

Research Article

A Hybrid Epigraph Directions Method for Nonsmooth and Nonconvex Constrained Optimization via Generalized Augmented Lagrangian Duality and a Genetic Algorithm

Wilhelm P. Freire,¹ Afonso C. C. Lemonge ,²
Tales L. Fonseca ,³ and Hernando J. R. Franco ³

¹Department of Mathematics, Institute of Exact Sciences, Federal University of Juiz de Fora, Brazil

²Department of Applied and Computational Mechanics, Faculty of Engineering, Federal University of Juiz de Fora, Brazil

³Postgraduate Program of Computational Modeling, Federal University of Juiz de Fora, Brazil

Correspondence should be addressed to Hernando J. R. Franco; hernando.franco@ifsudestemg.edu.br

Received 16 January 2018; Accepted 15 April 2018; Published 22 May 2018

Academic Editor: Guillermo Cabrera-Guerrero

Copyright © 2018 Wilhelm P. Freire et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The Interior Epigraph Directions (IED) method for solving constrained nonsmooth and nonconvex optimization problem via Generalized Augmented Lagrangian Duality considers the dual problem induced by a Generalized Augmented Lagrangian Duality scheme and obtains the primal solution by generating a sequence of iterates in the interior of the epigraph of the dual function. In this approach, the value of the dual function at some point in the dual space is given by minimizing the Lagrangian. The first version of the IED method uses the Matlab routine `fminsearch` for this minimization. The second version uses NFDNA, a tailored algorithm for unconstrained, nonsmooth and nonconvex problems. However, the results obtained with `fminsearch` and NFDNA were not satisfactory. The current version of the IED method, presented in this work, employs a Genetic Algorithm, which is free of any strategy to handle the constraints, a difficult task when a metaheuristic, such as GA, is applied alone to solve constrained optimization problems. Two sets of constrained optimization problems from mathematics and mechanical engineering were solved and compared with literature. It is shown that the proposed hybrid algorithm is able to solve problems where `fminsearch` and NFDNA fail.

1. Introduction

We present a new version of the Interior Epigraph Directions Method published by Burachik et al. [1] for solving constrained nonsmooth and nonconvex optimization problems of the following kind:

$$\text{minimize } f(x) \text{ over all } x \in X \text{ satisfying } g(x) = 0, \quad (P)$$

where $X \subset \mathbb{R}^n$ is compact and the functions $f: \mathbb{R}^n \rightarrow \mathbb{R}$ and $g: \mathbb{R}^n \rightarrow \mathbb{R}^m$ are continuous. Inequality constraints can be incorporated into (P) by using the operator $a^+ = \max\{a, 0\}$, $a \in \mathbb{R}$.

The IED method uses a Lagrangian duality technique, where each constraint function is appended to the objective function via a multiplier, or dual variable, to form the classical

Lagrangian. In a duality scheme, instead of problem (P), the problem solved is the one where the dual function is maximized. In this approach, the value of the dual function at a given point in the dual space is computed by minimizing the Lagrangian which is an unconstrained problem, and therefore unconstrained methods can be used.

It turns out that the optimal value of the dual problem associated with the classical Lagrangian is in general not the same as that of the primal problem (P), if (P) is nonconvex. The difference between the two optimal values is called duality gap.

Zero duality gap and saddle-point properties without convexity assumptions are known to hold for the following family of Lagrangian function [2, Remark 2.1].

$$L(x, (u, c)) = f(x) + c\sigma(g(x)) + \langle Au, g(x) \rangle, \quad (1)$$

where $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$, $c \in \mathbb{R}_+ = [0, \infty)$, $A \in \mathbb{R}^{m \times m}$ is a real symmetric matrix, and $\sigma : \mathbb{R}^m \rightarrow \mathbb{R}$ is a continuous function such that $\sigma(x) > 0$, $\forall x \in \mathbb{R}^m \setminus \{0\}$ and $\sigma(0) = 0$.

The dual problem generated by such Lagrangians as in (1) is a (nondifferentiable) convex problem, which is usually solved by nonsmooth convex optimization techniques, such as subgradient methods and their extensions [2–8].

The IED method uses the family of Lagrangians (1). These Lagrangians induce dual variables $(u, c) \in \mathbb{R}^m \times \mathbb{R}_+$, where u is the classical multiplier associated with the linear term in the Lagrangian and $c \geq 0$ is the penalty multiplier associated with the augmenting term.

A duality approach based on the Lagrangian L for solving (P) can be described as follows. Given a current dual iterate (u_k, c_k) , a primal iterate is computed through the rule

$$x_k \in \arg \min_{x \in X} L(x, (u_k, c_k)), \quad (2)$$

where L is as in (1).

The iterate x_k obtained in (2) can be used for computing the search direction which leads to the next dual iterate (u_{k+1}, c_{k+1}) . Indeed, x_k is used to obtain a deflection of the subgradient direction which in turn is utilized to improve the dual values. This is the idea behind the methods studied in [2–8]. These methods are referred to as deflected subgradient (DSG) methods.

The IED method [1] combines the Nonsmooth Feasible Directions Algorithm (NFDNA) for solving unconstrained nonsmooth convex problems, firstly presented in [9] and further studied in [10], with a certain DSG method.

Roughly speaking, the IED method works as follows: starting at a point in the interior of the epigraph of the dual function, a deflected subgradient direction is used to define a linear approximation to the epigraph. An auxiliary point is obtained by solving the optimality conditions of a resulting linear problem. If the auxiliary point belongs to the interior of the epigraph, the IED step is declared serious and the auxiliary point becomes the next iterate, from which the process is repeated. If the auxiliary point does not belong to the interior of the epigraph, the IED step is declared null and a suitable version of the DSG step is applied from the original point. Thanks to the properties of the DSG method, this step stays in the interior of the epigraph. The new point takes the place of the original point, from which the process is repeated, until a serious step is performed. If an infinite number of null steps occurs, the method converges to a primal solution, as a consequence of the convergence properties of the DSG.

An important aspect of the method is the minimization of the Lagrangian function. Although the solver used for this task does not affect the convergence of the IED method, as proved in [1], solving this problem effectively is fundamental for the success of the methods based on Lagrangian duality.

The first version of the IED method [1] employs the Matlab routine `fminsearch` for the Lagrangian minimization. The second version of the IED method [15] uses an algorithm called Nonsmooth Feasible Directions Nonconvex Algorithm (NFDNA) [16, 17] for solving nonsmooth and nonconvex unconstrained optimization problems which correspond to our problem (2). Although the previous versions of IED

were able to solve many problems, they failed some test problems and did not find the desired solutions as one can observe through some numerical experiments presented in this paper and found in [1, 15]. We do not know why `fminsearch` and `NFDNA` fail. For `fminsearch` is a routine of Matlab, we think that it does not make sense trying to fix or change it. The `NFDNA` performance needs deeper investigation. In this work, we have used a Genetic Algorithm (GA) for minimizing the Lagrangian function. With this modification, the method gained robustness and was able to solve all the test problems analyzed here. The hybridization of the IED method and the Genetic Algorithm is the main purpose of this work and can be considered innovative. Also, it is important to notice that the GA is free of any strategy to handle the constraints, a difficult task when a metaheuristic, such as the GA, is applied alone to solve constrained optimization problems [18].

We have solved twenty problems and compared the performance of the current version of the IED method with the performance of the first version. We also solved four real-world mechanical engineering optimization problems with the current version of the IED.

This paper is organized in 8 sections. In the next section, we state the primal and the dual problems and recall how to find a subgradient of the dual function. In Section 3, we show how the search direction used by IED is obtained. In Section 4, we present NFDNA, an algorithm for convex optimization problems that inspired the IED method. The IED method is then presented in Section 5. In Section 6, we present a brief description of the main characteristics of the Genetic Algorithm and its basic features adopted in this paper. Numerical results and comparisons are given in Section 7. Finally, conclusions and suggestions for future work are given in Section 8.

2. The Problem in Study

The problems we are interested in are of the same type of the primal problem (P).

The augmented Lagrangian function $L : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}_+ \rightarrow \mathbb{R}$ associated with (P) is defined as in (1). The corresponding concave dual function $H : \mathbb{R}^m \times \mathbb{R}_+ \rightarrow \mathbb{R}$ is defined as

$$H(u, c) = \min_{x \in X} [f(x) + c\sigma(g(x)) + \langle Au, g(x) \rangle]. \quad (3)$$

The dual problem is given by

$$\max_{(u, c) \in \mathbb{R}^m \times \mathbb{R}_+} H(u, c) \quad (D)$$

and, since the dual function H is concave, (D) is equivalent to the convex problem

$$\min_{(u, c) \in \mathbb{R}^m \times \mathbb{R}_+} -H(u, c). \quad (D^*)$$

For convenience, we introduce the set

$$X(u, c) = \arg \min_{x \in X} [f(x) + c\sigma(g(x)) + \langle Au, g(x) \rangle] \quad (4)$$

whose elements are obtained by the GA used in this work.

Figure 8 shows a flowchart that summarizes how such minimization is done.

For simplicity (D^*) will be our dual problem and we will refer to $q := -H$ as the dual function.

We recall now an important result that shows how, after solving problem (4), we can easily obtain a subgradient $s \in \partial q(u, c)$.

Theorem 1 (see [2], Lemma 2.1(a)). *Given the dual function q and a pair $v = (u, c) \in \mathbb{R}^m \times \mathbb{R}_+$, take $x \in X(v)$. Then $s = (-Ag(x), -\sigma(g(x))) \in \partial q(v)$.*

3. The IED Search Direction

In this section we describe how the IED search direction is obtained.

The methodology employed here was firstly used by an algorithm called Feasible Directions Interior Point Algorithm (FDIPA) [19] for smooth nonlinear problems.

Let us consider the inequality constrained optimization problem

$$\begin{aligned} & \text{minimize} && f(x), \\ & \text{subject to} && g(x) \leq 0, \end{aligned} \quad (P_3)$$

where $f : \mathbb{R}^n \rightarrow \mathbb{R}$ and $g : \mathbb{R}^n \rightarrow \mathbb{R}^m$ are continuously differentiable.

Let x^* be a regular point of problem (P_3). The Karush-Kuhn-Tucker (KKT) first order necessary optimality conditions are expressed as follows: If x^* is a local minimum of (P_3) then there exists $\lambda^* \in \mathbb{R}^m$ such that

$$\nabla f(x^*) + \nabla g(x^*) \lambda^* = 0 \quad (5)$$

$$G(x^*) \lambda^* = 0 \quad (6)$$

$$\lambda^* \geq 0 \quad (7)$$

$$g(x^*) \leq 0, \quad (8)$$

where $G(x)$ is a diagonal matrix with $G_{ii}(x) \equiv g_i(x)$.

The point x such that $g(x) \leq 0$ is called a ‘‘Primal Feasible Point’’ and $\lambda \geq 0$ a ‘‘Dual Feasible Point’’. Given an initial feasible pair (x^0, λ^0) , KKT points are found by solving iteratively the nonlinear system of equations (5) and (6) in (x, λ) , in such a way that all the iterates are primal and dual feasible.

A Newton-like iteration to solve the nonlinear system of equations (5) and (6) can be stated as

$$\begin{aligned} & \begin{pmatrix} B^k & \nabla g(x^k) \\ \Lambda^k \nabla g(x^k)^T & G(x^k) \end{pmatrix} \begin{pmatrix} x - x^k \\ \lambda - \lambda^k \end{pmatrix} \\ & = - \begin{pmatrix} \nabla f(x^k) + \nabla g(x^k) \lambda^k \\ G(x^k) \lambda^k \end{pmatrix}, \end{aligned} \quad (9)$$

where (x^k, λ^k) is the current point of the iteration k , Λ is a diagonal matrix with $\Lambda_{ii} \equiv \lambda_i$, and $B^k \equiv \nabla^2 f(x^k) +$

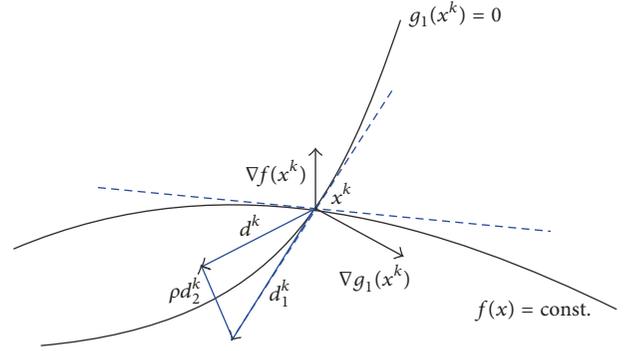


FIGURE 1: The search direction of FDIPA.

$\sum_{i=1}^m \lambda_i^k \nabla^2 g_i(x^k)$ is the Hessian of the Lagrangian function $L(x, \lambda) = f(x) + \lambda^T g(x)$ or some quasi-Newton approximation. However, B^k must be symmetric and positive definite in order to ensure convergence.

Calling $d = x - x^k$, the following system can be written from (9):

$$B^k d + \nabla g(x^k) \lambda = -\nabla f(x^k) \quad (10)$$

$$\Lambda^k \nabla g(x^k)^T d + G(x^k) \lambda = 0. \quad (11)$$

The solution (d_1^k, λ_1^k) of the system (10)-(11) provides a descent direction d_1^k as proved in [19]. However, d_1^k can not be employed as a search direction, since it is not necessarily a feasible direction because, in the case when $g_i(x^k) = 0$, it follows from (11) that $\nabla g_i(x^k)^T d_1^k = 0$.

To avoid this drawback, a new system with unknowns d and $\bar{\lambda}$ is defined by adding a negative matrix $-\rho^k \Lambda^k$, with $\rho^k > 0$, to the right side of (11),

$$B^k d + \nabla g(x^k) \bar{\lambda} = -\nabla f(x^k) \quad (12)$$

$$\Lambda^k \nabla g(x^k)^T d + G(x^k) \bar{\lambda} = -\rho^k \Lambda^k. \quad (13)$$

Now, (13) is equivalent to $\lambda_i^k \nabla g_i(x^k)^T d^k + g_i(x^k) \bar{\lambda}^k = -\rho^k \lambda_i^k$, $i = 1, 2, \dots, m$. Consequently, if $g_i(x^k) = 0$, the latest equation is reduced to $\nabla g_i(x^k)^T d^k = -\rho^k < 0$, which means that d^k is a feasible direction.

The addition of a negative term $-\rho^k \lambda^k$ produces a proportional deflexion of d^k into the feasible region, as one can see in Figure 1. The problem now is that d^k might not be a descent direction for the function f . However, we can keep such property if ρ^k is properly chosen. Notice that the direction d^k can be obtained by solving two systems with the same matrix

$$B^k d_1 + \nabla g(x^k) \lambda_1 = -\nabla f(x^k) \quad (14)$$

$$\Lambda^k \nabla g(x^k)^T d_1 + G(x^k) \lambda_1 = 0, \quad (15)$$

$$B^k d_2 + \nabla g(x^k) \lambda_2 = 0 \quad (16)$$

$$\Lambda^k \nabla g(x^k)^T d_2 + G(x^k) \lambda_2 = -\Lambda^k \quad (17)$$

and setting $d^k = d_1^k + \rho^k d_2^k$, which gives $(d^k)^T \nabla f(x^k) = (d_1^k)^T \nabla f(x^k) + \rho^k (d_2^k)^T \nabla f(x^k)$. In the case where $(d_2^k)^T \nabla f(x^k) \leq 0$ we have $(d^k)^T \nabla f(x^k) < 0, \forall \rho^k > 0$.

Thus, ρ^k must be well chosen if $(d_2^k)^T \nabla f(x^k) > 0$. In this case, imposing $(d^k)^T \nabla f(x^k) \leq \xi (d_1^k)^T \nabla f(x^k)$, where $\xi \in (0, 1)$, implies $(d_1^k)^T \nabla f(x^k) + \rho^k (d_2^k)^T \nabla f(x^k) \leq \xi (d_1^k)^T \nabla f(x^k)$ which leads to

$$\rho^k \leq \frac{(\xi - 1) (d_1^k)^T \nabla f(x^k)}{(d_2^k)^T \nabla f(x^k)}. \quad (18)$$

Thus, with ρ^k chosen as described above, we have that $d^k = d_1^k + \rho^k d_2^k$ is a feasible descent direction.

The ideas just described are sufficient for us to understand how the search direction used by the IED method is computed.

For more on FDIPA, its features, and convergence, see [19].

4. NFDA for Convex Problems

The Nonsmooth Feasible Directions Algorithm (NFDA), firstly presented by Freire [9], has been devised for solving unconstrained nonsmooth convex problems of the following kind:

$$\begin{aligned} \min \quad & F(x) \\ \text{subject to} \quad & x \in \mathbb{R}^n, \end{aligned} \quad (P_{4.1})$$

where $F : \mathbb{R}^n \rightarrow \mathbb{R}$ is a convex, not necessarily differentiable function.

Problem $(P_{4.1})$ can be replaced with the equivalent constrained problem

$$\begin{aligned} \min \quad & f(x, z) = z \\ \text{subject to} \quad & F(x) \leq z \\ & (x, z) \in \mathbb{R}^n \times \mathbb{R}. \end{aligned} \quad (P_{4.2})$$

It is assumed that one arbitrary subgradient $s \in \partial F(x)$ can be computed at any point $x \in \mathbb{R}^n$.

NFDA starts at a point (x^0, z^0) in the interior of the epigraph of the function F . At a point $(x^k, F(x^k))$, the method determines a supporting hyperplane to the epigraph of the function F

$$h^k(x) = F(x^k) + \langle s^k, (x - x^k) \rangle, \quad (19)$$

where $s^k \in \partial F(x^k)$ and defines an auxiliary constrained linear problem employing all the supporting hyperplanes computed so far, as follows:

$$\begin{aligned} \min \quad & f(x, z) = z \\ \text{subject to} \quad & g^k(x, z) \leq 0 \\ & (x, z) \in \mathbb{R}^n \times \mathbb{R}, \end{aligned} \quad (P_{4.3})$$

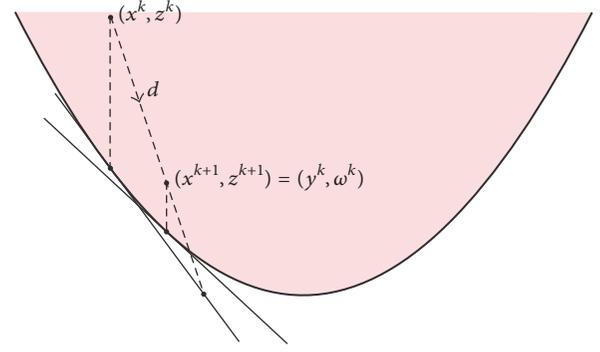


FIGURE 2: Serious step.

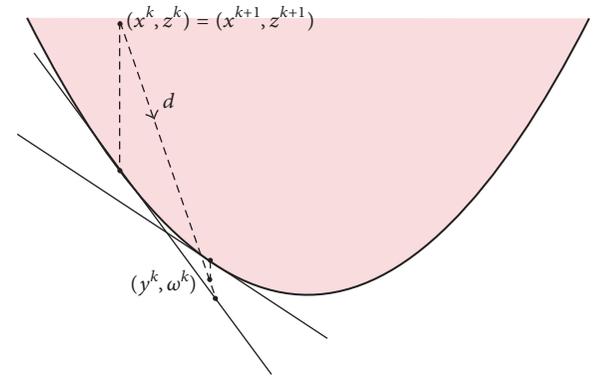


FIGURE 3: Null step.

where $g^k = (g_1^k, \dots, g_m^k) : \mathbb{R}^{n+1} \rightarrow \mathbb{R}^m$ is a vector function with $g_i^k : \mathbb{R}^{n+1} \rightarrow \mathbb{R}$ given by $g_i^k(x, z) = h_i^k(x) - z$.

Problem $(P_{4.3})$ is not solved. Indeed, it might not have a solution. NFDA uses its structure, which is similar to the structure of the problem (P_3) , to obtain a search direction by solving two systems and choosing the parameter ρ in the same way FDIPA does.

At the iterate (x^k, z^k) and having the search direction d^k at hand, a step t^k is obtained by the rule

$$t^k = \min \{t_{\max}, T\}, \quad (20)$$

where $t_{\max} = \max\{t \mid g_i^k((x^k, z^k) + td^k) \leq 0\}$ and $T > 0$ is a predefined parameter, since t_{\max} is not always finite.

A new point $(y^k, \omega^k) = (x^k, z^k) + \mu t^k d^k$ with $\mu \in (0, 1)$ is computed. If $F(y^k) < \omega^k$ then $(x^{k+1}, z^{k+1}) = (y^k, \omega^k)$ and the step is called *serious*, as shown in Figure 2. Otherwise, if $F(y^k) \geq \omega^k$ then $(x^{k+1}, z^{k+1}) = (x^k, z^k)$ and the step is called *null*; see Figure 3.

In any case, a new supporting hyperplane at the point $(y^k, F(y^k))$ is added to form a new auxiliary problem like $(P_{4.3})$.

These are the main ideas from NFDA; we need to understand the IED algorithm. For more on NFDA, see [9, 10].

5. The IED Method

In this section we describe the Interior Epigraph Directions (IED) method for solving constrained, nonsmooth, and nonconvex optimization problems.

As we have seen in Section 2, the dual problem (D) is equivalent to the problem

$$\begin{aligned} \min \quad & q(u, c) \\ & (u, c) \in \mathbb{R}^m \times \mathbb{R}_+ \end{aligned} \quad (D^{**})$$

which, in its turn, is equivalent to

$$\begin{aligned} \min \quad & \psi(u, c, z) = z \\ \text{subject to} \quad & q(u, c) \leq z \\ & (u, c, z) \in \mathbb{R}^m \times \mathbb{R}_+ \times \mathbb{R}. \end{aligned} \quad (ED^*)$$

One can notice that problems $(P_{4.2})$ and (ED^*) have the same structure. Therefore, we can apply to (ED^*) the ideas used by NFDA.

However, NFDA can only minimize nonsmooth convex functions which have bounded level sets. The convex function q , in general, does not enjoy this property. Therefore, the IED method uses directions which are suited to the function q we have. Indeed, the IED method takes advantage of both NFDA search direction and the special features of the dual function q .

At the step k of the IED method, we have a point $(v^k, z^k) = ((u^k, c^k), z^k) \in \text{int}(\text{epi}(q))$ and a primal iterate $x^k \in X(v^k)$ such that $g(x^k) \neq 0$.

Now, having x^k , according to Theorem 1, we can easily find a subgradient

$$s^k = (-Ag(x^k), -\sigma(g(x^k))) \in \partial q(v^k). \quad (21)$$

This subgradient defines a supporting hyperplane

$$h^k(v) = q(v^k) + \langle s^k, v - v^k \rangle \quad (22)$$

to the $\text{epi}(q)$ at the point $(v^k, q(v^k))$. By employing this supporting hyperplane, the following auxiliary problem is defined

$$\begin{aligned} \min \quad & \psi(u, c, z) = z \\ \text{subject to} \quad & g^k(u, c, z) \leq 0 \\ & (u, c, z) \in \mathbb{R}^m \times \mathbb{R}_+ \times \mathbb{R}, \end{aligned} \quad (AP_k)$$

where $g^k : \mathbb{R}^m \times \mathbb{R}_+ \times \mathbb{R} \rightarrow \mathbb{R}$ is given by $g^k(v, z) = h^k(v) - z$.

Solving two systems, similar to (14)-(15) and (16)-(17), obtained from (AP_k) , the directions d_1^k and d_2^k are computed. It is important to highlight that there is no duplication of the computational cost since the coefficient matrices of both systems are the same.

The search direction is thus defined by $d^k = d_1^k + \rho^k d_2^k$ where the parameter $\rho^k > 0$ must be chosen so that d^k is a

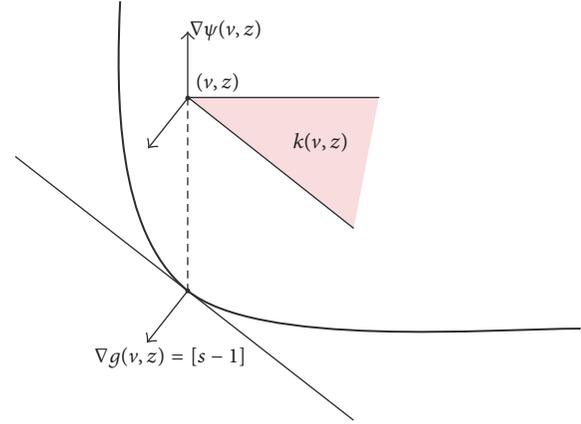


FIGURE 4: The cone $K(v, z)$.

descent direction for ψ , i.e., $(d^k)^T \nabla \psi(v^k, z^k) < 0$ and, at the same time, a local descent direction for the dual function q . This requires d^k to be a descent direction for g as well, i.e., $(d^k)^T \nabla g(v^k, z^k) < 0$.

In other words, IED finds a direction d which, at each iteration, belongs to the interior of the cone

$$\begin{aligned} K(v^k, z^k) = \{d \in \mathbb{R}^{m+1} \times \mathbb{R} \mid & d^T \nabla g(v^k, z^k) \\ & < 0, d^T \nabla \psi(v^k, z^k) < 0\}, \end{aligned} \quad (23)$$

according to Figure 4.

Burachik et al. [1] proved that if ρ satisfies

$$-\frac{d_1^T \nabla g(v, z)}{d_2^T \nabla g(v, z)} < \rho < (1 - \xi) \frac{d_1^T \nabla \psi(v, z)}{d_2^T \nabla \psi(v, z)} \quad (24)$$

then $d \in K(v, z)$.

From the current iterate $(v^k, z^k) \in \text{int}(\text{epi}(q))$, using a suitable stepsize t^k , an auxiliary point

$$(\bar{u}^k, \bar{c}^k, \bar{z}^k) = (\bar{v}^k, \bar{z}^k) = (v^k, z^k) + t^k d^k \quad (25)$$

is computed.

If $g(\bar{x}^k) = 0$ where $\bar{x}^k \in X(\bar{v}^k)$ then IED stops.

Otherwise, i.e., if $g(\bar{x}^k) \neq 0$ and $q(\bar{v}^k) < \bar{z}^k$ then $(v^{k+1}, z^{k+1}) = (\bar{v}^k, \bar{z}^k)$.

If $q(\bar{v}^k) \geq \bar{z}^k$ then a DSG step is performed in order to obtain the next iterate $(v^{k+1}, z^{k+1}) \in \text{int}(\text{epi}(q))$.

A new supporting hyperplane at the point $(v^{k+1}, q(v^{k+1}))$ is computed, a new auxiliary problem is defined, and the algorithm goes this way until a primal iterate x^k satisfying $g(x^k) = 0$ is found.

We now describe formally the algorithm.

5.1. The IED Algorithm

Step 0. Fix a sequence $\{\alpha_k\} \subset \mathbb{R}_+$ and a sequence $\{B^k\}$ of positive definite matrices. Fix $\beta \geq \eta > 0$, $T_0 \in (0, 1)$, $\hat{q} \leq \bar{q}$, where \bar{q} is the optimal dual value, a symmetric matrix A ,


```

(1) Input: Parameter  $popsiz$ ,  $gensiz$ ,  $u$ ,  $c$ ;
(2) Output: The best solution found in all generations;
(3)  $gen = 1$ ;
(4)  $Pop[gen] = InitializePopulation()$ ;
(5) Evaluate  $(L(x, (u, c)) = f(x) + c\sigma(g(x)) + \langle Au, g(x) \rangle)$ ,  $Pop[gen]$ ;
(6) while  $gen \leq gensiz$  do
(7)    $selPop[gen] = Selection(Pop[gen])$ ;
(8)    $newPop[gen] = Crossover(selPop[gen])$ ;
(9)    $Mutation(newPop[gen])$ ;
(10)  Evaluate  $(L(x, (u, c)) = f(x) + c\sigma(g(x)) + \langle Au, g(x) \rangle)$ ,  $newPop[gen]$ ;
(11)   $Pop[gen + 1] = NewGeneration(newPop[gen], Pop[gen])$ ;
(12)   $gen = gen + 1$ ;
(13) end
    
```

ALGORITHM 1: The generational Genetic Algorithm to minimize the Lagrangian function.

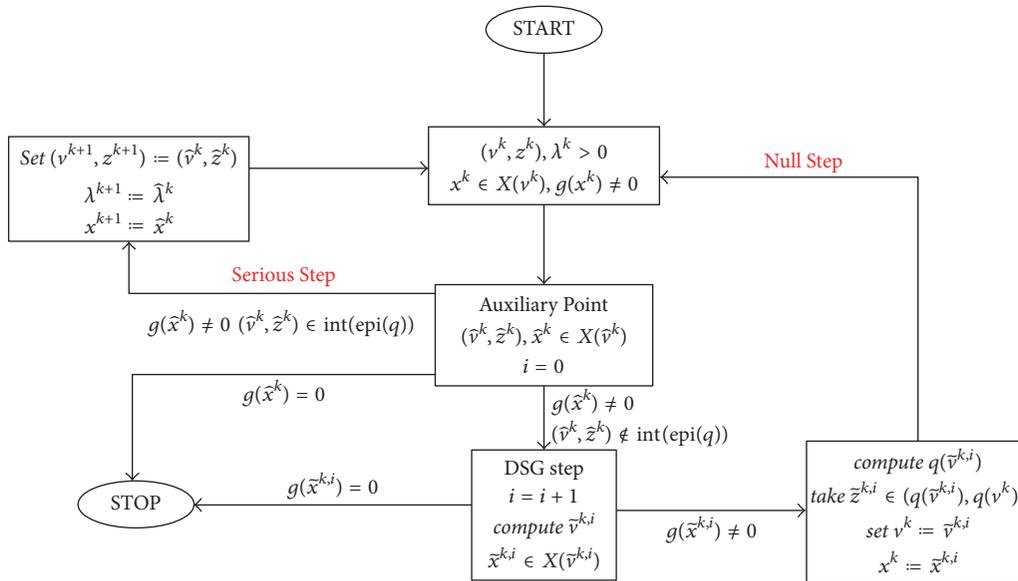


FIGURE 6: A flow chart for a step of the IED method with a Genetic Algorithms.

6. The Genetic Algorithm

Genetic algorithms (GAs) [20–22] were inspired by the Darwinian principles of natural selection and evolution. They are considered the most famous bioinspired metaheuristic and the basis of innumerable algorithms found in the related literature. GAs encode all the variables, using an adopted alphabet, as binary code, for example, corresponding to a candidate solution (defined as the phenotype) in a data structure (defined as the genotype). A population of genotypes evolves mimicking the evolutionary process found in the Nature. The candidate solutions are submitted to the evaluation of their quality and they are selected by a stochastic process favoring better solutions. The genetic materials of these are recombined and mutated by means of genetic operators generating a new population.

Basically, a GA presents the following five steps [21]: (i) a genetic coding of solutions to the problem; (ii) a procedure

for creating an initial population of solutions; (iii) an evaluation function that returns the quality of each individual (fitness function); (iv) genetic operators manipulating genetic material of parents during the reproduction process giving rise to new individuals; and (v) parameters to be used in the algorithm during the recombination and mutation operations.

The GAs are free-derivative algorithms and massively used to solve unconstrained and constrained optimization problems.

The link between IED and Genetic Algorithm recognized through the notation $x^k \in X(v^k)$ can be seen in Figures 6 and 7.

The pseudocode depicted in Algorithm 1 describes the main steps of a generational Genetic Algorithm for minimizing the Lagrangian.

In this paper, a Genetic Algorithm with a real code is adopted and the Simulated Binary Crossover (SBX) [23] and

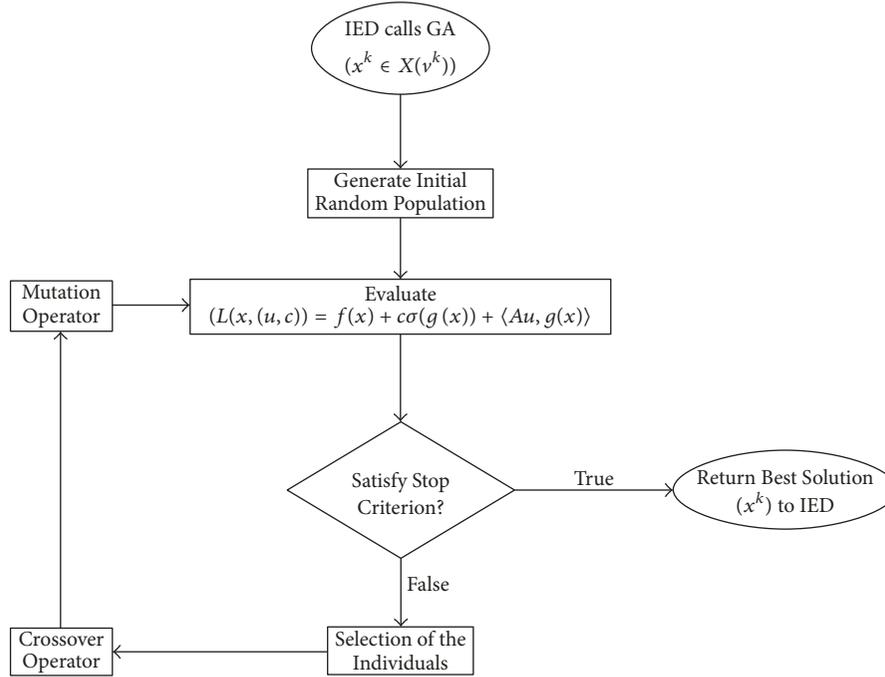


FIGURE 7: A flow chart of the Genetic Algorithm used inside the IED.

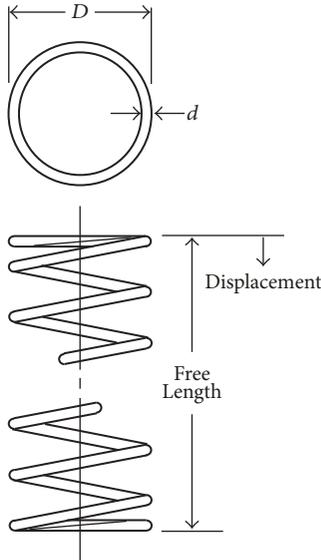


FIGURE 8: The Tension/Compression String.

the polynomial mutation [24] have been set as the genetic operators. The next subsections describe these operators. The baseline real-coded generational GA uses a rank-based selection and elitism (the best element is always copied into the next generation avoiding the loss of good solutions).

6.1. SBX Crossover. SBX is a crossover based on the properties presented by the one-point crossover in binary-coded. These properties are as follows:

- (i) Average: the average of the parents is equal to the average of the children after the crossover operation;
- (ii) Spread factor β : most of the crossover events result in a spread factor of $\beta \approx 1$, that is, children tend to be close to their parents.

To maintain the average property, the offspring values are calculated as

$$\begin{aligned} \text{ch}_1 &= \bar{q} - \frac{1}{2}\beta(p_2 - p_1) \\ \text{ch}_2 &= \bar{q} + \frac{1}{2}\beta(p_2 - p_1), \end{aligned} \quad (30)$$

where $\bar{q} = (1/2)(p_1 + p_2)$, $p_2 > p_1$, p_1 and p_2 are the parents and, ch_1 and ch_2 are the offspring. It can be observed that $\text{ch} = \bar{q}$. The parameter β is a random variable obtained from the following probabilistic distribution function:

$$f(\beta) = \begin{cases} 0.5(\eta + 1)\beta^\eta, & \text{if } \beta \leq 1, \\ 0.5(\eta + 1)\frac{1}{\beta^{\eta+2}} & \text{otherwise,} \end{cases} \quad (31)$$

where small values for η generate offspring which are more similar to their parents and, on the other hand, great values generate offspring which are quite different from their parents. In this work, we used $\eta = 2$, a value that works well according to the original reference [23].

6.2. Polynomial Mutation. Polynomial mutation is an operator that uses polynomial probability distribution to perturb a solution in a parent's proximity. Given a parent solution

$p \in [a, b]$, the polynomial mutation solution p' is created by using a random number $r \in [0, 1]$, for a given variable, as follows:

$$p'_i = \begin{cases} p_i + \bar{\delta}_L (p_i - x_i^{(L)}), & \text{for } r \leq 0.5, \\ p_i + \bar{\delta}_R (x_i^{(R)} - p_i) & \text{for } r > 0.5, \end{cases} \quad (32)$$

where $\bar{\delta}_L$ and $\bar{\delta}_R$ can be calculated as follows:

$$\begin{aligned} \bar{\delta}_L &= (2r)^{(1/1+\eta_m)} - 1, & \text{for } r \leq 0.5 \\ \bar{\delta}_R &= 1 - (2(1-r))^{(1/1+\eta_m)}, & \text{for } r > 0.5. \end{aligned} \quad (33)$$

It is important to note that no value outside the range $[a, b]$ is created by the polynomial mutation. In this work, we used $\eta_m = 100$, a value that works well according to the original reference [24].

7. Numerical Experiments

In this section, we present the results obtained by the current version of the IED method which uses genetic algorithms for minimizing the Lagrangian.

Regarding the function σ (see (1)), we have used seven norms in our first set of numerical experiments, namely, $\|\cdot\|_1$, $\|\cdot\|_2$, $\|\cdot\|_\infty$, $e^{\|\cdot\|_2} - 1$, $e^{\|\cdot\|_1} - 1$, $\sqrt{m} (e^{\|\cdot\|_\infty} - 1)$, and $\max(\|\cdot\|_2, \|\cdot\|_2^2)$. Usually, those norms are used in order to check the efficiency of algorithm [1, 2, 25]. Therefore, we have kept the norms to compare the results obtained by our algorithm.

With respect to the matrix A , we have considered $A = I$ which corresponds to the augmented Lagrangian case.

Inequality constraints $g(x) \leq 0$ have been replaced with their nonsmooth equivalent $\max\{g(x), 0\} = 0$ so that the problems have been converted into equality constrained optimization problems.

As the stopping criterion for IED we have used $\|g(x)\| \leq 10^{-6}$. A total of 10 independent runs have been executed by the IED + GA, extracting the best solution, the worst solution, the mean of the solutions, and the standard deviation (sd), for the first set of experiments. For the second set, we have executed 30 independent runs.

The algorithms have been implemented in Matlab - (R2012b) - in a microcomputer core (TM) i7 of 2.60 GHz with 8.00 GB of RAM.

This section is divided into two sets of test problems.

7.1. The First Set of Numerical Experiments. This set of numerical experiments corresponds to the analysis of 20 constrained optimization problems extracted from the Hock and Schittkowski collection [26] and all of them refer to minimization problems. They are described below were the objective $f(x)$ and the constraints are presented.

GLR-PI-1:

$$\begin{aligned} f(x) = & -32,174 \left(255 \right. \\ & \cdot \ln \left(\frac{(x_1 + x_2 + x_3 + 0,03)}{(0,09x_1 + x_2 + x_3 + 0,03)} \right) + 280 \\ & \cdot \ln \left(\frac{(x_2 + x_3 + 0,03)}{(0,07x_2 + x_3 + 0,03)} \right) + 290 \\ & \left. \cdot \ln \left(\frac{(x_3 + 0,03)}{(0,13x_3 + 0,03)} \right) \right) \end{aligned} \quad (34)$$

subject to

$$\begin{aligned} x_1 + x_2 + x_3 &= 0 \\ 0 \leq x_i &\leq 1, \quad i = 1, 2, 3 \end{aligned} \quad (35)$$

GQR-PI-1:

$$\begin{aligned} f(x) = & -75,196 + 3,8112x_1 + 0,0020567x_1^3 \\ & - 1,0345 (e - 5) x_1^4 + 6,8306x_2 \\ & - 0,030234x_1x_2 + 1,28134 (e - 3) x_2x_1^2 \\ & + 2,266 (e - 7) x_1^4x_2 - 0,25645x_2^2 \\ & + 0,0034604x_2^3 - 1,3514 (e - 5) x_2^4 \\ & + \frac{28,106}{(x_2 + 1)} + 5,2375e - 6x_1^2x_2^2 \\ & + 6,3 (e - 8) x_1^3x_2^2 - 7 (e - 10) x_1^3x_2^3 \\ & - 3,405 (e - 4) x_1x_2^2 + 1,6638 (e - 6) x_1x_2^3 \\ & + 2,8673 \exp(0,0005x_1x_2) \\ & - 3,5256 (e - 5) x_1^3x_2 \end{aligned} \quad (36)$$

subject to

$$\begin{aligned} x_1x_2 - 700 &\geq 0 \\ x_2 - \frac{x_1^2}{125} &\geq 0 \\ (x_2 - 50)^2 - 5(x_1 - 55) &\geq 0 \\ 0 \leq x_1 &\leq 75 \\ 0 \leq x_2 &\leq 65 \end{aligned} \quad (37)$$

PPR-PI-2:

$$\begin{aligned} f(x) = & 5x_1 + \frac{50000}{x_1} + 20x_2 + \frac{72000}{x_2} + 10x_3 \\ & + \frac{144000}{x_3} \end{aligned} \quad (38)$$

subject to

$$1 - \frac{4}{x_1} - \frac{32}{x_2} - \frac{120}{x_3} \quad (39)$$

$$1 \cdot 10^{-5} \leq x_i, \quad i = 1, 2, 3$$

PQR-TI-7:

$$f(x) = -x_1 x_2 x_3 \quad (40)$$

subject to

$$-x_1^2 - 2x_2^2 - 4x_3^2 + 48 \geq 0 \quad (41)$$

SQR-PI-1:

$$f(x) = \sum_{i=1}^{44} f_i(x)^2, \quad (42)$$

$$f_i(x) = b_i - x_1 - (0,49 - x_1) \exp(-x_2(a_i - 8)), \\ i = 1, \dots, 44$$

subject to

$$0,49x_2 - x_1 x_2 - 0,09 \geq 0 \\ 0,4 \leq x_1 \\ -4 \leq x_2 \quad (43)$$

LGR-PI-1:

$$f(x) = 0,2x_3 - 0,8x_1 \quad (44)$$

subject to

$$x_2 - \exp(x_1) \geq 0 \\ x_3 - \exp(x_2) \geq 0 \\ 0 \leq x_1 \leq 100 \\ 0 \leq x_2 \leq 100 \\ 0 \leq x_3 \leq 10 \quad (45)$$

QBR-TI-1:

$$f(x) = x_2 + 10^{-5}(x_2 - x_1)^2 \quad (46)$$

subject to

$$0 \leq x_2 \quad (47)$$

PBR-TI-1:

$$f(x) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2 \quad (48)$$

subject to

$$-x_2 - 1,5 \leq 0 \quad (49)$$

PBR-TI-2:

$$f(x) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2 \quad (50)$$

subject to

$$1,5 \leq x_2 \quad (51)$$

QQR-PI-3:

$$f(x) = (x_1 - x_2)^2 + \frac{(x_1 + x_2 - 10)^2}{9} + (x_3 - 5)^2 \quad (52)$$

subject to

$$48 - x_1^2 - x_2^2 - x_3^2 \geq 0 \\ -4,5 \leq x_i \leq 4,5, \quad i = 1, 2 \\ -5 \leq x_3 \leq 5 \quad (53)$$

QQR-TI-6:

$$f(x) = (x_1 - 2)^2 + (x_2 - 1)^2 \quad (54)$$

subject to

$$x_1^2 - x_2 \leq 0 \\ x_1 + x_2 - 2 \leq 0 \quad (55)$$

PLR-TI-1:

$$f(x) = \frac{1}{(27\sqrt{3})((x_1 - 3)^2 - 9)x_2^3} \quad (56)$$

subject to

$$\frac{x_1}{\sqrt{3}} - x_2 \geq 0 \\ x_1 + \sqrt{3}x_2 \geq 0 \\ -x_1 + \sqrt{3}x_2 - 6 \geq 0 \\ 0 \leq x_1 \\ 0 \leq x_2 \quad (57)$$

PBR-TI-3:

$$f(x) = \frac{1}{3(x_1 + 1)^3} + x_2 \quad (58)$$

subject to

$$1 \leq x_1 \\ 0 \leq x_2 \quad (59)$$

QLR-TI-1:

$$f(x) = 0,01x_1^2 + x_2^2 - 100 \quad (60)$$

subject to

$$\begin{aligned} 10x_1 - x_2 - 10 &\geq 0 \\ 2 &\leq x_1 \leq 50 \\ -50 &\leq x_2 \leq 50 \end{aligned} \quad (61)$$

PQR-TI-1:

$$f(x) = 100(x_2 - x_1^2)^2 + (1 - x_1)^2 \quad (62)$$

subject to

$$\begin{aligned} x_1x_2 - 1 &\geq 0 \\ x_1 + x_2^2 &\geq 0 \\ x_1 &\leq 0,5 \end{aligned} \quad (63)$$

QQR-TI-3:

$$f(x) = 0,5x_1^2 + x_2^2 - x_1x_2 - 7x_1 - 7x_2 \quad (64)$$

subject to

$$-25 + 4x_1^2 + x_2^2 \leq 0 \quad (65)$$

GBR-TI-1:

$$\begin{aligned} f(x) = \sin(x_1 + x_2) + (x_1 - x_2)^2 - 1,5x_1 + 2,5x_2 \\ + 1 \end{aligned} \quad (66)$$

subject to

$$\begin{aligned} -1,5 &\leq x_1 \leq 4 \\ -3 &\leq x_2 \leq 3 \end{aligned} \quad (67)$$

PQR-TI-4:

$$f(x) = (x_1 - 10)^3 + (x_2 - 20)^3 \quad (68)$$

subject to

$$\begin{aligned} -100 + (x_1 - 5)^2 + (x_2 - 5)^2 &\geq 0 \\ -(x_2 - 5)^2 - (x_1 - 6)^2 + 82,81 &\geq 0 \\ 13 &\leq x_1 \leq 100 \\ 0 &\leq x_2 \leq 100 \end{aligned} \quad (69)$$

QQR-TI-2:

$$f(x) = (x_1 - 5)^2 + x_2^2 - 25 \quad (70)$$

subject to

$$-x_1^2 + x_2^2 \geq 0 \quad (71)$$

QPR-TI-1:

$$f(x) = (x_1 - 2)^2 + x_2^2 \quad (72)$$

subject to

$$\begin{aligned} (1 - x_1)^3 - x_2 &\geq 0 \\ 0 &\leq x_1 \\ 0 &\leq x_2 \end{aligned} \quad (73)$$

7.1.1. *Comparison of Results of the First Set of Numerical Experiments.* Table 1 provides the initial guesses c_0 and u_0 used by IED + fmin and IED + GA, the size of the population, the number of generations, and bound constraints used by the Genetic Algorithm. The table also shows the optimal values.

For solving the first six problems, we have kept the initial guesses used by the IED in its previous version as described in [1]. For the rest of the problems, c_0 and u_0 were randomly chosen.

One can notice from Tables 2–21 that IED + GA has found the solution for all the problems regardless of the norms used whereas IED + fmin failed some of them.

It is interesting to highlight that the standard deviation of all the problems is small. This shows robustness of the version of the IED method proposed in this work.

Tables 2–21 show the results of the problems obtained by both versions of the algorithm. For the IED + GA algorithm, we present the best and the worst results, the mean and the standard deviation. Cases where a solution has been not found are indicated by a line struck through the entry of the table.

7.2. *The Second Set of Numerical Experiments.* In this section the algorithm proposed is applied to solve 4 real-world mechanical engineering optimization problems, largely used as a test-bed in the literature.

The Tension/Compression String Design. In this problem, the objective is to minimize the volume V of a coil spring, depicted in the Figure 8, under a constant tension/compression load. There are three design variables to be considered in this problem: The number $x_1 = N$ of active coils of the spring, the winding diameter $x_2 = D$ and the wire diameter $x_3 = d$. The volume of the coil to be minimized is written as [27]

$$V(x) = (x_1 + 2)x_2x_3^2 \quad (74)$$

and is subject to the constraints

$$g_1(x) = 1 - \frac{x_2^3x_1}{71785x_3^4} \leq 0$$

$$g_2(x) = \frac{4x_2^2 - x_3x_2}{12566(x_2x_3^3 - x_3^4)} + \frac{1}{5108x_3^2} - 1 \leq 0$$

TABLE 1: Parameters of the IED method.

Test-problem	c_0	u_0	Population Size	Generations	Lower Bound	Upper Bound	Optimal Value
GLR-PI-1	50	[0 0 0 0]	50	300	[0.01, 0.01, 0.01]	[1, 1, 1]	-2.6273×10^4
GQR-PI-1	1	[1 1 1 1 1 1 1]	50	100	[10, 10]	[60, 60]	-7.8028
PPR-PI-2	0.1	[1 1 1 1]	50	200	[0, 0, 0]	[300, 300, 300]	6.2998×10^3
PQR-TI-7	1.5	[1 1 1 1 1 1 1]	50	100	[-5, -5, -5]	[5, 5, 5]	-22.6274
SQR-PI-1	0.2	[0 0 0]	50	100	[-10, -10]	[10, 10]	0.0285
LGR-PI-1	0.1	[0 0 0 0 0 0 0 0]	50	100	[0, 0, 0]	[5, 5, 5]	0.5181
QBR-TI-1	1	[1 1 1 1 1 1 1 1]	50	100	[0, 0, 0]	[5, 5, 5]	0
PBR-TI-1	1	[1]	50	100	[-2, -2]	[2, 2]	0
PBR-TI-2	1	[1]	50	100	[-2, -2]	[2, 2]	0.0504
QQR-PI-3	1	[1 1 1 1 1 1 1]	50	100	[0, 0, 0]	[5, 5, 5]	0.9535
QQR-TI-6	1	[1 1]	50	100	[-2, -2]	[2, 2]	1
PLR-TI-1	1	[1 1 1 1 1]	50	100	[1, 1]	[4, 4]	-1
PBR-TI-3	1	[1 1]	50	100	[-1, -1]	[2, 2]	2.6666
QLR-TI-1	1	[1 1 1 1 1]	50	100	[-1, -1]	[3, 3]	-99.96
PQR-TI-1	5	[1 1 1]	50	100	[0, 0]	[3, 3]	306.5
QQR-TI-3	0.1	[1]	20	50	[1, 1]	[4, 4]	-30
GBR-TI-1	1	[1 1 1 1]	50	100	[-2, -2]	[0, 0]	-1.9132
PQR-TI-4	1	[1 1 1 1 1 1]	50	100	[0, 0]	[15, 15]	-6961.8138
QQR-TI-2	1	[1]	20	50	[0, 0]	[2, 2]	-8.4984
QPR-TI-1	0.1	[1 1 1]	50	100	[-1, -1]	[2, 2]	1

TABLE 2: Comparison of the results obtained by both versions of the IED for problem GLR-PI-1.

$\sigma(\cdot)$	GLR-PI-1				
	IED + fmin result	best	worst	IED + GA mean	sd
$\ \cdot\ _1$	-2.6273×10^4	-2.6272×10^4	-2.6267×10^4	-2.6271×10^4	2.0661
$\ \cdot\ _2$	-2.6273×10^4	-2.6272×10^4	-2.6267×10^4	-2.6271×10^4	2.0661
$\sqrt{m} \ \cdot\ _\infty$	—	-2.6272×10^4	-2.6267×10^4	-2.6271×10^4	2.0661
$e^{\ \cdot\ _2} - 1$	-2.6273×10^4	-2.6273×10^4	-2.6270×10^4	-2.6272×10^4	0.8555
$e^{\ \cdot\ _1} - 1$	-2.6273×10^4	-2.6273×10^4	-2.6270×10^4	-2.6272×10^4	0.8555
$\sqrt{m}(e^{\ \cdot\ _\infty} - 1)$	—	-2.6272×10^4	-2.6225×10^4	-2.6261×10^4	14.5763
$\max(\ \cdot\ _2, \ \cdot\ _2^2)$	—	-2.6273×10^4	-2.4727×10^4	-2.6049×10^4	477.8463

TABLE 3: Comparison of the results obtained by both versions of the IED for problem GQR-PI-1.

$\sigma(\cdot)$	GQR-PI-1				
	IED + fmin result	best	worst	IED + GA mean	sd
$\ \cdot\ _1$	—	-7.7978	-6.7495	-7.5575	0.4268
$\ \cdot\ _2$	—	-7.7978	-6.7495	-7.5572	0.4276
$\sqrt{m} \ \cdot\ _\infty$	-7.8028	-7.7984	-6.7495	-7.0554	0.4927
$e^{\ \cdot\ _2} - 1$	-7.8028	-7.7925	-6.7495	-7.4544	0.4872
$e^{\ \cdot\ _1} - 1$	-7.8028	-7.8012	-7.7443	-7.7801	0.0171
$\sqrt{m}(e^{\ \cdot\ _\infty} - 1)$	-7.8028	-7.7985	-6.7495	-7.4601	0.4914
$\max(\ \cdot\ _2, \ \cdot\ _2^2)$	-7.8028	-7.8024	-6.7495	-7.3560	0.5228

$$g_3(x) = 1 - \frac{140.45x_3}{x_2^2x_1} \leq 0$$

$$g_4(x) = \frac{x_2 + x_3}{1.5} - 1 \leq 0,$$

(75)

where

$$2 \leq x_1 \leq 15$$

$$0.25 \leq x_2 \leq 1.3$$

$$0.05 \leq x_3 \leq 2.$$

(76)

TABLE 4: Comparison of the results obtained by both versions of the IED for problem PPR-P1-2.

$\sigma(\cdot)$	PPR-P1-2				
	IED + fmin	IED + GA			
	result	best	worst	mean	sd
$\ \cdot\ _1$	6.2998×10^3	6.3002×10^3	6.3018×10^3	6.3008×10^3	0.4452
$\ \cdot\ _2$	6.2998×10^3	6.3002×10^3	6.3018×10^3	6.3008×10^3	0.4452
$\sqrt{m} \ \cdot\ _\infty$	6.2998×10^3	6.3001×10^3	6.3035×10^3	6.3015×10^3	1.0803
$e^{\ \cdot\ _2} - 1$	—	6.3000×10^3	6.3023×10^3	6.3007×10^3	0.7107
$e^{\ \cdot\ _1} - 1$	—	6.3000×10^3	6.3023×10^3	6.3007×10^3	0.7107
$\sqrt{m}(e^{\ \cdot\ _\infty} - 1)$	—	6.3000×10^3	6.3053×10^3	6.3012×10^3	1.6442
$\max(\ \cdot\ _2, \ \cdot\ _2^2)$	6.2999×10^3	6.2999×10^3	6.3045×10^3	6.3013×10^3	1.5406

TABLE 5: Comparison of the results obtained by both versions of the IED for problem PQR-T1-7.

$\sigma(\cdot)$	PQR-T1-7				
	IED + fmin	IED + GA			
	result	best	worst	mean	sd
$\ \cdot\ _1$	—	-22.5467	-22.1224	-22.4160	0.1450
$\ \cdot\ _2$	—	-22.6033	-22.1466	-22.4274	0.1383
$\sqrt{m} \ \cdot\ _\infty$	-22.6274	-22.6233	-22.3691	-22.5153	0.1151
$e^{\ \cdot\ _2} - 1$	-22.6262	-22.5424	-22.0668	-22.3474	0.1526
$e^{\ \cdot\ _1} - 1$	-22.6270	-22.5468	-21.9238	-22.2471	0.1891
$\sqrt{m}(e^{\ \cdot\ _\infty} - 1)$	-22.6274	-22.5215	21.6514	-17.8111	13.8676
$\max(\ \cdot\ _2, \ \cdot\ _2^2)$	-22.6156	-22.5704	-22.0331	-22.3381	0.1683

TABLE 6: Comparison of the results obtained by both versions of the IED for problem SQR-P1-1.

$\sigma(\cdot)$	SQR-P1-1				
	IED + fmin	IED + GA			
	result	best	worst	mean	sd
$\ \cdot\ _1$	0.0285	0.0285	0.0287	0.0286	8.4993×10^{-5}
$\ \cdot\ _2$	0.0285	0.0285	0.0287	0.0286	8.2863×10^{-5}
$\sqrt{m} \ \cdot\ _\infty$	0.0285	0.0285	0.0293	0.0287	2.3577×10^{-4}
$e^{\ \cdot\ _2} - 1$	0.0285	0.0285	0.0293	0.0287	5.7703×10^{-4}
$e^{\ \cdot\ _1} - 1$	0.0285	0.0285	0.0306	0.0289	6.5958×10^{-4}
$\sqrt{m}(e^{\ \cdot\ _\infty} - 1)$	0.0285	0.0285	0.0290	0.0286	1.8364×10^{-4}
$\max(\ \cdot\ _2, \ \cdot\ _2^2)$	0.0285	0.0285	0.0289	0.0286	1.2819×10^{-4}

TABLE 7: Comparison of the results obtained by both versions of the IED for problem LGR-P1-1.

$\sigma(\cdot)$	LGR-P1-1				
	IED + fmin	IED + GA			
	result	best	worst	mean	sd
$\ \cdot\ _1$	0.5187	0.5234	0.5407	0.5314	0.0061
$\ \cdot\ _2$	—	0.5216	0.5427	0.5301	0.0083
$\sqrt{m} \ \cdot\ _\infty$	0.5228	0.5213	0.5726	0.5364	0.0157
$e^{\ \cdot\ _2} - 1$	0.5199	0.5213	0.5726	0.5364	0.0157
$e^{\ \cdot\ _1} - 1$	0.5195	0.5232	0.5622	0.5384	0.0125
$\sqrt{m}(e^{\ \cdot\ _\infty} - 1)$	0.5183	0.5239	0.5677	0.5390	0.0161
$\max(\ \cdot\ _2, \ \cdot\ _2^2)$	0.5182	0.5201	0.5488	0.5313	0.0090

TABLE 8: Comparison of the results obtained by both versions of the IED for problem QBR-T1-1.

$\sigma(\cdot)$	QBR-T1-1				
	IED + fmin	IED + GA			
	result	best	worst	mean	sd
$\ \cdot\ _1$	—	5.4134×10^{-7}	1.5704×10^{-4}	6.7404×10^{-5}	5.3498×10^{-5}
$\ \cdot\ _2$	—	5.4134×10^{-7}	1.5704×10^{-4}	6.7404×10^{-5}	5.3498×10^{-5}
$\sqrt{m} \ \cdot\ _\infty$	—	5.4134×10^{-7}	1.5704×10^{-4}	6.7404×10^{-5}	5.3498×10^{-5}
$e^{\ \cdot\ _2} - 1$	3.668×10^{-17}	7.2718×10^{-6}	2.0872×10^{-4}	9.4788×10^{-5}	8.2756×10^{-5}
$e^{\ \cdot\ _1} - 1$	3.668×10^{-17}	7.2718×10^{-6}	2.0872×10^{-4}	9.4788×10^{-5}	8.2756×10^{-5}
$\sqrt{m}(e^{\ \cdot\ _\infty} - 1)$	3.668×10^{-17}	7.2718×10^{-6}	2.0872×10^{-4}	9.4788×10^{-5}	8.2756×10^{-5}
$\max(\ \cdot\ _2, \ \cdot\ _2^2)$	2.4563×10^{-14}	5.4134×10^{-7}	1.5704×10^{-4}	6.7404×10^{-5}	5.3498×10^{-5}

TABLE 9: Comparison of the results obtained by both versions of the IED for problem PBR-T1-1.

$\sigma(\cdot)$	PBR-T1-1				
	IED + fmin	IED + GA			
	result	best	worst	mean	sd
$\ \cdot\ _1$	$-9.0273x10^{-14}$	0	0.0134	0.0027	0.0052
$\ \cdot\ _2$	$-9.0273x10^{-14}$	0	0.0134	0.0027	0.0052
$\sqrt{m} \ \cdot\ _\infty$	$-9.0273x10^{-14}$	0	0.0134	0.0027	0.0052
$e^{\ \cdot\ _2} - 1$	$-9.0273x10^{-14}$	0	0.0134	0.0027	0.0052
$e^{\ \cdot\ _1} - 1$	$-9.0273x10^{-14}$	0	0.0134	0.0027	0.0052
$\sqrt{m}(e^{\ \cdot\ _\infty} - 1)$	$-9.0273x10^{-14}$	0	0.0134	0.0027	0.0052
$\max(\ \cdot\ _2, \ \cdot\ _2^2)$	$-9.0273x10^{-14}$	0	0.0134	0.0027	0.0052

TABLE 10: Comparison of the results obtained by both versions of the IED for problem PBR-T1-2.

$\sigma(\cdot)$	PBR-T1-2				
	IED + fmin	IED + GA			
	result	best	worst	mean	sd
$\ \cdot\ _1$	0.0504	0.0506	0.0677	0.0536	0.0052
$\ \cdot\ _2$	0.0504	0.0506	0.0677	0.0536	0.0052
$\sqrt{m} \ \cdot\ _\infty$	0.0504	0.0506	0.0677	0.0536	0.0052
$e^{\ \cdot\ _2} - 1$	0.0504	0.0508	0.0839	0.0593	0.0123
$e^{\ \cdot\ _1} - 1$	0.0504	0.0508	0.0839	0.0593	0.0123
$\sqrt{m}(e^{\ \cdot\ _\infty} - 1)$	0.0504	0.0508	0.0839	0.0593	0.0123
$\max(\ \cdot\ _2, \ \cdot\ _2^2)$	0.0504	0.0505	0.0790	0.0545	0.0088

TABLE 11: Comparison of the results obtained by both versions of the IED for problem QQR-P1-3.

$\sigma(\cdot)$	QQR-P1-3				
	IED + fmin	IED + GA			
	result	best	worst	mean	sd
$\ \cdot\ _1$	0.9535	0.9576	0.9837	0.9676	0.0099
$\ \cdot\ _2$	0.9535	0.9565	0.9853	0.9637	0.0890
$\sqrt{m} \ \cdot\ _\infty$	0.9535	0.9550	0.9881	0.9665	0.0115
$e^{\ \cdot\ _2} - 1$	0.9535	0.9550	0.9684	0.9603	0.0039
$e^{\ \cdot\ _1} - 1$	0.9535	0.9541	0.9860	0.9652	0.0096
$\sqrt{m}(e^{\ \cdot\ _\infty} - 1)$	-25.0694	0.9558	0.9683	0.9628	0.0040
$\max(\ \cdot\ _2, \ \cdot\ _2^2)$	0.9535	0.9568	1.0124	0.9708	0.0185

TABLE 12: Comparison of the results obtained by both versions of the IED for problem QQR-T1-6.

$\sigma(\cdot)$	QQR-T1-6				
	IED + fmin		IED + GA		
	result	best	worst	mean	sd
$\ \cdot\ _1$	1	1.0002	1.0185	1.0052	0.0057
$\ \cdot\ _2$	1	1.0023	1.0179	1.0101	0.0044
$\sqrt{m} \ \cdot\ _\infty$	1	1.0003	1.0404	1.0112	0.0125
$e^{\ \cdot\ _2} - 1$	1	1.0005	1.0287	1.0126	0.0093
$e^{\ \cdot\ _1} - 1$	1	1.0005	1.0400	1.0122	0.0137
$\sqrt{m}(e^{\ \cdot\ _\infty} - 1)$	1	1.0002	1.0366	1.0133	0.0106
$\max(\ \cdot\ _2, \ \cdot\ _2^2)$	1	1.0002	1.0097	1.0046	0.0037

TABLE 13: Comparison of the results obtained by both versions of the IED for problem PLR-T1-1.

$\sigma(\cdot)$	PLR-T1-1				
	IED + fmin		IED + GA		
	result	best	worst	mean	sd
$\ \cdot\ _1$	—	-0.9976	-0.9729	-0.9915	0.0071
$\ \cdot\ _2$	—	-0.9966	-0.9817	-0.9892	0.0052
$\sqrt{m} \ \cdot\ _\infty$	—	-0.9984	-0.9663	-0.9899	0.0102
$e^{\ \cdot\ _2} - 1$	-1	-0.9974	-0.9806	-0.9916	0.0057
$e^{\ \cdot\ _1} - 1$	-1	-0.9966	-0.9805	-0.9925	0.0048
$\sqrt{m}(e^{\ \cdot\ _\infty} - 1)$	-1	-0.9993	-0.9793	-0.9926	0.0061
$\max(\ \cdot\ _2, \ \cdot\ _2^2)$	-1	-0.9996	-0.9758	-0.9920	0.0073

TABLE 14: Comparison of the results obtained by both versions of the IED for problem PBR-T1-3.

$\sigma(\cdot)$	PBR-T1-3				
	IED + fmin		IED + GA		
	result	best	worst	mean	sd
$\ \cdot\ _1$	—	2.6686	2.7135	2.6827	0.0134
$\ \cdot\ _2$	—	2.6698	2.6855	2.6743	0.0050
$\sqrt{m} \ \cdot\ _\infty$	—	2.6697	2.6909	2.6784	0.0079
$e^{\ \cdot\ _2} - 1$	2.6667	2.6692	2.6947	2.6784	0.0082
$e^{\ \cdot\ _1} - 1$	2.6667	2.6685	2.7022	2.6809	0.0110
$\sqrt{m}(e^{\ \cdot\ _\infty} - 1)$	2.6667	2.6687	2.7059	2.6909	0.0125
$\max(\ \cdot\ _2, \ \cdot\ _2^2)$	2.6667	2.6684	2.7125	2.6847	0.0150

TABLE 15: Comparison of the results obtained by both versions of the IED for problem QLR-T1-1.

$\sigma(\cdot)$	QLR-T1-1				
	IED + fmin		IED + GA		
	result	best	worst	mean	sd
$\ \cdot\ _1$	-99.9600	-99.9600	-99.9597	-99.9599	8.9350×10^{-5}
$\ \cdot\ _2$	-99.9600	-99.9600	-99.9595	-99.9598	1.4311×10^{-4}
$\sqrt{m} \ \cdot\ _\infty$	-99.9600	-99.9600	-99.9599	-99.9599	5.1418×10^{-5}
$e^{\ \cdot\ _2} - 1$	-99.9600	-99.9600	-99.9594	-99.9599	1.5289×10^{-4}
$e^{\ \cdot\ _1} - 1$	-99.9600	-99.9600	-99.9595	-99.9599	1.3253×10^{-4}
$\sqrt{m}(e^{\ \cdot\ _\infty} - 1)$	-99.9600	-99.9600	-99.9594	-99.9598	1.8280×10^{-4}
$\max(\ \cdot\ _2, \ \cdot\ _2^2)$	-99.9600	-99.9599	-99.9598	-99.9599	3.7864×10^{-5}

TABLE 16: Comparison of the results obtained by both versions of the IED for problem PQR-T1-1.

$\sigma(\cdot)$	PQR-T1-1				
	IED + fmin	IED + GA			
	result	best	worst	mean	sd
$\ \cdot\ _1$	306.5	306.5146	326.6933	310.9483	6.2614
$\ \cdot\ _2$	306.5	308.7375	329.3517	312.3063	6.1269
$\sqrt{m} \ \cdot\ _\infty$	306.5	307.5067	327.2424	311.6510	5.8228
$e^{\ \cdot\ _2} - 1$	306.5	308.9435	317.7634	312.9006	2.7421
$e^{\ \cdot\ _1} - 1$	306.5	306.5408	315.6229	310.7872	3.0505
$\sqrt{m}(e^{\ \cdot\ _\infty} - 1)$	306.5	307.8027	316.7068	312.1473	2.7962
$\max(\ \cdot\ _2, \ \cdot\ _2^2)$	306.5	307.9935	315.4329	311.4265	2.4128

TABLE 17: Comparison of the results obtained by both versions of the IED for problem QQR-T1-3.

$\sigma(\cdot)$	QQR-T1-3				
	IED + fmin	IED + GA			
	result	best	worst	mean	sd
$\ \cdot\ _1$	—	-29.9998	-29.8286	-29.9714	0.0532
$\ \cdot\ _2$	—	-29.9998	-29.8286	-29.9714	0.532
$\sqrt{m} \ \cdot\ _\infty$	—	-29.9998	-29.8286	-29.9714	0.0532
$e^{\ \cdot\ _2} - 1$	-30	-29.9972	-29.3533	-29.8566	0.2162
$e^{\ \cdot\ _1} - 1$	-30	-29.9972	-29.3533	-29.8566	0.2162
$\sqrt{m}(e^{\ \cdot\ _\infty} - 1)$	-30	-29.9972	-29.3533	-29.8566	0.2162
$\max(\ \cdot\ _2, \ \cdot\ _2^2)$	-30	-29.9960	-29.2340	-29.8780	0.2316

TABLE 18: Comparison of the results obtained by both versions of the IED for problem GBR-T1-1.

$\sigma(\cdot)$	GBR-T1-1				
	IED + fmin	IED + GA			
	result	best	worst	mean	sd
$\ \cdot\ _1$	-1.9132	-1.9132	-1.9132	-1.9132	5.1999×10^{-7}
$\ \cdot\ _2$	-1.9132	-1.9132	-1.9132	-1.9132	5.1999×10^{-7}
$\sqrt{m} \ \cdot\ _\infty$	-1.9132	-1.9132	-1.9132	-1.9132	1.4981×10^{-6}
$e^{\ \cdot\ _2} - 1$	-1.9132	-1.9132	-1.9132	-1.9132	9.3127×10^{-7}
$e^{\ \cdot\ _1} - 1$	-1.9132	-1.9132	-1.9132	-1.9132	9.3127×10^{-7}
$\sqrt{m}(e^{\ \cdot\ _\infty} - 1)$	-1.9132	-1.9132	-1.9132	-1.9132	1.2166×10^{-6}
$\max(\ \cdot\ _2, \ \cdot\ _2^2)$	-1.9132	-1.9132	-1.9132	-1.9132	5.1999×10^{-7}

TABLE 19: Comparison of the results obtained by both versions of the IED for problem PQR-T1-4.

$\sigma(\cdot)$	PQR-T1-4				
	IED + fmin	IED + GA			
	result	best	worst	mean	sd
$\ \cdot\ _1$	—	-6791.5	173850	1237.6	5.6747×10^4
$\ \cdot\ _2$	—	-6936.9	1792.1	-5153.5	2.8982×10^3
$\sqrt{m} \ \cdot\ _\infty$	—	-6800.2	-2548.9	-5962.7	1.3694×10^3
$e^{\ \cdot\ _2} - 1$	-6961.8	-6874.3	-6258.0	-6737.2	188.5621
$e^{\ \cdot\ _1} - 1$	-6961.8	-6915.8	-1642.2	-6165.1	1606.1
$\sqrt{m}(e^{\ \cdot\ _\infty} - 1)$	-6961.8	-6777.9	-6159.4	-6590.4	222.8402
$\max(\ \cdot\ _2, \ \cdot\ _2^2)$	-6961.8	-6958.0	-1735.3	-6274.8	1609.0

TABLE 20: Comparison of the results obtained by both versions of the IED for problem QQR-TI-2.

$\sigma(\cdot)$	QQR-TI-2				
	IED + fmin result	best	worst	IED + GA mean	sd
$\ \cdot\ _1$	-8.4985	-8.4958	-8.3391	-8.4729	0.0481
$\ \cdot\ _2$	-8.4985	-8.4958	-8.3391	-8.4729	0.0481
$\sqrt{m} \ \cdot\ _\infty$	-8.4985	-8.4958	-8.3391	-8.4729	0.0481
$e^{\ \cdot\ _2} - 1$	-8.4985	-8.4958	-7.5694	-8.2675	0.3201
$e^{\ \cdot\ _1} - 1$	-8.4985	-8.4958	-7.5694	-8.2675	0.3201
$\sqrt{m}(e^{\ \cdot\ _\infty} - 1)$	-8.4985	-8.4958	-7.5694	-8.2675	0.3201
$\max(\ \cdot\ _2, \ \cdot\ _2^2)$	-8.4985	-8.4980	-8.0434	-8.3737	0.1564

TABLE 21: Comparison of the results obtained by both versions of the IED for problem QPR-TI-1.

$\sigma(\cdot)$	QPR-TI-1				
	IED + fmin result	best	worst	IED + GA mean	sd
$\ \cdot\ _1$	—	1.1349	1.4578	1.2706	0.0846
$\ \cdot\ _2$	—	1.0632	1.4943	1.2132	0.1356
$\sqrt{m} \ \cdot\ _\infty$	—	1.1317	1.6091	1.2670	0.1421
$e^{\ \cdot\ _2} - 1$	0.9978	1.1316	1.4839	1.2715	0.1220
$e^{\ \cdot\ _1} - 1$	—	1.0729	1.5440	1.2488	0.1375
$\sqrt{m}(e^{\ \cdot\ _\infty} - 1)$	0.9979	1.0818	1.4799	1.2300	0.1555
$\max(\ \cdot\ _2, \ \cdot\ _2^2)$	0.9987	1.0967	1.6843	1.2762	0.1853

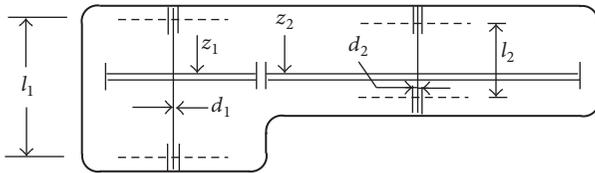


FIGURE 9: The Speed Reducer design.

The Speed Reducer Design. The objective of this problem is to minimize the weight W of the speed reducer [27] shown in the Figure 9. The design variables are the face width ($x_1 = b$), the module of teeth ($x_2 = m$), the number of teeth on pinion ($x_3 = n$), the length of the shaft 1 between the bearings ($x_4 = l_1$), the length of the shaft 2 between the bearings ($x_5 = l_2$), the diameter of the shaft 1 ($x_6 = d_1$), and, finally, the diameter of the shaft 2 ($x_7 = d_2$). The third variable is integer and all the others are continuous. The constraints include limitations on the bending and surface stress of the gear teeth, transverse deflections of the shafts 1 and 2 generated by the transmitted force, and, finally, the stress in the shafts 1 and 2. The weight of the speed reducer, to be minimized, is given by

$$\begin{aligned}
 W(x) &= 0.7854x_1x_2^2(3.3333x_3^2 + 14.9334x_3 - 43.0934) \\
 &\quad - 1.508x_1(x_6^2 + x_7^2) + 7.477(x_6^3 + x_7^3) \\
 &\quad + 0.7854(x_4x_6^2 + x_5x_7^2)
 \end{aligned} \tag{77}$$

subject to

$$\begin{aligned}
 g_1(x) &= 27x_1^{-1}x_2^{-2}x_3^{-1} \leq 1 \\
 g_2(x) &= 397.5x_1^{-1}x_2^{-2}x_3^{-2} \leq 1 \\
 g_3(x) &= 1.93x_2^{-1}x_3^{-1}x_4^3x_6^{-4} \leq 1 \\
 g_4(x) &= 1.93x_2^{-1}x_3^{-1}x_5^3x_7^{-4} \leq 1 \\
 g_5(x) &= \frac{[(745x_4/x_2x_3)^2 + 16.9 \times 10^6]^{0.5}}{0.1x_6^3} \leq 1100 \\
 g_6(x) &= \frac{[(745x_5/x_2x_3)^2 + 157.5 \times 10^6]^{0.5}}{0.1x_7^3} \leq 850 \\
 g_7(x) &= x_2x_3 \leq 40 \\
 g_8(x) &= \frac{x_1}{x_2} \geq 5 \\
 g_9(x) &= \frac{x_1}{x_2} \leq 12 \\
 g_{10}(x) &= (1.5x_6 + 1.9)x_4^{-1} \leq 1 \\
 g_{11}(x) &= (1.1x_7 + 1.9)x_5^{-1} \leq 1 \\
 2.6 &\leq x_1 \leq 3.6 \\
 0.7 &\leq x_2 \leq 0.8
 \end{aligned}$$

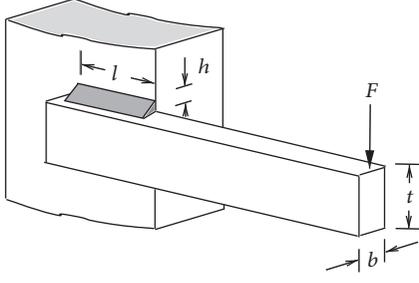


FIGURE 10: The welded beam.

$$17 \leq x_3 \leq 28$$

$$7.3 \leq x_4 \leq 8.3$$

$$7.8 \leq x_5 \leq 8.3$$

$$2.9 \leq x_6 \leq 3.9$$

$$5.0 \leq x_6 \leq 5.5.$$

(78)

The Welded Beam. This problem corresponds to the design of the welded beam [27] depicted in the Figure 10. The design variables are $\{h, l, t, b\}$, with bounds $0.125 \leq h \leq 10$, and $0.1 \leq l, t, b \leq 10$. The objective function to be minimized is the cost of the beam given as

$$C(h, l, t, b) = 1.10471h^2l + 0.04811tb(14.0 + l) \quad (79)$$

subject to

$$g_1(\tau) = 1 - \frac{\tau}{13600} \geq 0$$

$$g_2(\sigma) = 1 - \frac{\sigma}{30000} \geq 0$$

$$g_3(b, h) = b - h \geq 0 \quad (80)$$

$$g_4(P_c) = \frac{P_c}{6000} - 1 \geq 0$$

$$g_5(\delta) = 0.25 - \delta \geq 0.$$

The expressions for τ , σ , P_c , and δ involve the design variables and are given by

$$\tau = \sqrt{\tau'^2 + \tau''^2 + \frac{l\tau'\tau''}{\alpha}}$$

$$\alpha = \sqrt{0.25(l^2 + (h+t)^2)},$$

$$\sigma = \frac{504000}{t^2b}$$

$$P_c = 64746.022(1 - 0.0282346t)tb^3$$

$$\delta = \frac{2.1952}{t^3b},$$

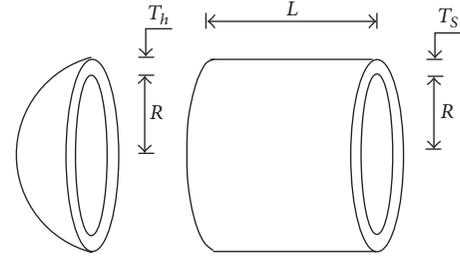


FIGURE 11: The pressure vessel.

$$\tau' = \frac{6000}{\sqrt{2hl}}$$

$$\tau'' = \frac{6000(14 + 0.5l)\alpha}{2(0.707hl(l^2/12 + 0.25(h+t)^2))}.$$

(81)

The Pressure Vessel. This problem corresponds to the weight minimization of a cylindrical pressure vessel with two spherical heads [27] as shown in Figure 11. The objective function involves four variables: the thickness of the pressure vessel (T_s), the thickness of the head (T_h), the inner radius of the vessel (R), and the length of the cylindrical component (L). Since there are two discrete variables (T_s and T_h) and two continuous variables (R and L), one has a nonlinearly constrained mixed discrete-continuous optimization problem.

The bounds of the design variables are $0.0625 \leq T_s, T_h \leq 5$ (in constant steps of 0.0625) and $10 \leq R, L \leq 200$. The design variables are given in inches and the weight is written as

$$w(T_s, T_h, R, L) = 0.6224T_sT_hRL + 1.7781T_hR^2 + 3.1661T_s^2L + 19.84T_s^2R \quad (82)$$

to be minimized subject to the constraints

$$g_1(T_s, R) = T_s - 0.0193R \geq 0$$

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

$$g_3(R, L) = \frac{\pi R^2L + 4/3\pi R^3}{1296000} - 1 \geq 0 \quad (83)$$

$$g_4(L) = -L + 240 \geq 0.$$

The first two constraints establish a lower bound to the ratios T_s/R and T_h/R , respectively. The third constraint corresponds to a lower bound for the volume of the vessel and the last one to an upper bound to length of the cylindrical component.

7.2.1. Comparison of Results of the Second Set of Numerical Experiments. Table 22 provides the initial guesses c_0 and u_0 used by IED + GA, the size of the population, and the number of generations used by the Genetic Algorithm.

Table 23 presents the best and the worst results, the mean, and the standard deviation. Tables 24–27 show the results found by the algorithm.

TABLE 22: Parameters of the IED method.

Problems	c_0	u_0	Population Size	Generations
The Tension/Compression String	10	[1 1 1 1]	100	300
The Speed Reducer	1	[1 1 1 1 1 1 1 1 1 1]	100	300
The Welded Beam	10	[1 1 1 1 1]	100	300
The Pressure Vessel	1	[1 1 1 1]	100	300

TABLE 23: Statistical Analysis obtained by IED + GA in the engineering problems.

Engineering problems	IED + GA			
	best	worst	mean	sd
The Tension/Compression String	0.0128	0.0164	0.0144	0.0009
The Speed Reducer	2996.7	3039.7	3015.0	13.7485
The Welded Beam	2.4151	5.4177	3.1995	0.7005
The Pressure Vessel	6477.2	8787.4	7454.1	636.1933

TABLE 24: Comparison for The Tension/Compression String problem.

Var.	Alg-1 [11]	Alg-2 [11]	IED + GA
x_1	11.852177	11.329555	11.184056
x_2	0.34747463	0.35603234	0.36004951
x_3	0.051301897	0.051660806	0.051889349
g_1	-0.00000012	-0.000006437	-0.003088921
g_2	-0.00000047	-0.000013709	-0.003358194
g_3	-4.03513200	-4.052324300	-4.026621039
g_4	-0.73414900	-0.728204600	-0.725374090
V	0.012668	0.012666	0.012781

TABLE 25: Comparison for the speed reducer problem.

Var.	Ref. [12]	Ref. [13]	Alg-1 [11]	Alg-2 [11]	IED + GA
x_1	3.506163	3.500000	3.500001	3.5	3.499684
x_2	0.700831	0.700000	0.700000	0.7	0.700011
x_3	17.0	17.0	17.0	17.0	17.290709
x_4	7.460181	7.300008	7.300017	7.3000035	7.302923
x_5	7.962143	7.715322	7.715326	7.7153225	7.806459
x_6	3.362900	3.350215	3.350216	3.3502147	3.350185
x_7	5.308949	5.286655	5.286654	5.2866545	5.286355
g_1	-0.077734	-0.07391524	-0.07391554	-0.07391524	-0.07386154
g_2	-0.201305	-0.19799852	-0.19799876	-0.19799852	-0.19795199
g_3	-0.474119	-0.49917084	-0.49916983	-0.49917156	-0.49856110
g_4	-0.897068	-0.90464383	-0.90464365	-0.90464383	-0.90120377
g_5	-0.011021	-0.00000023	-0.00000155	-0.000000119	0.000030538
g_6	-0.012500	-0.00000029	0.00000000	0.00000000	0.000187373
g_7	-0.702147	-0.70250000	-0.70250000	-0.702500000	-0.702495228
g_8	-0.000573	0.00000000	-0.00000029	0.00000000	0.00010613
g_9	-0.583095	-0.58333333	-0.58333320	-0.583333300	-0.583377553
g_{10}	-0.069144	-0.051326692	-0.05132753	-0.051326156	-0.051711423
g_{11}	-0.027920	-0.00000018	-0.00000077	-0.000000357	-0.011717075
V	3025.0051	2994.4717	2994.4720	2994.4712	2996.7085

TABLE 26: Comparison for the welded beam problem.

Var.	Ref. [14]	Alg-1 [11]	Alg-2 Ref. [11]	IED + GA
x_1	0.2442949	0.24432427	0.24438575	0.240091515
x_2	6.2116738	6.2201996	6.2183037	6.390631032
x_3	8.3015486	8.291464	8.291165	8.314142911
x_4	0.2443003	0.24436942	0.24438748	0.246214266
g_1	0.0004447	0.000000000	0.001953125	0.004494538
g_2	64.378068	0.001953125	0.056640625	0.012900125
g_3	0.0000054	0.000045150	0.000001728	0.006122751
g_4	0.0002553	0.029785156	1.210937500	0.024765021
g_5	0.2342937	0.234240830	0.234240280	0.234486548
V	2.38159	2.381246	2.3812175	2.4151069

TABLE 27: Comparison for the pressure vessel problem.

Var.	Ref. [13]	Ref. [14]	Alg-1 [11]	Alg-2 [11]	IED + GA
T_s	0.8125	0.8125	0.8125	0.8125	0.90625
T_h	0.4375	0.4375	0.4375	0.4375	0.46250
R	42.086994	42.0946558	42.093082	42.094967	46.189759
L	176.779128	176.684062	176.70308	176.67972	138.552311
g_1	0.000221	0.000073	0.0001035	0.000007	0.014787
g_2	0.035990	0.035917	0.0359320	0.035914	0.021849
g_3	3.219817	2.929000	0.1562500	0.0625	0.035064
g_4	63.220872	63.315938	63.296920	63.320282	101.447688
V	6061.1229	6060.187934	6060.3677	6060.138	6477.182

8. Conclusions

We have proposed a version of the IED method for nonsmooth and nonconvex optimization problems that employs a Genetic Algorithm for minimization the Lagrangian function.

We have solved twenty test problems for comparison between the first version and the current version to verify the effectiveness of the new approach proposed.

We also solved four classical engineering problems to show that the method can be applied to more complex problems.

We have evidence that shows that the more effectively we minimize the Lagrangian the better the IED method will be. Therefore, more powerful methods must be sought for this task. We also believe that the IED method can be bettered by establishing rules to update the matrices B^k and employing different types of augmented Lagrangians with different matrices A and different functions σ .

Finally, the hybridization presented in this work combines two algorithms, namely, the IED and a Genetic Algorithm. This approach frees the Genetic Algorithm of handling with constraints which is troublesome for these type of methods and, at the same time, provides IED with a good technique for minimizing the Lagrangian.

Data Availability

Data are available at https://www.dropbox.com/sh/5nahzdoy-u37ug7z/AABYgt71fc_UB1EsP38xD08fa?dl=0 or from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

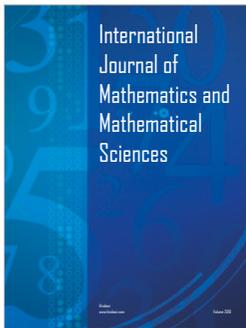
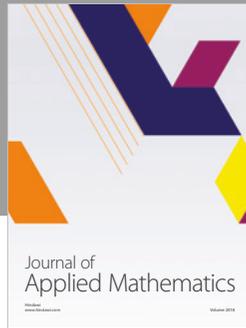
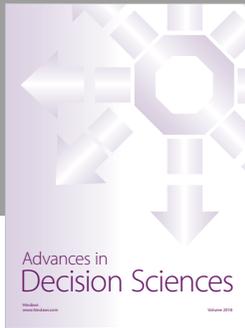
Acknowledgments

The authors thank CNPq (Grant 306186/2017-9), FAPEMIG, and FADEPE (Grant 01/17 STRICTO SENSU).

References

- [1] R. S. Burachik, W. P. Freire, and C. Y. Kaya, "Interior Epigraph Directions method for nonsmooth and nonconvex optimization via generalized augmented Lagrangian duality," *Journal of Global Optimization*, vol. 60, no. 3, pp. 501–529, 2014.
- [2] R. S. Burachik and C. Y. Kaya, "A deflected subgradient method using a general augmented Lagrangian duality with implications on penalty methods," *Springer Optimization and Its Applications*, vol. 47, pp. 109–132, 2010.
- [3] R. S. Burachik and C. Y. Kaya, "An update rule and a convergence result for a penalty function method," *Journal of Industrial and Management Optimization*, vol. 3, no. 2, pp. 381–398, 2007.
- [4] R. S. Burachik, R. N. Gasimov, N. A. Ismayilova, and C. Y. Kaya, "On a modified subgradient algorithm for dual problems via sharp augmented Lagrangian*," *Journal of Global Optimization*, vol. 34, no. 1, pp. 55–78, 2006.
- [5] R. S. Burachik, A. N. Iusem, and J. G. Melo, "A primal dual modified subgradient algorithm with sharp Lagrangian," *Journal of Global Optimization*, vol. 46, no. 3, pp. 347–361, 2010.

- [6] R. S. Burachik, A. N. Iusem, and J. G. Melo, "Duality and Exact Penalization for General Augmented Lagrangians," *Journal of Optimization Theory and Applications*, vol. 147, no. 1, pp. 125–140, 2010.
- [7] R. S. Burachik, C. Y. Kaya, and M. Mammadov, "An inexact modified subgradient algorithm for nonconvex optimization," *Computational Optimization and Applications*, vol. 45, no. 1, pp. 1–24, 2010.
- [8] R. N. Gasimov, "Augmented Lagrangian duality and nondifferentiable optimization methods in nonconvex programming," *Journal of Global Optimization*, vol. 24, no. 2, pp. 187–203, 2002.
- [9] P. W. Freire, *A Feasible Directions Algorithm for Convex Nondifferentiable Optimization [Ph.D. thesis]*, 2005.
- [10] J. Herskovits, W. P. Freire, M. Tanaka Fo, and A. Canelas, "A feasible directions method for nonsmooth convex optimization," *Structural and Multidisciplinary Optimization*, vol. 44, no. 3, pp. 363–377, 2011.
- [11] S. H. Bernardino, J. C. H. Barbosa, and C. C. A. Lemonge, "Constraint handling in genetic algorithms via artificial immune systems," in *In Late-breaking paper at Genetic and Evolutionary Computation Conference (GECCO 2006)*, 2006.
- [12] E. Mezura-Montes, C. A. C. Coello, and R. Landa-Becerra, "Engineering Optimization Using a Simple Evolutionary Algorithm," in *Proceedings of the 15th IEEE International Conference on Tools with artificial Intelligence*, pp. 149–156, November 2003.
- [13] C. A. C. Coello and N. C. Cortés, "Hybridizing a genetic algorithm with an artificial immune system for global optimization," *Engineering Optimization*, vol. 36, no. 5, pp. 607–634, 2004.
- [14] A. C. C. Lemonge and H. J. C. Barbosa, "An adaptive penalty scheme for genetic algorithms in structural optimization," *International Journal for Numerical Methods in Engineering*, vol. 59, no. 5, pp. 703–736, 2004.
- [15] P. B. Luis, F. F. Camila, P. F. Wilhelm, and J. R. Hernando, "An improvement on the ied method for solving constrained nonsmooth and nonconvex optimization problems," in *Proceedings of the In EngOpt 2016 - 5th International Conference on Engineering Optimization - Proceedings*, pp. 434–443, 2017.
- [16] M. T. Filho, *Algorithms of feasible directions to nonsmooth optimization [Ph.D. thesis]*, 2011, COPPE/UFRJ.
- [17] N. Karmitsa, M. Tanaka Filho, and J. Herskovits, "Globally Convergent Cutting Plane Method for Nonconvex Nonsmooth Minimization," *Journal of Optimization Theory and Applications*, vol. 148, no. 3, pp. 528–549, 2011.
- [18] R. Datta and K. Deb, *Evolutionary Constrained Optimization*, Springer, 2015.
- [19] J. Herskovits, "Feasible direction interior-point technique for nonlinear optimization," *Journal of Optimization Theory and Applications*, vol. 99, no. 1, pp. 121–146, 1998.
- [20] C. Darwin, *On the Origin of Species by Means of Natural Selection*, Murray, London, 1859.
- [21] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, vol. 412, Addison-Wesley Reading, Menlo Park, Calif, USA, 1989.
- [22] J. H. Holland, *Adaptation in Natural and Artificial Systems*, The University of Michigan Press, 1975.
- [23] K. Deb and R. B. Agrawal, "Simulated binary crossover for continuous search space," *Complex Systems*, vol. 9, no. 2, pp. 115–148, 1995.
- [24] K. Deb and M. Goyal, "A combined genetic adaptive search (geneas) for engineering design," *Computer Science and Informatics*, vol. 26, pp. 30–45, 1996.
- [25] R. S. Burachik, A. N. Iusem, and J. G. Melo, "An Inexact Modified Subgradient Algorithm for Primal-Dual Problems via Augmented Lagrangians," *Journal of Optimization Theory and Applications*, vol. 157, no. 1, pp. 108–131, 2013.
- [26] W. Hock and K. Schittkowski, *Test examples for nonlinear programming codes*, Lecture Notes in Economics and Mathematical Systems, Springer, New York, NY, USA, 1981.
- [27] V. S. Aragón, S. C. Esquivel, and A. C. C. Carlos, "A modified version of a t-cell algorithm for constrained optimization problems," *International Journal for Numerical Methods in Engineering*, vol. 84, no. 3, pp. 351–378, 2010.



Hindawi

Submit your manuscripts at
www.hindawi.com

