Grouping-based Evolutionary Algorithm: Seeking Balance Between Feasible and Infeasible Individuals of Constrained Optimization Problems

Ming Yuchi
Dept. of Electrical Engineering and Computer Science
373-1, Guseong-dong, Yuseong-gu
Daejeon 305-701, Republic of Korea
Email: ycm@rit.kaist.ac.kr

Jong-Hwan Kim
Dept. of Electrical Engineering and Computer Science
373-1, Guseong-dong, Yuseong-gu
Daejeon 305-701, Republic of Korea
Email: johkim@rit.kaist.ac.kr

Abstract—Most of the optimization problems in the real world have constraints. In recent years, evolutionary algorithms caught a lot of researchers’ attention for solving constrained optimization problems. Infeasible individuals are often underrated by most of the current evolutionary algorithms when evolutionary algorithms are used for solving constraint optimization problems. This paper proposes a novel approach to balance the feasible and infeasible individuals. Feasible and infeasible individuals are divided into two groups: feasible group and infeasible group. The evaluation and ranking of these two groups are performed separately. Parents for reproduction are selected from the two groups by a novel parent selection method. Objective function and bubble sort method are selected as the fitness function and ranking method for the feasible group. One existing evolutionary algorithms are used for solving constraint optimization problems. Infeasible individuals are often underrated by most of the current evolutionary algorithms when evolutionary algorithms are used for solving constraint optimization problems. The inequalities $g_k(\vec{x}) \leq 0$, $k = 1, \ldots, l$, and $h_k(\vec{x}) = 0$, $k = l + 1, \ldots, m$. The problem can be stated as follows:

$\text{Minimize } f(\vec{x}) = 1.10471x_1^2 x_2 + 0.04811x_3 x_4(14 + x_2),$
subject to

\begin{align*}
g_1(\vec{x}) &= \tau(\vec{x}) - \tau_{\text{max}} \leq 0, \\
g_2(\vec{x}) &= \sigma(\vec{x}) - \sigma_{\text{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 + x_2) - 5 \leq 0, \\
g_5(\vec{x}) &= 0.125 - x_1 \leq 0, \\
g_6(\vec{x}) &= \delta(\vec{x}) - \delta_{\text{max}} \leq 0, \\
g_7(\vec{x}) &= P - P_c(\vec{x}) \leq 0,
\end{align*}

and bounds

\begin{align*}
0.1 &\leq x_1 \leq 2.0, 0.1 \leq x_2 \leq 2, \\
0.1 &\leq x_3 \leq 10, 0.1 \leq x_4 \leq 2,
\end{align*}

where

\begin{align*}
\tau &= \sqrt{\frac{P}{2x_1x_2} + \frac{\tau''}{2R}} + (\tau'')^2, \\
\tau' &= \frac{P}{\sqrt{2x_1x_2}}, \tau'' = \frac{MR}{J}, M = P(L + x_2), \\
R &= \sqrt{\frac{x_2^2}{4} + \frac{(x_1 + x_4)^2}{2}}, \\
J &= 2 \left\{ \sqrt{2x_1x_2} \left[ \frac{x_2^2}{2} + \frac{(x_1 + x_4)^2}{2} \right] \right\}, \\
\sigma(\vec{x}) &= \frac{6PL}{x_4x_3^3}, \delta(\vec{x}) = \frac{4PL^3}{Ex_3x_4}, \\
P_c(\vec{x}) &= 4.013E \frac{\sqrt{x_1^2 + x_4^2}}{L^2} \left( 1 - \frac{x_3}{2L} \sqrt{\frac{E}{4G}} \right),
\end{align*}

\begin{align*}
P &= 6000\text{lb}, L = 14\text{in}, E = 30 \times 10^6\text{psi}, G = 12 \times 10^6\text{psi}, \\
\tau_{\text{max}} &= 13, 600\text{psi}, \sigma_{\text{max}} = 30, 000\text{psi}, \delta_{\text{max}} = 0.25\text{in}.
\end{align*}

Due to the simplicity of this problem, it was not used as test problem in this paper.

B. Evolutionary Algorithms for Constrained Optimization Problems

Constrained optimization problem \((P)\), in general, is intractable. If the objective function and constraint functions are arbitrary, then there is little choice apart from methods based on exhaustive search. In general, it is impossible to develop a deterministic method for \(P\) in the global optimization category, which would be better than the exhaustive search. As stated by Gregory [5]:

It’s unrealistic to expect to find one general nonlinear programming code that’s going to work for every kind of nonlinear model. Instead, you should try to select a code that fits the problem you are solving. If your problem doesn’t fit in any category except ‘general’ or if you insist on a globally optimal solution (except when there is no chance of encountering multiple local optima), you should be prepared to have to use a method that boils down to exhaustive search, i.e. you have an intractable problem.

Evolutionary algorithms are global methods, which aim at complex objective functions (e.g., non differentiable or discontinuous). It has received a lot of attention regarding their potential for solving the constrained optimization problems. In the survey paper by Michalewicz and Schoenauer [6], the existing evolutionary algorithms developed for solving the constrained optimization problems are classified into four categories: 1) methods based on preserving feasibility of solutions, 2) methods based on penalty functions, 3) methods which make a clear distinction between feasible and infeasible solutions, and 4) other hybrid methods. 11 different types of constrained optimization problems were also provided in [6] and have become a commonly used test suit for comparing the effectiveness of different evolutionary algorithms [7], [8], [9], [10], [11], [12].

In recent years, some researchers proposed several new evolutionary algorithms, thus extended the range of the approaches for constrained optimization problems. Based on the augmented Lagrangian method, Tahk and Sun [9] introduced a coevolutionary method which is taken to transform a constrained optimization problem to a zero-sum game with the saddle-point solution. Hinterding and Michalewicz [13] used preferences rather than numeric penalties and matched an infeasible parent with a parent that satisfied the constraints that it did not. Koziel and Michalewicz [11] incorporated a decoder to establish a homomorphous mapping between \(n\)-dimensional cube \([-1, 1]^n\) and the feasible part of the search space to guarantee a feasible solution. Runarsson and Yao [10] presented a simple but efficient stochastic ranking method. In their paper, the dominance of the penalty and objective functions was analyzed and a probability \(P_f\) was introduced to balance the penalty and objective functions.

When we use evolutionary algorithms to solve the constrained optimization problems, the relation between feasible and infeasible populations is a critical problem. Intuitively, one would say that feasible individuals are fitter than infeasible individuals because our goal is to find solutions that minimize the objective function and satisfy all the constraints. The feasible populations have satisfied all the constraints and the work left seems to be one minimization problem. This kind of view ignores one important thing that the evolutionary algorithm is a probabilistic method not a deterministic method. Considering the complexity and variety of different constrained optimization problems, there exists the possibility that the infeasible populations are fitter than the feasible populations at some generations during evolution computation. How to deal with the relation between the feasible and infeasible individuals and how to process the infeasible individuals become very important for solving constrained optimization problems using evolutionary techniques.

Most of the existing evolutionary algorithms for constrained optimization problems, more or less, underrated the importance of the infeasible populations. For example, nearly all of the penalty methods add some ‘penalties’ to the fitness func-
tions of the infeasible individuals [8], [14], [15]. Some other methods [16], [17] directly assume that feasible populations are always fitter than infeasible populations.

The novel evolutionary algorithm proposed in this paper is based on the idea that the infeasible populations be treated more fairly. For this purpose, the feasible and infeasible populations are divided into two groups. The evaluation and ranking of these two groups are performed separately. The selection of parents for reproduction from the feasible and infeasible populations will be adjusted by one tuning parameter $S_p$.

The rest of this paper is organized as follows. Section II gives a deeper investigation about the relation between feasible and infeasible populations. Then, the grouping-based evolutionary algorithm (GEA) is introduced. Two important aspects of this method are put forward: ranking of infeasible group and how to select parents from feasible and infeasible groups. Section III tests the GEA with one ranking methods for infeasible group: stochastic ranking. Finally, Section V concludes with some remarks and future research directions.

II. GROUPING-BASED EVOLUTIONARY ALGORITHM

A. Motivation

In evolutionary algorithms, fitness function is used to evaluate how an individual of the population fits with the optimization problem. According to the fitness values, the individuals are ranked with some ranking method. And following the selection strategy, the parents are chosen out for reproduction. Fitness function, ranking and selection decide the fate of the individual: survival, reproduction or death. Thus, how to design the fitness function and adjust the relation between ranking and selection directly influence the outcome (success or failure) of the algorithm. For constrained optimization problems, our aim is to find $\vec{x}$ satisfying all constraints (feasible solutions) and minimize $f(\vec{x})$. Intuitively, feasible population could be thought to have better fitness values than infeasible population (here ‘better’ means to have more chance to survive and reproduce). Since all the constrains have already been satisfied for the feasible population, the only aim left is to find $\vec{x}$ minimize $f(\vec{x})$. Most of the current evolutionary algorithms for constrained optimization problems follow the idea that feasible individuals are, in some sense, better than the infeasible individuals. The fitness values of the infeasible population are degraded intentionally in different kinds of strategies. However, this is not always a good idea. Let’s see one example from [7]. Figure 2 shows a search space with feasible and infeasible subspaces, $F$ and $U$, respectively. These subspaces need not to be convex or connected. The feasible part $F$ of the search space consists of four disjointed subsets. During the search process, we have to deal with various feasible and infeasible individuals. For example, at one certain stage of the evolution process, a population may contain some feasible ($c, d, f, h, k, l$) and infeasible individuals ($a, b, e, g, i, j$), while the global optimum solution is marked with ‘X’ (Figure 3).

Apparently, the infeasible individual ‘j’ is the closest to the optimal solution ‘X’, and its offspring have more chance to reach ‘X’. It is not reasonable to give feasible individual ‘d’ a better fitness value than the infeasible individual ‘j’.

Michalewicz [7] gave a deep investigation in solving the constrained optimization problem with evolutionary algorithms. 12 important questions have been addressed and discussed in detail. Nearly all of these 12 questions are talking about different aspects of the relationship between the feasible and infeasible populations.

B. GEA

In order to treat the infeasible individuals fairly as the feasible individuals, and at the same time, to evolve successfully (find one feasible solution to minimize $f(x)$), in this paper, a grouping-based EA (GEA) is proposed for constrained optimization problems. GEA focuses on building the relationship between the feasible and infeasible solutions so as to avoid the case that mentioned in Figure 3 (‘b’ has better fitness value than ‘j’). The main idea of GEA is to divide feasible and infeasible populations into two groups: feasible group and infeasible group. The evaluation and ranking of these two groups are performed in parallel and separately. In order to exchange useful information of these two groups, the best feasible and infeasible individuals are combined as a parent group and reproduce together through one predefined selection strategy. The offspring of the parent group are separated as feasible group and infeasible group again. Figure 4 shows the flowchart of GEA.

GEA provides an open structure for the designer. Different types of EA or GA can be directly adopted to operate the evolution process. The user can also design different evaluation, ranking and parents selection strategies.

1) Evaluation and Ranking: Let’s see the evaluation and ranking part first. For feasible group, the objective function can be directly selected as the fitness function. And the ‘bubble sort’ (rank from the smallest fitness value to the biggest)
is adopted in this paper for feasible group. For infeasible
group, the evaluation and rank is a little complicated. One
of the simple way is to adopt the existing methods. Here,
we will try the stochastic ranking method [10]. Stochastic
ranking method is simple to implement and showed good
results for one test function suit including 13 benchmark
functions. This method was originally employed to evaluate
and rank the whole population and will be modified to apply
to infeasible group in GEA. Next subsection will introduce
stochastic ranking method in detail and give the simulation
results of GEA with this method.

2) Parent Selection: Another important part of GEA is
how to select the parents from the two separated groups.
Normally, the number of parents and the number of individuals
of evolutionary algorithm are predefined. Since the feasible
and infeasible individuals in GEA are separated, we need
one strategy to decide the number of feasible parents and the
number of infeasible parents for reproduction. Here, a novel
parent selection method is proposed, where the number of feasible
parents and infeasible individuals in GEA are separated, we need
to design the corresponding numbers of best individuals from the feasible
and infeasible groups, respectively, according to their rankings.
To explain this more clearly, assume that \((\mu, \lambda) - ES\) is used
for computation, \(\mu\) is parent size, \(\lambda\) is population size. \((\mu, \lambda)\) is
set as \((30, 200)\). If \(S_p = 5\) and \(numFeaInd = 20\), then by (5),
we can get \(numFeaPar = 4\). And by (7), \(numInfeaPar = 30 - 4 = 26\). Therefore we select the best 4 individuals from the feasible
group and best 26 individuals from the infeasible
group to form the parent group for reproduction.

### III. Evaluation and Ranking of GEA

As the population of GEA is divided into two groups: feasible
group and infeasible group, it is necessary to design the
evaluation and ranking of these two groups separately. Since
the individuals in feasible group satisfies all the constraints,
the job is relatively simple. In this paper, we directly choose
the objective function (1) as the fitness function and the bubble
ranking as the ranking method for the feasible group. For
infeasible group, stochastic ranking method is applied.

A. Stochastic Ranking Method for Infeasible Group

Most of the penalty methods use some penalty coefficients
to adjust the extent of ‘penalty’ the infeasible individuals
deserve. However, deciding optimal values of penalty coeffi-
cients turns out to be a difficult problem. Runarsson and
Yao gave a deep investigation of the penalty method and the
penalty coefficients in [10]. Also, a stochastic ranking method
different from any penalty method was proposed to balance the
dominance of the objective function (1) and penalty function
(8). The main idea was to use the objective function for
comparison with a probability \(P_f\). That is, given any pair of
two adjacent individuals, the probability of comparing them
(in order to determine which is fitter) according to the objective
function was 1 if both individuals were feasible; otherwise it
was \(P_f\).

In order to adopt this stochastic ranking method for the
infeasible group of GEA, a small modification is made. The
modified stochastic ranking is shown in Figure 5. \(\phi > 0\) is
a real-valued function called the penalty function defined as follows:

\[
\phi(g_k(\bar{x}), h_k(\bar{x})) = \sum_{k=1}^{l} \max\{0, g_k(\bar{x})\}^2 + \sum_{k=l+1}^{m} |h_k(\bar{x})|^2.
\]

The difference is that the condition in step 5 of the original
stochastic ranking is ‘if \(\phi(I_j) = \phi(I_{j+1}) = 0\) or \(u < P_f\)
then’, while in the modified stochastic ranking, the condition
‘if \(\phi(I_j) = \phi(I_{j+1}) = 0\) then’ is removed because this
condition will never happen in infeasible group. In Figure 5,
\(\lambda\) is the population size, \(N\) is the number of sweeps going
through the population. It is fixed to be equal to \(\lambda\) as in [10].
In the later parts of this paper, we will call the GEA with stochastic ranking method ‘GEA_S’.

B. Experimental Studies

Thirteen benchmark functions were tested. The details of these functions are listed in Appendix. Problems G2, G3, G8 and G12 are maximization problems. They were transformed into minimization problems using \(-f(x)\). Problems G3, G5, G11 and G13 include one or several equality constraints. All of these equality constraints were converted into inequality constraints, \(|h(x)| - \delta \leq 0\), using the degree of violation \(\delta = 0.0001\).

A \((\mu, \lambda)\)-ES [18] was employed for recombination and mutation. For fair comparison, all parameters of the ES used here are the same as those of [10]. For each of the benchmark problem, 30 independent runs were performed using a \((30, 200)\)-ES. All runs were terminated after 2000 generations. The simulation program was coded in C.

Table I (data from [10]) and Table II show the experimental results of stochastic ranking method and GEA_S, respectively. \(P_f\) was set as 0.45 for these two algorithms and \(S_p\) was set as 5 for GEA_S which was the same as that of GEA_D. The median number of generations for finding the best solution in each run is indicated by \(g_m\) in the tables. The tables also show the known ‘optimal’ solution for each problem and statistics for the 30 independent runs. These include the best, median, worst and mean objective values and the standard deviation found.

Comparing Table I and Table II, we could find that GEA_S has an advantage over stochastic ranking method for most problems. For problems G1, G4, G8, G11 and G12, both two algorithms performed well and found the optimal solutions for all 30 runs. For problem G2, best of GEA_S was a little worse than that of stochastic ranking method, while median, worst and mean were much better. For problem G3, stochastic ranking method was more consistent. All of the four criteria terms (best, median, worst and mean) could reach the optimal value \(-1.000\). Although GEA_S also found the optimal value on best, median, worst and mean still had a gap about 0.001 \(~0.002\) to the optimal value. For the rest of problems, GEA_S performed better than the stochastic ranking method except that several criteria terms were a little worse: worst of G5, worst and mean of G13. For problems G5 and G13, the best results of GEA_S were even better than the optimal solution. This is the consequence of converting equality constraints into inequality constraints, although a very small \(\delta\) was used. For problems G6, G7 and G9, both of these two algorithms found the same best values, but GEA_S performed much better on the other three criteria terms median, mean and worst. For problem G10, GEA_S performed significantly better than the stochastic ranking method on all criteria terms: best, mean and worst of stochastic ranking method were 7054.316, 7559.192 and 8835.655, while the corresponding values of GEA_S were 7050.965, 7181.072 and 7526.509. It is interesting to note that the worst of GEA_S was even better than the mean of stochastic ranking method.

In summary, the numerical experiments with the benchmark problems suggest that GEA provides an efficient structure to improve the performance of evolutionary algorithms with stochastic ranking method for constrained optimization problems.

IV. CONCLUSION AND FUTURE RESEARCH DIRECTIONS

In this paper, we proposed a new constraint handling technique: grouping-based evolutionary algorithm. This method divides the population into two groups: feasible group and infeasible group according to the feasibility of the individuals. The evaluation and ranking of these two groups are performed in parallel and separately. Parent selection from these two groups is tuned by one parameter \(S_p\) which determines the numbers of feasible parents and infeasible parents for reproduction. In addition, stochastic ranking method is modified and included into GEA_S to evaluate and rank the infeasible group. GEA_S was tested on a set of 13 benchmark problems. Future research includes \(S_p\) tuning; GEA for multi-objective constrained optimization problems.

REFERENCES

### TABLE I

**Experimental Results of Stochastic Ranking Method; \( P_f = 0.45 \); Data is From [10].**

<table>
<thead>
<tr>
<th>Function</th>
<th>( g_m )</th>
</tr>
</thead>
<tbody>
<tr>
<td>G1</td>
<td>7125</td>
</tr>
<tr>
<td>G2</td>
<td>1271</td>
</tr>
<tr>
<td>G3</td>
<td>1033</td>
</tr>
<tr>
<td>G4</td>
<td>341</td>
</tr>
<tr>
<td>G5</td>
<td>300</td>
</tr>
<tr>
<td>G6</td>
<td>172</td>
</tr>
<tr>
<td>G7</td>
<td>1142</td>
</tr>
<tr>
<td>G8</td>
<td>400</td>
</tr>
<tr>
<td>G9</td>
<td>690</td>
</tr>
<tr>
<td>G10</td>
<td>920</td>
</tr>
<tr>
<td>G11</td>
<td>119</td>
</tr>
<tr>
<td>G12</td>
<td>86</td>
</tr>
<tr>
<td>G13</td>
<td>411</td>
</tr>
</tbody>
</table>

### TABLE II

**Experimental Results of GEA_S; \( S_p = 5 \); \( P_f = 0.45 \).**

<table>
<thead>
<tr>
<th>Function</th>
<th>( g_m )</th>
</tr>
</thead>
<tbody>
<tr>
<td>G1</td>
<td>7125</td>
</tr>
<tr>
<td>G2</td>
<td>1271</td>
</tr>
<tr>
<td>G3</td>
<td>1033</td>
</tr>
<tr>
<td>G4</td>
<td>341</td>
</tr>
<tr>
<td>G5</td>
<td>300</td>
</tr>
<tr>
<td>G6</td>
<td>172</td>
</tr>
<tr>
<td>G7</td>
<td>1142</td>
</tr>
<tr>
<td>G8</td>
<td>400</td>
</tr>
<tr>
<td>G9</td>
<td>690</td>
</tr>
<tr>
<td>G10</td>
<td>920</td>
</tr>
<tr>
<td>G11</td>
<td>119</td>
</tr>
<tr>
<td>G12</td>
<td>86</td>
</tr>
<tr>
<td>G13</td>
<td>411</td>
</tr>
</tbody>
</table>

---


---

**APPENDIX**

Problem G1: Minimize

\[
 f(\vec{x}) = 5 \sum_{j=1}^{4} x_j - 5 \sum_{j=1}^{4} x_j^2 - \sum_{j=5}^{13} x_j,
\]

subject to

\[
 g_1(\vec{x}) = 2x_1 + 2x_2 + x_10 + x_{11} - 10 \leq 0,
 g_2(\vec{x}) = x_1 + x_3 + x_10 + x_{12} - 10 \leq 0,
 g_3(\vec{x}) = 2x_2 + x_3 + x_{11} + x_{12} - 10 \leq 0,
 g_4(\vec{x}) = -8x_1 + x_{10} \leq 0,
 g_5(\vec{x}) = -8x_2 + x_{11} \leq 0,
 g_6(\vec{x}) = -8x_3 + x_{12} \leq 0,
 g_7(\vec{x}) = 2x_4 - x_5 + x_{10} \leq 0,
 g_8(\vec{x}) = -2x_6 + x_{11} \leq 0,
 g_9(\vec{x}) = -2x_8 - x_9 + x_{12} \leq 0,
\]
and bounds
\[ 0 \leq x_j \leq 1 \quad (j = 1, \ldots, 9), \]
\[ 0 \leq x_j \leq 100 \quad (j = 10, 11, 12), \]
\[ 0 \leq x_{13} \leq 1. \]
The global minimum is at \( \bar{x}^* = (1, 1, 1, 1, 1, 1, 1, 3, 3, 1, 3, 3), \) and \( f(\bar{x}^*) = -15. \)

Problem G2 : Maximize

\[
f(\bar{x}) = \left| \sum_{j=1}^{n} \cos(x_j) - 2 \prod_{j=1}^{n} \cos(x_j) \right|,\]
subject to
\[
g_1(\bar{x}) = 0.75 - \prod_{j=1}^{n} x_j \leq 0,\]
\[
g_2(\bar{x}) = \sum_{j=1}^{n} x_j - 7.5n \leq 0,\]
and bounds
\[ 0 \leq x_j \leq 10 \quad (j = 1, \ldots, n), \]
where \( n = 20. \) The global maximum is unknown; the known solution is \( f(\bar{x}^*) = 0.803619. \)

Problem G3 : Maximize

\[
f(\bar{x}) = (\sqrt{n})^n \prod_{j=1}^{n} x_j,\]
subject to
\[
h_1(\bar{x}) = \sum_{j=1}^{n} x_j^2 - 1 = 0,\]
and bounds
\[ 0 \leq x_j \leq 1 \quad (j = 1, \ldots, n), \]
where \( n = 10. \) The global minimum is at \( x_j^* = 1/\sqrt{n} \quad (j = 1, \ldots, n), \) and \( f(\bar{x}^*) = 1. \)

Problem G4 : Minimize

\[
f(\bar{x}) = 5.3578547x_1^3 + 0.8356891x_1x_5 + 37.2932391x_1 - 40792.141,\]
subject to
\[
g_1(\bar{x}) = 85.334407 + 0.0056858x_2x_5 + 0.0006262x_1x_4 - 0.002265x_4x_5 \leq 92, \]
\[
g_2(\bar{x}) = -85.334407 - 0.0056858x_2x_5 - 0.0006262x_1x_4 + 0.002265x_4x_5 \leq 0, \]
\[
g_3(\bar{x}) = 80.51249 + 0.0071317x_2x_5 + 0.0029955x_1x_2 + 0.0021813x_2^2 \leq 110, \]
\[
g_4(\bar{x}) = -80.51249 - 0.0071317x_2x_5 - 0.0029955x_1x_2 - 0.0021813x_2^2 \leq 0, \]
\[
g_5(\bar{x}) = 9.300961 + 0.0047026x_3x_5 + 0.0012547x_1x_3 + 0.0019085x_3x_2 - 25 \leq 0, \]
\[
g_6(\bar{x}) = -9.300961 - 0.0047026x_3x_5 - 0.0012547x_1x_3 - 0.0019085x_3x_2 + 20 \leq 0, \]
and bounds
\[ 78 \leq x_1 \leq 102, 33 \leq x_2 \leq 45, 27 \leq x_3 \leq 45, \quad (j = 3, 4, 5). \]
The optimal solution is at \( \bar{x}^* = (78.33, 29.995256025682, 45, 36.775819205788), \) and \( f(\bar{x}^*) = -30665.539. \)

Problem G5 : Minimize

\[
f(\bar{x}) = 3x_1 + 0.0000001x_1^3 + 2x_2 + (0.000002/3)x_2^3,\]
subject to
\[
g_1(\bar{x}) = -x_4 + x_3 - 0.55 \leq 0, \]
\[
g_2(\bar{x}) = -x_3 + x_4 - 0.55 \leq 0, \]
\[
h_1(\bar{x}) = 1000 \sin(-x_3 - 0.25) + 1000 \sin(-x_4 - 0.25) + 894.8 - x_1 = 0, \]
\[
h_2(\bar{x}) = 1000 \sin(x_3 - 0.25) + 1000 \sin(x_4 - x_4 - 0.25) + 894.8 - x_2 = 0, \]
\[
h_3(\bar{x}) = 1000 \sin(x_4 - 0.25) + 1000 \sin(x_4 - x_3 - 0.25) + 1294.8 = 0, \]
and bounds
\[ 0 \leq x_1 \leq 1200, 0 \leq x_2 \leq 1200, \]
\[ -0.55 \leq x_3 \leq 0.55, -0.55 \leq x_4 \leq 0.55. \]
The best known solution is at \( \bar{x}^* = (679.9453, 1026.067, 0.1188764, -0.3962336), \) and \( f(\bar{x}^*) = 5126.4981. \)

Problem G6 : Minimize

\[
f(\bar{x}) = (x_1 - 10)^3 + (x_2 - 20)^3,\]
subject to
\[
g_1(\bar{x}) = -(x_1 - 5)^2 - (x_2 - 5)^2 + 100 \leq 0, \]
\[
g_2(\bar{x}) = (x_1 - 6)^2 + (x_2 - 5)^2 - 82.81 \leq 0, \]
and bounds
\[ 13 \leq x_1 \leq 100, 0 \leq x_2 \leq 100. \]
The known global solution is \( \bar{x}^* = (14.095, 0.84296), \) and \( f(\bar{x}^*) = -6961.81388. \)

Problem G7 : Minimize

\[
f(\bar{x}) = x_1^3 + x_2^3 + x_1x_2 - 14x_1 - 16x_2 + (x_3 - 10)^2 + 4(x_4 - 5)^2 + (x_5 - 3)^2 + 2(x_6 - 1)^2 + 5x_7^2 + 7(x_8 - 11)^2 + 2(x_9 - 10)^2 + (x_{10} - 7)^2 + 45,\]
subject to
\[
g_1(\bar{x}) = -105 + 4x_1 + 5x_2 - 3x_7 + 9x_8 \leq 0, \]
\[
g_2(\bar{x}) = 10x_1 - 8x_2 - 17x_7 + 2x_9 \leq 0, \]
\[
g_3(\bar{x}) = -8x_1 + 2x_2 + 5x_9 - 2x_5 - 12 \leq 0, \]
\[
g_4(\bar{x}) = 3(x_1 - 2)^2 + 4(x_2 - 3)^2 + 2x_3^2 - 7x_4 - 120 \leq 0, \]
\[
g_5(\bar{x}) = 5x_1^2 + 8x_2 + (x_3 - 6)^2 - 2x_4 - 40 \leq 0, \]
\[
g_6(\bar{x}) = x_1^2 + 2(x_2 - 2)^2 - 2x_1x_2 + 14x_5 - 6x_6 \leq 0, \]
\[
g_7(\bar{x}) = 0.5(x_1 - 8)^2 + 2(x_2 - 4)^2 + 3x_3^2 - x_6 - 30 \leq 0, \]
\[
g_8(\bar{x}) = -3x_1 + 6x_2 + 12(x_9 - 8)^2 - 7x_{10} \leq 0, \]
and bounds 

\[-10 \leq x_j \leq 10 \quad (j = 1, \ldots, 10).\]

The optimal solution is at \( \bar{x}^* = (2.171996, 2.363683, 8.773926, 5.095984, 0.906548, 1.430574, 1.321644, 9.828726, 8.280092, 8.375927) \), and \( f(\bar{x}^*) = 24.3062091 \).

Problem G8 : Maximize

\[
f(\bar{x}) = \sin^3(2\pi x_1) \sin(2\pi x_2),
\]

subject to

\[
g_1(\bar{x}) = x_1^2 - x_2 + 1 \leq 0,
g_2(\bar{x}) = 1 - x_1 + (x_2 - 4)^2 \leq 0,
\]

and bounds 

\[-10 \leq x_1 \leq 10, -10 \leq x_2 \leq 10.\]

The optimal solution is at \( \bar{x}^* = (1.2279713, 4.2453733) \), and \( f(\bar{x}^*) = 0.095825 \).

Problem G9 : Minimize

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

subject to

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

and bounds 

\[-10 \leq x_j \leq 10 \quad (j = 1, \ldots, 7).\]

The known global solution is at \( \bar{x}^* = (2.330499, 1.951372, -0.4775414, 4.365726, -0.6244870, 1.038131, 1.594227) \), and \( f(\bar{x}^*) = 680.6300573 \).

Problem G10 : Minimize

\[
f(\bar{x}) = x_1 + x_2 + x_3,
\]

subject to

\[
g_1(\bar{x}) = -1 + 0.0025(x_2 + x_6) \leq 0,
g_2(\bar{x}) = -1 + 0.0025(x_5 + x_7 - x_4) \leq 0,
g_3(\bar{x}) = -1 + 0.01(x_8 - x_5) \leq 0,
g_4(\bar{x}) = -x_1x_6 + 833.33252x_4 + 100x_1 - 83333.333 \leq 0,
g_5(\bar{x}) = -x_2x_7 + 1250x_5 + x_2x_4 - 1250x_4 \leq 0,
g_6(\bar{x}) = -x_3x_8 + 125000 + x_3x_5 - 2500x_5 \leq 0,
\]

and bounds 

\[100 \leq x_1 \leq 10000, 1000 \leq x_j \leq 10000 \quad (j = 2, 3),
10 \leq x_j \leq 1000 \quad (j = 4, \ldots, 8).\]

The optimal solution is at \( \bar{x}^* = (579.3167, 1359.943, 5110.071, 182.0174, 295.5985, 217.9799, 286.4162, 395.5979) \), and \( f(\bar{x}^*) = 7049.3307 \).

Problem G11 : Minimize

\[
f(\bar{x}) = x_2^2 + (x_2 - 1)^2,
\]

subject to 

\[
h_1(\bar{x}) = x_2 - x_1^2 = 0,
\]

and bounds 

\[-1 \leq x_1 \leq 1, -1 \leq x_2 \leq 1.\]

The optimal solution is at \( \bar{x}^* = (\pm 1/\sqrt{2}, 1/\sqrt{2}) \), and \( f(\bar{x}^*) = 0.75 \).

Problem G12 : Maximize

\[
f(\bar{x}) = (100 - (x_1 - 5)^2 - (x_2 - 5)^2 - (x_3 - 5)^2)/100,
\]

subject to 

\[
g(\bar{x}) = (x_1 - p)^2 + (x_2 - q)^2 + (x_3 - r)^2 - 0.0625 \leq 0,
\]

and bounds 

\[0 \leq x_j \leq 10 \quad (j = 1, 2, 3),\]

where \( p, q, r = 1, 2, \ldots, 9 \). The feasible region of the search space consists of 9 disjointed spheres. A point \( (x_1, x_2, x_3) \) is feasible if and only if there exists \( p, q, r \) such that the above inequality holds. The optimal solution is at \( \bar{x}^* = (5, 5, 5) \), and \( f(\bar{x}^*) = 1 \).

Problem G13 : Minimize

\[
f(\bar{x}) = e^{x_1x_2x_3x_4x_5},
\]

subject to 

\[
h_1(\bar{x}) = x_2^2 + x_3^2 + x_4^2 + x_5^2 - 10 = 0,
h_2(\bar{x}) = x_2x_3 - 5x_4x_5 = 0,
h_3(\bar{x}) = x_3^2 + x_4^2 + 1 = 0,
\]

and bounds 

\[-2.3 \leq x_j \leq 2.3 \quad (j = 1, 2), -3.2 \leq x_j \leq 3.2 \quad (j = 3, 4, 5).\]

The optimal solution is at \( \bar{x}^* = (-1.717143, 1.595709, 1.827247, -0.7636413, -0.763645) \), and \( f(\bar{x}^*) = 0.0539498 \).