results of the 30 independent runs are shown in Table 5.

'[Insert Table 5 around here]'

Based on the results obtained, our approach was able to provide the most robust ("better" mean and standard deviation values) and the best quality results ("better" best solution found) in all four engineering design problems adopted. In fact, none of the penalty-based approaches was able to find feasible solutions for the speed reducer design problem. Besides, the dynamic penalty approach could not find feasible solutions for the pressure vessel problem and it could not find feasible solutions in two and eight runs for the welded beam and the pressure vessel problems, respectively. Furthermore, the adaptive approach could not find feasible solutions for the welded beam and for the pressure vessel problems in three and nine runs, respectively. On the other hand, our approach consistently found feasible solutions in each run for all design problems.

The overall results suggest that our approach provided the most consistent performance, while the penalty-based approaches were competitive in some problems, but in others the results were poor. This behavior indeed reflects the need to update the penalty factors according to the problem to be solved. This negative effect is not present in our approach, because with the same set of parameters, avoiding the use of a penalty function and with the diversity mechanism, the approach finds good feasible solutions consistently.

For the third set of experiments we compared the performance of our approach against those provided by the last two approaches discussed in Section 2. We used He's and Ray's approaches because they solved the same set of test problems. In this case, the number of evaluations required by each approach is included and it is used as a comparison criterion. We also added the results obtained with a traditional DE approach with the same set of initial parameters used by our approach and with the same constraint handling mechanism. The only difference between traditional DE and our approach is that at the beginning, only one offspring per parent is generated and in our approach we generate 5 offspring per parent. The parameter values used in our approach are exactly the same used in the previous experiment. The comparison of the statistical results and the number of evaluations required by each approach are presented in Table 6. In Tables 7, 8, 9 and 10, we provide the details of the best solution found by each state-of-the-art technique and our approach.

'[Insert Table 6 around here]'

The results show that our approach clearly outperforms the two compared approaches in the welded beam design problem ("better" best, mean and standard deviation values, requiring less evaluations to provide such a performance). For the pressure vessel problem, Ray's approach required the lowest number of evaluations. However, our approach provided clearly "better" best, mean and standard deviation values than those provided by Ray's technique while just performing 4000 more evaluations. For the spring design problem, He's approach provided the same best solution found by our approach by using just $15,000$ evaluations of the objective function. Nonetheless, it is important to note that He's approach requires to generate an initial feasible population. In contrast, our approach starts with random solutions, regardless of feasibility. Furthermore, our approach provided "better" mean and standard deviation values, which imply more robustness of our approach. For the speed reducer problem, Ray's approach provided the "best" best result, but it required more than twice the number of evaluations used by our approach. Besides, our best solution is close to the value provided by Ray's approach and also our mean and standard deviation values are clearly better, showing again, the robustness of the approach. Finally, our approach provided better results in all cases in all criteria with respect to the traditional DE approach despite the fact that both share the same diversity mechanism. The details of the best solutions found shown in Tables 7, 8, 9 and 10 seem to emphasize the ability of the approach, based on the intensive use of the DE operator in one parent, to explore solutions close to already good solutions and to improve the quality of the final result. This behavior was found mostly in the beam and the pressure vessel design problems.

The overall results of this third experiment suggest that our approach was able to provide very competitive results compared with those provided by two state-of-the-art approaches based on the quality, robustness and computational cost measured by the number of evaluations of the objective function. Besides, the chance to generate more offspring per parent provided an improvement in the quality and robustness of the results obtained in the two DE-based algorihtms compared.

### 4.1 *Remarks*

Based on the results of the three experiments developed, we summarize the following findings:

- The effect of generating more than one offspring per parent in DE seems to improve the quality and robustness of the solutions obtained significantly.
- The quality and robustness presented by the approach is competitive, and better in some cases, with respect to those provided by state-of-the-art techniques when solving a set of benchmark problems.
- The proposed approach required less evaluations of the objective function of the problem, compared to the two state-of-the-art proposals used for comparison when solving the benchmark problems.
- The diversity mechanism coupled with DE in the proposed approach improves the capabilities of the approach to sample the constrained search space as to reach better solutions than those obtained by the traditional penalty-based techniques.
- The results obtained by our approach to solve four engineering design

problems, compared with two state-of-the-art techniques, were competitive, based on quality, robustness and computational cost (measured by the number of evaluations of the objective function).

'[Insert Table 7 around here]'

'[Insert Table 8 around here]'

'[Insert Table 9 around here]'

'[Insert Table 10 around here]'

## 5 Conclusions and Future Work

We have presented a DE-based approach to solve engineering design problems. The approach was based on the more frequent use of the DE operator per parent. The constraint handling technique adopted was based on simple selection criteria and a diversity mechanism to maintain promising solutions regardless on feasibility. The results obtained in three different experiments show that the proposed approach provided a very competitive performance with respect to those provided by two state-of-the-art evolutionary-based approaches and also with respect to a traditional differential evolution approach to solve several constrained optimization test problems. Also, our technique clearly outperformed four different typical penalty-based techniques when tested in four engineering design problems. Also, in the last experiment, the approach obtained better results than a traditional DE approach and it provided very competitive results against other two state-of-the-art approaches. As future paths of research we will explore a mechanism to adapt the parameters that control the number of offspring per parent and the diversity in the population. Finally, we aim to test our approach in real-world problems.

***Acknowledgements.*** The second author acknowledges support from CONACyT through project no. 45683-Y.

## Appendix

The details of the benchmark functions and the set of engineering design problems used in the experiments are the following:

- **g01**:
Minimize: $f(\vec{x}) = 5\sum_{i=1}^{4} x_i - 5\sum_{i=1}^{4} x_i^2 - \sum_{i=5}^{13} x_i$
subject to:
$g_1(\vec{x}) = 2x_1 + 2x_2 + x_{10} + x_{11} - 10 \leq 0$
$g_2(\vec{x}) = 2x_1 + 2x_3 + x_{10} + x_{12} - 10 \leq 0$
$g_3(\vec{x}) = 2x_2 + 2x_3 + x_{11} + x_{12} - 10 \leq 0$
$g_4(\vec{x}) = -8x_1 + x_{10} \leq 0$
$g_5(\vec{x}) = -8x_2 + x_{11} \leq 0$
$g_6(\vec{x}) = -8x_3 + x_{12} \leq 0$
$g_7(\vec{x}) = -2x_4 - x_5 + x_{10} \leq 0$
$g_8(\vec{x}) = -2x_6 - x_7 + x_{11} \leq 0$
$g_9(\vec{x}) = -2x_8 - x_9 + x_{12} \leq 0$

  where the bounds are $0 \leq x_i \leq 1$ $(i = 1, \ldots, 9)$, $0 \leq x_i \leq 100$ $(i = 10, 11, 12)$ and $0 \leq x_{13} \leq 1$. The global optimum is located at $x^* = (1,1,1,1,1,1,1,1,1,3,3,3,1)$ where $f(x^*) = -15$. Constraints $g_1$, $g_2$, $g_3$, $g_4$, $g_5$ and $g_6$ are active.

- **g02**:
Maximize: $f(\vec{x}) = \left| \frac{\sum_{i=1}^{n} \cos^4(x_i) - 2\prod_{i=1}^{n} \cos^2(x_i)}{\sqrt{\sum_{i=1}^{n} i x_i^2}} \right|$
subject to:
$g_1(\vec{x}) = 0.75 - \prod_{i=1}^{n} x_i \leq 0$
$g_2(\vec{x}) = \sum_{i=1}^{n} x_i - 7.5n \leq 0$

  where $n = 20$ and $0 \leq x_i \leq 10$ $(i = 1, \ldots, n)$. The global maximum is unknown; the best reported solution is (Runarsson and Yao 2000): $f(x^*) = 0.803619$. Constraint $g_1$ is close to being active $(g_1 = -10^{-8})$.

- **g03**:
Maximize: $f(\vec{x}) = (\sqrt{n})^n \prod_{i=1}^{n} x_i$

  subject to:
$h(\vec{x}) = \sum_{i=1}^{n} x_i^2 - 1 = 0$

  where $n = 10$ and $0 \leq x_i \leq 1$ $(i = 1, \ldots, n)$. The global maximum is located at $x_i^* = 1/\sqrt{n}$ $(i = 1, \ldots, n)$ where $f(x^*) = 1$.

- **g04**:
Minimize: $f(\vec{x}) = 5.3578547x_3^2 + 0.8356891x_1x_5 + 37.293239x_1 - 40792.141$
subject to:
$g_1(\vec{x}) = 85.334407 + 0.0056858x_2x_5 + 0.0006262x_1x_4 - 0.0022053x_3x_5 - 92 \leq 0$

$g_2(\vec{x}) = -85.334407 - 0.0056858x_2x_5 - 0.0006262x_1x_4 + 0.0022053x_3x_5 \leq 0$
$g_3(\vec{x}) = 80.51249 + 0.0071317x_2x_5 + 0.0029955x_1x_2 + 0.0021813x_3^2 - 110 \leq 0$
$g_4(\vec{x}) = -80.51249 - 0.0071317x_2x_5 - 0.0029955x_1x_2 - 0.0021813x_3^2 + 90 \leq 0$
$g_5(\vec{x}) = 9.300961 + 0.0047026x_3x_5 + 0.0012547x_1x_3 + 0.0019085x_3x_4 - 25 \leq 0$
$g_6(\vec{x}) = -9.300961 - 0.0047026x_3x_5 - 0.0012547x_1x_3 - 0.0019085x_3x_4 + 20 \leq 0$

where: $78 \leq x_1 \leq 102$, $33 \leq x_2 \leq 45$, $27 \leq x_i \leq 45$ $(i = 3, 4, 5)$. The global optimum is located at $x^* = (78, 33, 29.995256025682, 45, 36.775812905788)$ where $f(x^*) = -30665.539$. Constraints $g_1$ and $g_6$ are active.

- **g05**
  Minimize: $f(\vec{x}) = 3x_1 + 0.000001x_1^3 + 2x_2 + (0.000002/3)x_2^3$

  subject to:
  $g_1(\vec{x}) = -x_4 + x_3 - 0.55 \leq 0$
  $g_2(\vec{x}) = -x_3 + x_4 - 0.55 \leq 0$
  $h_3(\vec{x}) = 1000\sin(-x_3 - 0.25) + 1000\sin(-x_4 - 0.25) + 894.8 - x_1 = 0$
  $h_4(\vec{x}) = 1000\sin(x_3 - 0.25) + 1000\sin(x_3 - x_4 - 0.25) + 894.8 - x_2 = 0$
  $h_5(\vec{x}) = 1000\sin(x_4 - 0.25) + 1000\sin(x_4 - x_3 - 0.25) + 1294.8 = 0$

  where $0 \leq x_1 \leq 1200$, $0 \leq x_2 \leq 1200$, $-0.55 \leq x_3 \leq 0.55$, and $-0.55 \leq x_4 \leq 0.55$. The best known solution is $x^* = (679.9453, 1026.067, 0.1188764, -0.3962336)$ where $f(x^*) = 5126.4981$.

- **g06**
  Minimize: $f(\vec{x}) = (x_1 - 10)^3 + (x_2 - 20)^3$
  subject to:
  $g_1(\vec{x}) = -(x_1 - 5)^2 - (x_2 - 5)^2 + 100 \leq 0$
  $g_2(\vec{x}) = (x_1 - 6)^2 + (x_2 - 5)^2 - 82.81 \leq 0$

  where $13 \leq x_1 \leq 100$ and $0 \leq x_2 \leq 100$. The global optimum is located at $x^* = (14.095, 0.84296)$ where $f(x^*) = -6961.81388$. Both constraints are active.
- **g07**
  Minimize: $f(\vec{x}) = x_1^2 + x_2^2 + x_1x_2 - 14x_1 - 16x_2 + (x_3 - 10)^2 + 4(x_4 - 5)^2 + (x_5 - 3)^2 + 2(x_6 - 1)^2 + 5x_7^2 + 7(x_8 - 11)^2 + 2(x_9 - 10)^2 + (x_{10} - 7)^2 + 45$
  subject to:
  $g_1(\vec{x}) = -105 + 4x_1 + 5x_2 - 3x_7 + 9x_8 \leq 0$
  $g_2(\vec{x}) = 10x_1 - 8x_2 - 17x_7 + 2x_8 \leq 0$
  $g_3(\vec{x}) = -8x_1 + 2x_2 + 5x_9 - 2x_{10} - 12 \leq 0$
  $g_4(\vec{x}) = 3(x_1 - 2)^2 + 4(x_2 - 3)^2 + 2x_3^2 - 7x_4 - 120 \leq 0$

$g_5(\vec{x}) = 5x_1^2 + 8x_2 + (x_3 - 6)^2 - 2x_4 - 40 \leq 0$
$g_6(\vec{x}) = x_1^2 + 2(x_2 - 2)^2 - 2x_1x_2 + 14x_5 - 6x_6 \leq 0$
$g_7(\vec{x}) = 0.5(x_1 - 8)^2 + 2(x_2 - 4)^2 + 3x_5^2 - x_6 - 30 \leq 0$
$g_8(\vec{x}) = -3x_1 + 6x_2 + 12(x_9 - 8)^2 - 7x_{10} \leq 0$

where $-10 \leq x_i \leq 10$ ($i = 1, \ldots, 10$). The global optimum is located at $x^* = (2.171996, 2.363683, 8.773926, 5.095984, 0.9906548, 1.430574, 1.321644, 9.828726, 8.280092, 8.375927)$ where $f(x^*) = 24.3062091$. Constraints $g_1$, $g_2$, $g_3$, $g_4$, $g_5$ and $g_6$ are active.

- **g08**
  Maximize: $f(\vec{x}) = \frac{\sin^3(2\pi x_1)\sin(2\pi x_2)}{x_1^3(x_1+x_2)}$
  subject to:
  $g_1(\vec{x}) = x_1^2 - x_2 + 1 \leq 0$
  $g_2(\vec{x}) = 1 - x_1 + (x_2 - 4)^2 \leq 0$

  where $0 \leq x_1 \leq 10$ and $0 \leq x_2 \leq 10$. The global optimum is located at $x^* = (1.2279713, 4.2453733)$ where $f(x^*) = 0.095825$.

- **g09**
  Minimize: $f(\vec{x}) = (x_1 - 10)^2 + 5(x_2 - 12)^2 + x_3^4 + 3(x_4 - 11)^2 + 10x_5^6 + 7x_6^2 + x_7^4 - 4x_6x_7 - 10x_6 - 8x_7$
  subject to:

  $g_1(\vec{x}) = -127 + 2x_1^2 + 3x_2^4 + x_3 + 4x_4^2 + 5x_5 \leq 0$
  $g_2(\vec{x}) = -282 + 7x_1 + 3x_2 + 10x_3^2 + x_4 - x_5 \leq 0$
  $g_3(\vec{x}) = -196 + 23x_1 + x_2^2 + 6x_6^2 - 8x_7 \leq 0$
  $g_4(\vec{x}) = 4x_1^2 + x_2^2 - 3x_1x_2 + 2x_3^2 + 5x_6 - 11x_7 \leq 0$

  where $-10 \leq x_i \leq 10$ ($i = 1, \ldots, 7$). The global optimum is located at $x^* = (2.330499, 1.951372, -0.4775414, 4.365726, -0.6244870, 1.038131, 1.594227)$ where $f(x^*) = 680.6300573$. Two constraints are active ($g_1$ and $g_4$).

- **g10**
  Minimize: $f(\vec{x}) = x_1 + x_2 + x_3$
  subject to: $g_1(\vec{x}) = -1 + 0.0025(x_4 + x_6) \leq 0$
  $g_2(\vec{x}) = -1 + 0.0025(x_5 + x_7 - x_4) \leq 0$
  $g_3(\vec{x}) = -1 + 0.01(x_8 - x_5) \leq 0$
  $g_4(\vec{x}) = -x_1x_6 + 833.33252x_4 + 100x_1 - 83333.333 \leq 0$
  $g_5(\vec{x}) = -x_2x_7 + 1250x_5 + x_2x_4 - 1250x_4 \leq 0$
  $g_6(\vec{x}) = -x_3x_8 + 1250000 + x_3x_5 - 2500x_5 \leq 0$

where $100 \leq x_1 \leq 10000$, $1000 \leq x_i \leq 10000$, $(i = 2,3)$, $10 \leq x_i \leq 1000$, $(i = 4,\ldots,8)$. The global optimum is located at $x^* = (579.19, 1360.13, 5109.92, 182.0174, 295.5985, 217.9799, 286.40, 395.5979)$, where $f(x^*) = 7049.25$. $g_1$, $g_2$ and $g_3$ are active.

- **g11**
  Minimize: $f(\vec{x}) = x_1^2 + (x_2 - 1)^2$
  subject to:
  $h(\vec{x}) = x_2 - x_1^2 = 0$

  where: $-1 \leq x_1 \leq 1$, $-1 \leq x_2 \leq 1$. The global optimum is located at $x^* = (\pm 1/\sqrt{2}, 1/2)$ where $f(x^*) = 0.75$.

- **g12**
  Maximize: $f(\vec{x}) = \frac{100-(x_1-5)^2-(x_2-5)^2-(x_3-5)^2}{100}$
  subject to:
  $g_1(\vec{x}) = (x_1 - p)^2 + (x_2 - q)^2 + (x_3 - r)^2 - 0.0625 \leq 0$

  where $0 \leq x_i \leq 10$ $(i = 1,2,3)$ and $p, q, r = 1, 2, \ldots, 9$. The feasible region of the search space consists of $9^3$ disjointed spheres. A point $(x_1, x_2, x_3)$ is feasible if and only if there exist $p, q, r$ such the above inequality (5) holds. The global optimum is located at $x^* = (5,5,5)$ where $f(x^*) = 1$.

- **g13**
  Minimize: $f(\vec{x}) = e^{x_1 x_2 x_3 x_4 x_5}$
  subject to:
  $g_1(\vec{x}) = x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_5^2 - 10 = 0$
  $g_2(\vec{x}) = x_2 x_3 - 5 x_4 x_5 = 0$
  $g_3(\vec{x}) = x_1^3 + x_2^3 + 1 = 0$

  where $-2.3 \leq x_i \leq 2.3$ $(i = 1,2)$ and $-3.2 \leq x_i \leq 3.2$ $(i = 3,4,5)$. The global optimum is located at $x^* = (-1.717143, 1.595709, 1.827247, -0.7636413, -0.763645)$ where $f(x^*) = 0.0539498$.

- **Design of a Welded Beam**
  '[Insert figure 1 of the appendix about here]'

A welded beam is designed for minimum cost subject to constraints on shear stress ($\tau$), bending stress in the beam ($\sigma$), buckling load on the bar

($P_c$), end deflection of the beam ($\delta$), and side constraints (Rao 1996). There are four design variables as shown in Figure 5 (Rao 1996): $h$ ($x_1$), $l$ ($x_2$), $t$ ($x_3$) and $b$ ($x_4$). The problem can be stated as follows:
Minimize: $f(\vec{x}) = 1.10471x_1^2x_2 + 0.04811x_3x_4(14.0 + x_2)$
Subject to:
$g_1(\vec{x}) = \tau(\vec{x}) - \tau_{max} \leq 0$
$g_2(\vec{x}) = \sigma(\vec{x}) - \sigma_{max} \leq 0$
$g_3(\vec{x}) = x_1 - x_4 \leq 0$
$g_4(\vec{x}) = 0.10471x_1^2 + 0.04811x_3x_4(14.0 + x_2) - 5.0 \leq 0$
$g_5(\vec{x}) = 0.125 - x_1 \leq 0$
$g_6(\vec{x}) = \delta(\vec{x}) - \delta_{max} \leq 0$
$g_7(\vec{x}) = P - P_c(\vec{x}) \leq 0$

where $\tau(\vec{x}) = \sqrt{(\tau')^2 + 2\tau'\tau''\frac{x_2}{2R} + (\tau'')^2}$
$\tau' = \frac{P}{\sqrt{2}x_1x_2}, \tau'' = \frac{MR}{J}, M = P\left(L + \frac{x_2}{2}\right)$
$R = \sqrt{\frac{x_2^2}{4} + \left(\frac{x_1+x_3}{2}\right)^2}$
$J = 2\left\{\sqrt{2}x_1x_2\left[\frac{x_2^2}{12} + \left(\frac{x_1+x_3}{2}\right)^2\right]\right\}$
$\sigma(\vec{x}) = \frac{6PL}{x_4x_3^2}, \delta(\mathbf{X}) = \frac{4PL^3}{Ex_3^3x_4}$
$P_c(\vec{x}) = \frac{4.013E\sqrt{\frac{x_3^2x_4^6}{36}}}{L^2}\left(1 - \frac{x_3}{2L}\sqrt{\frac{E}{4G}}\right)$
$P = 6000\ lb, \quad L = 14\ in, \quad E = 30 \times 10^6\ psi, \quad G = 12 \times 10^6\ psi$
$\tau_{max} = 13,600\ psi, \quad \sigma_{max} = 30,000\ psi, \quad \delta_{max} = 0.25\ in$

where $0.1 \leq x_1 \leq 2.0$, $0.1 \leq x_2 \leq 10.0$, $0.1 \leq x_3 \leq 10.0$ y $0.1 \leq x_4 \leq 2.0$.

- **Design of a Pressure Vessel**
  '[Insert figure 2 of the appendix about here]'

A cylindrical vessel is capped at both ends by hemispherical heads as shown in Figure 5. The objective is to minimize the total cost, including the cost of the material, forming and welding. There are four design variables: $T_s$ (thickness of the shell), $T_h$ (thickness of the head), $R$ (inner radius) and $L$ (length of the cylindrical section of the vessel, not including the head). $T_s$ and $T_h$ are integer multiples of 0.0625 inch, which are the available thicknesses of rolled steel plates, and $R$ and $L$ are continuous. Using the same notation given by Kannan and Kramer (1994), the problem can be stated as follows:
Minimize : $f(\vec{x}) = 0.6224x_1x_3x_4 + 1.7781x_2x_3^2 + 3.1661x_1^2x_4 + 19.84x_1^2x_3$

Subject to :
$g_1(\vec{x}) = -x_1 + 0.0193x_3 \leq 0$
$g_2(\vec{x}) = -x_2 + 0.00954x_3 \leq 0$
$g_3(\vec{x}) = -\pi x_3^2 x_4 - \frac{4}{3}\pi x_3^3 + 1,296,000 \leq 0$
$g_4(\vec{x}) = x_4 - 240 \leq 0$

where $1 \leq x_1 \leq 99$, $1 \leq x_2 \leq 99$, $10 \leq x_3 \leq 200$ y $10 \leq x_4 \leq 200$.

- **Minimization of the Weight of a Tension/Compression Spring**
'[Insert figure 3 of the appendix about here]'

This problem was described by Arora (1989) and Belegundu (1982), and it consists of minimizing the weight of a tension/compression spring (see Figure 5) subject to constraints on minimum deflection, shear stress, surge frequency, limits on outside diameter and on design variables. The design variables are the mean coil diameter $D$ ($x_2$), the wire diameter $d$ ($x_1$) and the number of active coils $N$ ($x_3$). Formally, the problem can be expressed as:
Minimize: $(N+2)Dd^2$
Subject to:
$g_1(\vec{x}) = 1 - \frac{D^3 N}{71785 d^4} \leq 0$
$g_2(\vec{x}) = \frac{4D^2 - dD}{12566(Dd^3 - d^4)} + \frac{1}{5108 d^2} - 1 \leq 0$
$g_3(\vec{x}) = 1 - \frac{140.45 d}{D^2 N} \leq 0$
$g_4(\vec{x}) = \frac{D+d}{1.5} - 1 \leq 0$

where $0.05 \leq x_1 \leq 2$, $0.25 \leq x_2 \leq 1.3$ y $2 \leq x_3 \leq 15$.

- **Problem 4: (Minimization of the Weight of a Speed Reducer)** The weight of the speed reducer is to be minimized subject to constraints on bending stress of the gear teeth, surfaces stress, transverse deflections of the shafts and stresses in the shafts. The variables $x_1, x_2, \cdots, x_7$ are the face width, module of teeth, number of teeth in the pinion, length of the first shaft between bearings, length of the second shaft between bearings and the diameter of the first and second shafts. The third variable is integer, the rest of them are continuous.
Minimize : $f(\vec{x}) = 0.7854x_1x_2^2(3.3333x_3^2 + 14.9334x_3 - 43.0934) - 1.508x_1(x_6^2 + x_7^2) + 7.4777(x_6^3 + x_7^3) + 0.7854(x_4x_6^2 + x_5x_7^2)$
Subject to :
$g_1(\vec{x}) = \frac{27}{x_1 x_2^2 x_3} - 1 \leq 0$
$g_2(\vec{x}) = \frac{397.5}{x_1 x_2^2 x_3^2} - 1 \leq 0$

$g_3(\vec{x}) = \frac{1.93x_4^3}{x_2x_3x_6^4} - 1 \leq 0$
$g_4(\vec{x}) = \frac{1.93x_5^3}{x_2x_3x_7^4} - 1 \leq 0$
$g_5(\vec{x}) = \frac{\left(\left(\frac{745x_4}{x_2x_3}\right)^2 + 16.9\times10^6\right)^{1/2}}{110.0x_6^3} - 1 \leq 0$
$g_6(\vec{x}) = \frac{\left(\left(\frac{745x_5}{x_2x_3}\right)^2 + 157.5\times10^6\right)^{1/2}}{85.0x_7^3} - 1 \leq 0$
$g_7(\vec{x}) = \frac{x_2x_3}{40} - 1 \leq 0$
$g_8(\vec{x}) = \frac{5x_2}{x_1} - 1 \leq 0$
$g_9(\vec{x}) = \frac{x_1}{12x_2} - 1 \leq 0$
$g_{10}(\vec{x}) = \frac{1.5x_6+1.9}{x_4} - 1 \leq 0$
$g_{11}(\vec{x}) = \frac{1.1x_7+1.9}{x_5} - 1 \leq 0$

where $2.6 \leq x_1 \leq 3.6$, $0.7 \leq x_2 \leq 0.8$, $17 \leq x_3 \leq 28$, $7.3 \leq x_4 \leq 8.3$, $7.8 \leq x_5 \leq 8.3$, $2.9 \leq x_6 \leq 3.9$ and $5.0 \leq x_7 \leq 5.5$.

## REFERENCES


Arora, J. S. (1989), *Introduction to Optimum Design*, McGraw-Hill, New York.

Bäck, T. (1996), *Evolutionary Algorithms in Theory and Practice*, Oxford University Press, New York.

Belegundu, A. D. (1982), A Study of Mathematical Programming Methods for Structural Optimization, Department of civil and environmental engineering, University of Iowa, Iowa.

Coello, C. A. C. (2002), 'Theoretical and Numerical Constraint Handling Techniques used with Evolutionary Algorithms: A Survey of the State of the Art', *Computer Methods in Applied Mechanics and Engineering* **191**(11-12), 1245–1287.

Deb, K. (2000), 'An Efficient Constraint Handling Method for Genetic Algorithms', *Computer Methods in Applied Mechanics and Engineering* **186**(2/4), 311–338.

Hadj-Alouane, A. B. and Bean, J. C. (1997), 'A Genetic Algorithm for the Multiple-Choice Integer Program', *Operations Research* **45**, 92–101.

He, S., Prempain, E. and Q.H.Wu (2004), 'An Improved Particle Swarm Optimizer for Mechanical Design Optimization Problems', *Engineering Optimization* **36**(5), 585–605.

Hoffmeister, F. and Sprave, J. (1996), Problem-independent handling of constraints by use of metric penalty functions, *in* L. J. Fogel, P. J. Angeline and T. Bäck, eds, 'Proceedings of the Fifth Annual Conference on Evolutionary Programming (EP'96)', The MIT Press, San Diego, California, pp. 289–294.

Joines, J. and Houck, C. (1994), On the use of non-stationary penalty functions to solve nonlinear constrained optimization problems with GAs, *in* D. Fogel, ed., 'Proceedings of the first IEEE Conference on Evolutionary Computation', IEEE Press, Orlando, Florida, pp. 579–584.

Kannan, B. K. and Kramer, S. N. (1994), 'An Augmented Lagrange Multiplier Based Method for Mixed Integer Discrete Continuous Optimization and Its Applications to Mechanical Design', *Journal of Mechanical Design. Transactions of the ASME* **116**, 318–320.

Mezura-Montes, E. and Coello Coello, C. A. (2005), 'A Simple Multimembered Evolution Strategy to Solve Constrained Optimization Problems', *IEEE Transactions on Evolutionary Computation* **9**(1), 1–17.

Michalewicz, Z. and Schoenauer, M. (1996), 'Evolutionary Algorithms for Constrained Parameter Optimization Problems', *Evolutionary Computation* **4**(1), 1–32.

Price, K. V. (1999), An Introduction to Differential Evolution, *in* D. Corne, M. Dorigo and F. Glover, eds, 'New Ideas in Optimization', Mc Graw-Hill, UK, pp. 79–108.

Rao, S. S. (1996), *Engineering Optimization*, third edn, John Wiley and Sons.

Ray, T. and Liew, K. (2003), 'Society and Civilization: An Optimization Algorithm Based on the Simulation of Social Behavior', *IEEE Transactions on Evolutionary Computation* **7**(4), 386–396.

Runarsson, T. P. and Yao, X. (2000), 'Stochastic Ranking for Constrained Evolutionary Optimization', *IEEE Transactions on Evolutionary Computation* **4**(3), 284–294.

Schwefel, H.-P. (1981), *Numerical Optimization of Computer Models*, John Wiley & Sons, Great Britain.

Smith, A. E. and Coit, D. W. (1997), Constraint Handling Techniques—Penalty Functions, *in* T. Bäck, D. B. Fogel and Z. Michalewicz, eds, 'Handbook of Evolutionary Computation', Oxford University Press and Institute of Physics Publishing, chapter C 5.2.

Storn, R. (1999), 'System Design by Constraint Adaptation and Differential Evolution', *IEEE Transactions on Evolutionary Computation* **3**(1), 22–34.

Tessema, B. and Yen, G. G. (2006), A self adaptive penalty function based algorithm for constrained optimization, *in* 'Proceedings of the IEEE Congress on Evolutionary Computation 2006 (CEC'2006)', Vancouver, Canada, IEEE Service Center, Piscataway, New Jersey, pp. 950–957.

Figure 1: Our algorithm. The steps modified with respect to the original DE algorithm are marked with an arrow. $randint(min, max)$ returns an integer value between $min$ and $max$. rand$[0, 1)$ returns a real number between 0 and 1. Both functions adopt a uniform probability distribution. flip$(W)$ returns 1 with probability $W$.

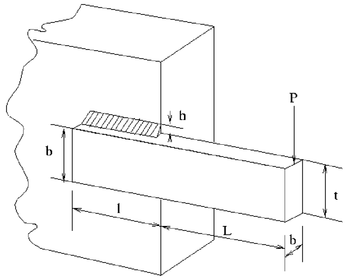
Appendix

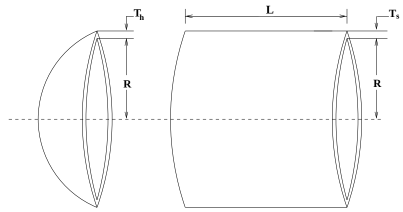
Figure 1: The welded beam used for the first engineering design problem

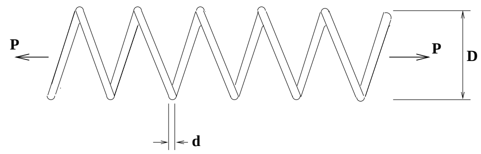
Figure 2: Center and end section of the pressure vessel used for the second engineering design problem.

Figure 3: Tension/compression spring used for the third engineering design problem.

```
    Begin
        G=0
        Create a random initial population x⃗^i_G ∀i, i = 1, ..., NP
        Evaluate f(x⃗^i_G) ∀i, i = 1, ..., NP
        For G=1 to MAX_GENERATIONS Do
            F=rand[0.3,0.9]
            For i=1 to NP Do
⇒               For k=1 to n_o Do
                    Select randomly r_1 ≠ r_2 ≠ r_3 ≠ i
                    j_rand = randint(1, D)
                    For j=1 to n_o Do
                        If (rand_j[0,1) < CR or j = j_rand) Then
                            child_j = x^{r3}_{j,G} + F(x^{r1}_{j,G} − x^{r2}_{j,G})
                        Else
                            child_j = x^i_{j,G}
                        End If
                    End For
                    If k > 1 Then
⇒                       If (child is better than u⃗^i_{G+1}
                         (based on the three selection criteria)) Then
                            u⃗^i_{G+1}=child
                        End If
                    Else
                        u⃗^i_{G+1}=child
                End For
⇒               If flip(S_r) Then
⇒                   If (f(u⃗^i_{G+1}) ≤ f(x⃗^i_G)) Then
                        x⃗^i_{G+1} = u⃗^i_{G+1}
                    Else
                        x⃗^i_{G+1} = x⃗^i_G
                    End If
                Else
⇒                   If (u⃗^i_{G+1} is better than x⃗^i_G
                     (based on the three selection criteria)) Then
                        x⃗^i_{G+1} = u⃗^i_{G+1}
                    Else
                        x⃗^i_{G+1} = x⃗^i_G
                    End If
                End If
            End For
            G = G + 1
        End For
    End
```


P
h
b
t
l
L
b


T_h
L
T_s
R
R


P
P
D
d

| Problem | n | Type of function | $\rho$ | LI | NI | LE | NE |
|---|---|---|---|---|---|---|---|
| g01 | 13 | quadratic | 0.0003% | 9 | 0 | 0 | 0 |
| g02 | 20 | nonlinear | 99.9973% | 2 | 0 | 0 | 0 |
| g03 | 10 | nonlinear | 0.0026% | 0 | 0 | 0 | 1 |
| g04 | 5 | quadratic | 27.0079% | 4 | 2 | 0 | 0 |
| g05 | 4 | nonlinear | 0.0000% | 2 | 0 | 0 | 3 |
| g06 | 2 | nonlinear | 0.0057% | 0 | 2 | 0 | 0 |
| g07 | 10 | quadratic | 0.0000% | 3 | 5 | 0 | 0 |
| g08 | 2 | nonlinear | 0.8581% | 0 | 2 | 0 | 0 |
| g09 | 7 | nonlinear | 0.5199% | 0 | 4 | 0 | 0 |
| g10 | 8 | linear | 0.0020% | 6 | 0 | 0 | 0 |
| g11 | 2 | quadratic | 0.0973% | 0 | 0 | 0 | 1 |
| g12 | 3 | quadratic | 4.7697% | 0 | $9^3$ | 0 | 0 |
| g13 | 5 | nonlinear | 0.0000% | 0 | 0 | 1 | 2 |

Table 1. Main features for each benchmark problem used in the first set of experiments. $\rho$ is the estimated size of the feasible region with respect to the whole search space (Michalewicz and Schoenauer 1996), $n$ is the number of decision variables, LI is the number of linear inequality constraints , NI the number of nonlinear inequality constraints, LE is the number of linear equality constraints and NE is the number of nonlinear equality constraints.

| Problem & Best Known Sol. | Stats. | Tessema Yen (2006) | Mezura & Coello (2005) | Traditional DE | Our approach |
|---|---|---|---|---|---|
| g01 | Best | **-15.000** | **-15.000** | - | **-15.000** |
| -15.000 | Mean | 14.552 | **-15.000** | - | **-15.000** |
| | St. Dev. | 0.7 | **0** | - | **0** |
| g02 | Best | 0.803202 | 0.803601 | 0.624750 | **0.803610** |
| 0.803619 | Mean | 0.755798 | 0.785238 | 0.512637 | **0.796437** |
| | St. Dev. | 0.13321 | 0.0167 | 0.049706 | **0.008929** |
| g03 | Best | **1.000** | **1.000** | - | **1.000** |
| 1.000 | Mean | 0.964 | **1.000** | - | **1.000** |
| | St. Dev. | 0.301 | **0.000209** | - | **0** |
| g04 | Best | -30665.401 | **-30665.539** | -297979.515 | **-30665.539** |
| -30665.539 | Mean | -306659.221 | **-30665.539** | -30135.475 | **-30665.539** |
| | St. Dev. | 2.043 | **0** | 140.167 | **0** |
| g05 | Best | 5126.907 | 5241.599 | - | **5126.497** |
| 5126.498 | Mean | 5214.232 | 5174.492 | - | **5126.497** |
| | St. Dev. | 247.476 | 50.06 | - | **0** |
| g06 | Best | -6961.046 | **-6961.814** | -6127.775 | **-6961.814** |
| -6961.814 | Mean | -6953.061 | -6961.284 | -6741.669 | **-6961.814** |
| | St. Dev. | 5.876 | 1.85 | 186.275 | **0** |
| g07 | Best | 24.838 | 24.327 | 58.184 | **24.306** |
| 24.306 | Mean | 27.328 | 24.475 | 39.135 | **24.306** |
| | St. Dev. | 2.172 | 0.132 | 5.672 | **0** |
| g08 | Best | **0.095825** | **0.095825** | **0.095825** | **0.095825** |
| 0.095825 | Mean | 0.095635 | **0.095825** | **0.095825** | **0.095825** |
| | St. Dev. | 0.001055 | **0** | **0** | **0** |
| g09 | Best | 680.773 | **680.632** | 762.947 | **680.630** |
| 680.63 | Mean | 681.246 | 680.643 | 707.190 | **680.630** |
| | St. Dev. | 0.332 | 0.0155 | 15.936 | **0** |
| g10 | Best | 7069.981 | 7051.903 | - | **7049.25** |
| 7049.25 | Mean | 7238.964 | 7253.047 | - | **7049.35** |
| | St. Dev. | 137.773 | 136.02 | - | **0.1945110** |
| g11 | Best | 0.749 | **0.75** | 0.7483 | **0.75** |
| 0.75 | Mean | 0.751 | **0.75** | 0.8376 | **0.75** |
| | St. Dev. | 0.002 | **0.000152** | 0.0738 | **0** |
| g12 | Best | **1.000** | **1.000** | 1.000 | **1.000** |
| 1.000 | Mean | 0.99994 | **1.000** | 1.000 | **1.000** |
| | St. Dev. | 0.000141 | **0** | 0 | **0** |
| g13 | Best | **0.053941** | 0.053986 | - | 0.053942 |
| 0.0539498 | Mean | 0.28627 | 0.166385 | - | **0.053942** |
| | St. Dev. | 0.275463 | 0.177 | - | **0** |

Table 2. Comparison of statistical results on 13 benchmark problems obtained by some state-of-the-art approaches, a traditional DE and our proposed approach. "-" means no feasible solutions were found.

| **Problem** | **Best Known sol.** | **Confidence intervals** |
|---|---|---|
| | | **Our approach** |
| g01 | -15.000 | [**-15.000, -15.000**] |
| g02 | 0.803619 | [ 0.7925494,0.7988851] |
| g03 | 1.000 | [**1.000,1000**] |
| g04 | -30665.539 | [**-30665.539,-30665.539**] |
| g05 | 5126.498 | [5126.497,5126.497] |
| g06 | -6961.814 | [**-6961.814,-6961.814**] |
| g07 | 24.306 | [**24.306,24.306**] |
| g08 | 0.095825 | [0.095826,0.095826] |
| g09 | 680.63 | [**680.630,680.63**] |
| g10 | 7049.25 | [7049.289,7049.401] |
| g11 | 0.75 | [**0.75,0.75**] |
| g12 | 1.000 | [**1.000,1.000**] |
| g13 | 0.053949 | [0.0539415,0.0539415] |

Table 3. 95%-confidence intervals for the mean statistic per test problem obtained by our approach.

| Problem | n | Type of function | $\rho$ | LI | NI | LE | NE |
|---|---|---|---|---|---|---|---|
| beam | 4 | quadratic | 2.6859% | 6 | 1 | 0 | 0 |
| vessel | 4 | quadratic | 39.6762% | 3 | 1 | 0 | 0 |
| spring | 3 | quadratic | 0.7537% | 1 | 3 | 0 | 0 |
| truss | 10 | nonlinear | 46.8070% | 0 | 22 | 0 | 0 |

Table 4. Main features for each engineering design problem used in the second set of experiments. $\rho$ is the estimated size of the feasible region with respect to the whole search space (Michalewicz and Schoenauer 1996), $n$ is the number of decision variables, LI is the number of linear inequality constraints , NI the number of nonlinear inequality constraints, LE is the number of linear equality constraints and NE is the number of nonlinear equality constraints.

| | a) Welded beam design | | | | |
|---|---|---|---|---|---|
| | Death Penalty | Static | Dynamic *(28) | Adaptive *(27) | Our approach |
| Best | 1.739736 | 1.792248 | 1.781232 | 1.792266 | **1.724852** |
| Mean | 2.104756 | 2.023434 | 2.138872 | 2.164542 | **1.724853** |
| Worst | 2.803005 | 2.739448 | 2.904370 | 3.018553 | **1.724854** |
| St. Dev | 3.0E-1 | 2.2E-1 | 2.8E-1 | 3.4E-1 | **1.0E-15** |

| | b) Pressure vessel design | | | | |
|---|---|---|---|---|---|
| | Death Penalty | Static | Dynamic *(22) | Adaptive *(21) | Our approach |
| Best | 6172.421387 | – | 6162.862793 | 6292.51022 | **6059.701660** |
| Mean | 7417.028727 | – | 7042.828564 | 7703.780354 | **6059.701660** |
| Worst | 10477.677734 | – | 7798.198242 | 10830.894278 | **6059.701660** |
| St. Dev | 9.6E+2 | – | 5.3E+2 | 1.6E+3 | **1.0E-12** |

| | c) Ten./Comp. spring design | | | | |
|---|---|---|---|---|---|
| | Death Penalty | Static | Dynamic | Adaptive | Our approach |
| Best | 0.012719 | 0.012753 | 0.012702 | 0.012692 | **0.012665** |
| Mean | 0.014665 | 0.014636 | 0.013998 | 0.014002 | **0.012666** |
| Worst | 0.018139 | 0.018918 | 0.017044 | 0.016661 | **0.012674** |
| St. Dev | 1.4E-3 | 1.6E-3 | 1.0E-3 | 1.2E-4 | **2.0E-6** |

| | d) Speed reducer design | | | | |
|---|---|---|---|---|---|
| | Death Penalty | Static | Dynamic | Adaptive | Our approach |
| Best | – | – | – | – | **2996.356689** |
| Mean | – | – | – | – | **2996.367220** |
| Worst | – | – | – | – | **2996.390137** |
| St. Dev | – | – | – | – | **8.2E-3** |

Table 5. Comparison of statistical results for the penalty-based approaches and our approach. "-" means no feasible solutions found. A result in **boldface** means a better result. "$*(X)$" means that only in $X$ runs (out of 30) feasible solutions were found.

| Problem | Stats | Ray & Liew (2003) | He et al. (2004) | Our approach | Traditional DE |
|---|---|---|---|---|---|
| Welded beam | best | 2.385435 | 2.380957 | **1.724852** | 1.904312 |
| | mean | 3.255137 | 2.381932 | **1.724853** | 2.237370 |
| | St. Dev | 9.6E-1 | 5.2E-3 | **1.0E-15** | 2.3E-1 |
| | Evals | 33000 | 30000 | **24000** | 24000 |
| Pressure vessel | best | 6171.00 | 6059.7143 | **6059.701660** | 7247.938477 |
| | mean | 6335.05 | 6289.92881 | **6059.701660** | 8854.318896 |
| | St. Dev | NA | 3.1E+2 | **1.0E-12** | 1.3E+3 |
| | Evals | **20000** | 30000 | 24000 | 24000 |
| Ten/Comp. spring | best | 0.012669 | **0.012665** | **0.012665** | 0.012851 |
| | mean | 0.012923 | 0.012702 | **0.012666** | 0.014119 |
| | St. Dev | 5.9E-4 | 4.1E-5 | **2.0E-6** | 1.0E-3 |
| | Evals. | 25167 | **15000** | 24000 | 24000 |
| Speed reducer | best | **2994.744241** | NA | 2996.356689 | 3064.211426 |
| | mean | 3001.758264 | NA | **2996.367220** | 3244.569010 |
| | St. Dev | 4.0E+0 | NA | **8.2E-3** | 2.0E+2 |
| | Evals. | 54456 | NA | **24000** | 24000 |

Table 6. Comparison of results with respect to two state-of-the-art approaches and a traditional DE approach. A result in **boldface** means a better result. "NA" means not available.

| Welded beam | Problem 1 | | |
|---|---|---|---|
| | Ray & Liew (2003) | He et al. (2004) | Our Approach |
| $x_1$ | 0.244438 | 0.244369 | 0.205730 |
| $x_2$ | 6.237967 | 6.217520 | 3.470489 |
| $x_3$ | 8.288576 | 8.291471 | 9.036624 |
| $x_4$ | 0.244566 | 0.244369 | 0.205730 |
| $g_1(x)$ | −5760.110471 | −5741.176933 | −0.000335 |
| $g_2(x)$ | −3.245428 | 0.000001 | −0.000753 |
| $g_3(x)$ | −0.000128 | 0.000000 | −0.000000 |
| $g_4(x)$ | −3.020055 | −3.022955 | −3.432984 |
| $g_5(x)$ | −0.119438 | −0.119369 | −0.080730 |
| $g_6(x)$ | −0.234237 | −0.234241 | −0.235540 |
| $g_7(x)$ | −13.079305 | −0.000309 | −0.000882 |
| **f(x)** | **2.38119** | **2.380956** | **1.724852** |

Table 7. Details of the best solution found by each compared state-of-the-art technique and our approach for the welded beam design problem.

| **Pressure vessel** | **Problem 2** | | |
|---|---|---|---|
| | **Ray & Liew (2003)** | **He et al. (2004)** | **Our Approach** |
| $x_1$ | 0.8125 | 0.8125 | 0.8125 |
| $x_2$ | 0.4375 | 0.4375 | 0.4375 |
| $x_3$ | 41.9768 | 42.098446 | 42.098446 |
| $x_4$ | 182.2845 | 176.636052 | 176.636047 |
| $g_1(x)$ | $-0.0023$ | $-0.000000$ | 0.000000 |
| $g_2(x)$ | $-0.0370$ | $-0.035881$ | $-0.035881$ |
| $g_3(x)$ | $-23420.5966$ | $-0.000000$ | $-0.000002$ |
| $g_4(x)$ | $-57.7155$ | $-63.363948$ | $-63.363949$ |
| $\mathbf{f(x)}$ | **6171.0** | **6059.7143** | **6059.701660** |

Table 8. Details of the best solution found by each compared state-of-the-art technique and our approach for the pressure vessel design problem.

| **Ten./Comp. spring** | **Problem 3** | | |
|---|---|---|---|
| | **Ray & Liew (2003)** | **He et al. (2004)** | **Our Approach** |
| $x_1$ | 0.0521602 | 0.051690 | 0.051688 |
| $x_2$ | 0.368159 | 0.356750 | 0.356692 |
| $x_3$ | 10.648442 | 11.287126 | 11.290483 |
| $g_1(x)$ | $-0.000000$ | $-0.000000$ | $-0.000000$ |
| $g_2(x)$ | $-0.000000$ | 0.000000 | $-0.000000$ |
| $g_3(x)$ | $-4.075805$ | $-4.053827$ | $-0.727747$ |
| $g_4(x)$ | $-0.719787$ | $-0.727706$ | $-4.053734$ |
| $\mathbf{f(x)}$ | **0.012669** | **0.012665** | **0.012665** |

Table 9. Details of the best solution found by each compared state-of-the-art technique and our approach for the tension/compression spring design problem.

| **Speed reducer** | **Problem 4** | |
|---|---|---|
| | **Ray & Liew (Ray and Liew 2003)** | **Our approach** |
| $x_1$ | 3.500000 | 3.500010 |
| $x_2$ | 0.700000 | 0.700000 |
| $x_3$ | 17 | 17 |
| $x_4$ | 7.327602 | 7.300156 |
| $x_5$ | 7.715321 | 7.800027 |
| $x_6$ | 3.350267 | 3.350221 |
| $x_7$ | 5.286655 | 5.286685 |
| $g_1(x)$ | NA | $-0.073918$ |
| $g_2(x)$ | NA | $-0.198001$ |
| $g_3(x)$ | NA | $-0.499144$ |
| $g_4(x)$ | NA | $-0.901471$ |
| $g_5(x)$ | NA | $-0.000005$ |
| $g_6(x)$ | NA | $-0.000001$ |
| $g_7(x)$ | NA | $-0.702500$ |
| $g_8(x)$ | NA | $-0.000003$ |
| $g_9(x)$ | NA | $-0.583332$ |
| $g_{10}(x)$ | NA | $-0.051345$ |
| $g_{11}(x)$ | NA | $-0.010856$ |
| $\mathbf{f(x)}$ | **2994.744241** | **2996.356689** |

Table 10. Details of the best solution found by each compared state-of-the-art technique and our approach for the speed reducer design problem.