



Mechanics Based Design of Structures and Machines

An International Journal

ISSN: 1539-7734 (Print) 1539-7742 (Online) Journal homepage: <http://www.tandfonline.com/loi/lmbd20>

Structure-control design of a mechatronic system with parallelogram mechanism using an estimation of distribution algorithm

S. Iván Valdez, E. Chávez-Conde, Eusebio E. Hernandez & M. Ceccarelli

To cite this article: S. Iván Valdez, E. Chávez-Conde, Eusebio E. Hernandez & M. Ceccarelli (2016) Structure-control design of a mechatronic system with parallelogram mechanism using an estimation of distribution algorithm, *Mechanics Based Design of Structures and Machines*, 44:1-2, 58-71, DOI: [10.1080/15397734.2015.1035785](https://doi.org/10.1080/15397734.2015.1035785)

To link to this article: <http://dx.doi.org/10.1080/15397734.2015.1035785>



Published online: 06 Apr 2016.



Submit your article to this journal [↗](#)



Article views: 2



View related articles [↗](#)



View Crossmark data [↗](#)

Structure-control design of a mechatronic system with parallelogram mechanism using an estimation of distribution algorithm

S. Ivvan Valdez^a, E. Chávez-Conde^b, Eusebio E. Hernandez^c, and M. Ceccarelli^d

^aCenter for Research in Mathematics (CIMAT), Department of Computer Science, Guanajuato, Mexico; ^bUniversity of Papaloapan Campus Loma Bonita, Loma Bonita, Mexico; ^cNational Polytechnic Institute, ESIME-UPT, Section of Graduate Studies and Research, Mexico City, Mexico; ^dLaboratory of Robotics and Mechatronics, University of Cassino and Southern Lazio, Cassino, Italy

ABSTRACT

In this paper, a structure-control design methodology for simultaneously optimizing both mechanical structure and control of a parallelogram linkage robot is proposed. It takes into count the dynamical model and the mechanical parameters for the optimization process along with the controller. Thus, proportional-integral-derivative (PID) control and geometric variables are optimized in a simultaneously way. Through the concurrent procedure an optimal combination of the robot structure and controller gains is obtained. The global optimization problem is tackled by using an estimation of distribution algorithm (EDA) based on the Boltzmann distribution. The EDA seeks for the global optimum by estimating and sampling a probability distribution. The proposed methodology is verified through simulation experiments and applied to the design process of a parallelogram linkage system. The results obtained in experiments show the effectiveness of the proposal. This approach is generic and could be applied to other mechanisms in similar way when for concurrent process both kinematic and dynamic models are available along with the controller. In particular, the results are promising when the optimization parameters are uncorrelated, namely control and mechanical parameters.

ARTICLE HISTORY

Received 26 March 2015
Accepted 26 March 2015

KEYWORDS

Concurrent design;
estimation of distribution
algorithms; optimization;
parallelogram linkage

1. Introduction

Products or systems dedicated for applications requiring high performance, require a multidisciplinary approach in their design process with an integrated approach. It can be considered that the mechatronic design is the integrated design of a mechanical system and its embedded control system (see, van Amerongen, 2003). In particular, the design process of robot manipulators can benefit greatly from the perspective of mechatronic design (see, Youcef-Toumi, 1996).

In recent years, parallel mechanisms have attracted increasing attention from research community and industry. It is well known that compared with traditional serial mechanisms, parallel mechanisms have potential advantages in accuracy, dynamic, and stiffness. Many architectures of robots using parallel mechanisms have been proposed and analyzed. However, the practical parallel structures show a significant difference between the expected and actual performance, due to passive joints that introduce geometric errors and deformations. In addition, most existing structures have limited workspace and singularities. This is a very demanding task for the design process of a parallel structure. Traditionally, the mechanism to be controlled is designed first and independently of the control scheme (see, for example

CONTACT Eusebio E. Hernandez  euhernandezm@ipn.mx  National Polytechnic Institute, ESIME-UPT, Section of Graduate Studies and Research, Av. Ticomán, Col. San José Ticomán, Mexico City 07340, Mexico.

Color versions of one or more of the figures in the article can be found online at www.tandfonline.com/lmbd.

Communicated by Marco Ceccarelli.

© 2016 Taylor & Francis

Delchev and Zahariev, 2008 and Valles et al., 2012). In this case, a strategy similar to the fixed point method is used, it is to say, first we optimize about the lengths, once we get optimal lengths we intend to optimize control parameters. This kind of approaches, are not actually looking for a global optimum, but find the optimum in projections of the objective function with respect of a subset of variables, and then report the optimum in such projections as an approximation of the global optimum. In our case, we tackle the actual global optimization problem, without any assumption about independence of the control and length parameters.

Thus, the design and development of these kind of structures requires a multidisciplinary approach. The optimal design of these kind of structures, considered as mechatronic systems, can be stated as a traditional optimization problem. However, the optimization techniques to address these problems can be attracted by local minima, are sensitive to initial conditions or involve the computation of gradients of the objective function.

By instance, in Pil and Asada (1996), an integrated structure/control design method is proposed, based on an iterative algorithm, for the development of a robotic system. The mechanical structure is modified iteratively by using the structure reinforcement and rapid prototyping techniques, coupled with the adjustment of the control parameters according to the mechanical structure updated. This kind of approaches requires continuity and differentiability in all the configurations that are tested iteratively, it is quite possible that these characteristics are not accomplished, hence methods to circumvent numerical problems must be proposed. In Pil and Asada (1996) some methods to circumvent these problems are proposed, such as robust numerical estimations of jacobian matrices. Even though it is not sufficient to always adequately solve this problem, even if the problem of differentiability, continuity, and numerical singularities are solved, this kind of methods can converge to a local optimum. In Li (2006), an approach to the methodology of design for control is presented, applied to a five-bar mechanism. In the optimization process is considering the mass balancing in order to simplify as far as possible, the dynamic model, and so design a controller for tracking path, simultaneously.

Recently, with the aim to circumvent these problems, optimization techniques based on metaheuristic methods have been used in the design process of different mechatronic systems. In Ravichandran et al. (2006), optimization techniques based on a heuristic evolutionary algorithm are selected for optimizing both the design parameters and non-linear PD controller gains of a two-link planar manipulator. In Portilla-Flores et al. (2011), a method based on the differential evolution algorithm is implemented to address the optimization problem in the concurrent structure and control design of mechatronic systems.

Others works about structure-control design that have been presented in the last decade use genetic algorithm (see, Affi et al., 2007) or non-dominated sorting genetic algorithm (NSGA-II) for the optimization process (see, da Silva et al., 2009, EL-Kribi et al., 2013, and Liu et al., 2012). It is remarkable that until now all integrated design approaches are applied to simple mechanisms such as four-bar mechanisms (EL-Kribi et al., 2013; Affi et al., 2007), five-bar linkages (Villarreal-Cervantes et al., 2014, 2013), and for a Cartesian robot based on three linear motor motion system (see, da Silva et al., 2009).

Most of the above design optimization problems are solved by widely used stochastic methods, which have been successful in approaching the problem, but lack of a consistent interpretation of its way of working from a mathematical viewpoint. Actually they are consistent since the artificial evolution point of view, additionally neither apriori information nor a posteriori information can be introduced or obtained, respectively, from these algorithms.

We propose the use of Estimation of Distribution Algorithms (EDAs), (see, Mühlenbein and Paaß, 1996), for addressing the design optimization problem. The EDAs maintain the advantages of other evolutionary algorithms, it is to say, they approach the global optimum, and do not require gradients or continuity, while they have a more explainable way of working from the mathematical point of view. Additionally, the EDA proposed to be used in this work has proven statistically to require less evaluations than similar algorithms and to provide high quality of solutions, when the optimization parameters are considered uncorrelated. In this work we use an EDA based on the Boltzmann

distribution: the Boltzmann Univariate Marginal Distribution Algorithm (BUMDA), (see, Valdez et al., 2013). It has been proven that conceptual EDAs based on the Boltzmann selection, such as the Boltzmann EDA (BEDA), (see, Mühlenbein et al., 1999), converge to the optimum (see, Zhang and Mühlenbein, 2004). The BUMDA uses the Boltzmann distribution through a Gaussian model, fitted according to the Kullback-Leibler divergence. The BUMDA aim is to preserve the desired characteristics of the Boltzmann distribution, while maintaining a low computational cost in the estimation and sampling steps.

In this paper, the design of a parallelogram linkage robot is carried out by this approach which integrates in a simultaneous way the structure design parameters of the robot and the PID controller gains, in order to optimize its mechanical design and to minimize the position error for a specific path. Thus, both the structure design and the controller design are simultaneously evaluated in order to obtain an appropriate system performance. The aim of this paper is to obtain the set of optimal mechanical and controller parameters in a concurrent way. The paper is organized as follows: Section 2 introduces the proposal of our concurrent design methodology. Section 3 describes the estimation of distribution algorithm-based approach. Section 4 presents the kinematic and dynamic models, and the motion controller for a Parallelogram Linkage System (PLS) as a case of study, while the optimization problem is depicted in Section 5. The results of experiments are showed and analyzed in Section 6. Finally, the conclusions are presented in Section 7.

2. Concurrent design methodology

The concurrent design methodology proposed in this paper, considers the design process as a mapping from a requirement space to a structural space. In Li et al. (2001), is suggested dividing the requirement space into two subspaces: real-time behaviors (e.g., desired path, desired tracking speeds), and non-real-time behaviors (e.g., workspace, maximum payload). Following this division, the system parameters in the structural space can also be divided into two subspaces: real-time parameters (e.g., controller gains), and non-real-time parameters (e.g., kinematics and dynamics parameters). This viewpoint has been adapted for the optimal integrated design of a parallelogram linkage robot proposed in this paper, to optimize the structural parameters and the controller gains simultaneously, satisfying the restrictions set imposed, by using an estimation distribution algorithm. It is recommended to include constraints for mass and force balance, or constraints for dynamic model simplification.

The optimization process could consider the following criteria:

- **Design objectives:** (1) Minimizing position error $e_i(t)$ for a given path. Considering, $E = \min \sum_{i=1}^n |e_i(t)|$ with $e_i(t) = q_i^* - q_i$, where q_i^* and q_i are the i -th generalized coordinates desired and actual, respectively, for n generalized coordinates. (2) Maximizing a manipulability measure $w(\mathbf{q})$ (away from singularities). Considering, $S = \max w(\mathbf{q})$, with $w(\mathbf{q}) = \sqrt{\det(J(\mathbf{q})J^T(\mathbf{q}))}$, where $J(\mathbf{q})$ is the Jacobian matrix of robotic mechanism.
- **Design constraints:** (1) Inequalities design constraints (e.g., kinematic and dynamic parameters, actuator power, workspace). (2) Equality design constraints (e.g., dynamic model simplification, desired path).
- **Design variables:** Design parameters vector \mathbf{X} (e.g., kinematic and dynamic parameters, controller gains), $\mathbf{X} \in \mathbb{R}^n$.

The actual objective function depends on the particular problem. Thus, in the case of study we will provide the specific model for such application.

Figure 1 shows a flow diagram of the process performed by the estimation of distribution algorithm, used in this concurrent design methodology. Each of the blocks in the diagram are explained in more detail in the following sections.

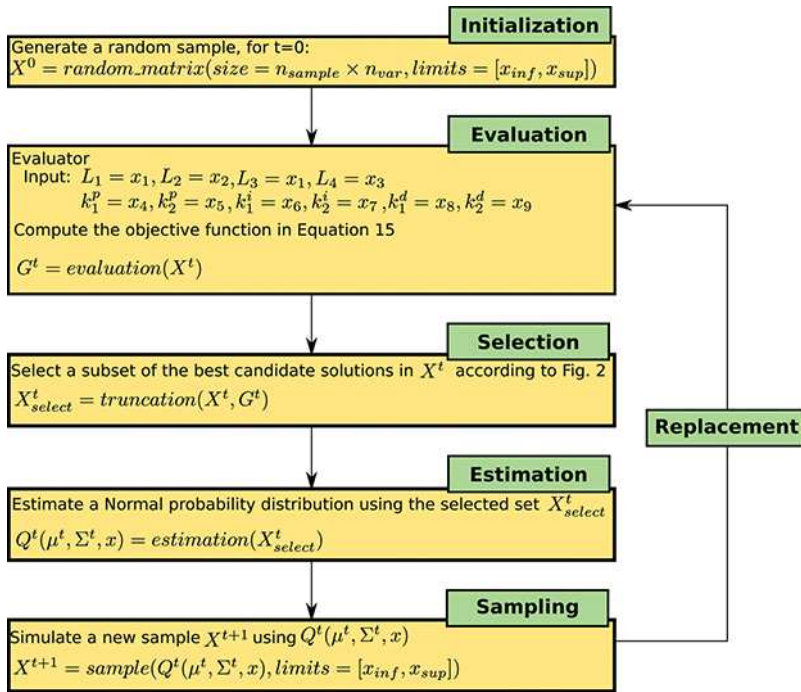


Figure 1. Flow diagram of the automated design process using an Estimation of Distribution Algorithm (BUMDA).

3. The Boltzmann Univariate Marginal Distribution Algorithm (BUMDA)

Estimation of Distribution Algorithms estimate and sample from a probability distribution. An EDA usually starts with an uniform initial sample, then it is evaluated, the best candidate solutions in the sample are selected, the selected set is used to re-estimate the probability distribution, and so on. The BUMDA (see, Valdez et al., 2013) is an EDA based on the Boltzmann distribution, the Boltzmann distribution is approximated by a Gaussian distribution because of its characteristics for estimating its parameters and sampling.

The BUMDA accomplishes two desired characteristics in any optimization algorithm:

- It maintains a non-decreasing sequence of the expected value of the objective function. In order to obtain better samples than the generation before.
- It converges to the best solution found. This allows to refine the solution, and to determine when the algorithm rarely will improve the best solution known.

The convergence is achieved by applying a truncation selection method which increases the mean of the objective value of the current sample, such as explained in Fig. 2. We ensure that it is always at least one element in the selected set by preserving the elite candidate solution.

The whole BUMDA is shown in Fig. 3. Notice that the BUMDA uses the truncation selection method to ensure a non-decreasing average (mean estimator) of the objective function of the sample. In addition, the objective function of the selected set is used to incorporate information about the fitness landscape into the Gaussian model. As can be seen it requires of two parameters: a sample size and a minimum variance, in comparison with other similar optimization algorithms the number of parameters is quite small. As suggested in Valdez et al. (2013), the sample size is chosen by setting an initial value, then increasing it until the algorithm performance does not change. The minimum variance allowed is of 5×10^{-5} and is related with the maximum error allowed.

Truncation Selection Method

Consider a sample of decreasingly sorted candidate solutions (maximization case), such that x_1 are the decision variables of the candidate solution with the maximum objective function in the sample:

1. For the initial generation $t = 0$, let be $g(x_i, 0)$ for $i = 1..N$, the objective values of the initial sample. Define: $\theta_0 = \min g(x_i, 0)$.
2. For $t > 0$, set:
 $\theta_t = \max (g(x_{N/2}, t), \min(g(x_i, t) | g(x_i, t) \geq \theta_{t-1}))$.
3. Truncate the sample such that $g(x_s, t) \geq \theta_t$. Where x_s are all the candidate solutions whose objective values are equal or greater than θ_t .

Figure 2. Truncation method to ensure convergence in the EDA.

BUMDA

1. Give the parameter and stopping criterion:
nsample \leftarrow Number of candidate solutions to be sample.
minvar \leftarrow Minimum variance allowed.
2. Uniformly generate the initial sample X_0 , set $t = 0$.
3. While $v > \text{minvar}$ for all dimensions
 - (a) $t \leftarrow t + 1$
 - (b) Evaluate and truncate the sample according to the algorithm in Fig. 2.
 - (c) Compute the approximation to μ and v (for all dimensions) by using the selected set (of size n_{selec}), as follows:

$$\mu \approx \frac{\sum_1^{n_{selec}} x_i \bar{g}(x_i)}{\sum_1^{n_{selec}} \bar{g}(x_i)}, \quad v \approx \frac{\sum_1^{n_{selec}} \bar{g}(x_i)(x_i - \mu)^2}{1 + \sum_1^{n_{selec}} \bar{g}(x_i)},$$

where $\bar{g}(x_i) = g(x) - g(x_{n_{selec}}) + 1$.

Note: the candidate solutions can be sorted to simplify the computation, and $g(x_{n_{selec}})$ is the minimum (for maximization case) objective value of the selected.

- (d) Generate $n_{sample} - 1$ candidate solutions from the new model $Q(x, t) = \text{Normal}(\mu, v)$, and insert the elite candidate solution.
4. Return the elite candidate solution as the best approximation to the optimum.

Figure 3. Pseudo-code for BUMDA.

Some of the BUMDA advantages are the following:

- The BUMDA converges to the best approximation to the optimum.
- The variance tends to 0 for a large number of generations.
- The BUMDA only needs **one** parameter (sample size).
- The estimation of the search distribution parameters results in a fast automatic adaptation. The variance could be increased or decreased, according to the solutions in the selected set and their objective values, and the mean moves fast to the region where the best solutions are.

4. Case of study: A parallelogram linkage system

The proposal is applied for the design process of a parallelogram linkage system, which can be considered as a parallel robot. Research on parallel robots have been increased recently due to their advantages in comparison with serial robots, such as high stiffness, elevate speed and accelerations, and a precise positioning capability. However, they show some disadvantages too, as singularities, friction losses or reduced workspace. Additionally, there is not a well developed standard method for analysis, synthesis, and control. The mechanism to be controlled is very often designed first and independently of the control scheme. Thus, the parallel robot design remains a challenge for optimizing both mechanical and control parameters in a concurrent way. Figure 4 shows a diagram of the parallelogram robot considered in this paper.

4.1. Kinematic model of PLS

The direct kinematic is given as,

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} L_1 \cos \varphi_1 - L_4 \cos \varphi_2 \\ L_1 \sin \varphi_1 - L_4 \sin \varphi_2 \end{bmatrix} \quad (1)$$

The solution of the inverse kinematic problem can be computed as,

$$\varphi_1 = \tan^{-1} \left(\frac{y}{x} \right) - \tan^{-1} \left(\frac{L_4 \sin \gamma}{L_1 + L_4 \cos \gamma} \right), \quad \gamma = \tan^{-1} \left(\frac{-\sqrt{1-d^2}}{d} \right) \quad (2)$$

$$\varphi_2 = \varphi_1 + (\gamma + \pi), \quad d = \cos \gamma = \frac{x^2 + y^2 - L_1^2 - L_4^2}{2L_1L_4}, \quad (3)$$

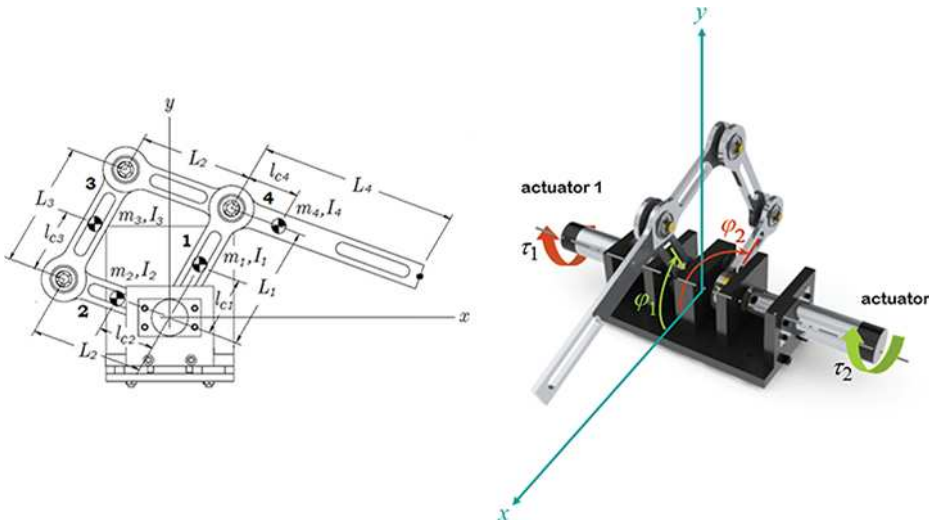


Figure 4. A schematic diagram of the parallelogram linkage system.

where $L_1, L_2, L_3, L_2 + L_4$ are the link lengths, $l_{c1}, l_{c2}, l_{c3}, l_{c4}$ are the locations of the mass centers for each link, φ_1 and φ_2 are the angular positions for the link 1 and link 2, respectively. In the same form, the differential kinematics can be computed as,

$$\dot{\mathbf{x}} = J(\mathbf{q})\dot{\mathbf{q}}, \quad (4)$$

where, $\dot{\mathbf{x}} = [\dot{x} \ \dot{y}]^T$ is the vector of Cartesian velocities in the end-effector, $\dot{\mathbf{q}} = [\dot{\varphi}_1 \ \dot{\varphi}_2]^T$ is the vector of angular velocities of the link 1 and link 2, and the Jacobian matrix is given by,

$$J(\mathbf{q}) = \begin{bmatrix} -L_1 \sin \varphi_1 & L_4 \sin \varphi_2 \\ L_1 \cos \varphi_1 & -L_4 \cos \varphi_2 \end{bmatrix}$$

4.2. Dynamic model of PLS and its controller

The dynamic model of the PLS without friction parameters has been reported in Rivin (1987), Spong and Vidyasagar (2008), Sciavicco and Siciliano (2000), which can be written as,

$$d_{11}\ddot{\varphi}_1 + d_{12}\ddot{\varphi}_2 + (m_3L_2l_{c3} - m_4L_1l_{c4})\sin(\varphi_2 - \varphi_1)\dot{\varphi}_2^2 + \phi_1(\varphi_1) = \tau_1, \quad (5)$$

$$d_{21}\ddot{\varphi}_1 + d_{22}\ddot{\varphi}_2 - (m_3L_2l_{c3} - m_4L_1l_{c4})\sin(\varphi_2 - \varphi_1)\dot{\varphi}_1^2 + \phi_2(\varphi_2) = \tau_2. \quad (6)$$

with,

$$d_{11} = m_1l_{c1}^2 + m_3l_{c3}^2 + m_4L_1^2 + I_1 + I_3,$$

$$d_{12} = d_{21} = (m_3L_2l_{c3} - m_4L_1l_{c4})\cos(\varphi_2 - \varphi_1),$$

$$d_{22} = m_2l_{c2}^2 + m_3L_2^2 + m_4l_{c4}^2 + I_2 + I_4,$$

$$\phi_1(\varphi_1) = g \cos \varphi_1 (m_1l_{c1} + m_3l_{c3} + m_4L_1),$$

$$\phi_2(\varphi_2) = g \cos \varphi_2 (m_2l_{c2} + m_3L_2 + m_4l_{c4}).$$

where, $\mathbf{q} = [\varphi_1 \ \varphi_2]^T$ is the vector of generalized coordinates, τ_1 and τ_2 are the input torques, m_1, m_2, m_3, m_4 are the masses of each of the links of the mechanism, I_1, I_2, I_3, I_4 are the moments of inertia at the center of mass of each of the links, and g is the acceleration of gravity. And the other parameters have been described above.

If the next terms of mathematical model,

$$m_3L_2l_{c3} = m_4L_1l_{c4}, \quad (7)$$

the inertial matrix is diagonal and constant. If this condition is satisfied, the dynamic model of the manipulator can be simplified and this is given by a set of decoupled equations without centrifugal terms as the next form,

$$d_{11}\ddot{\varphi}_1 + \phi_1(\varphi_1) = \tau_1, \quad (8)$$

$$d_{22}\ddot{\varphi}_2 + \phi_2(\varphi_2) = \tau_2. \quad (9)$$

Due to these considerations, the parallelogram linkage system has been adopted by industrial robots and can be considered very popular in this context. In fact, the angles φ_1, φ_2 can be adjusted in independent form. Considering that the gravity terms of the dynamic model have been eliminated in mechanical way by means of a force-balancing, it is possible to use a proportional-derivative (PD) controller, considering not exist applied loads to mechanism and it is friction free. On the other hand, if the gravity terms are not eliminated, a PID controller can be considered which can be implemented for minimizing the steady state error that is generated by these terms.

In this case study is considered a classic PID controller for each input torque, for the motion control of mechanism, given by,

$$u_i(t) = k_i^P e_i(t) + k_i^I \int_0^t e_i(\tau) d\tau + k_i^D \dot{e}_i(t); \quad i = 1, 2. \quad (10)$$

$$e_i(t) = \varphi_i^* - \varphi_i,$$

where, $u_i(t)$ with $i = 1, 2$, are the input torque, τ_1 and τ_2 , respectively; $e_i(t)$ and $\dot{e}_i(t)$ with $i = 1, 2$, are the position and velocity errors of the link 1 and link 2 (desired φ_i^* and actual φ_i); k_i^P , k_i^I and k_i^D , are the controller gains, proportional, integral, and derivative, respectively. In Qu and Dorsey (1991), Cervantes and Alvarez-Ramirez (2001) can see proof of stability of PID controllers for trajectory tracking of robot manipulators, and have been widely applied in industrial robots.

5. Optimization problem

As it has been mentioned before, the design process for both the geometric and controller parameters ought to be performed in a simultaneously way. Thus, the design process can be expressed as an optimization problem as follows:

5.1. Problem definition

The design variables vector $\mathbf{X} \in \mathbb{R}^{22}$ is given by,

$$\mathbf{X} = [m_i, L_i, l_{ci}, I_i, k_1^P, k_2^P, k_1^I, k_2^I, k_1^D, k_2^D]^T; \quad i = 1, 2, 3, 4. \quad (11)$$

This variables are kinematic and dynamic parameters of the parallelogram linkage, and the controller gains. Table 1 shown the values of the parameters of the parallelogram linkage that were employed by practical issues of academic type.

The following constraints are considered:

$$L_{i,\min} \leq L_i \leq L_{i,\max} \quad (12)$$

$$L_1 = L_3 \quad (13)$$

$$m_3 L_2 l_{c3} = m_4 L_1 l_{c4} \quad (14)$$

The first three are the search limits, hence we only draw random values inside such limits. The last one is a model simplification considered in the simulation.

The objective function is the following:

$$\begin{aligned} \max f(m, L, l_c, I, k) = W - & \left[\lambda (E(|\tau_1, k|) + E(|\tau_2, k|)) \right. \\ & \left. + E\left(\left|\int e_1(t) dt\right|\right) + E\left(\left|\int e_2(t) dt\right|\right) \right]. \end{aligned} \quad (15)$$

where $E(x) = 1/T \cdot \int_0^T x dt$. And $\lambda = 1/10$ is a scaling factor. $W = 50$ is a translation parameter. Notice that the BUMDA is a maximization algorithm, hence the function to minimize is multiplied by -1 and translated in order to a turn it into a maximization problem with an objective function that only returns positive values.

5.2. Discussion about the optimization process

The kind of algorithm used in this paper is also known as blackbox optimization. This kind of algorithms do not require information about a mathematical expression of the objective function. They only used an evaluator (or blackbox) from which they obtain evaluations of candidate solutions. The advantages are that a simulator, or numerical solver can be an evaluator (also a physical experimental test). Blackbox

Table 1. Kinematic and dynamic parameters of the parallelogram linkage system.

$m_1 = 2$ [kg]	$m_2 = 0.5$ [kg]	$m_3 = 2$ [kg]	$m_4 = 2$ [kg]
$L_1 = 0.3$ [m]	$L_2 = 0.15$ [m]	$L_3 = 0.3$ [m]	$L_4 = 0.4$ [m]
$l_{c1} = 0.15$ [m]	$l_{c1} = 0.075$ [m]	$l_{c3} = 0.15$ [m]	$l_{c4} = 0.075$ [m]
$I_1 = 0.0226$ [kg.m ²]	$I_2 = 0.00641$ [kg.m ²]	$I_3 = 0.0226$ [kg.m ²]	$I_4 = 0.0226$ [kg.m ²]

optimization algorithms do not assume convexity or derivability of the objective function. Notice that proving convexity of such function is not straight forward Eq. (15), numerically the objective function seems not to be convex neither derivable in the whole domain, for example, consider a mechanism with unconnected elements, such candidate solutions could be sampled by the BUMDA, such mechanism represents a discontinuity of the objective function. In such case, we assign a very low value to the objective function. Hence, classical optimization algorithms such as gradient based ones, can not be applied in this case. Equation (15) minimizes the power (τ_1 and τ_2) used for manipulating the mechanism (which is related with the proportional, integral, and derivative errors), as well as the integral of the error in the simulation interval.

The λ value is set to scale the τ term with the integral error term. It was set after revising numerical experiments. Notice that the objective function is basically composed of two terms: the τ_i term and the error term. As λ is introduced to scale the terms. Notice that only one scaling factor is needed, for example, consider the maximization problem as: $\max C_0 - (C_1(\tau \text{ term}) + C_2(\text{error term}))$, it has the same maximum than $\max C_0/C_2 - (C_1/C_2(\tau \text{ term}) + (\text{error term}))$. In our problem $C_0/C_2 = 50$ and $C_1/C_2 = \lambda$.

When a mechanism presents any singularity, or the system of equations can not be numerically solved, it is set to 0 (a low value), such solutions are automatically disregarded by the truncation procedure of the BUMDA.

6. Results

In this section we test *three* different paths in order to show the robustness of the algorithm. The paths are listed in Table 2. The simulation time in the optimization process is of 20 sec and the step size for the

Table 2. Paths considered for testing the proposal.

$X_1(t) = \delta x + R \cos(\omega t),$	Circle path
$Y_1(t) = \delta y + R \sin(\omega t),$	$\delta x = 0.4, \delta y = 0.3, \omega = 1, R = 0.05$
$X_2(t) = \delta x + a \cos(3\omega t) \cos(\omega t),$	Three petal flower
$Y_2(t) = \delta y + a \cos(3\omega t) \sin(\omega t),$	$\delta x = 0.4, \delta y = 0.3, \omega = 1, a = 0.05$
$X_3(t) = \delta x + (b - a \cos(\omega t)) \cos(\omega t),$	Snail Pascal
$Y_3(t) = \delta y + (b - a \cos(\omega t)) \sin(\omega t),$	$\delta x = 0.47, \delta y = 0.3, a = 0.06, b = 0.02$

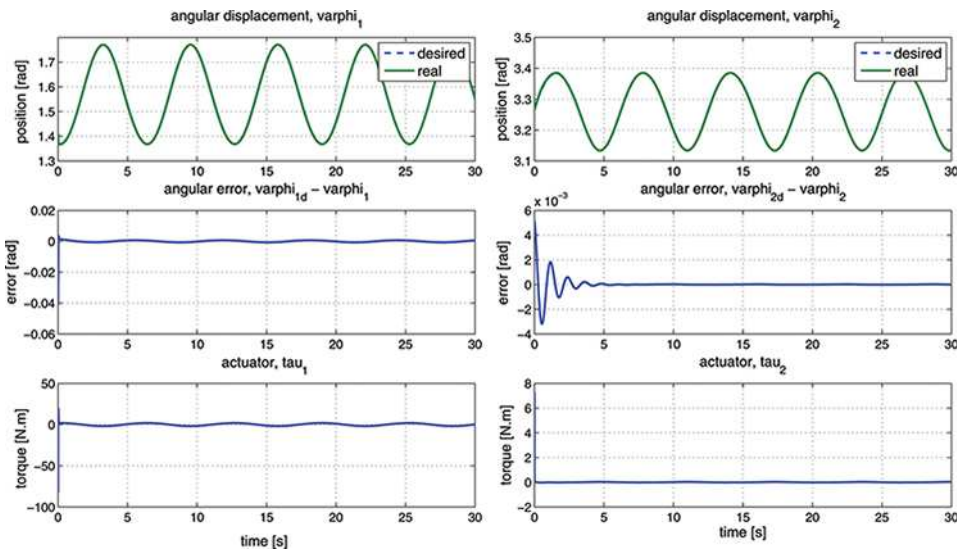


Figure 5. Circle path: Desired φ_{id} versus computed φ_i , angular error $\varphi_{id} - \varphi_i$, torque τ_1, τ_2 .

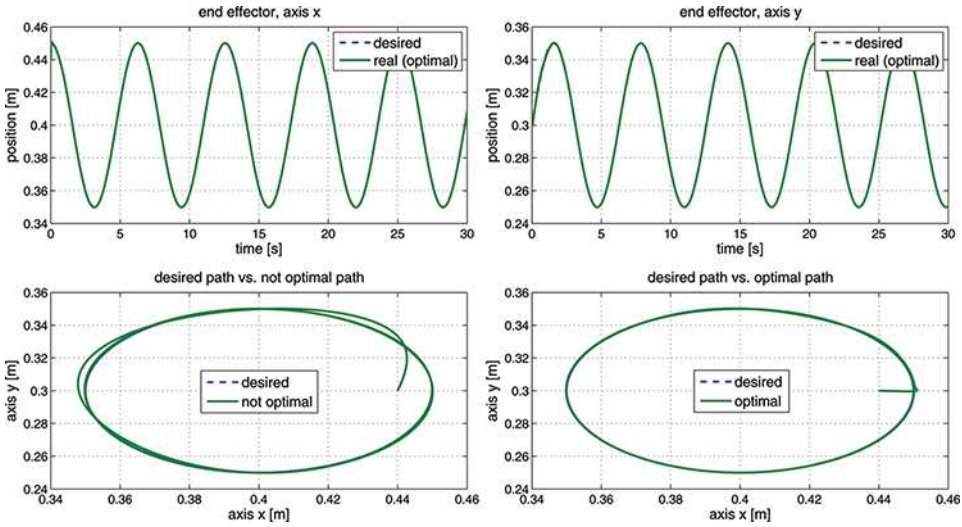


Figure 6. Circle path: Desired *versus* computed path, end effector.

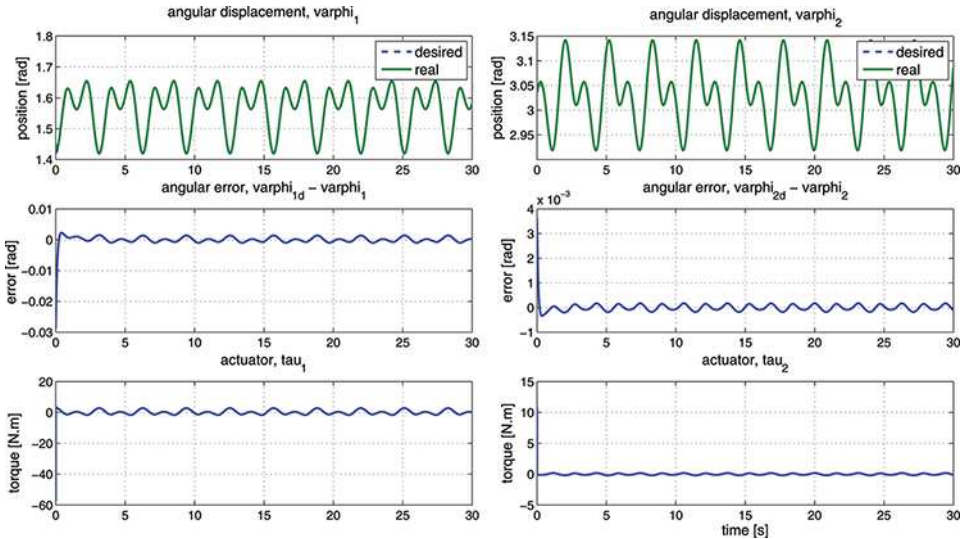


Figure 7. Path of a three petal flower: Desired φ_{id} *versus* computed φ_i , angular error $\varphi_{id} - \varphi_i$, torque τ_1, τ_2 .

Runge–Kutta method is of 0.001 sec. The optimization algorithm uses a sample size of 73. The stopping criterion is when the sum of the standard deviations of all variables reaches a value less than $1.0 \times E^{-4}$. In the optimization process all the variables were considered in the interval $[0, 1]$, then the values are mapped to the corresponding intervals simply by scaling and translating them. The execution of the algorithm is about 1 min 56 sec. In a personal computer with Intel i7 processor and 8GB of RAM, using OpenSuse Linux and GNU compiler (GCC) and C program. The number of evaluations of the objective function in a typical program execution is about 6745.

In Figs. 5, 6, 7, 8, 9, and 10 shows the error of φ_1 and φ_2 , the τ_1 and τ_2 values, and desired path *versus* the delivered by the BUMDA, for a circle path, a three petal flower and snail Pascal, respectively.

6.1. Improvement of the initial design

In Fig. 11(a) we can observe the evolution of the best and worst solution as well as the mean of the objective value per iteration of the BUMDA. Hence, it shows the improvement of the solutions thought

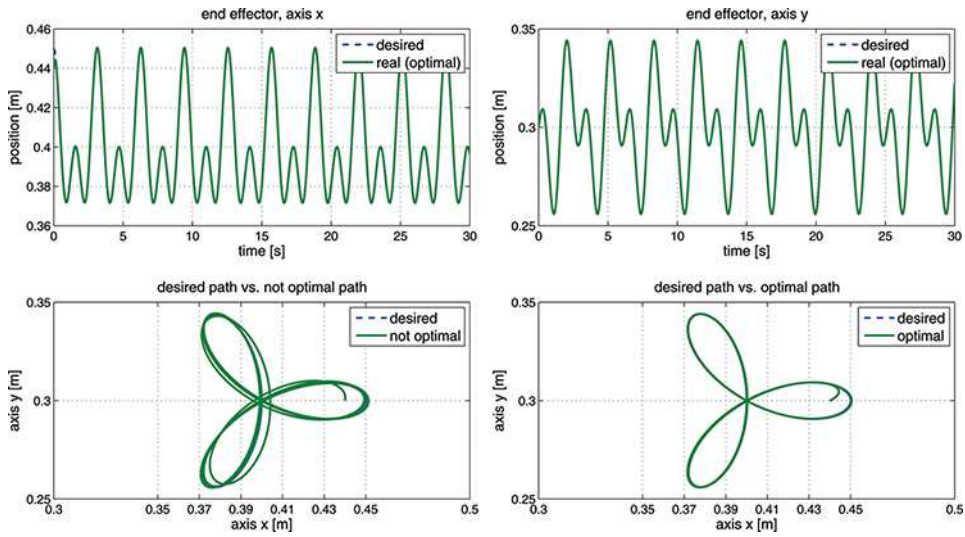


Figure 8. Path of a three petal flower: Desired versus computed path, end effector.

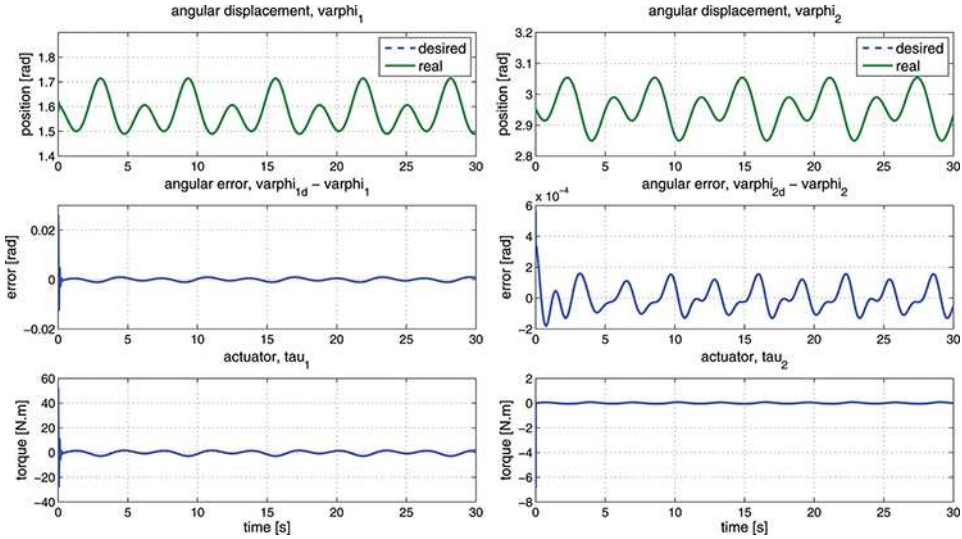


Figure 9. Snail Pascal path: Desired φ_{id} versus computed φ_i , angular error $\varphi_{id} - \varphi_i$, torque τ_1 , τ_2 .

the iterations. In Fig. 11(b) we can observe the improvement for the last iterations. We divide in this two phases the algorithm behavior, because in the first iterations BUMDA finds an approximation of the global solutions, it is to say, it explores the whole search space, in the second phase the solutions is almost the final design but it refines the design, even though the improvement can be consider marginal, it shows the capacity of the algorithm to evolve solutions with a high precision, the figures were computed using Circle Path, but a similar behavior is presented by the other.

Table 3 shows the best solutions known at the first iteration of the BUMDA, and at the end of the optimization process, notice that nevertheless the objective function is not quite different, the initial and final design is completely different.

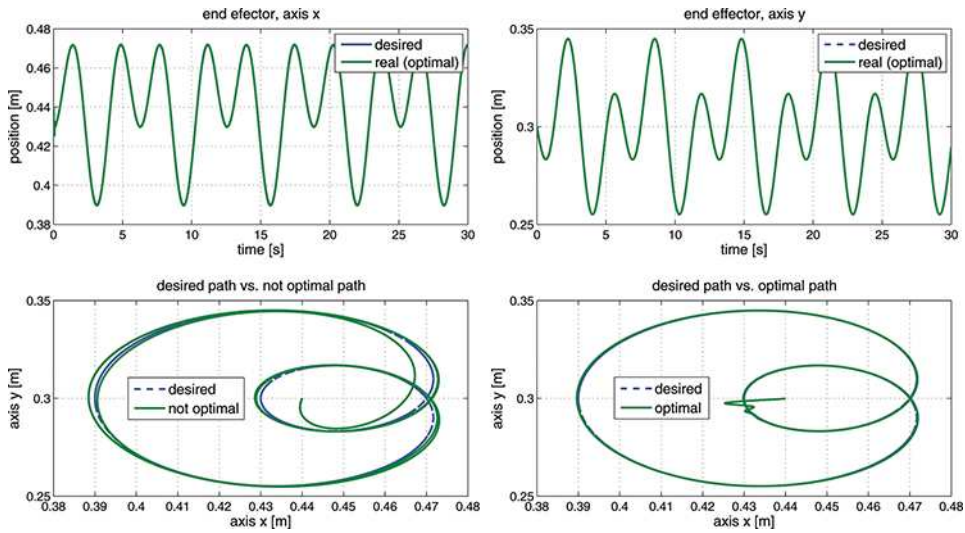


Figure 10. Snail Pascal path: Desired versus computed path, end effector.

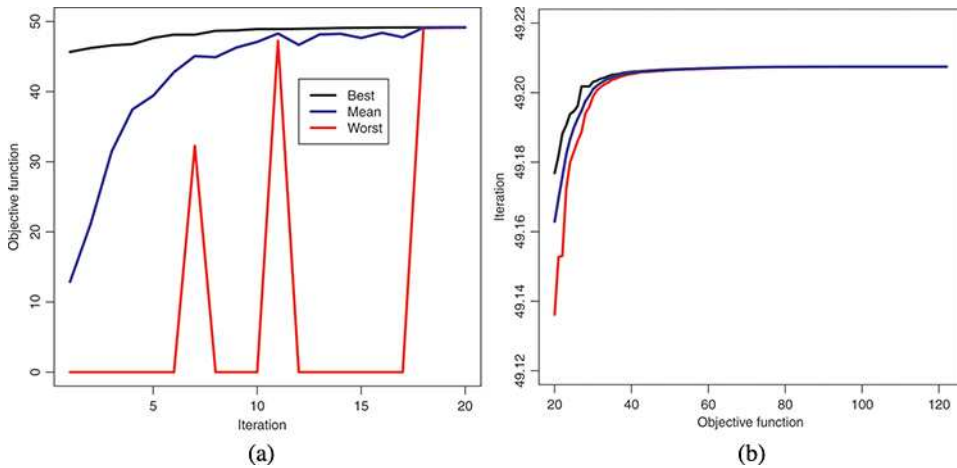


Figure 11. (a) Evolution of the design in the first 20 iterations. Notice that a significant improvement is achieved in this phase. (b) Improvement of the design for the last iterations. Notice that the design can be refined with a high precision.

Table 3. Initial and final designs for a typical run, for each of the paths. The values are taken as the best solution known in the first generation and the best solution at the end of the optimization process.

Path	L_1	L_2	L_4	k_1^P	k_2^P	k_1^I	k_2^I	k_1^D	k_2^D	f_{obj}
Circle Initial	0.9	0.0888	0.614	1177.7	1912.3	1995.5	1986.4	1215.4	741.89	38.798
Circle Final	0.583	0.148	0.496	2000	150.15	2000	2000	41.972	11.646	48.842
Three Petal Ini.	0.202	0.199	0.37	1208.6	203.94	1969.2	1834.4	28.576	44.614	48.975
Three Petal Final	0.286	0.182	0.401	2000	788.27	2000	1982.6	32.432	64.44	49.397
Snail Pascal Ini.	0.568	0.159	0.451	373.21	1768.8	919.46	559.11	420.98	662.06	45.509
Snail Pascal Final	0.638	0.155	0.528	2000	1810.1	2000	1951	52.778	476.67	49.226

7. Conclusion

In this proposal, a structure-control design for a parallelogram linkage robot with trajectory tracking is presented. This is formulated as a concurrent process where both the kinematic and the dynamic models of the mechanical structure and the controller gains are considered for optimizing in a simultaneously way. An estimation of distribution algorithm was adopted for solving the optimization problem. This

algorithm successfully found well performed solutions. One of the most remarkable characteristics of this algorithm is that practically it uses a unique parameter: the sample size, because the stopping criterion is set according to the maximum allowed numerical error, which is a problem parameter. The sample size is quite small for this kind of algorithms.

The BUMDA performs quite well and we expect to tackle similar problems with this algorithm. The BUMDA samples the whole search space using an approximation of the Boltzmann distribution, in such distribution the probability of sampling the global optimum is the maximum. Hence, we look for the global optimum, and it is quite possible that we find a near approximation, and on the other hand, when the algorithm converges, this approximation is refined by locally sampling in a small neighborhood. Notice that the BUMDA uses a Gaussian model, which is a soft distribution, when this Gaussian model has a standard deviation less than ϵ , most of the samples from it will be in a radius of 3ϵ from the mean, hence if ϵ is a small number the Gaussian distribution provides a way of looking for solutions in a close neighborhood, even this is not a prove of local optimality, it is a strong argument about our procedure, which samples intensively the neighborhood of the best solution when the algorithm converges. In consequence we have a strong argument to say that very possibly the solutions delivered by the BUMDA are local optima. On the other hand, when the algorithm is not in the convergence phase (when it has a large variance) it is performing a global exploration. Then it is possible that BUMDA finds the global optimum, but it is highly possible that at least it provides a local optimum.

The experiments are developed using different kinds of paths of the parallelogram linkage robot, in all cases the algorithm is capable of solving satisfactorily all the cases. It can be observed that structural and controller parameters of the mechanism fulfill desired motions. In a subsequent work, other parallel mechanisms will be studied, even those with more complex architecture where optimization parameters are uncorrelated, in order to test the performance of the concurrent optimization methodology.

References

- Affi, Z., EL-Kribi, B., Romdhane, L. (2007). Advanced mechatronic design using a multi-objective genetic algorithm optimization of a motor-driven four-bar system. *Mechatronics* 17(9):489–500.
- Cervantes, I., Alvarez-Ramirez, J. (2001). On the pid tracking control of robot manipulators. *Systems and Control Letters* 42:37–46.
- da Silva, M., Brils, O., Desmet, W., Van Brussel, H. (2009). Integrated structure and control design for mechatronic systems with configuration-dependent dynamics. *Mechatronics* 19(6):1016–1025.
- Delchev, K., Zahariev, E. (2008). Computer simulation-based synthesis of learning-control law of robots. *Mechanics Based Design of Structures and Machines* 36(3):225–248.
- EL-Kribi, B., Houidi, A., Affi, Z., Romdhane, L. (2013). Application of multi-objective genetic algorithms to the mechatronic design of a four bar system with continuous and discrete variables. *Mechanism and Machine Theory* 61:68–83.
- Li, Q. (2006). Experimental validation on the integrated design and control of a parallel robot. *Robotica* 24(2):173–181.
- Li, Q., Zhang, W., Chen, L. (2001). Design for control-a concurrent engineering approach for mechatronic systems design. *Mechatronics, IEEE/ASME Transactions on* 6(2):161–169.
- Liu, D., Li, G., Shang, Y. (2012). Dynamic modeling and multiobjective optimization of forging manipulator during one forging stroke. *Mechanics Based Design of Structures and Machines* 40(2):241–252.
- Mühlenbein, H., Mahnig, T., Ochoa-Rodriguez, A. (1999). Schemata, distributions and graphical models in evolutionary optimization. *Journal of Heuristics* 5(2):215–247.
- Mühlenbein, H., Paaß, G. (1996). From recombination of genes to the estimation of distributions i. binary parameters. In: Voigt, H.-M., Ebeling, W., Rechenberg, I., Schwefel, H.-P., eds. *Parallel Problem Solving from Nature PPSN IV*, Vol. 1141 of *Lecture Notes in Computer Science*. Heidelberg: Springer Berlin, 178–187.
- Pil, A., Asada, H. (1996). Integrated structure/control design of mechatronic systems using a recursive experimental optimization method. *Mechatronics, IEEE/ASME Transactions on* 1(3):191–203.
- Portilla-Flores, E. A., Mezura-Montes, E., Alvarez-Gallegos, J., Coello-Coello, C. A., Cruz-Villar, C. A., Villarreal-Cervantes, M. G. (2011). Parametric reconfiguration improvement in non-iterative concurrent mechatronic design using an evolutionary-based approach. *Engineering Applications of Artificial Intelligence* 24(5):757–771.
- Qu, Z., Dorsey, J. (1991). Robust pid control of robots. *International Journal of Robotics and Automation* 6(4):228–235.
- Ravichandran, T., Wang, D., Heppler, G. (2006). Simultaneous plant-controller design optimization of a two-link planar manipulator. *Mechatronics* 16(3–4):233–242.
- Rivin, E. I. (1987). *Mechanical Design of Robots*. New York: McGraw-Hill, Inc.

- Sciavicco, L., Siciliano, B. (2000). *Modelling and Control of Robot Manipulators*. New York: Springer.
- Spong, M. W., Vidyasagar, M. (2008). *Robot Dynamics and Control*. New Jersey: John Wiley & Sons.
- Valdez, S. I., Hernández, A., Botello, S. (2013). A boltzmann based estimation of distribution algorithm. *Information Sciences* 236:126–137.
- Valles, M., Diaz-Rodriguez, M., Valera, A., Mata, V., Page, A. (2012). Mechatronic development and dynamic control of a 3-DOF parallel manipulator. *Mechanics Based Design of Structures and Machines* 40(4):434–452.
- van Amerongen, J. (2003). Mechatronic design. *Mechatronics* 13(1):1045–1066.
- Villarreal-Cervantes, M., Cruz-Villar, C., Alvarez-Gallegos, J., Portilla-Flores, E. (2013). Robust structure-control design approach for mechatronic systems. *IEEE/ASME Transactions on Mechatronics* 18(5):1592–1601.
- Villarreal-Cervantes, M., De la Cruz-Mucio, D., Portilla-Flores, E. (2014). Optimum design of parallelogram five-bar manipulator for dexterous workspace by using ELEMAEF in differential evolution. *Applied Mathematics and Information Sciences* 8(5):2129–2140.
- Youcef-Toumi, K. (1996). Modeling, design, control integration: A necessary step in mechatronics. *Mechatronics, IEEE/ASME Transactions on* 1(1):29–38.
- Zhang, Q., Mühlenbein, H. (2004). On the Convergence of a Class of Estimation of Distribution Algorithms. *IEEE Transactions on Evolutionary Computation* 8(2):127–136.