Applied Mathematics & Information Sciences
An International Journal

http://dx.doi.org/10.12785/amis/080506

Optimum Design of Parallelogram Five-bar Manipulator for Dexterous Workspace by using ELEMAEF in Differential Evolution

Miguel G. Villarreal-Cervantes, Daniel De la Cruz-Muciño and Edgar A. Portilla-Flores*

Postgraduate Department, Instituto Politécnico Nacional, Centro de Innovación y Desarrollo Tecnológico en Cómputo, 07700, Mexico

Received: 9 Aug. 2013, Revised: 6 Nov. 2013, Accepted: 7 Nov. 2013
Published online: 1 Sep. 2014

Abstract: The kinematic design of mechanism is an important stage in the design methodology. A dexterous workspace for a manipulator is an outstanding characteristic that must be considered in it. Hence, a mono-objective constraint optimization problem (MOCOP) for the kinematic design of a manipulator with three revolute joints (3R robot), that fulfils a defined dexterous workspace, is formulated. The MOCOP is solved by proposing a mechanism in the differential evolution (DE) algorithm called exhaustive local exploitation mechanism with adaptive scale factor (ELEMAEF). This mechanism exhaustively exploits a local region in the search space with the information of the base and the difference vectors of good trial vector, in an attempt to generate better individuals in the same direction. In addition, the ELEMAEF guides the evolution of the population toward a better zone without sacrificing the search capabilities of the DE algorithm. A comparison of the DE algorithm with and without the ELEMAEF for this particular design problem is presented. The use of the ELEMAEF gives a superior performance in the DE algorithm.

Keywords: Evolutionary algorithm, differential evolution, dexterous manipulator, kinematic design, parallelogram manipulator

1 Introduction

Analysis and synthesis of mechanisms [1] are the most important stages in the design methodology of parallel manipulators. The dimensional synthesis and workspace are two main characteristics which define a mechanism. They are the most studied issues in the field [2] [3]. The dimensional synthesis of a mechanism can be developed with graphical, analytical and numerical methods [4], [5]. On the other hand, the current methods for determining the manipulator's workspace and its boundary are classified as [6]: geometrical methods [7] [8], discretization methods [9] and numerical methods [10], [11]. Several robotic manipulators have a mechanism (closed kinematic chain) in their structure in order to improve the design performance. Those manipulators are called parallel one and present some advantages over serial manipulators, such as rigidity, dexterity, precision, velocity and acceleration [12]. One important characteristic that must be considered in the design of manipulators is the dexterous workspace. The dexterous workspace represents the region that can be reached by a point on the end-effector with any orientation, i.e., it is the volume or area which its end-effector can reach in the Cartesian space with different orientation [13], [14].

The design of a manipulator that meets one or several performance criteria such as trajectory accuracy, workspace, stiffness, singularity, dexterity, accuracy, etc., is a challenge because most of the performance criteria presents tradeoff among them and there is not just one solution that meets the aforementioned requirements. Hence, the manipulator design have been stated as an optimization problem where optimization techniques such as, heuristic algorithms [15], [16], [17], [18], [19],[20] and gradient based algorithms [21], have been used. Nevertheless, if the optimization problem is nonlinear or discontinuous one, gradient based algorithms are not suitable to solve the problem because they converge to local minima near the initial condition (sensitive to initial condition) [22], [23], then the design solution will perform poorly. So, it is important to have an algorithm that efficiently search in the design space to obtain a feasible solution, i.e., to obtain a set of parameters that describe the system and meet the design requirements.

* Corresponding author e-mail: mvillarrealc@ipn.mx

Heuristic optimization techniques such as evolutionary algorithms (EAs), genetic algorithms (GAs) or particle swarm optimization (PSO) have been developed to solve complex engineering optimization problems. Some advantages of these approaches are: i) These are population-based methods, therefore they can produce several possible solutions. ii) They do not require additional information to start the search, i.e. the information of the gradient, the Hessian matrix, the initial search points, etc. iii) They do not require the objective functions and constraints to be continuous and/or differentiable. Finally, iv) They can be used and/or adapted to a large set of problems, because they do not need special mathematical formulation (problem transformation) in order to obtain a set of solutions.

Since Storn and Price proposed the algorithm of Differential Evolution (DE) in the middle of 90's [24], it has proven to be a powerful computational tool to solve optimization problems. The advantages of the DE is the easy computational implementation, the great adaptability to different kinds of optimization problems and the reasonable computing time, among others. However, this algorithm does not guarantee the convergence to the global optimum.

Studies [25], [26] have established that the mutation factor and the crossover probability influence in the DE performance. The mutation factor controls the rate at which the population evolves and the crossover probability controls the probability for a component to be selected from the mutant vector. Hence, several research works improve the mutation operator since it plays a key role in the DE performance. Efforts to improve the mutation operator were addressed considering the tradeoff between convergence speed and robustness [27]. Adaptive parameter control schemes [28] were introduced in the mutation strategy. Other researches combine different mutation strategies [22], [29]. Finally, some other researchers deals with the implementation of mechanisms and its effects to ensure better exploration and exploitation capabilities in the solution space such that the convergence to the global optimum may be found [30],[31], [32].

An efficient exploration mechanism in the search space and an effective exploitation mechanism in a region of the search space, would be desirable into the optimization algorithm. The exploration can widely search different regions in it, while the exploitation accelerates the convergence to the optimum solution in the region. Several researchers have explained the relationship between the exploration and exploitation [32], [33].

In this paper, an optimization problem for the kinematic design of a three revolute joint manipulator with a parallelogram five-bar mechanism that fulfils a defined dexterous workspace is stated. A mechanism to promote the local exploitation of the individuals that present an appropriate fitness is proposed and included in the DE/Rand/1/Bin algorithm. This mechanism uses the vectors that form the mutant vector and an adaptive scale factor, in an attempt to guide the evolution of the DE algorithm to better zones. Hence, this mechanism is called Exhaustive Local Exploitation Mechanism with Adaptive Scale Factor (ELEMAEF). The DE algorithm with ELEMAEF is compared with the traditional DE/Rand/1/Bin algorithm (latter named it, DE algorithm without ELEMAEF) in order to show its performance in a particular optimization problem.

The main motivations of this work are: $1)$ the formal formulation, as an optimization problem, of the optimum design of the link lengths of a 3R manipulator which fulfills a defined dexterous workspace and $2)$ the proposal of the exhaustive local exploitation mechanism with adaptive scale factor in the DE/Rand/1/Bin which improves the DE performance for this particular problem.

The rest of the paper is organized as follows: The kinematic design of a 3R manipulator with a parallelogram five-bar mechanism is stated in section II. In section III the ELEMAEF in the differential evolution is explained. The experiments are detailed and discussed in section IV. Finally, the conclusions are commented in section V.

2 Design problem statement

The $3R$ manipulator with a parallelogram five-bar mechanism presents three degree of freedom in the joint space which provide the ability to move the tip of the end-effector (point $(\bar{x}_{i,j,k}, \bar{z}_{i,j,k}$ of the link 4 which is represented by an asterisk in Fig. 1) in the plane $X-Z$ with an orientation $\phi_{i,j,k}$ with respect to the X axis of the inertial coordinate system $X-Z$. The parallelogram five-bar mechanism, included into the $3R$ robot, achieves a higher precision and a higher stiffness than a $3R$ robot without the parallelogram five-bar mechanism [12].The $3R$ manipulator is shown in Fig. 1, where $l_i\ \forall\ \underline{i} = 1,2,\ldots,4$ is the $i-th$ link length, $(\bar{x}_{i,j,k}, \bar{z}_{i,j,k})$ and $\bar{\phi}_{i,j,k}$ are the Cartesian coordinate of the manipulator's end-effector and the angular position of the manipulator's end-effector, respectively. The desired dexterous workspace [13] is represented by an area with vertices $(\bar{x}_{d_{i,j,k}}, \bar{z}_{d_{i,j,k}})\ \forall\ i,j = 1,2$, where $k = 1,2,3$ is the $k-th$ orientation that must fulfil the end-effector for each vertex. The $k-th$ desired orientation for each vertex is represented by the angle $\phi_{d_{i,j,k}}$ as it is shown in Fig. 1.

The reachable workspace of a manipulator is the volume that its end-effector can reach in the Cartesian space. The dexterous workspace can be defined as the volume or area which its end-effector can reach in the Cartesian space with different orientation [13], [14]. Several robotic tasks require the manipulation of objects with different orientation (due to obstacles in the workspace or for positioning the end-effector tool). In this problem statement it is considered that the designer (user) requires a $3R$ robot with a parallelogram five-bar

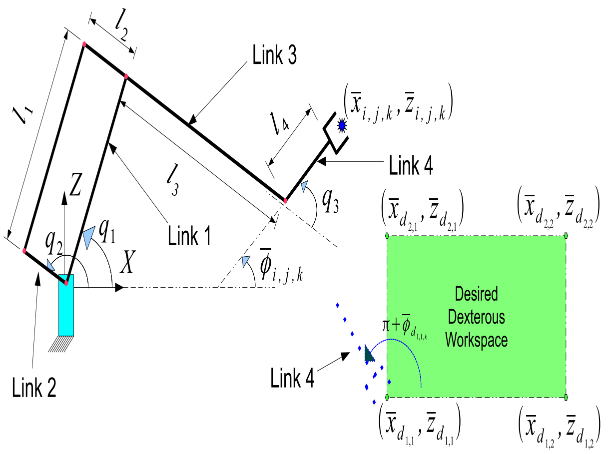


Fig. 1: Schematic diagram of the 3R manipulator with parallelogram five-bar mechanism.

mechanism to handle an object in a specific squared workspace (desired squared workspace) with different orientations, i.e. a robot with dexterous workspace is requested. The desired squared workspace is given by its vertexes. Then, the design problem is to find the link lengths of the 3R robot such that they fulfils the desired dexterous workspace subject to inherent design constraints.

In order to formulate the design problem, the following assumptions are considered:

a)The desired workspace is squared shape with four vertices $(\bar{x}_{d_{i,j,k}}, \bar{z}_{d_{i,j,k}})\ \forall\ i, j = 1,2$.

b)If the tip of the end-effector of the link 4 reaches the four vertices of the desired squared workspace with three different orientations (angle $\bar{\phi}_{i,j,k} \forall k = 1,2,3$ of the link 4), then the end-effector of the robot can reach any interior points in the desired squared workspace with at least the proposed three different orientations. Hence, a dexterous workspace is promoted.

In the next subsections, the design variables, objective function, constraints and the formal optimization problem statement for the design of the 3R robot with a desired dexterous workspace are detailed.

2.1 Design variables

In this paper the design variables of the 3R robot are the link lengths (l_1, l_2, l_3) and the angular position of the links $(q_{1_{i,j,k}}, q_{2_{i,j,k}}, q_{3_{i,j,k}})$ to reach the vertices of the desired workspace $((\bar{x}_{d_{i,j,k}}, \bar{z}_{d_{i,j,k}}))$ with k different orientations $\bar{\phi}_{d_{i,j,k}}$. Hence, the design variable vector is shown in (1) $\forall$ $i, j = 1,2$ and $k = 1,2,3$.

$$p = [l_1, l_3, l_4, q_{1_{i,j,k}}, q_{2_{i,j,k}}, q_{3_{i,j,k}}]^T \in R^{39} \tag{1}$$

2.2 Objective function

The purpose for establishing an optimization problem in this paper is to design the robot with a desired dexterous workspace. It is clear that the objective function is established as the sum of the square of the Cartesian position error between the desired workspace vertex and the tip of the end-effector, plus the square of the angular position error between the desired orientation and the end-effector orientation. Hence, the objective function can be described in (2). The vertex positions of the desired dexterous workspace are chosen as $(\bar{x}_{d_{1,1,k}}, \bar{z}_{d_{1,1,k}}) = (0.25m, 0.10m)$, $(\bar{x}_{d_{1,2,k}}, \bar{z}_{d_{1,2,k}}) = (0.65m, 0.10m)$, $(\bar{x}_{d_{2,1,k}}, \bar{z}_{d_{2,1,k}}) = (0.25m, 0.40m)$ and $(\bar{x}_{d_{2,2,k}}, \bar{z}_{d_{2,2,k}}) = (0.65m, 0.40m)$. The subscript k indicate three different orientations in each vertex position. They are selected as $\bar{\phi}_{d_{i,j,1}} = -\frac{\pi}{2} rad$, $\bar{\phi}_{d_{i,j,2}} = 0 rad$ and $\bar{\phi}_{d_{i,j,3}} = \frac{\pi}{2} rad\ \forall\ i, j = 1,2$.

$$J = \int_w \left(\bar{x}_{d_{i,j,k}} - \bar{x}_{i,j,k}\right)^2 dw + \int_w \left(\bar{z}_{d_{i,j,k}} - \bar{z}_{i,j,k}\right)^2 dw + \frac{18}{\pi} \int_w \left(\bar{\phi}_{d_{i,j,k}} - \bar{\phi}_{i,j,k}\right)^2 dw \tag{2}$$

In equation (2), the first two terms (Cartesian error) are equally weighted because they have the same units (meters). But, the last term a weight value of $\frac{18}{\pi}$ is selected by assuming that one degree is proporcional to one millimeter (0.001m). This weight value efficiently weights the last term with the other two in order to provide good solutions.

The direct kinematic (3)-(5) of the manipulator is used to define the Cartesian position of the end-effector in (2).

$$\bar{x}_{i,j,k} = l_1 \cos q_{1_{i,j,k}} - l_4 \cos\left(q_{2_{i,j,k}}\right) - l_5 \cos\left(q_{2_{i,j,k}} + q_{3_{i,j,k}}\right) \tag{3}$$

$$\bar{z}_{i,j,k} = l_1 \sin q_{1_{i,j,k}} - l_4 \sin\left(q_{2_{i,j,k}}\right) - l_5 \sin\left(q_{2_{i,j,k}} + q_{3_{i,j,k}}\right) \tag{4}$$

$$\bar{\phi}_{i,j,k} = q_{2_{i,j,k}} + q_{3_{i,j,k}} - \pi \tag{5}$$

2.3 Constraints

The parallel structure of the manipulator formed by the rectangle at links l_1 and l_2, presents mobility constraints between them. Those constraints must be considered in the manipulator design and they are presented when the links 1 and 2 collide each other. The mobility constraints are stated in (6)-(7), which involves a total of 24 inequality constraints.

$$g_{1-12}: Tol_{Max2} - q_{2_{i,j,k}} + q_{1_{i,j,k}} \leq 0 \quad (6)$$

$$g_{13-24}: q_{2_{i,j,k}} - q_{1_{i,j,k}} - \pi + Tol_{Max2} \leq 0 \quad (7)$$

In addition, other 39 constraints must be included to bound the design variable vector p, i.e. limits in the link angle and in the link lengths. In (8)-(13), those constraints are shown, where $Tol_{Max1} = \frac{\pi}{36} rad$ and $Tol_{Max2} = \frac{5\pi}{36} rad$ are the minimum security angle that should have the links to prevent the impact.

$$g_{25-36}: 0 \leq q_{1_{i,j,k}} \leq \pi - Tol_{Max2} \quad (8)$$

$$g_{37-48}: Tol_{Max2} \leq q_{2_{i,j,k}} \leq \frac{3}{2}\pi - Tol_{Max2} \quad (9)$$

$$g_{49-60}: -\pi + Tol_{Max1} \leq q_{3_{i,j,k}} \leq \pi - Tol_{Max1} \quad (10)$$

$$g_{61}: l_{1_{Min}} \leq l_1 \leq l_{1_{Max}} \quad (11)$$

$$g_{62}: l_{3_{Min}} \leq l_3 \leq l_{3_{Max}} \quad (12)$$

$$g_{63}: l_{4_{Min}} \leq l_4 \leq l_{4_{Max}} \quad (13)$$

2.4 Optimization problem statement

The optimization problem for the 3R manipulator consists on finding the optimal design parameter vector $p* \in R^{39}$ (link lengths and angular positions) such that its end-effector (point $(\bar{x},\bar{y})$ of the link 4) can reach a desired workspace with different orientations of the link 4 (dexterous workspace). Hence the optimization problem statement can be generally expressed as in (14)-(15), where $J \in R$ (2) is the objective function to be minimized and the equation $g \in R^{63}$ (15) is the inequality constraint vector (6)-(13) .

$$\underset{p}{Min}\ J \quad (14)$$

Subject to:

$$g(p) \leq 0 \in R^{63} \quad (15)$$

3 Optimization algorithms

The optimization problem (14)-(15) is solved by using eight variants of the differential evolution (DE) algorithm [34], [35] and by using the proposed exhaustive local exploitation mechanism in the eight variants of the differential evolution (DE) algorithm.

3.1 DE algorithm

The differential evolution (DE) algorithm consists of NP individuals $x_{i,G} = [x_{1,i,G}, x_{2,i,G}, ..., x_{j,i,G}, ..., x_{D,i,G},]^T$ $\forall\ i = 1,2,..,NP,\ G = 1,2,...,GenMax$ called target population. Each individual $x_{i,G}$ contains D design variables which are limited by their bounds $[x_j^{\min}, x_j^{\max}]$ $\forall\ j = 1,2,...,D$. The initial population $x_{j,i,G=0}$ $\forall\ j = 1,...,D,\ i = 1,..,NP$ is randomly selected considering their limits as follows: $x_{j,i,G=0} = x_j^{\min} + rand_j(0,1)(x_j^{\max} - x_j^{\min})$, where $rand_j$ is a uniformly distributed random number in the interval $[0,1]$.

3.1.1 Mutation and crossover

In each generation G, the mutation and crossover operator are used in the target population $x_{i,G}$ in order to generate other population called trial population with trial vectors $u_{i,G} = [u_{1,i,G}, u_{2,i,G}, ..., u_{j,i,G}, ..., u_{D,i,G},]^T$ $\forall\ i = 1,2,..,NP$ as their individuals. There are several variants of the DE algorithm [34] which allow the exploration and exploitation of the search space for the DE algorithm. The main differences among them are in the mutation and crossover operator. The use of the DE variants depends on the problem at hand as it is stated in [36] and in [22]. In this paper the mutation and crossover operator of the DE/rand/1/bin, DE/rand/1/exp, Best/1/bin, Best/1/exp, Current to rand 1, Current to best 1, Current to rand 1 bin, Rand 2 Dir are used. The mutation and crossover variants are summarized in Fig. 2.

The scale factor $F \in (0,1]$ and $K \in (0,1)$ in the mutation process, are used to control the influence of the selected individuals in order to generate the mutant vector. The indexes r_1, r_2 and r_3 are randomly chosen from the range $[1,NP]$ and the index *best* represent the individual with the best objective function.

The uniform crossover generates trial vectors $u_{i,G}$ from the mutant vector $v_{i,G}$ or the target vector $x_{i,G}$ depending on the crossover probability CR (higher values mean less influence of the target vector, hence higher influence of the mutant vector). The crossover stage is not required in current-to-rand/1, current-to-best/1 and rand/2/dir, as it is observed in Fig. 2.

3.1.2 Constrained selection mechanism

Given that traditional DE [34] does not handle constrained optimization problem, the technique proposed in [37] is used to provide an elitism constrained selection mechanism (ECSM) in the DE algorithm ([38],[39],[40]). The ECSM determines between the trial vector $u_{i,G}$ and the target one $x_{i,G}$, which of them pass to the next generation $x_{i,G+1}$ and it depends on their fitness. This decision is based on the following statements:

–Feasible solutions are preferred to any infeasible solution
–Between two feasible solutions, the one having better objective function value is preferred.
–Between two infeasible solutions, the one having smaller constraint violation is preferred.

Nomenclature	Variant
rand/1/bin	$u_j^i = \begin{cases} v_j^i = x_j^{r_3} + F(x_j^{r_1} - x_j^{r_2}) & \text{if } \mathrm{rand}_j(0,1) < CR \text{ or } j = j_{rand} \\ x_j^i & \text{otherwise} \end{cases}$
rand/1/exp	$u_j^i = \begin{cases} v_j^i = x_j^{r_3} + F(x_j^{r_1} - x_j^{r_2}) & \text{from } \mathrm{rand}_j(0,1) < CR \text{ or } j = j_{rand} \\ x_j^i & \text{otherwise} \end{cases}$
best/1/bin	$u_j^i = \begin{cases} v_j^i = x_j^{best} + F(x_j^{r_1} - x_j^{r_2}) & \text{if } \mathrm{rand}_j(0,1) < CR \text{ or } j = j_{rand} \\ x_j^i & \text{otherwise} \end{cases}$
best/1/exp	$u_j^i = \begin{cases} v_j^i = x_j^{best} + F(x_j^{r_1} - x_j^{r_2}) & \text{from } \mathrm{rand}_j(0,1) < CR \text{ or } j = j_{rand} \\ x_j^i & \text{otherwise} \end{cases}$
current-to-rand/1	$\mathbf{u}^i = v_j^i = \mathbf{x}^i + K(\mathbf{x}^{r_3} - \mathbf{x}^i) + F(\mathbf{x}^{r_1} - \mathbf{x}^{r_2})$
current-to-best/1	$\mathbf{u}^i = v_j^i = \mathbf{x}^i + K(\mathbf{x}^{best} - \mathbf{x}^i) + F(\mathbf{x}^{r_1} - \mathbf{x}^{r_2})$
current-to-rand/1/bin	$u_j^i = \begin{cases} v_j^i = x_j^i + K(x_j^{r_3} - x_j^i) + F(x_j^{r_1} - x_j^{r_2}) & \text{if } \mathrm{rand}_j(0,1) < CR \text{ or } j = j_{rand} \\ x_j^i & \text{otherwise} \end{cases}$
rand/2/dir	$\mathbf{u}^i = v_j^i = \mathbf{w}^1 + \frac{F}{2}(\mathbf{w}^1 - \mathbf{w}^2 + \mathbf{w}^3 - \mathbf{w}^4)$ where $f(\mathbf{w}^1) < f(\mathbf{w}^2)$ and $f(\mathbf{w}^3) < f(\mathbf{w}^4)$

Fig. 2: DE variants.

3.1.3 Exhaustive local exploitation mechanism with adaptive scale factor

The exhaustive local exploitation mechanism with adaptive scale factor (ELEMAEF) consist on deeply searching on a previous direction if it presents better fitness. This mechanism favors the search in the neighborhood of the trial vector promoting efficient individual (local) exploitation. The exploration capabilities of the DE variants are not substantially diminishing because the ELEMAEF only enters when the trial vector presents better fitness (considering the constrained selection operation) than the target vector. When the ELEMAEF is activated, a new individual $\hat{\mathbf{u}}$ is generated by using the same mutation and crossover operation (the same base vector $x_{j,r_1,G}$ and the same difference vector $(x_{j,r_2,G} - x_{j,r_3,G})$) considering the adaptive scale factor F_2 proposed in [28]. If the new individual $\hat{\mathbf{u}}$ is better than $\mathbf{u}_{i,G+1}$, the new individual passes to the next generation. This is done repeatedly until the maximum number of searching N_w is fulfilled or when the new individual $\hat{\mathbf{u}}$ is worse than previous one $\mathbf{u}_{i,G+1}$. It is important to note that the proposed DE algorithm with the exhaustive local exploitation mechanism requires an additional parameter F_2 in the mutation process. The pseudocode of the exhaustive exploitation mechanism is observed in Fig. 3. In the next paragraph, the parameter adaptation of F_2 is introduced.

Parameter adaptation of F_2: The mutation factor F_2 in the exhaustive local exploitation mechanism is self generated according to [28]. This factor uses a Cauchy distribution with a location parameter μ_{F_2}, specifying the location of the peak of the distribution and the scale parameter of 0.1 which specifies the half-width at half-maximum as it is observed in (16).

$$y = randc(\mu_{F_2}, 0.1) \tag{16}$$

$$F_2 = \begin{cases} 1, & if \;\; y \geq 1 \\ Recompute \;\; y, & if \;\; y \leq 0 \\ y, & else \end{cases} \tag{17}$$

The location parameter μ_{F_2} is initialized as 0.5 and then updated at the end of each generation according to (18), where the parameter S_{F_2} denote the set of all successful mutation factors in the ELEMAEF and $c = 0.1$ is chosen. The $mean_L(\cdot)$ is the Lehmer mean stated in (19).

$$\mu_{F_2} = (1-c)\cdot\mu_{F_2} + c\cdot mean_L(S_{F_2}) \tag{18}$$

$$mean_L(S_{F_2}) = \frac{\sum S_{F_2}^2}{\sum S_{F_2}} \tag{19}$$

The pseudocode of the DE algorithm with ELEMAEF is shown in Fig. 4.

4 Experiments and results

The proposed DE algorithm with ELEMAEF is programmed in Matlab Release 7.9 on a Windows platform. The experiments are performed on a PC with a 1.83 *GHz* Core 2 Duo with 2 *GB* of RAM. The population size *NP* consists on 36 individuals, the algorithm stops when the number of generations exceed $G_{Max} = 6500$ generations or when the mean of the objective function of the individuals in the current generation is smaller than $1e-4$. In order to analyze the results, the proposed DE algorithm with ELEMAEF is compared with the DE algorithm without ELEMAEF, by using constant scaling factor F and constant crossover CR.

Ten independent runs are carried out for each DE variants and for each of the following scale factor $F = 0.3, 0.6, 0.9$ and crossover parameters $CR = 0.3, 0.6, 0.9$. When the DE algorithm with ELEMAEF is used, the maximum number of local search $N_w = 10$ is chosen for each selected scale factor F and crossover parameter CR. Therefore, different "*cases*" are shown below in Table 1 and in Table 2 for each DE variants. Each *case* comprising ten independent runs.

In Table 1 and Table 2 the experimental results with different DE variants with and without ELEMAEF are shown, respectively. The meaning of the abbreviations in Tables is explained as follows: J_{mean} is the mean of the objective function values for the individuals in the last generation considering the ten runs. $\sigma(J)_{mean}$ is mean from the ten runs, of the standard deviation of the objective function values for the last generation. "Time" is the mean of the convergence time from the ten runs. $MaxG_{mean}$ is the mean of the maximum number of generation for the ten runs. $\#FunJ_{mean}$ is the mean of the times that the objective function is evaluated for the ten runs. $\#ImpJ_{mean}$ is the mean of the sum of the times that the ELEMAEF improves the individual in each generation for the ten runs. $J^*\%$ is the percentage from the ten runs, which the optimum objective function value is reached in the population. It is considered that the optimum solution is reached when the objective function is less than $1e-4$ ($J^* < 1e-4$).

DE variants without ELEMAEF

Based on the column $\sigma(J)_{mean}$ and the column J_{mean} in Table 1, it is observed that the low standard deviation

For $w = 1$ **to** N_w **Do**
  For $j = 1$ **to** D **Do**
    If $u_{j,i,G+1} == x_{j,i,G}$ **Then**
      $\hat{u}_j = x_{j,i,G+1}$
    Else
      $F_2 = randc_i(\mu_{F_2}, 0.1)$
      $\hat{u}_j = x_{j,r_1,G} + F_2(x_{j,r_2,G} - x_{j,r_3,G})$
    End If
  End For
  If $\hat{\mathbf{u}}$ is better than $\mathbf{u}_{i,G+1}$ (Based on CSM) **Then**
    $\mathbf{u}_{i,G+1} = \hat{\mathbf{u}}$
    $F_2 \rightarrow S_{F_2}$
  Else
    $\mathbf{u}_{i,G+1} = \mathbf{u}_{i,G+1}$
    Break
  End If
End For

Fig. 3: Exhaustive exploitation mechanism added to the DE algorithm.

BEGIN
  $G = 0$; $\mu_{F_2} = 0.5$
  Create a random population $\mathbf{x}_{i,G}$ $\forall i = 1, \ldots, NP$
  Evaluate $J(\mathbf{x}_{i,G})$, $g(\mathbf{x}_{i,G})$, $\forall i = 1, \ldots, NP$
  Do
    $S_{F_2} = \emptyset$
    For $i = 1$ to NP **Do**
    Select randomly $\{r_1 \neq r_2 \neq r_3\} \in \mathbf{x}_G$.
    j_{rand} =randint$(1, D)$
      For $j = 1$ to D **Do**
        Mutation and crossover
      End For
      Evaluate $J(\mathbf{u}_{i,G+1})$, $g(\mathbf{u}_{i,G+1})$
      If $\mathbf{u}_{i,G+1}$ is better than $\mathbf{x}_{i,G}$ (Based on CSM) **Then**
        Exhaustive local exploitation mechanism
        $\mathbf{x}_{i,G+1} = \mathbf{u}_{i,G+1}$
      Else
        $\mathbf{x}_{i,G+1} = x_{i,G}$
      End
        $\mu_{F_2} = (1-c) \cdot \mu_{F_2} + c \cdot mean_L(S_{F_2})$
        $G = G + 1$
  While $(G \leq G_{Max})$
END

Fig. 4: Pseudocode of the DE algorithm with the exhaustive local exploitation mechanism.

($\sigma(J)_{mean} \leq 0$) indicates that most of the runs for each case converge to a solution (local solution whether J_{mean} is large and $\sigma(J)_{mean} \leq 0$ or global solution whether J_{mean} is small and $\sigma(J)_{mean} \leq 0$). The DE/Current to Rand/1, DE/Current to Best/1, DE/Current to Rand/1/Bin and DE/Rand/2/Dir converge in general to a local solution far away from the optimal one (see column J_{mean} and $\sigma(J)_{mean} \leq 0$). High standard deviation indicating that the individuals in the population are spread out over a large range of values. Then, DE/Rand/1/Exp and DE/Best/1/Exp diverge in most of the cases (see column $\sigma(J)_{mean} > 0$). In this particular design problem, the use of exponential crossover implies the large exploration of the search space, such that DE/Rand/1/Exp and DE/Best/1/Exp diverge in general.

The boldface rows in Table 1 indicate the best DE variants without ELEMAEF which solves the particular design problem. Those are DE/Best/1/Bin, DE/Rand/1/Bin and DE/Best/1/Exp because they find the optimum objective function at 44%, 22% and 11% of the cases (see column $J^*\%$), respectively. This indicates that the use of binomial crossover with the use of random or best individual as the base vector, promotes the search of the optimum solution. As it is previously commented, the use of exponential crossover implies the large exploration of the search space but only for the case $8D$ when $CR = 0.9$ and $F = 0.6$, the DE/Best/1/Exp can find the optimum solution at 90% of the runs.

Table 1: Experimental results for the DE variants without ELEMAEF.

Algorithm	*Case*	*CR*	*F*	J_{mean}	$\sigma(J)_{mean}$	*Time*	$MaxG_{mean}$	$\#FunJ_{mean}$	$J^{*}\%$
Rand 1 Bin	1A	0.3	0.3	0.361753	$8.8282e^{-10}$	266.21	6500	234000	0%
Rand 1 Bin	2A	0.3	0.6	0.027353	0.900604	261.60	6500	234000	0%
Rand 1 Bin	3A	0.3	0.9	5.007168	147.8961	261.04	6500	234000	0%
Rand 1 Bin	4A	0.6	0.3	1.052967	$3.2852e^{-11}$	261.40	6500	234000	0%
Rand 1 Bin	**5A**	**0.6**	**0.6**	**0.002540**	$\mathbf{5.8938e^{-5}}$	**212.46**	**5794.5**	**208602**	**80%**
Rand 1 Bin	6A	0.6	0.9	17.30592	251.8817	261.87	6500	234000	0%
Rand 1 Bin	7A	0.9	0.3	21.85020	$1.8188e^{-5}$	262.06	6500	234000	0%
Rand 1 Bin	**8A**	**0.9**	**0.6**	**0.109544**	$\mathbf{5.6835e^{-4}}$	**162.09**	**4719**	**169884**	**50%**
Rand 1 Bin	9A	0.9	0.9	0.159810	0.062602	262.62	6500	234000	0%
Rand 1 Exp	1B	0.3	0.3	472.0137	283.3315	186.91	6500	234034	0%
Rand 1 Exp	2B	0.3	0.6	538.1129	303.1237	186.95	6500	234034	0%
Rand 1 Exp	3B	0.3	0.9	541.7425	267.4548	187.01	6500	234035	0%
Rand 1 Exp	4B	0.6	0.3	438.5806	294.4790	188.21	6500	234035	0%
Rand 1 Exp	5B	0.6	0.6	527.3437	275.8388	188.28	6500	234036	0%
Rand 1 Exp	6B	0.6	0.9	569.5916	268.5908	188.25	6500	234035	0%
Rand 1 Exp	7B	0.9	0.3	54.90616	0	196.60	6500	234017	0%
Rand 1 Exp	8B	0.9	0.6	0.700875	0.137024	196.07	6500	234035	0%
Rand 1 Exp	9B	0.9	0.9	544.2769	221.4091	196.28	6500	234036	0%
Best 1 Bin	1C	0.3	0.3	689.6785	0	157.96	6500	62222	0%
Best 1 Bin	**2C**	**0.3**	**0.6**	**15.65933**	**6.895569**	**152.59**	**6042**	**217530**	**20%**
Best 1 Bin	3C	0.3	0.9	9.096234	7.431126	164.17	6500	234036	0%
Best 1 Bin	4C	0.6	0.3	2563.211	0	155.92	6500	23386	0%
Best 1 Bin	**5C**	**0.6**	**0.6**	**1.111960**	**0.000127**	**117.95**	**4678**	**168440**	**50%**
Best 1 Bin	**6C**	**0.6**	**0.9**	**6.671277**	**0.158473**	**158.03**	**6284**	**226274**	**30%**
Best 1 Bin	7C	0.9	0.3	6595.621	0	154.20	6500	3211	0%
Best 1 Bin	8C	0.9	0.6	929.9061	0	163.17	6500	234036	0%
Best 1 Bin	9C	0.9	0.9	53.78760	0.001746	156.89	6270	225759	10%
Best 1 Exp	1D	0.3	0.3	627.9030	456.1873	188.01	6500	234012	0%
Best 1 Exp	2D	0.3	0.6	535.1099	318.0737	188.06	6500	234010	0%
Best 1 Exp	3D	0.3	0.9	546.6142	275.7620	188.06	6500	234011	0%
Best 1 Exp	4D	0.6	0.3	575.4174	386.8449	189.37	6500	234025	0%
Best 1 Exp	5D	0.6	0.6	451.7013	331.9883	189.36	6500	234019	0%
Best 1 Exp	6D	0.6	0.9	563.4715	304.5132	189.34	6500	234021	0%
Best 1 Exp	7D	0.9	0.3	987.5407	0	191.57	6500	64915	0%
Best 1 Exp	**8D**	**0.9**	**0.6**	**2.474110**	**0.004745**	**107.96**	**3548**	**127768**	**90%**
Best 1 Exp	9D	0.9	0.9	6.560564	0.157588	197.53	6500	234025	0%
Current to Rand 1	1E	0.3	0.3	6686.528	0	137.39	6500	84511	0%
Current to Rand 1	2E	0.3	0.6	3414.828	0	142.92	6500	234036	0%
Current to Rand 1	3E	0.3	0.9	1288.256	0.000285	142.65	6500	234036	0%
Current to Rand 1	4E	0.6	0.3	7916.050	13.40574	135.70	6500	42207	0%
Current to Rand 1	5E	0.6	0.6	2944.626	0	142.90	6500	234036	0%
Current to Rand 1	6E	0.6	0.9	1298.420	0.005519	142.60	6500	234036	0%
Current to Rand 1	7E	0.9	0.3	7355.543	0	137.20	6500	82625	0%
Current to Rand 1	8E	0.9	0.6	3093.975	8.023190	142.87	6500	234036	0%
Current to Rand 1	9E	0.9	0.9	1124.290	0	142.67	6500	234036	0%
Current to best 1	1F	0.3	0.3	6594.853	0	129.94	6500	108142	0%
Current to best 1	2F	0.3	0.6	3764.453	0	134.97	6500	234036	0%
Current to best 1	3F	0.3	0.9	2573.786	0	134.70	6500	234036	0%
Current to best 1	4F	0.6	0.3	6701.564	0	127.97	6500	49401	0%
Current to best 1	5F	0.6	0.6	3644.155	40.42009	135.23	6500	234036	0%
Current to best 1	6F	0.6	0.9	3359.008	0.000003	134.86	6500	234036	0%
Current to best 1	7F	0.9	0.3	6670.760	0	128.69	6500	71767	0%
Current to best 1	8F	0.9	0.6	3065.115	0	135.21	6500	234036	0%
Current to best 1	9F	0.9	0.9	3191.107	0.001585	134.91	6500	234036	0%
Current to rand 1 Bin	1G	0.3	0.3	1242.558	11.53473	163.10	6500	97103	0%
Current to rand 1 Bin	2G	0.3	0.6	0.154404	0.079587	167.97	6500	234036	0%
Current to rand 1 Bin	3G	0.3	0.9	467.4536	97.92991	168.02	6500	234036	0%
Current to rand 1 Bin	4G	0.6	0.3	3166.224	0	164.09	6500	51426	0%
Current to rand 1 Bin	5G	0.6	0.6	0.157648	0.000035	170.71	6500	234036	0%
Current to rand 1 Bin	6G	0.6	0.9	1203.751	177.3152	170.97	6500	234036	0%
Current to rand 1 Bin	7G	0.9	0.3	6110.778	0	167.23	6500	72893	0%
Current to rand 1 Bin	8G	0.9	0.6	20.48629	0.014726	173.40	6500	234036	0%
Current to rand 1 Bin	9G	0.9	0.9	0.627232	0.019767	173.17	6500	234036	0%
Rand 2 Dir	1H	0.3	0.3	6060.665	0	137.87	6500	183369	0%
Rand 2 Dir	2H	0.3	0.6	3987.782	0	139.84	6500	234036	0%
Rand 2 Dir	3H	0.3	0.9	2086.551	0	139.51	6500	234035	0%
Rand 2 Dir	4H	0.6	0.3	6718.683	0	139.86	6500	232804	0%
Rand 2 Dir	5H	0.6	0.6	4508.276	0	139.88	6500	234036	0%
Rand 2 Dir	6H	0.6	0.9	2201.797	0	139.70	6500	234036	0%
Rand 2 Dir	7H	0.9	0.3	7158.456	0	138.57	6500	195107	0%
Rand 2 Dir	8H	0.9	0.6	3920.266	0	139.98	6500	234036	0%
Rand 2 Dir	9H	0.9	0.9	2100.025	0	139.73	6500	234036	0%

DE algorithm with ELEMAEF

Based on the column $\sigma(J)_{mean}$ and the column J_{mean} in Table 2, it is observed that the worst DE variants are: DE/Current to Rand/1, DE/Current to Best/1 and DE/Rand/2/Dir because they converge to a local solution at 44%, 55% and 100% of the cases, respectively and the rest of the cases diverge (see column $\sigma(J)_{mean} > 0$).

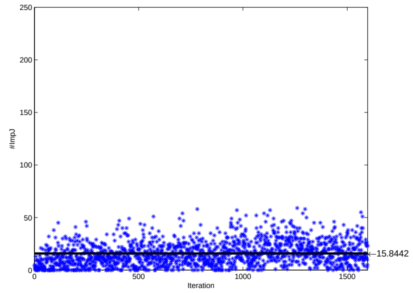
In Table 2 the best DE variants are marked in boldface. Those are: DE/Rand/1/Bin, DE/Best/1/Bin, DE/Current to Rand/1/Bin, DE/Rand/1/Exp, DE/Best/1/Exp. They find the optimum solution at 88%, 77%, 33%, 22%, 22% of the cases, respectively. In addition, in column $\#ImpJ_{mean}$, it is observed that larger improvement of the individual by using the ELEMAEF, results the convergence to local solutions far away from the optimum one.

The number of times that the individuals are improved by the ELEMAEF for one run with different DE variants is shown in Fig. 5. The results in Fig. 5 find the global optimum in the last generation. The horizontal line in the figures represent the mean of the times that the ELEMAEF improves the individual in the generation. It is observed in the first generations, the algorithm explores the solution space such that the ELEMAEF generates few improved solutions. Then, after the first generations, the ELEMAEF exhaustively search in the vicinity of the individuals with better fitness such that more improved individuals are found. It is important to note, that the ELEMAEF and the selection of the parameter adaptation of F_2 do not guide the exhaustive search towards local regions.

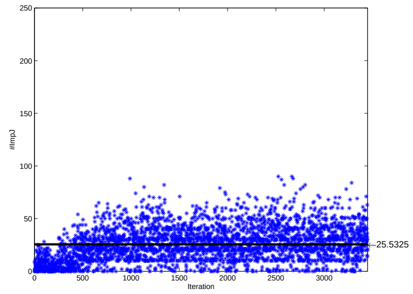
Comparative results

In Table 3, comparative results between the DE variants with the use of ELEMAEF and without it are shown. The results are marked in boldface to indicate the **DE variant with the use of ELEMAEF** and in italic to indicate *DE variant without the use of ELEMAEF*. The column "% Optimum solution" indicates the percentage of the cases for each DE variant where the algorithm converge to the optimum solution. It is observed that the use of ELEMAEF increases the times that the DE variant converge to the optimum solution. This fact indicates that the ELEMAEF promotes the exploitation of the individuals in the population such that better individuals are found. On the other hand, the column "Best C. Time" represents the DE variant that provide less convergence time. It is clear that the incorporation of the ELEMAEF in the DE variants requires more computing time due to more objective function evaluation is required.

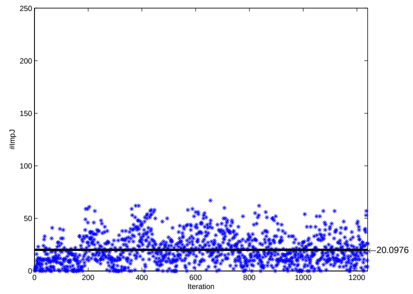
It is important to point out that the use of ELEMAEF occasionally improves the individual in each generation (see $\#ImpJ_{mean}$ in Table 2), but this is sufficient to improve the DE variant behavior with respect to the DE variant without ELEMAEF. This indicates that a trade off



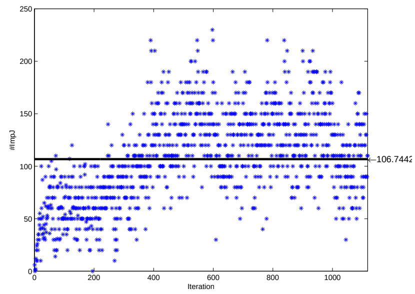
a) CR=0.6, F=0.6 Rand 1 Bin.



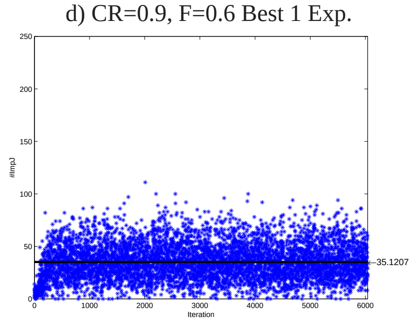
b) CR=0.9, F=0.3 Rand 1 Exp.



c) CR=0.3, F=0.9 Best 1 Bin.



d) CR=0.9, F=0.6 Best 1 Exp.



e) CR=0.9, F=0.6 Current to rand 1 Bin.

Fig. 5: Number of times that the individuals are improved by the Exhaustive Local Mechanism.

Table 2: Experimental results for the DE algorithm with ELEMAEF.

Algorithm	Case	CR	F	J_{mean}	$\sigma(J)_{mean}$	Time	$MaxG_{mean}$	$\#FunJ_{mean}$	$J^{*}\%$	$\#ImpJ_{mean}$
Rand 1 Bin	**1A**	**0.3**	**0.3**	**0.071963**	**0.000080**	**161.87**	**4532**	**297048**	**60%**	**16.13**
Rand 1 Bin	**2A**	**0.3**	**0.6**	**0.012789**	**0.000049**	**137.01**	**4047**	**234573**	**70%**	**15.14**
Rand 1 Bin	**3A**	**0.3**	**0.9**	**7.674573**	**1.876054**	**191.05**	**6500**	**253746**	**10%**	**2.55**
Rand 1 Bin	**4A**	**0.6**	**0.3**	**0.033166**	**0.000149**	**243.76**	**6143**	**388596**	**40%**	**13.41**
Rand 1 Bin	**5A**	**0.6**	**0.6**	**0.021057**	**0.000047**	**75.79**	**2030**	**116272**	**90%**	**15.40**
Rand 1 Bin	**6A**	**0.6**	**0.9**	**0.086668**	**0.003166**	**153.12**	**4718**	**212453**	**70%**	**8.13**
Rand 1 Bin	7A	0.9	0.3	51.95243	0.006939	308.83	6500	449920	0%	19.13
Rand 1 Bin	**8A**	**0.9**	**0.6**	**0.471681**	**0.000510**	**230.48**	**4809**	**337603**	**50%**	**25.78**
Rand 1 Bin	**9A**	**0.9**	**0.9**	**0.006594**	**0.000026**	**165.76**	**4174**	**230469**	**60%**	**16.90**
Rand 1 Exp	1B	0.3	0.3	538.7221	292.1712	215.85	6500	250933	0%	1.29
Rand 1 Exp	2B	0.3	0.6	549.9120	282.5619	215.63	6500	249592	0%	1.15
Rand 1 Exp	3B	0.3	0.9	561.9273	259.6416	215.74	6500	249834	0%	1.28
Rand 1 Exp	4B	0.6	0.3	535.3635	343.9401	216.79	6500	249037	0%	1.18
Rand 1 Exp	5B	0.6	0.6	544.3737	287.7287	216.85	6500	247690	0%	1.08
Rand 1 Exp	6B	0.6	0.9	590.3187	255.9502	216.82	6500	247460	0%	1.15
Rand 1 Exp	**7B**	**0.9**	**0.3**	**0.018500**	**0.000034**	**188.79**	**4651**	**343525**	**70%**	**26.03**
Rand 1 Exp	**8B**	**0.9**	**0.6**	**0.015631**	**0.000023**	**213.38**	**5758**	**304331**	**50%**	**10.92**
Rand 1 Exp	9B	0.9	0.9	242.0042	87.92096	224.36	6500	247735	0%	1.45
Best 1 Bin	**1C**	**0.3**	**0.3**	**42.94217**	**0.009001**	**253.15**	**6085**	**569326**	**20%**	**41.79**
Best 1 Bin	**2C**	**0.3**	**0.6**	**0.132729**	**0.203463**	**209.66**	**4999**	**476598**	**30%**	**43.16**
Best 1 Bin	**3C**	**0.3**	**0.9**	**0.622174**	**1.413398**	**179.06**	**5438**	**307992**	**50%**	**18.21**
Best 1 Bin	4C	0.6	0.3	107.5080	0.013497	409.51	6500	816277	0%	71.20
Best 1 Bin	**5C**	**0.6**	**0.6**	**4.175193**	**0.001241**	**333.49**	**6204**	**642686**	**10%**	**42.45**
Best 1 Bin	**6C**	**0.6**	**0.9**	**0.018921**	**0.097005**	**178.08**	**4388**	**302293**	**40%**	**19**
Best 1 Bin	7C	0.9	0.3	1538.257	0.387008	730.11	6500	1230698	0%	140.66
Best 1 Bin	**8C**	**0.9**	**0.6**	**84.81393**	**0.003082**	**407.47**	**6435**	**655529**	**10%**	**39.66**
Best 1 Bin	**9C**	**0.9**	**0.9**	**75.57373**	**0.000008**	**467.14**	**6434**	**777978**	**10%**	**72.60**
Best 1 Exp	1D	0.3	0.3	744.2973	1184.047	219.45	6500	234202	0%	15.43
Best 1 Exp	2D	0.3	0.6	587.5616	348.9203	218.44	6500	234537	0%	12.65
Best 1 Exp	3D	0.3	0.9	583.1925	285.3257	218.00	6500	235038	0%	11.06
Best 1 Exp	4D	0.6	0.3	547.9760	584.6983	221.10	6500	234236	0%	15.72
Best 1 Exp	5D	0.6	0.6	549.1788	368.3518	219.27	6500	234587	0%	10.65
Best 1 Exp	6D	0.6	0.9	635.6344	302.6703	218.61	6500	235061	0%	8.74
Best 1 Exp	7D	0.9	0.3	865.6287	0.000000	556.31	6500	73758	0%	318.07
Best 1 Exp	**8D**	**0.9**	**0.6**	**0.002355**	**0.000049**	**142.78**	**2976**	**107584**	**90%**	**94.73**
Best 1 Exp	**9D**	**0.9**	**0.9**	**1.774057**	**0.284646**	**232.16**	**6475**	**233948**	**10%**	**16.75**
Current to Rand 1	1E	0.3	0.3	6613.680	2018.004	504.97	6500	808565	0%	78.45
Current to Rand 1	2E	0.3	0.6	4719.736	0	332.63	6500	524796	0%	30.90
Current to Rand 1	3E	0.3	0.9	2648.037	0	301.53	6500	479674	0%	29.52
Current to Rand 1	4E	0.6	0.3	5907.370	1781.042	504.61	6500	808436	0%	78.34
Current to Rand 1	5E	0.6	0.6	4336.297	3.845465	327.83	6500	516579	0%	30.02
Current to Rand 1	6E	0.6	0.9	2842.934	0	301.59	6500	479803	0%	29
Current to Rand 1	7E	0.9	0.3	7404.721	998.5876	512.49	6500	822221	0%	80.31
Current to Rand 1	8E	0.9	0.6	5123.504	6.872291	328.42	6500	517954	0%	30.16
Current to Rand 1	9E	0.9	0.9	2666.130	0	301.58	6500	480051	0%	29.54
Current to best 1	1F	0.3	0.3	9518.423	4079.354	1261.14	6500	57945	0%	316.64
Current to best 1	2F	0.3	0.6	5315.249	0	1007.94	6500	1043266	0%	224.35
Current to best 1	3F	0.3	0.9	4772.813	0	278.74	6500	436361	0%	20.89
Current to best 1	4F	0.6	0.3	9741.819	2552.972	1286.55	6500	53482	0%	324.01
Current to best 1	5F	0.6	0.6	5787.309	3.508465	1011.69	6500	1051785	0%	225.39
Current to best 1	6F	0.6	0.9	4481.553	0	278.36	6500	435845	0%	20.86
Current to best 1	7F	0.9	0.3	9941.413	2505.349	1295.63	6500	62168	0%	325.63
Current to best 1	8F	0.9	0.6	4851.411	0	1015.62	6500	1043572	0%	226.08
Current to best 1	9F	0.9	0.9	4965.898	0	278.51	6500	436788	0%	20.96
Current to rand 1 Bin	1G	0.3	0.3	2137.612	7321.067	473.64	6500	572493	0%	196.04
Current to rand 1 Bin	**2G**	**0.3**	**0.6**	**0.792429**	**0.222792**	**186.74**	**6254**	**246512**	**10%**	**3.13**
Current to rand 1 Bin	3G	0.3	0.9	556.7300	161.8644	189.46	6500	236271	0%	0.11
Current to rand 1 Bin	4G	0.6	0.3	2755.634	6087.996	805.56	6500	737530	0%	245.59
Current to rand 1 Bin	**5G**	**0.6**	**0.6**	**0.069026**	**0.000066**	**163.34**	**4090**	**251626**	**60%**	**14.77**
Current to rand 1 Bin	6G	0.6	0.9	1101.626	164.5283	192.38	6500	235359	0%	0.03
Current to rand 1 Bin	7G	0.9	0.3	5435.739	3452.629	1231.17	6500	1067167	0%	291.07
Current to rand 1 Bin	**8G**	**0.9**	**0.6**	**4.293221**	**0.001469**	**360.52**	**6454**	**520077**	**10%**	**34.11**
Current to rand 1 Bin	**9G**	**0.9**	**0.9**	**0.029537**	**0.000654**	**193.79**	**6377**	**236592**	**50%**	**.037**
Rand 2 Dir	1H	0.3	0.3	12243.65	0	1402.93	6500	196155	0%	338.31
Rand 2 Dir	2H	0.3	0.6	5569.629	0	1071.48	6500	234550	0%	248.05
Rand 2 Dir	3H	0.3	0.9	3138.329	0.000009	653.92	6500	234699	0%	134.86
Rand 2 Dir	4H	0.6	0.3	12132.54	0	1357.02	6500	216758	0%	325.81
Rand 2 Dir	5H	0.6	0.6	6260.418	0	1073.60	6500	234488	0%	248.69
Rand 2 Dir	6H	0.6	0.9	3300.208	0.000001	657.54	6500	234734	0%	135.83
Rand 2 Dir	7H	0.9	0.3	10512.23	0	1370.32	6500	210425	0%	329.57
Rand 2 Dir	8H	0.9	0.6	5508.342	0	1079.23	6500	234523	0%	249.57
Rand 2 Dir	9H	0.9	0.9	2718.728	0.000194	657.87	6500	234691	0%	135.56

Table 3: Comparative results between the DE algorithm with ELEMAEF and DE algorithm without it.

DE variant	% Optimum solution **DE with ELEMAEF** / *DE without ELEMAEF*	Best C. Time
Rand 1 Bin	**88%** / *22%*	**DE with ELEMAEF**
Rand 1 Exp	**22%** / *0%*	*DE without ELEMAEF*
Best 1 Bin	**77%** / *44%*	*DE without ELEMAEF*
Best 1 Exp	**22%** / *11%*	*DE without ELEMAEF*
Current to Rand 1	**0%** / *0%*	*DE without ELEMAEF*
Current to best 1	**0%** / *0%*	*DE without ELEMAEF*
Current to rand 1 Bin	**33%** / *0%*	*DE without ELEMAEF*
Rand 2 Dir	**0%** / *0%*	*DE without ELEMAEF*

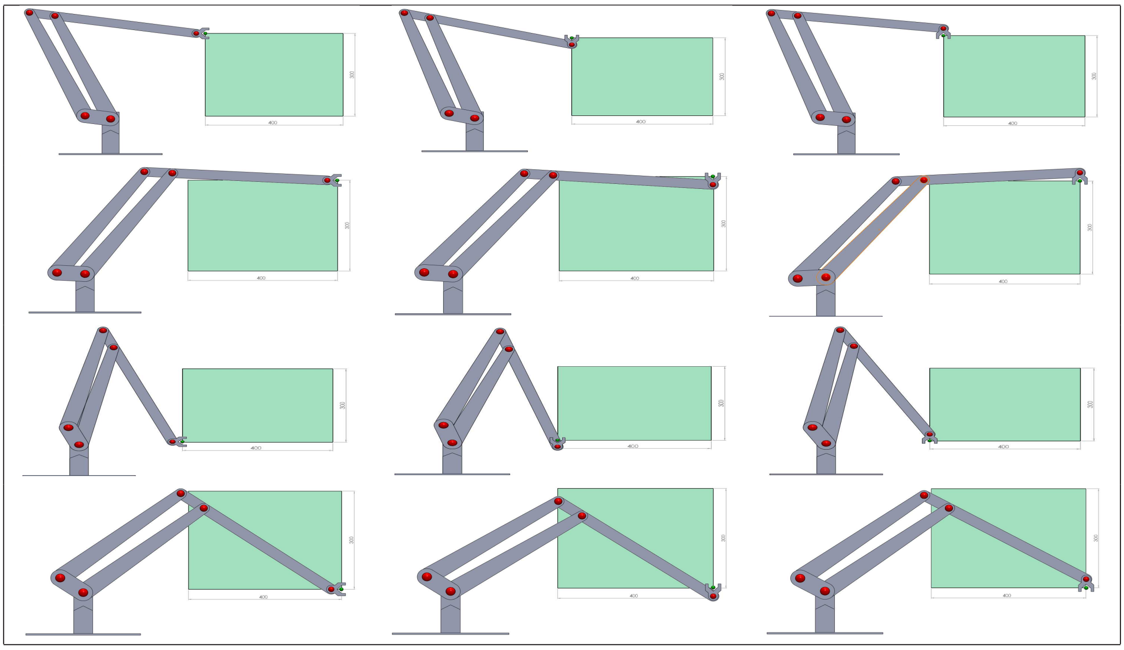


Fig. 6: Optimum design of the 3R manipulator with dexterous workspace.

between the exploration and exploitation mechanism in the DE algorithm is an outstanding characteristic to be considered in order to find the optimum solution in this particular design problem.

Design results

The optimal link lengths are provided by using the DE/Rand/1/Bin with ELEMAEF, those are: $l_1^{*ELEMAEF} = 0.407$ $l_3^{*ELEMAEF} = 0.415$ $l_4^{*ELEMAEF} = 0.0264$. The optimal design meets the objective function and constraints, such that its end-effector can reach the defined workspace with different orientation in the interval $[-\frac{pi}{2}, \frac{pi}{2}]$. Hence, the manipulator designed by this approach presents a defined dexterous workspace. In Fig. 6 the optimum design $p^{*ELEMAEF}$ of the 3R manipulator with a parallelogram mechanism is displayed. The optimum design guarantees the end-effector can reach the vertices of the workspace with at least three different orientations. Hence, the optimum design fulfills the objective design, i.e., it has a desired dexterous workspace.

5 Conclusion

In this paper an optimization problem to optimal design the link length of a 3R manipulator with a parallelogram five-bar mechanism is stated. The design involves that the manipulator must fulfil a defined dexterous workspace. An exhaustively local mechanism is incorporated in the differential evolution algorithm to solve the problem. The final optimal design results in a 3R manipulator which its end-effector can be moved in the desired workspace with different orientations, fulfilling the design objective and constraints.

According to the empirical results, the ELEMAEF can search into the design space such that better solutions can be found. The DE with ELEMAEF improves the convergence to the optimal solution compared with the DE algorithm without ELEMAEF. The DE with ELEMAEF does not require an additional parameter to be tuned. The main drawback of the algorithm is that it requires more functions to be evaluated, such that it requires more computational time.

The strategy of applying a local exploitation mechanism is a very important factor to be included in the metaheuristic algorithm in order to promote the exhaustively search of the optimal solution. Nevertheless, the way that the local exploitation mechanism is working must be carefully analyzed due to the problem of the premature convergence to a local solution.

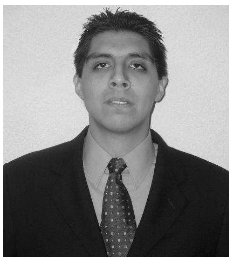
Acknowledgment

This work was supported in part by Consejo Nacional de Ciencia y Tecnología (CONACyT) under Grant 182298 and in part by the Operation and Development Committee of Academic Activities and the Secretariat of Research and Posgrade of the Instituto Politécnico Nacional under Grant 20131053 and 20131350.

References

[1] A. G. Erdman, G. N. Sandor, S. Kota, Mechanism Design: Analysis And Synthesis, Prentice Hall, (2001).
[2] C. M. Gosselin, J. Angeles, Singularity Analysis Of Closed Loop Kinematic Chains, IEEE Transactions On Robotics And Automation, **6**, 281-290 (1990).
[3] K. M. Lee, D. K. Shah, Part 1, Kinematic Analysis Of A 3-Dof In-Parallel Actuated Manipulator. IEEE Journal Of Robotics And Automation, **4**, 354-360 (1988).
[4] Robert L. Norton, Design of Machinery: An Introduction to the Synthesis and Analysis of Mechanisms and Machines, McGraw-Hill, (2003).
[5] J. E. Shigley, J. J. Uicker, Theory Of Machines And Mechanisms, Mc. Graw Hill, (1995).
[6] J. P. Merlet, Parallel Robots, Springer, (2006).
[7] C. M. Gosselin, Determination Of The Workspace Of 6-Dof Parallel Manipulators, Asme Journal Of Mechanical Design, **112**, 331-336 (1990).
[8] A. Gallant, R. Boudreau, M. Gallant, Geometric Determination Of The Dexterous Workspace Of N-Rrrr And N-Rrpr Manipulators, Mechanism And Machine Theory, **51**, 159-171 (2012).
[9] R. E. Stamper, L. W. Tsai, G. C. Walsh, Optimization Of A Three Dof Translational Platform For Well-Conditioned Workspace, IEEE International Conference On Robotics & Automation, **4**, 3250-3255 (1997).
[10] Z. Wang, Z. Wang, W. Liu, Y. Lei, A Study On Workspace, Boundary Workspace Analysis And Workpiece Positioning For Parallel Machine Tools. Mechanism Machine Theory, **36**, 605-622 (2001).
[11] E. J. Haug, C. M. Luh, F. A. Adkins, J. Y. Wang, Numerical Algorithms For Mapping Boundaries Of Manipulator Workspaces. Asme Journal Of Mechanical Design, **118**, 228-234 (1996).
[12] T. Lung-Wen, Robot analysis: The mechanics of serial and parallel manipulators, John Wiley & Sons, (1999).
[13] M. W. Spong, M. Vidyasagar, Robot Dynamics and Control, John Wiley & Sons, (2004).
[14] A. Kumar, K. J. Waldron, The workspaces of a mechanical manipulator, Journal of Mechanical Design, **103**, 665-672 (1981).
[15] Miguel G. Villarreal-Cervantes, Carlos A. Cruz-Villar, Jaime Alvarez-Gallegos, Edgar A. Portilla-Flores, Robust Structure-Control Design Approach for Mechatronic Systems. IEEE/ASME Transactions on Mechatronics, **18**, 1592-1601 (2013).
[16] S. B. Matekar, G. R. Gogate, Optimum Synthesis Of Path Generating Four-Bar Mechanisms Using Differential Evolution And A Modified Error Function, Mechanism And Machine Theory, **52**, 158-179 (2012).
[17] R. R. Bulatovic, S. R. Dordevic, Control Of The Optimum Synthesis Process Of A Four-Bar Linkage Whose Point On The Working Member Generates The Given Path, Applied Mathematics And Computation, **23**, 9765-9778 (2011).
[18] E. A. Portilla, et. al., Integration Of Structure And Control Using An Evolutionary Approach: An Application To The Optimal Concurrent Design Of A Cvt, International Journal For Numerical Methods In Engineering, **71**, 883-890 (2007).
[19] A. Ortiz, J.A. Cabrera, F. Nadal, A. Bonilla, Dimensional synthesis of mechanisms using Differential Evolution with auto-adaptive control parameters, Mechanism and Machine Theory, **64**, 210-229 (2013).
[20] Prasad Vilas Chanekar, Michael Angelo Amith Fenelon, Ashitava Ghosal, Synthesis of Adjustable Spherical Four-Link Mechanisms for Approximate Multi-Path Generation, Mechanism and Machine Theory, **70**, 538-552 (2013).
[21] Singiresu S. Rao, Engineering Optimization: Theory and Practice, John Wiley & Sons, (2009).
[22] M. G. Villarreal, C. A. Cruz, J. Alvarez, E. A. Portilla, Kinematic Dexterity Maximization Of An Omnidirectional Wheeled Mobile Robot: A Comparison Of Metaheuristic And Sqp Algorithms, International Journal Of Advanced Robotic Systems, **9**, 1-12 (2012).
[23] Paulo Roberto Bergamaschi, Sezimária de Fátima Pereira Saramago, Leandro dos Santos Coelho, Comparative Study of SQP and Metaheuristics for Robotic Manipulator Design, Applied Numerical Mathematics, **58**, 1396-1412 (2008).
[24] R. Storn, K. Price, A Simple And Efficient Heuristic For Global Optimization Over Continuous Spaces, Journal Of Global Optimization, **11**, 341-359 (1997).
[25] Roger Gämperle , Sibylle D. Müller , Petros Koumoutsakos, A parameter study for differential evolution, Int. Conf. on Advances in Intelligent Systems, Fuzzy Systems, Evolutionary Computation, 293-298 (2002).
[26] Karin Zielinski, Petra Weitkemper, Rainer Laur, Karl-Dirk Kammeyer, Parameter Study for Differential Evolution Using a Power Allocation Problem Including Interference Cancellation, IEEE Congress on Evolutionary Computation, 1857-1864 (2006).
[27] Hui-Yuan Fan, Jouni Lampinen, A Trigonometric Mutation Operation to Differential Evolution, Journal of Global Optimization, **27**, 105-129 (2003).
[28] Jingqiao Zhang, Arthur C. Sanderson, JADE: Adaptive Differential Evolution with Optional External Archive, IEEE Transactions on Evolutionary Computation, **13**, 945-958 (2009).

[29] Yong Wang, Zixing Cai, Qingfu Zhang, Differential Evolution with Composite Trial Vector Generation Strategies and Control Parameters, IEEE Transactions on Evolutionary Computation **15**, 55-66 (2011).
[30] J. Dongli, Z. Guoxing, K.K. Muhammad, An Effective Memetic Differential Evolution Algorithm Based On Chaotic Local Search, Information Sciences, **181**, 3175-3187 (2011).
[31] W. Chunquiu, et. al., Wind Farm Micro-Siting By Gaussian Particle Swarm Optimization With Local Search Strategy, Renewable Energy, **48**, 276-286 (2012).
[32] M. G. Epitropakis, et. al., Enhancing Differential Evolution Utilizing Proximity-Based Mutation Operators, IEEE Transaction On Evolutionary Computation, **15**, 99-119 (2011).
[33] M. Weber, F. Neri, V. Tirronen, Distributed Differential Evolution With Explorative: Exploitative Population Families, Genetic Programming Evolvable, **10**, 343-371 (2009).
[34] K. V. Price, R. M. Storn, J. A. Lampinen, Differential Evolution: A Practical Approach to Global Optimization. Springer, (2005).
[35] V. Feoktistov and S. Janaqi, Generalization of the strategies in differential evolution, Proceedings of the 18th International parallel and distributed processing symposium, (2004).
[36] E. Mezura, J. Velázquez, C. A. Coello, A Comparative Study of Differential Evolution Variants For Global Optimization, Genetic and Evolutionary Computation Conference, 485-492 (2006).
[37] K. Deb, An Efficient Constraint Handling Method For Genetic Algorithms, Computer Methods In Applied Mechanics and Engineering, **186**, 311-338 (2000).
[38] S. Kukkonen, J. Lampinen, Constrained Real-Parameter Optimization With Generalized Differential Evolution. IEEE Congress On Evolutionary Computation, 911-918 (2006).
[39] E. Mezura-Montes, J. Velázquez-Reyes, C.A. Coello Coello, Comparing Differential Evolution Models For Global Optimization. Genetica And Evolutionary Computation Conference, 485-492 (2006).
[40] Zielinski, K., Laur, R., Stopping Criteria For Differential Evolution In Constrained Single-Objective Optimization, Advancesin Differential Evolution. Springer, 111-138 (2008).



Miguel G. Villarreal-Cervantes received the B. S. degree in electronics engineering from the Instituto Tecnológico de Veracruz in 2003 and the M. Sc. and Ph. D. degrees in electrical engineering from the Centro de Investigación y de Estudios Avanzados (CINVESTAV), Mexico in 2005 and 2010, respectively. He is currently a Researcher of the Mechatronic Section in the Centro de Innovación y Desarrollo Tecnológico en Cómputo del Instituto Politécnico Nacional (CIDETEC-IPN). He is member of the National System of Researcher (S.N.I.) in Mexico. His research interests include robotics and the system optimization, dynamics and control of electro-mechanical systems.

Daniel De la Cruz-Muciño received the B. S. degree in mechanical and electrical engineering from the Instituto Politécnico Nacional (IPN) in 2009. He is currently studied a master program in the mechatronic section in CIDETEC-IPN.

Edgar A. Portilla-Flores Received the B.S. degree in Communications and Electronic engineering from Universidad Autónoma Metropolitana, Mexico in 1992 and the M.S. degree in mechanical engineering from Instituto Tecnológico de Puebla, Mexico in 2002, Ph.D. degree in Electrical engineering from CINVESTAV-IPN in 2006, and a Postdoc at Universidad Estatal de Campinas, Brasil in 2012 . He is a researcher in Optimization and Concurrent Design of Mechatronic Systems in CIDETEC-IPN. He is member of the S.N.I. in Mexico.