ISSN 2395-8618

Influence of the Binomial Crossover in the DE Variants Based on the Robot Design with Optimum Mechanical Energy

Miguel G. Villarreal-Cervantes, Daniel De-la-Cruz-Muciño, Carlos Ricaño-Rea, Jesus Said Pantoja-García

***Abstract*—Differential evolution (DE) is a powerful algorithm to find an optimal solution in real world problems. Nevertheless, the binomial crossover parameter is an important issue for the success of the algorithm. The proper selection of the binomial crossover parameter depends on the problem at hand. In this work, the effect of the binomial crossover in the DE/Rand/1/bin, DE/Best/1/Bin and DE/Current to rand/1/Bin is empirically studied and analyzed in the optimum design of the kinematic and the dynamic parameters of links for a parallel robot. The optimum design minimizes mechanical energy and consequently reduces the energy provided by the actuator. Based on the experimental results, the range of crossover parameter values that properly explores the search space is obtained. The importance of finding a proper crossover parameter is highlighted. In addition, the optimal design shows a decrease in the parallel robot mechanical energy compared with non-optimal design.**

***Index Terms*—Differential evolution, binomial crossover, optimum design, mechatronic design.**

I. Introduction

DIFFERENTIAL Evolution (DE) has been proved to be a powerful evolutionary algorithm in many real-world problems due to it being highly flexible to adapt to diverse problems (nonlinear, discontinuos, etc.). It presents a superior performance in the majority of applications and it is easy to program. In [1], DE is used and modified to parameterise an equivalent circuit model of lithium-ion batteries. A boundary evolution strategy (BES) is developed and incorporated into the DE to update the parameter boundaries during the parameterizations. The method can parameterize the model without extensive data preparation.The efficiency of the approach is verified through two battery packs, one is an 8-cell battery module and the other from an electrical vehicle. In [2], DE is used to search a global optimum solution for ball bearings link system assembly weight with constraints and mixed design variables. The implementation of the DE algorithm into the particular mechanical design shows a robust performance and obtains an efficient solution to the problem. Beside, the comparisons with other algorithms confirm the effectiveness and the superiority of the DE in terms of the quality of the obtained solution. In [3], DE solves the dimensional synthesis of four and six-bar mechanisms for path generation. In [4], the simulation-optimization approach to determine the optimum location of groundwater production wells is stated as an optimization problem. The DE algorithm and the Broyden-Fletcher-Goldfarb-Shanno (BFGS) technique is used to solve it. A significant conclusion is that the simulation-optimization model consistently finds well locations in less vulnerable areas of the model domain. Nevertheless, in the previous works [1]–[4], the selection of the parameter for the DE algorithm is a crucial factor to find better solutions in such problems. An important open issue is that the performance of the DE algorithm is highly dependent on a mutation and crossover parameter [5]. The binomial (uniform) crossover operator allows the generation of a new individual, called trial vector, from the target and mutant vectors, according to a uniform probability given by the crossover constant $CR \in [0, 1]$. Thus, the crossover constant controls which and how many elements from the current population are mutated. The right selection of the mutation and crossover parameters is a very important factor to determine the quality of the obtained solution and the efficiency of the search [6]. The selection of the suitable parameters depends on the specific problem and the previous experience of the user [7]. Unfortunately, there is no methodology to determine the mutation and crossover parameter. In this paper, an empirical study of the binomial crossover parameters on three different differential evolution variants based on the parallel robot design with optimum mechanical energy is presented.

On the other hand, the demand on high performance mechatronic systems has been a crucial factor to study the mechatronic design approach [8]–[11]. The general philosophy from the mechatronic design approach is to create an integrated design environment which promotes simultaneous design among mechanical engineering, electrical engineering, control engineering and computer engineering. Nevertheless, lately only the mechanical structure design and the control system design have been simultaneously integrated to achieve an optimal system performance due to the complexity for integrating all areas. Therefore, mechatronic system performance not only relies on its controller, but also on its mechanical structure design. Some mechatronic

Manuscript received on July 02, 2014, accepted for publication on January 20, 2015, published on June 15, 2015.

The authors are with the Instituto Politécnico Nacional, CIDETEC, Mechatronic Section, Postgraduate Department, Juan de Dios Bátiz s/n, 07700, DF, Mexico (e-mail: {mvillarrealc, ddelacruzm, cricanor, jpantjag}@ipn.mx).

ISSN 2395-8618

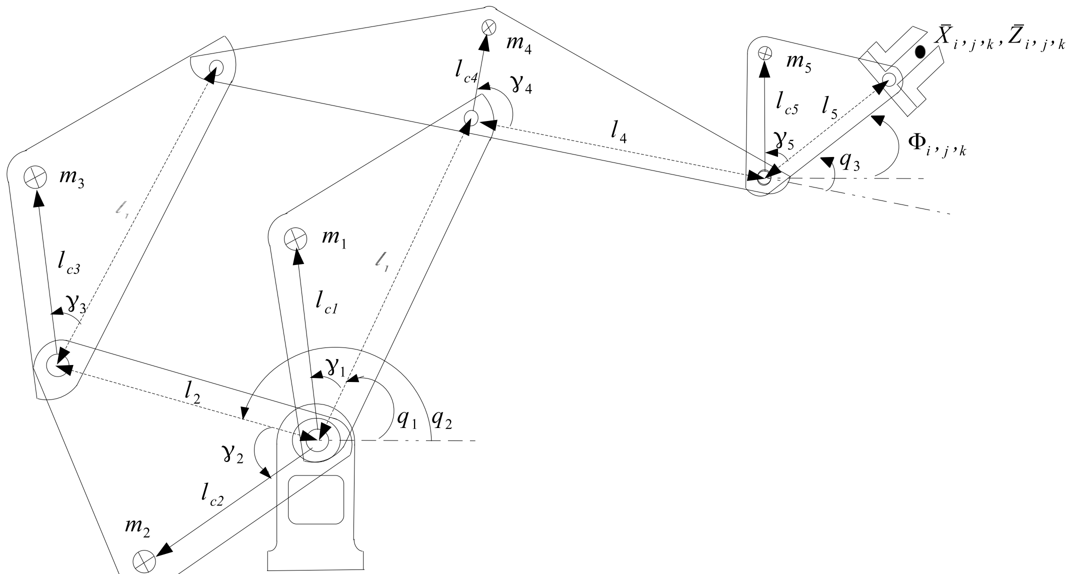

Fig. 1. Schematic diagram of the parallel robot

design works establish optimization problems to integrate both designs due to the non-linear dynamic/static nature. Meta-heuristic algorithms have been used to solve such problems due to the complex relationship in the mechatronic design. Nevertheless, few works are related to the performance of the meta-heuristic algorithm, which is an important issue to be analyzed in order to improve the obtained solutions in the mechatronic design framework.

In [12], a differential evolution algorithm with a constraint handling mechanism is proposed to simultaneously solve the design of the mechanical structure parameters of a parallel robot and the design of the proportional-integral-derivative control system required to perform a task in the Cartesian space. In [13], an approach based on a differential evolution algorithm to promote parametric reconfiguration characteristics on a continuously variable transmission $C.V.T.$ and on a parallel robot optimal design is presented. In [14], a hybrid evolutive-gradient optimization technique is proposed with the purpose of finding the optimal solutions in the search space of the synergetic design of a planar parallel robot and its control system.

Considering the mechatronic design approach, in this paper the mechanical structure of a parallel robot is designed such that the control system is improved from the energy consumption point of view. The parallel robot design is stated as an optimization problem and is solved by using three different variants of the DE algorithm. The selection of the binomial crossover parameter is analyzed to show the importance of adequately selecting such parameter in the mechatronic design framework.

The paper is organized as follows: In Section 2 the design variables, objective function and constraints of the optimization problem are described. The differential evolution algorithm is explained in Section 3. The results obtained by using the DE algorithm are described and discussed in Section 4 as well as the optimum design performance. Finally, in Section 5 the conclusions are drawn.

II. Robot design approach with optimum mechanical energy

The present work states, based on an optimization problem, the optimal dynamic and kinematic design parameters of a parallel robot which reduce the mechanical energy in a defined workspace and guarantee a dextrous workspace. The defined workspace must be described by its vertices $(\bar{x}_{d_{i,j,k}}, \bar{z}_{d_{i,j,k}})\,\forall\, i,j = 1,2$. In order to ensure a dextrous workspace, the end-effector of the parallel robot must reach three different desired orientations (described by $\bar{\phi}_{i,j,k}$ $\forall$ $k = 1,2,3$) for each vertex. The parallel robot has three degree of freedom and the end-effector can move in the X–Z plane, as is shown in Fig. 1, where q_i, $\dot{q}_i$, $\ddot{q}_i$, are the joint angular position, joint velocity and joint acceleration. The Cartesian coordinate and the angular position of the manipulator end-effector are represented by $(\bar{x}_{i,j,k}, \bar{z}_{i,j,k})$ and $\bar{\phi}_{i,j,k}$, respectively. The dynamic parameters of the i-th link are the mass m_i, mass center length l_{c_i} and inertia I_i. The kinematic parameter is the i-th link length l_i.

The next sections describe the design variables, objective function, and constraints involved in the optimization problem.

A. Design variable vector

The kinematic and dynamic parameters of the links are considered as the design variables

$$p_m = [l_1, l_2, l_4, l_5, m_1, ..., m_5, l_{c_1}, ..., l_{c_5}, \gamma_1, ..., \gamma_5]^T \in R^{19},$$

ISSN 2395-8618

```
BEGIN
  G = 0;
  Create a random population x_{i,G} ∀i = 1,...,NP
  Evaluate J(x_{i,G}), g(x_{i,G}), ∀i = 1,...,NP
  Do
    For i = 1 to NP Do
    Select randomly {r1 ≠ r2 ≠ r3} ∈ x_G.
    j_rand = randint(1, D)
      For j = 1 to D Do
        Mutation and crossover
      End For
      Evaluate J(u_{i,G+1}), g(u_{i,G+1})
      If u_{i,G+1} is better than x_{i,G} (Based on CHM) Then
        x_{i,G+1} = u_{i,G+1}
      Else
        x_{i,G+1} = x_{i,G}
      End
        G = G + 1
  While (G ≤ G_Max)
END
```

Fig. 2. The DE algorithm with the constraint handling mechanism

because these modify the mechanical structure of the parallel robot. It is assumed that the length of the third link is $l_3 = l_2$, such that it is not included in the design variable vector. In addition, the joint angle configurations of the robot are chosen as other design variables

$$p_q = [q_{1_{i,j,k}}, q_{2_{i,j,k}}, q_{3_{i,j,k}}]^T \in R^{36} \subseteq W$$

$\forall\, i,j = 1,2,\ k = 1,2,3$, where the space W is defined as $W = \{q | q \in p_q\}$. Hence the design variable vector is described in (1):

$$p = [p_m, p_q]^T \in R^{55}. \quad (1)$$

B. Objective Function

One way to optimize the mechanical energy of the parallel robot is to minimize the robot dynamic load. Then, the sum of the Frobenious norm of the inertia matrix $\|M\|_F$ and the potential energy V of the parallel robot is proposed as the objective function to be minimized. The objective function is shown in (2). The potential energy and the inertia matrix can be obtained in [15]:

$$J(p) = \int_W (\|M(p_m, q)\|_F + V(p_m, q))dW. \quad (2)$$

C. Design constraints

The dextrous workspace of a robot is defined as the set of all reachable points in the Cartesian space by its end-effector with different orientations [15]. Thus, the desired dextrous workspace is bounded by the vertices $(\bar{x}_{d_{i,j,k}}, \bar{z}_{d_{i,j,k}}) \forall i,j = 1,2$ and is assumed that if the end-effector reaches the four vertices with three different orientations $\bar{\phi}_{d_{i,j,k}} \forall k = 1,2,3$, then any point inside the workspace is reachable with at least three different orientations. Therefore a dextrous workspace is promoted. According to the previous comments, the inequality constraint described in (3) is chosen to guarantee a desired dextrous workspace. The vertices of the desired dextrous workspace (see Fig. 1) are chosen as $(\bar{x}_{d_{1,1,k}}, \bar{z}_{d_{1,1,k}}) = (0.2m, -0.23m)$, $(\bar{x}_{d_{1,2,k}}, \bar{z}_{d_{1,2,k}}) = (0.2m, 0.26m)$, $(\bar{x}_{d_{2,1,k}}, \bar{z}_{d_{2,1,k}}) = (0.5m, 0.26m)$ y $(\bar{x}_{d_{2,2,k}}, \bar{z}_{d_{2,2,k}}) = (0.5m, -0.23m)$. An additional point $(\bar{x}_{d_{3,1,k}}, \bar{z}_{d_{3,1,k}}) = (0.2m, 0m)$ is chosen in order to fulfill the three different orientations in the workspace. The subindex k indicates the three different desired orientation for each vertex. The orientations are defined as $\bar{\phi}_{d_{i,j,1}} = -\frac{\pi}{2}\, rad$, $\bar{\phi}_{d_{i,j,2}} = 0\, rad$ and $\bar{\phi}_{d_{i,j,3}} = \frac{\pi}{2}\, rad\ \forall\ i,j = 1,2$:

$$\begin{aligned} g_1: \ & \int_w \begin{pmatrix} \bar{x}_{d_{i,j,k}} - (l_1 \cos q_{1_{i,j,k}} - l_4 \cos q_{2_{i,j,k}} \\ -l_5 \cos(q_{2_{i,j,k}} + q_{3_{i,j,k}})) \end{pmatrix}^2 dW + \\ & \int_w \begin{pmatrix} \bar{z}_{d_{i,j,k}} - (l_1 \sin q_{1_{i,j,k}} - l_4 \sin q_{2_{i,j,k}} \\ -l_5 \sin(q_{2_{i,j,k}} + q_{3_{i,j,k}})) \end{pmatrix}^2 dW + \\ & \frac{1.8}{\pi} \int_w \left(\bar{\phi}_{d_{i,j,k}} - (q_{2_{i,j,k}} + q_{3_{i,j,k}} - \pi)\right)^2 dW - \\ & 1 \times 10^{-6} \le 0 \end{aligned} \quad (3)$$

Another important constraint in the parallel robot design is to avoid the collision of links in the parallel structure. Hence, the angular motion must be bounded. The inequality constraints described in (4)-(5) are included to avoid collisions between links, where $Tol_{Max2} = \frac{5\pi}{36}\ rad$ is the minimum security angle between two links:

$$g_{2\to13} : Tol_{Max2} - q_{2_{i,j,k}} + q_{1_{i,j,k}} \le 0 \quad (4)$$
$$g_{14\to25} : q_{2_{i,j,k}} - q_{1_{i,j,k}} - \pi + Tol_{Max2} \le 0 \quad (5)$$

The last constraints involve the bound in the design variables vector p. Those are stated in (6)-(9), where the maximum and minimum values are shown in Table I:

$$g_{26\to37} : 0 < q_{1_{i,j,k}} < \pi - Tol_{Max2} \quad (6)$$
$$g_{38\to49} : Tol_{Max2} \le q_{2_{i,j,k}} \le \frac{3}{2}\pi - Tol_{Max2} \quad (7)$$
$$g_{50\to61} : -\pi + Tol_{Max1} \le q_{3_{i,j,k}} \le \pi - Tol_{Max1} \quad (8)$$
$$g_{62\to80} : p_{m_{Min}} \le p_m \le p_{m_{Max}} \quad (9)$$

TABLE I
DESIGN VARIABLE VECTOR BOUNDS

Design variable	Min	Max
$q_1\ [rad]$	0	$\frac{31\pi}{36}$
$q_2\ [rad]$	$\frac{5\pi}{36}$	$\frac{49\pi}{36}$
$q_3\ [rad]$	$-\frac{35\pi}{36}$	$\frac{35\pi}{36}$
$l_i\ [m]\ i \in \{1,2,4,5\}$	0.01	0.5
$m_i\ i \in \{1,2,3\}$	0.1	0.35
m_4	0.3	0.35
m_5	0.2	0.35
$l_{c_i}\ i \in \{1,2,...,5\}$	0	0.4
$\gamma_i\ i \in \{1,2,...,5\}$	$-\pi$	π

ISSN 2395-8618

Nomenclature	Variant
rand/1/bin	$u_j^i = \begin{cases} x_j^{r_3} + F(x_j^{r_1} - x_j^{r_2}) & \text{if } \text{rand}_j(0,1) < CR \text{ or } j = j_{rand} \\ x_{i,j} & \text{otherwise} \end{cases}$
best/1/bin	$u_j^i = \begin{cases} x_j^{best} + F(x_j^{r_1} - x_j^{r_2}) & \text{if } \text{rand}_j(0,1) < CR \text{ or } j = j_{rand} \\ x_j^i & \text{otherwise} \end{cases}$
current-to-rand/1/bin	$u_j^i = \begin{cases} x_j^i + K(x_j^{r_3} - x_j^i) + F(x_j^{r_1} - x_j^{r_2}) & \text{if } \text{rand}_j(0,1) < CR \text{ or } j = j_{rand} \\ x_j^i & \text{otherwise} \end{cases}$

Fig. 3. DE variants with binomial crossover

D. Optimization problem statement

The optimization problem for the parallel robot design consists in finding the optimal design parameter vector p^* which minimizes the mechanical energy of the robot (2) subject to inequality constraints related to the design such as to have a desired dextrous workspace (3), to avoid the collision between links (4)-(5) and to limit the design variable vector (6)-(9). Then, the optimization problem can be formally stated as in (10)–(11):

$$\underset{p \in R^{55}}{Min}\ J \tag{10}$$

subject to:

$$g(p) \leq 0 \in R^{65}. \tag{11}$$

III. Differential evolution algorithm

The differential evolution (DE) algorithm is a stochastic, population-based algorithm developed by Storn and Price [5], designed for optimization problems in continuous search space. DE is a real-valued number encoded evolutionary strategy for global optimization. It has been shown to be an efficient, effective and robust optimization algorithm. The main advantages of the DE algorithm are: $i)$ The DE is a population based algorithm, $ii)$ No additional computation is needed to define the search direction, such as, gradient vector, Hessian matrix, $iii)$ The DE can be used for different kinds of optimization problems, such as, continuos, discontinuous, etc. Nevertheless, the original DE algorithm lacks a constraint handling mechanism. In this paper a constraint handling mechanism is included into the DE algorithm [16]. The key parameters are: NP - the population size that is the set of individuals, CR - the crossover constant that controls the influence of the parent in the generation of the offspring (higher values mean less influence of the parent), F - the weight applied to the influence of two of the three individuals selected at random in order to generate the offspring (scaling factor). The DE algorithm with the constraint handling mechanism works as follows: the initial population vector called parent is randomly generated. It is mutated and recombined in order to produce another population vector called mutant vector. The offspring vector will inherit features from the mutant vector or from its parent which depends on the uniform crossover. Finally, the new population for the next generation is selected between the parent and the offspring vector taking into account the constraint handling mechanism [16]:

- Any feasible solution is preferred to any infeasible solution.
- Among two feasible solutions, the one having better objective function value is preferred.
- Among two infeasible solutions, the one having smaller constraint violation is preferred.

Once the new population is created, all process (mutation, recombination and selection) are repeated until a pre-specified termination criterion is satisfied. The DE algorithm with the constraint handling mechanism (CHM) is described in Fig. 2.

Different DE variants with binomial crossover are used in this paper. The differences among those variants are in the recombination operator and in the way of selecting the elements in the individual. A summarize of the DE variants used in this paper is shown in Fig. 3.

IV. Results and discussion

The experiments are programmed in Matlab on a windows platform on a PC with 2.8 GHz core $i - 7$ with 16GB of RAM. The population size NP of the DE algorithm consists of 36 individuals, the maximum generation is $G_{Max} = 50000$. Five independent runs are carried out with ten different values of crossover parameter $CR = [0, 0.2, ..., 0.9, 1]$.

In Tables II, III and IV, the empirical results of the DE variants with different crossover factor for the optimum parallel robot design are shown. The term J_{mean}, $\sigma(J)_{mean}$ is the mean and the standard deviation of the best objective function in the runs, J_{Best} is the best objective function found in all runs, $Time$ is the mean of the convergence time in the runs and $\#gUF$ is the percentage from the five runs which does not find feasible solution.

In Table II the empirical behavior of the Rand 1 Bin is displayed. It it observed that the best performance function values is $J^* = 0.0742$ and it will be considered as the optimum one. There are runs when $CR \in \{[0, 0.1, 0.2], 1\}$ that neither find feasible solution nor converge to the optimum solution J^*. Moreover, the individuals of those results are dispersed in the search space as is observed in the standard deviation. On the other hand, some runs with $CR \in \{[0.3, 0.5], 0.9\}$ find feasible solution and only with $CR = 0.9$ the converge to the optimal solution is given (see J_{mean} and $\sigma(J)_{mean}$). The results with $CR \in \{[0.3, 0.5]\}$ indicate that

ISSN 2395-8618

TABLE II
EMPIRICAL BEHAVIOR OF THE RAND 1 BIN ALGORITHM

$Algorithm$	CR	J_{mean}	$\sigma(J)_{mean}$	J_{Best}	$Time$ [hr]	$\#gUF$
Rand 1 Bin	0.00	3.0417	1.2605	1.0550	0.32	100%
Rand 1 Bin	0.10	3.1331	0.4526	2.7439	0.34	100%
Rand 1 Bin	0.20	3.6704	0.9251	2.7961	0.33	100%
Rand 1 Bin	0.30	2.6865	1.7699	0.1040	0.33	80%
Rand 1 Bin	0.40	0.6523	0.9613	0.1200	0.33	20%
Rand 1 Bin	0.50	1.2072	2.4889	0.0808	0.33	20%
Rand 1 Bin	**0.60**	**0.0744**	**0.0001**	**0.0742**	**0.33**	**0%**
Rand 1 Bin	**0.70**	**0.0742**	**0.0000**	**0.0742**	**0.33**	**0%**
Rand 1 Bin	**0.80**	**0.0745**	**0.0003**	**0.0742**	**0.34**	**0%**
Rand 1 Bin	0.90	2.8655	1.8061	0.0742	0.33	80%
Rand 1 Bin	1.00	22.2395	17.6979	1.8977	0.33	100%

TABLE III
EMPIRICAL BEHAVIOR OF THE BEST 1 BIN ALGORITHM

$Algorithm$	CR	J_{mean}	$\sigma(J)_{mean}$	J_{Best}	$Time$ [hr]	$\#gUF$
Best 1 Bin	0.00	2.6063	1.1047	1.4722	0.33	100%
Best 1 Bin	0.10	1.8437	0.8270	1.2125	0.32	100%
Best 1 Bin	0.20	0.8046	0.6280	0.3204	0.32	100%
Best 1 Bin	0.30	1.1269	0.5443	0.4283	0.33	100%
Best 1 Bin	0.40	0.4054	0.0803	0.3312	0.32	100%
Best 1 Bin	0.50	0.8405	0.3533	0.4492	0.33	100%
Best 1 Bin	0.60	0.4034	0.3700	0.1189	0.32	100%
Best 1 Bin	0.70	0.1026	0.0924	0.0103	0.32	100%
Best 1 Bin	0.80	0.1035	0.1647	0.0069	0.32	100%
Best 1 Bin	0.90	0.3434	0.2422	0.0310	0.32	100%
Best 1 Bin	1.00	7.5324	5.8768	1.5052	0.32	100%

TABLE IV
EMPIRICAL BEHAVIOR OF THE CURRENT TO RAND 1 BIN ALGORITHM

$Algorithm$	CR	J_{mean}	$\sigma(J)_{mean}$	J_{Best}	$Time$ [hr]	$\#gUF$
Current to rand 1 Bin	0.00	2.7856	1.2458	1.0478	0.32	100%
Current to rand 1 Bin	**0.10**	**0.9398**	**1.1931**	**0.0758**	**0.32**	**40%**
Current to rand 1 Bin	**0.20**	**3.6631**	**2.4611**	**0.0978**	**0.32**	**80%**
Current to rand 1 Bin	**0.30**	**3.6965**	**2.3941**	**0.0753**	**0.33**	**80%**
Current to rand 1 Bin	0.40	4.6247	1.1783	2.9582	0.33	100%
Current to rand 1 Bin	0.50	4.0629	1.5545	2.6370	0.34	100%
Current to rand 1 Bin	0.60	3.6810	0.6680	2.7071	0.34	100%
Current to rand 1 Bin	0.70	4.2893	0.3206	3.9128	0.34	100%
Current to rand 1 Bin	0.80	6.1524	1.0623	4.7779	0.34	100%
Current to rand 1 Bin	0.90	4.5003	2.3716	0.9021	0.34	100%
Current to rand 1 Bin	1.00	27.2530	20.8946	7.8482	0.34	100%

in spite of producing feasible solutions the converge to the optimum one is not reached, which means that suboptimal solutions are found. The best results are given with $CR \in [0.6, 0.8]$ because they find feasible solution in all runs and the convergence to the optimal solution is always reached in all runs (see J_{mean} and $\sigma(J)_{mean}$).

Tables III and IV show the Best 1 Bin and the Current to rand 1 Bin behaviors, respectively. It is observed that the Best 1 Bin algorithm performs poorly. The convergence is towards unfeasible solutions. This indicates that the use of the best individuals in the mutation process accelerate the convergence to unfeasible solutions and there is a lack of diversity in the solution. On the other hand, Current to rand 1 Bin finds local solutions near the optimum one with $CR \in \{[0.1, 0.3]\}$ and those solutions do not converge to a similar performance function value (see the standard deviation).

In all DE variants, the convergence time of the results is competitive among different crossover values. Clearly, selection of the crossover parameter CR is a very important factor in the parallel robot design, because different values of the crossover parameter are required to find feasible solutions among the DE variants. The optimal empirical results indicate that the best DE variant among those analyzed is the DE Rand 1 Bin with the best crossover probability 0.6%–0.8%. Hence the influence of the mutant vector in the generation of the child vector (offspring) must be larger than the parent (target) vector influence. A tradeoff in the selection between mutant and parent vectors must be considered in order to obtain the best solution, and it depends on the problem at hand. A suitable selection of the crossover parameter promotes a better exploration of the search space and the success of the DE variant to find the optimal design parameters of the parallel robot.

ISSN 2395-8618

TABLE V
OPTIMUM PARAMETERS OF THE PARALLEL ROBOT

Mass [Kg]	$m_1 = 0.3499$	$m_2 = 0.3499$	$m_3 = 0.3499$	$m_4 = 0.3$	$m_5 = 0.2$
Length [m]	$l_1 = 0.2263$	$l_2 = 0.0765$	$l_4 = 0.3514$	$l_5 = 0.03$	
Mass center length [m]	$l_{c_1} = 0.1566$	$l_{c_2} = 0.0546$	$l_{c_3} = 0.1520$	$l_{c_4} = 0.0783$	$l_{c_5} = 0.0095$
Mass center angle [rad]	$\gamma_1 = -3.1415$	$\gamma_2 = 0.0842$	$\gamma_3 = 3.1215$	$\gamma_4 = -3.1284$	$\gamma_5 = -1.7150$

TABLE VI
NON-OPTIMUM PARAMETERS OF THE PARALLEL ROBOT

Mass [Kg]	$m_1 = 0.3$	$m_2 = 0.25$	$m_3 = 0.16$	$m_4 = 0.35$	$m_5 = 0.13$
Length [m]	$l_1 = .2$	$l_2 = 0.05$	$l_4 = .25$	$l_5 = 0.072$	
Mass center length [m]	$l_{c_1} = 0.0524$	$l_{c_2} = 0.0114$	$l_{c_3} = 0.1$	$l_{c_4} = 0.0643$	$l_{c_5} = .0185$
Mass center angle [rad]	$\gamma_1 = 0$	$\gamma_2 = 0$	$\gamma_3 = 0$	$\gamma_4 = \pi$	$\gamma_5 = 0$

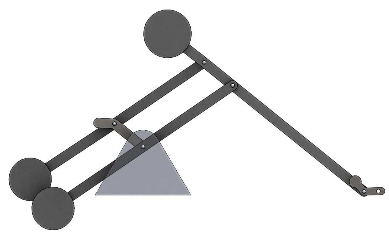

Fig. 4. Parallel robot with optimum links

On the other hand, the optimum design parameter vector is shown in Table V. In Fig. 4 the shape of the links of the parallel robot with the optimum design parameter are displayed. The shapes of the links are obtained by considering the optimum design parameter vector and making empirical Computer Aided Designs (CAD) in Solidworks until the design fulfills the optimum design parameter vector.

In order to verify the mechanical energy of the optimum parallel robot design, simulation results were used. In this case, a circle in the X–Z plane and a sinusoidal signal are chosen as the desired position and orientation to be followed by the end-effector of the parallel robot, respectively. The desired trajectory is shown in (12)-(14). A proportional-integral-derivative (PID) control is selected for this goal:

$$X_d = 0.35 + 0.1\cos(0.6283t) \tag{12}$$

$$Z_d = 0.1\sin(0.6283t) \tag{13}$$

$$\phi_d = 0.0872\sin(2.0943t) \tag{14}$$

The PID gains are selected by a trial and error procedure. Those gains are: $k_{p_1} = 20$, $k_{i_1} = 5$, $k_{d_1} = 3$, $k_{p_2} = 15.8$, $k_{i_2} = 5.4$, $k_{d_2} = 1.1$, $k_{p_3} = 0.8$, $k_{i_3} = 0.8$, $k_{d_3} = 0.005$. In Fig. 5 the trajectory tracking of the end-effector is given. It is observed that the end-effector trajectory is in the desired workspace (bounded by a squared continuous line) and the PID control system stabilizes the end-effector in the trajectory. The control signal (applied torque) to follow the desired trajectory is shown in Fig. 6. It is observed, after the first second, the control torque is low, such that, the mechanical energy of the parallel robot is low too.

TABLE VII
COMPARISON OF THE CONTROL SIGNAL NORM WITH BOTH APPROACHES

Design approach	$\|u_1\|$	$\|u_2\|$	$\|u_3\|$
Optimum	1.1999	0.3245	0.1293
Non optimum	33.0886	0.7917	1.0584

In order to compare the proposed optimum design of the parallel robot, comparative results with a non optimum design are carried out. The non optimum design parameters are chosen with the consideration that the total mass of the parallel robot is smaller than the total mass of the optimum design of the parallel robot. Simulation results are performed with both designs and the comparative results are carried out by analyzing the norm of the control signal of the tracking trajectory. The non optimum design are chosen accordingly to Table VI and its PID gains are proposed as: $k_{p_1} = 130$, $k_{i_1} = 35$, $k_{d_1} = 3$, $k_{p_2} = 55.8$, $k_{i_2} = 5.4$, $k_{d_2} = 1.1$, $k_{p_3} = 0.8$, $k_{i_3} = 0.8$, $k_{d_3} = 0.005$.

In Table VII the norm of the control signals are shown. It is observed that the norm of the control signal in the optimum design is smaller than the non optimum design, in spite of having more total mass. Hence, the optimum mechanical structure of the parallel robot minimizes the mechanical energy, resulting that the torque provided by the control system is reduced. Then, the proposed design approach promotes the mechatronic design approach because the optimum mechanical structure improves the energy efficiency of the control system.

V. CONCLUSIONS

In this work, an optimum design approach for a parallel robot is stated as an optimization problem. This approach finds the dynamic and kinematic parameters of links that fulfill with a structure with less mechanical energy. Hence, as a consequence, the mechanical structure improves the control system behavior w.r.t. the energy consumption in the trajectory tracking.

The main highlights in the selection of the crossover parameter in DE variants are:

– The DE best 1 bin presents a high premature convergence to unfeasible solution in spite of the crossover parameter selection.

ISSN 2395-8618

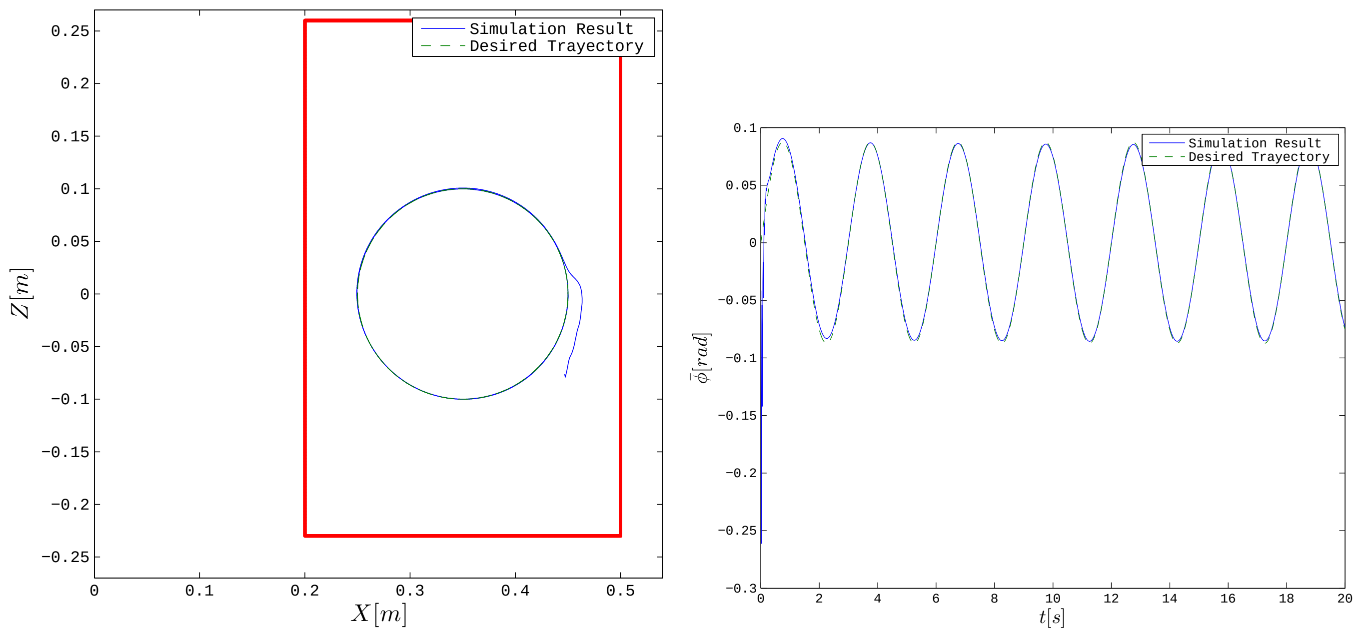

Fig. 5. Trajectory tracking of the optimum parallel robot with the PID control system

– The DE current to Rand 1 Bin converge to a suboptimal solution near the optimum one and it highly depends on the crossover parameter selection.
– The DE Rand 1 Bin presents a good convergence to the optimal solution with $CR = [0.6, 0.8]$. It promotes a better exploration of the search space without converging to local minima and without exhaustive exploration.
– The success of the DE variant to solve the optimum design problem effectively depends on the selection of the crossover parameter and is related to the optimization problem.

ACKNOWLEDGEMENTS

This work is supported by the Secretaría de Investigación y Posgrado del Instituto Politécnico Nacional (SIP-IPN) under project number SIP-20151212 and the CONACYT under project number 182298. The second to the fourth authors acknowledge support from CONACYT through a scholarship to pursue graduate studies at Instituto Politécnico Nacional.

REFERENCES

[1] G. Yang, "Battery parameterisation based on differential evolution via a boundary evolution strategy," *Journal of Power Sources*, vol. 245, pp. 583–593, 2014.
[2] H. Saruhan, "Differential evolution and simulated annealing algorithms for mechanical systems design," *Engineering Science and Technology, an International Journal*, 2014.
[3] A. Ortiz, J. Cabrera, F. Nadal, and A. Bonilla, "Dimensional synthesis of mechanisms using differential evolution with auto-adaptive control parameters," *Mechanism and Machine Theory*, vol. 64, pp. 210–229, 2013.
[4] A. Elci and M. T. Ayvaz, "Differential-evolution algorithm based optimization for the site selection of groundwater production wells with the consideration of the vulnerability concept," *Journal of Hydrology*, vol. 511, pp. 736–749, 2014.
[5] K. Price, R. M. Storn, and J. A. Lampinen, *Differential Evolution: A Practical Approach to Global Optimization*, ser. Natural Computing Series. Springer-Verlag New York, Inc., 2005.
[6] A. E. Eiben, R. Hinterding, and Z. Michalewicz, "Parameter control in evolutionary algorithms," *IEEE Transaction on Evolutionay Computation*, vol. 3, no. 2, pp. 124–141, 1999.
[7] D. Zaharie, "Influence of crossover on the behavior of differential evolution algorithms," *Applied Soft Computing*, vol. 9, no. 3, pp. 1126–1138, 2009.
[8] M. Villarreal-Cervantes, C. Cruz-Villar, J. Alvarez-Gallegos, and E. Portilla-Flores, "Robust structure-control design approach for mechatronic systems," *IEEE/ASME Transactions on Mechatronics*, vol. 18, no. 5, pp. 1592–1601, Oct 2013.
[9] C. A. Cruz-Villar, J. Alvarez-Gallegos, and M. G. Villarreal-Cervantes, "Concurrent redesign of an underactuated robot manipulator," *Mechatronics*, vol. 19, no. 2, pp. 178–183, 2009.
[10] Q. Li, W. Zhang, and L. Chen, "Design for control-a concurrent engineering approach for mechatronic systems design," *IEEE/ASME Transactions on Mechatronics*, vol. 6, no. 2, pp. 161–169, Jun 2001.
[11] S. Alyaqout, P. Papalambros, and A. Ulsoy, "Combined robust design and robust control of an electric DC motor," *IEEE/ASME Transactions on Mechatronics*, vol. 16, no. 3, pp. 574–582, June 2011.
[12] M. G. Villarreal-Cervantes, C. A. Cruz-Villar, J. Alvarez-Gallegos, and E. A. Portilla-Flores, "Differential evolution techniques for the structure-control design of a five-bar parallel robot," *Engineering Optimization*, vol. 42, no. 6, pp. 535–565, 2010.
[13] E. A. Portilla-Flores, E. Mezura-Montes, J. Alvarez-Gallegos, C. A. Coello-Coello, C. A. Cruz-Villar, and M. G. Villarreal-Cervantes, "Parametric reconfiguration improvement in non-iterative concurrent mechatronic design using an evolutionary-based approach," *Engineering Applications of Artificial Intelligence*, vol. 24, no. 5, pp. 757–771, 2011.
[14] M. G. Villarreal-Cervantes, C. A. Cruz-Villar, and J. Alvarez-Gallegos, "Synergetic structure-control design via a hybrid gradient-evolutionary algorithm," *Optimization & Engineering*, 2014.
[15] M. Spong and S. Hutchinson, *Robot Modeling and Control*. Wiley, 2005.
[16] K. Deb, "An efficient constraint handling method for genetic algorithms," *Computer methods in applied mechanics and engineering*, vol. 186, no. 2/4, pp. 311–338, 2000.

ISSN 2395-8618

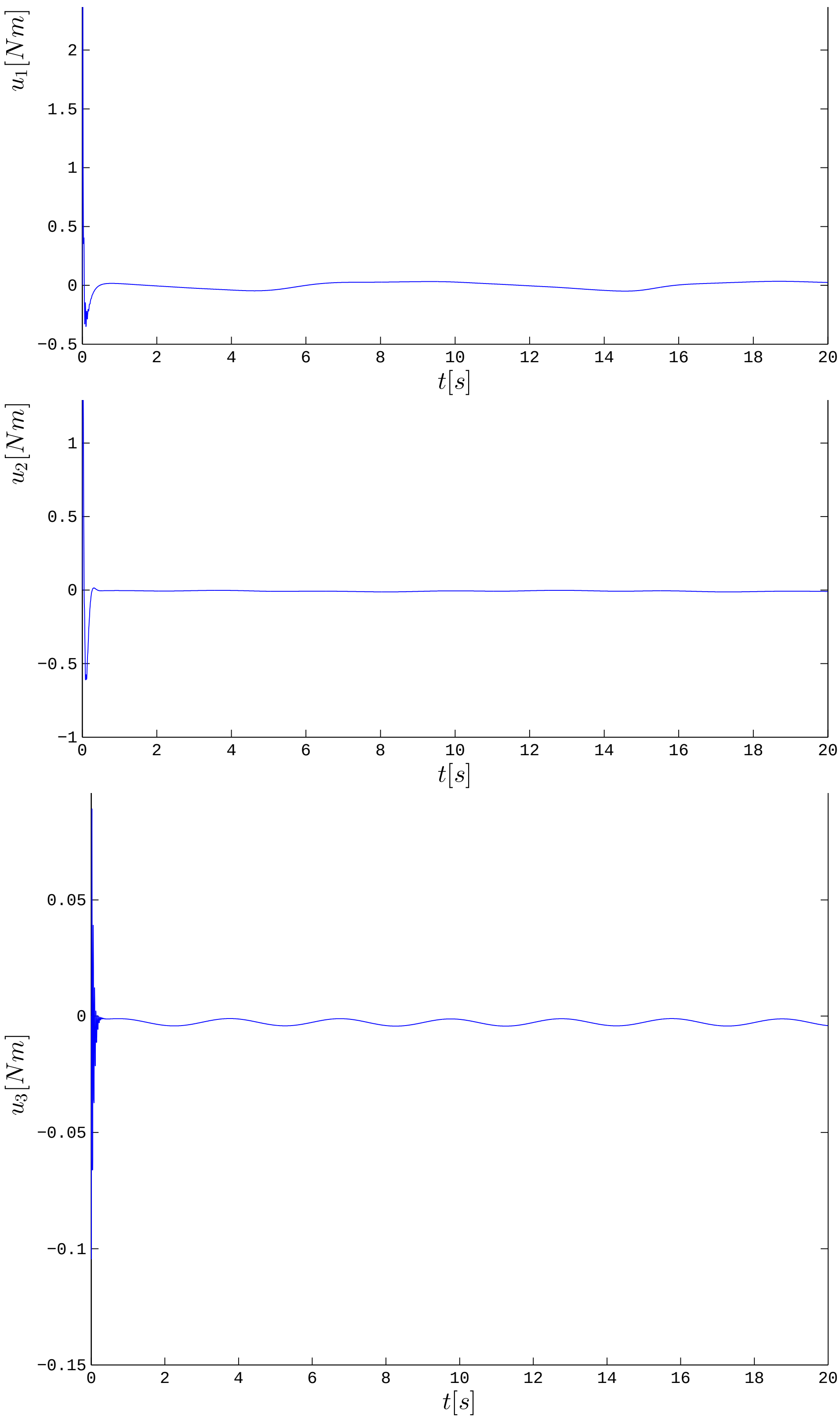

Fig. 6. PID control signal behavior