Latin American Journal of Solids and Structures
www.lajss.org

Multi-objective optimization of a parallel manipulator for the design of a prosthetic arm using genetic algorithms

Abstract
This paper presents a synthesis of a spherical parallel manipulator for a shoulder of a seven-degrees-of-freedom prosthetic human arm using a multi-objective optimization. Three design objectives are considered, namely the workspace, the dexterity, and the actuators torques. The parallel manipulator is modelled considering 13 design parameters in an optimization procedure. Due to the non-linearity of the design problem, genetic algorithms are implemented. The outcomes show that a suitable performance of the manipulator is achieved using the proposed optimization.

Keywords
Prosthetic arm, Biomechanics, Multi-objective optimization, Genetic algorithms.

José-Alfredo Leal-Naranjo[a,b*]
Marco Ceccarelli[b]
Christopher-René Torres-San-Miguel[a]
Luis-Antonio Aguilar-Perez[a]
Guillermo Urriolagoitia-Sosa[a]
Guillermo Urriolagoitia-Calderón[a]

[a]Instituto Politécnico Nacional, Sección de estudios de posgrado de la Escuela Superior de Ingeniería Mecánica y Eléctrica Unidad Zacatenco, Mexico City, Mexico. E-mails: lealnaranjo@gmail.com, ctorress@ipn.mx, laguilarpe@gmail.com, guiurri@hotmail.com, urrio332@hotmail.com

[b]LARM: Laboratorio di Robotica e Meccatronica, DiCEM, Universitá di Cassino e del Lazio Meridionale, Cassino, Italy. E-mails: ceccarelli@unicas.it

*Corresponding author

http://dx.doi.org/10.1590/1679-78254044

Received: May 19, 2017
In Revised Form: November 26, 2017
Accepted: November 26, 2017
Available online: February 05, 2018

1 INTRODUCTION

The success of a prosthetic human arm design can be evaluated mainly in terms of the functionality of the design and its cost. For these reasons the design of the mechanism that performs the movement of the prosthesis is an issue that impacts greatly in the final product. Upper limb disarticulation is one of the cases with less incidences, therefore the people with this disability have fewer options of prostheses available (Dillingham et al., 2002). One of the most advanced prosthetic arm can be found in (Johannes et al., 2011) that is a prosthetic arm with 2 DOF at the shoulder, a humeral rotator, elbow and three DOF at wrist. This device can be used for different levels of amputation. In (Resnik et al., 2014) the Deka arm is presented, which is a prosthetic arm with 6 DOF at the arm and 4 at hand and a weight of 4.45 kg. He et al. (2014) presented a prosthetic arm with 7 DOF that uses underactuated mechanisms and weighs 4.45 kg. Most of these solutions are based on serial kinematic chain configurations. In this work, a parallel configuration is addressed for a novel design.

Main advantages of parallel manipulators are that the payload is shared between the actuators, the actuators are located in the base so that the inertia is reduced and parallel manipulators exhibit stiff behavior. Disadvantages of the parallel manipulators are the small workspace and limited dexterity as compared to the serial manipulators (Ceccarelli, 2004).

For the design of a prosthetic arm, design criteria can be considered in workspace, stiffness, and dexterity among others that are used also in design procedures of manipulators.

Many authors have investigated the optimization of mechanisms using different techniques. In (Boudreau and Turkkan, 1996) the forward kinematics of three different planar parallel manipulators are solved using a genetic algorithm (GA). The optimization of two different parallel planar manipulators is achieved as well using a genetic

(cc) BY

algorithm in (Boudrear and Gosselin, 1999) where the objective function is formulated to get a workspace as close as possible to a predefined one. In Cabrera et al. (2002) the path synthesis of a four-bar mechanism is worked out using genetic algorithms. Ceccarelli and Lanni (2004) worked out the optimization of the workspace and the dimensions of general three revolute joints manipulator using sequential quadratic programming. In (Hernandez et al., 2015) the optimization of a cable-based parallel manipulator is worked out using evolutionary algorithms. The workspace of a spherical parallel mechanism for laparoscopic surgery is optimized using a genetic algorithm in (Li and Payandeh, 2002). In (Castejón et al., 2010) a multi-objective optimization was performed in order to design a robotic arm for service tasks. In Zhen et al. (2010) the optimization of the stiffness and dexterity of a six DOFs parallel manipulator is presented using genetic algorithms and neural network. Chaker et al. (2012) performed the synthesis of a spherical manipulator for surgeries by optimizing the workspace and dexterity of the mechanism using GA. In (Essomba et al., 2016) it is presented the synthesis of a spherical manipulator used as a probe holder considering a multi-objective optimization and GA. The synthesis of a 3-RRR spherical parallel manipulator is worked out in (Wu, 2012) using a multi-objective optimization considering as objective function the dynamic dexterity and the isotropy. In Ramana et al. (2016) it is worked out the optimization of a 3 DOF parallel manipulator using as objectives the dexterity, work space and stiffness.

An arising problem in the design of a parallel mechanism comes from the fact that their performance is highly dependent on their geometric parameters and configurations as well as the different performance measures (workspace, dexterity, etc) are mutually dependent (Wu, 2012). In this work, the synthesis of a parallel spherical manipulator is carried out using a multi-objective optimization combined with GA. The proposed spherical manipulator is used as a mechanism to replicate the shoulder movement in a prosthetic human arm. The shoulder of the prosthetic arm is analyzed as one of the most complex part of the prosthetic device since it requires a great mobility and volume restrictions. Three objective functions are taken in to account in the proposed optimization. For the first objective, the specified trajectory must be inside the available workspace; the second objective is the maximization of the mechanism accuracy (measured by the dexterity); while the third objective is the minimization of the torques (measured here by the maximum values) of the three actuators. This is done since a reduction in the torque could yield to a reduction in the actuators and thus reduction in weight and power consumption. With the obtained results, a parallel manipulator is designed and a set of dynamic simulations are performed to validate the results of the optimization and to characterize the design solution.

2 Genetic algorithms and multi-objective optimization

Genetic algorithms (GA), developed in Holland (1992), are heuristic methods that consist in optimization procedures as inspired in natural evolution. In GA, an initial random population needs to be created. The characteristics of each solution are used in equivalent chromosomes. Each solution is evaluated and classified according to how well it satisfies the objective function and then it is assigned a probability of reproduction. The fittest individuals are more likely to be reproduced (selection), and thus they inherit those characteristics. The combination (crossover) of the parent genes yields to a consecutive generation (replacement). Mutation can occur in the chromosomes of some individuals. It is projected that some individuals of this new generation will have inherited the best characteristics of their parents and will be a better solution to the problem. This new population goes through the same process and this cycle is repeated until all the members share the same genetic information. This last generation is the best solution to the optimization problem (Fogel, 1994). A genetic algorithm is stopped when the population converges to an optimal solution or when a maximum number of generations is reached, Figure 1.

Genetic algorithms require objective and fitness functions. The objective function defines the optimal condition and the fitness function assesses how well a specific solution satisfies the objective function and assigns a real value to that solution (Coello et al., 2007).

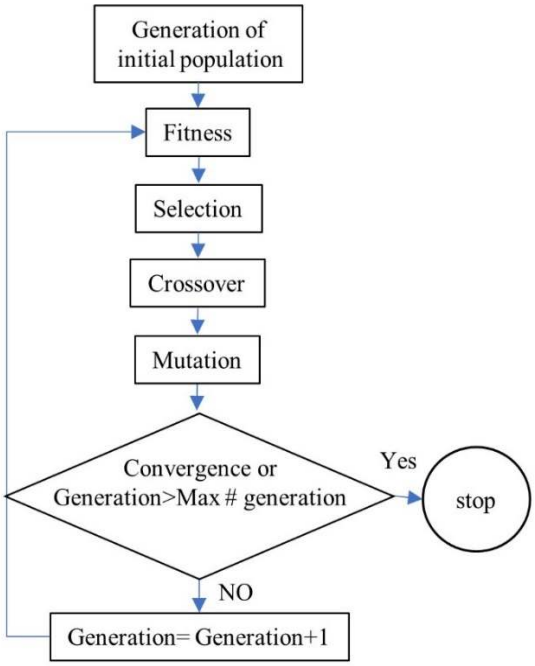


Figure 1: *A flow chart of a genetic algorithm optimization.*

Formerly, the information of the individuals was encoded in bit strings called Binary-coded, but nowadays the individuals are coded using real numbers (Deb and Kumar, 1995). Real-coded GA usually achieves better results than Binary-coded GA (Davis, 1991) .

The selection process consists in taking two parents to create offspring. The objective is to provide to the fitter individuals a greater chance of reproduction expecting that their offspring will have higher fitness. Typical types of selection scheme are the proportionate selection and the ordinal-based selection. In the proportionate based selection, the individuals are picked up based on their fitness values that are relative to the fitness of the other individuals. The ordinal-based selection selects individuals by considering their rank within the population (Sivanandam and Deepa, 2007).

The crossover combines two different individuals to generate new offspring. Many crossover methods exist, and the most common is single-point crossover (Blum et al., 2012). In this work, a linear crossover is used. Two parents are selected as p_1 and p_2 and three offsprings are generated as $0.5\ p_1 + 0.5p_2$, $1.5\ p_1 - 0.5p_2$, $-0.5\ p_1 + 1.5p_2$ respectively (Herrera et al., 1998). Then, the three offspring are evaluated and the best two are selected for the next generation.

Mutation consists in a random modification of a gene during reproduction (Cabrera et al., 2002). In general, the mutation prevents convergence to a local optimum (Li and Payandeh, 2002). The mutation operator is applied with a small probability (Wang et al., 2004). In this work, a non-uniform mutation is used (Michalewicz, 1996).

Multi-objective optimization commonly involves multiple conflicting objectives that must be considered simultaneously and there is a set of mathematically equally good solutions (Miettinen, 2008). These set of solutions are known as nondominated or Pareto optimal set. The set of solutions forms the Pareto optimal front. The characteristic of a Pareto optimal solution is that any change in the values of the solution will not improve any of the objective functions.

The concept of dominance or domination is used in most of the optimization algorithms. If there are Nobjective functions, a solution $x^{(1)}$ dominates another solution $x^{(2)}$ if the two following conditions are true (Deb, 2014):

1. The solution $x^{(1)}$ is no worse than $x^{(2)}$ in all objectives.
2. The solution $x^{(1)}$ is better than $x^{(2)}$ in at least one objective.

A solution $x^{(1)}$ is Pareto-optimal if is nondominated with respect to the search space (or feasible region)(Coello, 2015).

Many evolutionary algorithms have been proposed over the years to solve multi-objective optimization problems. In this work, it is used a controlled elitist non-dominated sorting GA (CENSGA)(Deb and Goel, 2001). The algorithm of CENSGA is based in NSGA-II (Deb et al., 2002), but there is a difference in the selection approach that improves the convergence to the Pareto front.

The CENSGA algorithm works as follows. First, a random P_t population of size N is formed. From this population, Q offsprings are created using the common GA operators (crossover, mutation). Then, a combined R_t population ($P_t \cup Q_t$) of size 2N is formed. The population R_t is sorted according to non-domination and different non-dominated fronts are created. The first non-dominated front is formed only by elements non-dominated with respect to R_t. The second non-dominated front is formed by elements dominated by just one solution. All the elements of R_t are assigned a front in a similar fashion (Elarbi et al., 2017). The crowding distance is calculated for every element of R_t (Tan et al., 2006). Then N elements are selected from R_t to form the next generation P_{t+1}. The maximum number of elements selected from i-th non-dominated front is defined by a geometric distribution

$$n_i = N \frac{1-r}{1-r^K} r^{i-1} \tag{1}$$

where r is the reduction ratio (r<1) and K is the number of non-dominated fronts.

n_i denotes the maximum allowable number of individuals taken from the i-th front. If n_i is larger than the number of elements in the i-th front, then all the elements of this front are chosen and the remaining slots are added to n_{i+1}. But if n_i is smaller than the number of elements in the front, n_i the elements are chosen using a crowded binary tournament selection. The crowded binary tournament takes two elements and returns the one with the bigger crowding distance. GA operators are applied to the new population of P_{t+1} to form Q_{t+1} and are combined to form R_{t+1} and the described process is repeated for a specific number of generations. The result of this process is the Pareto front, this means a set of optimal solutions. CENSGA algorithm is summarized in Figure 2.

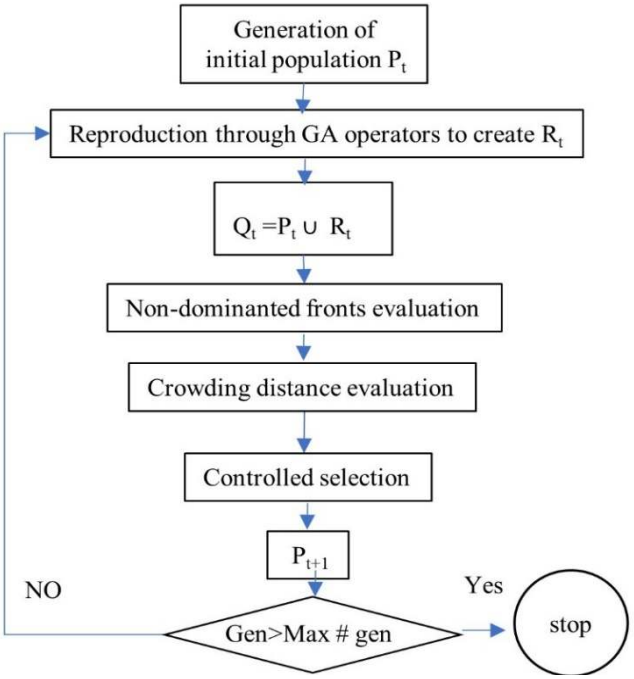


Figure 2: *A flow diagram of the CENSGA algorithm.*

3 Kinematic design and its analysis

The prosthetic device considered in this work is based in the design in (Leal-Naranjo et al., 2016), Figure 3(a). This device has seven degrees of freedom, with three DOFs that are in the shoulder, one in the elbow and three in the wrist. The shoulder of this prototype is driven by a 3DOF parallel spherical manipulator, Figure 3(b). The mechanism of the elbow is a 4bar mechanism. The wrist mechanism is also a parallel spherical manipulator. The weight of this prototype is 1250 g including the hand. The main issue of prosthetic arms is the functionality and weight of the device. An improvement in the prosthetic kinematic design could potentially yield in improvements in the functionality and also in the weight due to lower power requirements.

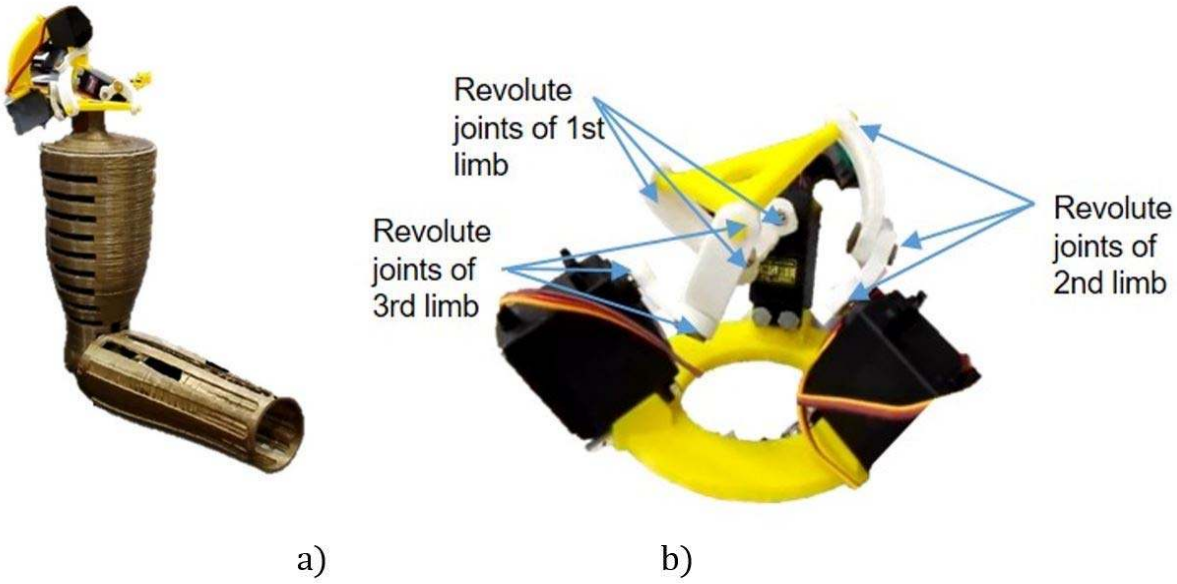


Figure 3: *3D printed prototype: a) a prosthetic human arm; b) shoulder mechanism.*

The shoulder of the prosthesis is modelled as a three DOFs spherical manipulator, Figure 4(a). The manipulator consists of three legs with three revolute joints each, whose axes converge at one point that is the center of rotation. The axes of these revolute joints are defined by the unit vectors $\mathbf{u}_i$, $\mathbf{v}_i$ and $\mathbf{w}_i$. The links a_i (attached to the lower platform) are characterized by the angle α_{1_i}, which represents the angle between the joints of the link. The links b_i (attached to the mobile platform) are characterized by the angle α_{2_i}, which represent the angle between the joints of the link, Figure 4 (b).

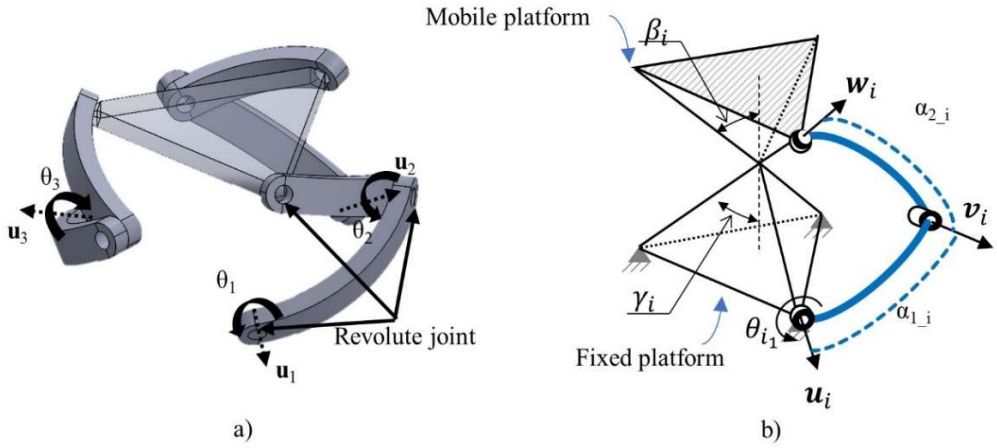


Figure 4:*A spherical parallel manipulator for the shoulder joint: a) a CAD design; b) a scheme with design parameters.*

The mobile and the fixed platforms are commonly triangular pyramids defined by β_i and γ_i, with an equilateral triangle as a base. Due to the shape of the base of the pyramids, the attachment points (vertexes of the base) are spaced every 120° but in this work instead of regular pyramids, the attachment points in the fixed and mobile platforms are considered as a parameter that can take any value. This means that the base is a scalene triangle and the attachment points are defined by the angle δ_i (i refers to the i-th leg; i=1,2,3) which is considered as another parameter of the mechanism, Figure 5. Angle δ_i is the angle between the Y axis and the axis of the attachment point.

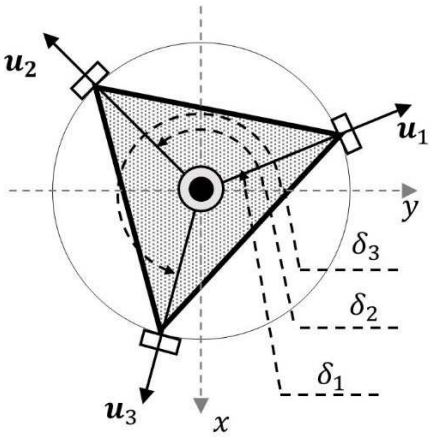


Figure 5: *Distribution of the attachment points at the fixed and mobile platforms.*

For this parallel manipulator, the inverse kinematic analysis is pretty straight forward in comparison to the direct kinematic. The inverse analysis is as follows. Vector $\mathbf{u}_i$ is defined as

$$u_i = R_z(\delta_i)R_x(\gamma - \pi/2)[0 \quad 1 \quad 0]^T \quad , i = 1,2,3 \tag{2}$$

where $R_a(b)$ represents a rotation of "b" degrees around the "a" axis.

Unit vector $\mathbf{v}_i$ is given by

$$\boldsymbol{v}_i = R_z(\delta_i)R_x\left(\gamma - \frac{\pi}{2}\right)R_y(\theta_i)R_z(\alpha_{1_i})\left[0 \quad 1 \quad 0\right]^T \tag{3}$$

unit vector $\mathbf{w}_i$ is a function of the orientation of the mobile platform, thus this vector is defined as

$$\boldsymbol{w}_i = \boldsymbol{Q}\,\boldsymbol{w}_i' \tag{4}$$

where $\mathbf{w}_i'$ is the initial orientation of the vectors when the manipulator is in its home configuration and is given by $\boldsymbol{w}_i' = R_{z(\delta_i)}\left[0 \quad \sin(\beta) \quad \cos(\beta)\right]^T$ and $\mathbf{Q}$ is the orientation matrix of the mobile platform.

For the second link this relation follows

$$\boldsymbol{v}_i \cdot \boldsymbol{w}_i = \cos(\alpha_{2_i}) \tag{5}$$

substituting $\mathbf{w}_i$ and $\mathbf{v}_i$ in Equation 5, and making algebraic simplifications it yields to

$$AX^2 + BX + C = 0 \tag{6}$$

Where

$$A = w_{i_x}(-a_1 + a_3) + w_{i_y}(a_4 - a_5) + w_{i_z}(a_7) - \cos(\alpha_{2_i}) \tag{7}$$

$$B = w_{i_x}(2a_2) + w_{i_y}(2a_6) + w_{i_z}(2a_8) \tag{8}$$

$$C = w_{i_x}(a_1 + a_3) + w_{i_y}(a_4 + a_5) + w_{i_z}(a_7) - \cos(\alpha_{2_i}) \tag{9}$$

$$X = \tan(\theta_i/2) \tag{10}$$

$$a_1 = -\sin(\alpha_{1_i})\cos(\delta_i) \tag{11}$$

$$a_2 = -\sin(\alpha_{1_i})\sin(\delta_i)\cos(\gamma_i) \tag{12}$$

$$a_3 = -\cos(\alpha_{1_i})\sin(\delta_i)\sin(\gamma_i) \tag{13}$$

$$a_4 = \cos\left(\alpha_{1_i}\right)\cos\left(\delta_i\right)\sin\left(\gamma_i\right) \tag{14}$$

$$a_5 = -\sin\left(\alpha_{1_i}\right)\sin\left(\delta_i\right) \tag{15}$$

$$a_6 = \sin\left(\alpha_{1_i}\right)\cos\left(\delta_i\right)\cos\left(\gamma_i\right) \tag{16}$$

$$a_7 = -\cos\left(\alpha_{1_i}\right)\cos\left(\gamma_i\right) \tag{17}$$

$$a_8 = \sin\left(\alpha_{1_i}\right)\sin\left(\gamma_i\right) \tag{18}$$

Equation 6 is a quadratic equation that provides the angle of the input links of the spherical manipulator. It can be seen from equation 6 that for each leg, two solutions exist. Thus, for every position in the workspace of this manipulator can be designed for eight possible assembly solutions, Figure 6.

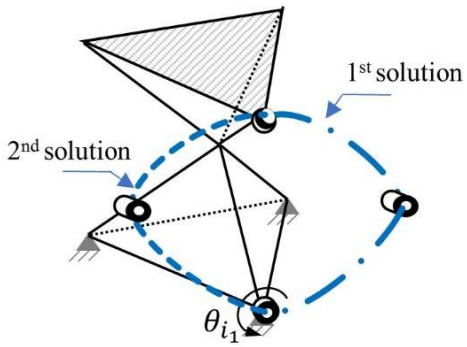


Figure 6: *Representation of the two possible solutions for each leg of the mechanism.*

4 Modelling and design optimization procedure

One of the aims of the optimization of this parallel manipulator is to maximize the workspace in a way that the prosthetic arm should be able to describe a humeral extension movement of 20 degrees and a humeral flexion movement of 90 degrees, Figure 7(a).

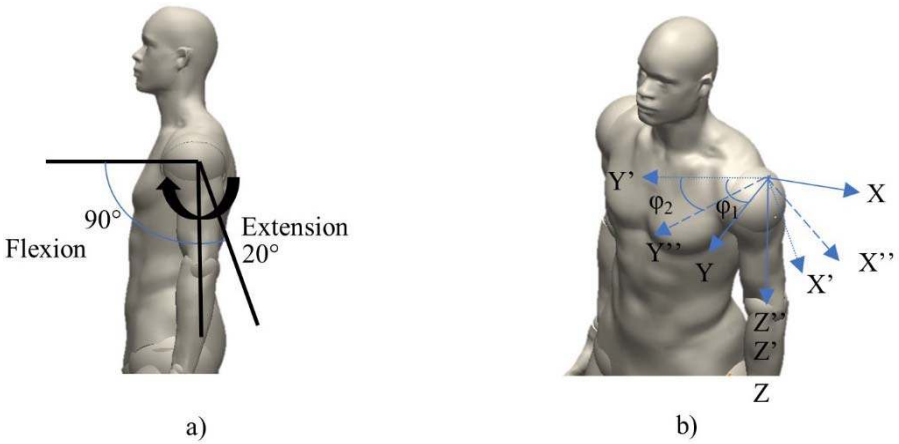


Figure 7*: Motion requirements: a) Flexion-Extension movement of the humerus; b) reference frames.*

To fulfill this requirement, it is necessary the manipulator will to be able to perform a rotation from -20° to 90° around the axis X", which is perpendicular to the sagittal plane of the body, Figure 7(b). The axis X", where the rotation occurs, not necessarily coincides with the home position of the manipulator. For this reason, it was considered that the manipulator is rotated an angle ϕ_1 around the Z axis from its reference position, and the manipulator

rotates (doing the flexion-extension) around the X" axis which is rotated from X' an angle ϕ_2, so the orientation of the end-effector is given by

$$\boldsymbol{Q} = R_Z\left(\varphi_1\right) R_{x''}\left(\varepsilon\right) \tag{19}$$

with

$$R_{x''} = \begin{bmatrix} k_x k_x\left(1-\cos\left(\varepsilon\right)\right)+\cos\left(\varepsilon\right) & k_x k_y\left(1-\cos\left(\varepsilon\right)\right) & k_y \sin\left(\varepsilon\right) \\ k_x k_y\left(1-\cos\left(\varepsilon\right)\right) & k_y k_y\left(1-\cos\left(\varepsilon\right)\right)+\cos\left(\varepsilon\right) & -k_x \sin\left(\varepsilon\right) \\ -k_y \sin\left(\varepsilon\right) & k_x \sin\left(\varepsilon\right) & \cos\left(\varepsilon\right) \end{bmatrix} \tag{20}$$

where $k_x = -\sin(\phi_2)$, $k_y = \cos(\phi_2)$ and ε is a variable with values from -20° to 90°.

The required movement describes an arc of 110°. This arc was discretized in segments of 5°. The objective function with the workspace can be defined as

$$g_1 = \sum h(\boldsymbol{Q}), h = \begin{cases} 1 if\ X\ in\ Eq.6\ is\ real \\ 0 if\ X\ in\ Eq.6\ is\ not\ real \end{cases} \tag{21}$$

where g_1 is evaluated in the mentioned arc. Thus, due to the adopted discretization, the maximum value for this function is 23. For simplicity, this function is multiplied by -1 and it is tried to be minimized.

The second objective of the optimization is related to the kinematic accuracy of the manipulator. The kinematic accuracy is associated with the conditioning number of the Jacobian matrix (also related to dexterity). The dexterity is defined as (Gosselin and Angeles, 1987)

$$k = \left\| -\boldsymbol{K}_q^{-1} \boldsymbol{J}_x \right\| \left\| -\boldsymbol{J}_x^{-1} \boldsymbol{K}_q \right\| \quad 1 \le k < \infty \tag{22}$$

with

$$\boldsymbol{J}_x = \begin{bmatrix} \left(\boldsymbol{v}_1 \times \boldsymbol{w}_1\right)^T \\ \left(\boldsymbol{v}_2 \times \boldsymbol{w}_2\right)^T \\ \left(\boldsymbol{v}_3 \times \boldsymbol{w}_3\right)^T \end{bmatrix} \tag{23}$$

$$\boldsymbol{K}_q = diag\left(\boldsymbol{v}_1 \times \boldsymbol{u}_1 \cdot \boldsymbol{w}_1 \quad \boldsymbol{v}_2 \times \boldsymbol{u}_2 \cdot \boldsymbol{w}_2 \quad \boldsymbol{v}_3 \times \boldsymbol{u}_3 \cdot \boldsymbol{w}_3\right) \tag{24}$$

where || || denotes the Euclidean norm of its matrix argument and × denotes the vector product (or cross product).

The value of k=1 corresponds to a configuration with good kinematic accuracy, and does not have an upper limit. The second objective function is expressed as

$$g_2 = k \tag{25}$$

The third objective is related to the torque that is required to perform the flexion-extension movement. Due to the low accelerations that are required during the movement and the low mass of the prosthetic device, the inertial effects can be ignored and a static analysis can be performed for each position in the defined trajectory with limited computational costs.

Using the virtual work approach, the end-effector output forces (forces and torques) is given by (Tsai, 1999)

$$F = J^T \tau \tag{26}$$

thus, the motor torques is calculated as

$$\tau = \left(J^T\right)^{-1} F \tag{27}$$

where J is the jacobian and its given by

$$\boldsymbol{J} = -\boldsymbol{K}_q^{-1} \boldsymbol{J}_x \tag{28}$$

and F are the torques due to the weight of the prosthetic device (15 N), the hand (5 N) and the load carried (5 N) and considering its corresponding lever arm.

By using Equation (27), the torque that is required for the actuators can be evaluated as the third objective function in the form

$$g_3 = MaxTorque \tag{29}$$

where Max Torque is the maximum value of the torque of the motors in the prescribed trajectory to be minimized.

Since the inverse kinematic of this manipulator it has eight different solutions (two different solutions for each leg), so it is necessary to stablish an equation to limit the mechanism to be constrained during the entire optimization process for the same branch. Furthermore, from Figure 4 it can be seen that in order to change from one branch to another it can only be possible if the links of a leg are aligned, thus the mechanism passes through a singular position. To constrain the mechanism in the same branch, the sign of the cosine between the plane formed by the unit vectors $\mathbf{v}_i$ and $\mathbf{u}_i$ and the unit vector $\mathbf{w}_i$ needs to remain constant. Such a condition can be expressed as

$$sign\left(\left(\boldsymbol{v}_i \times \boldsymbol{u}_i\right) \cdot \boldsymbol{w}_i\right) \tag{30}$$

In order to assure that the position of the motors is not overlapped, a restriction between δ_1, δ_2 and δ_3 are stablished as:

$$h_1 = \left|\delta_1 - \delta_2\right| \geq 40° \tag{31}$$

$$h_2 = \left|\delta_2 - \delta_3\right| \geq 40° \tag{32}$$

$$h_3 = \left|\delta_1 - \delta_3\right| \geq 40° \tag{33}$$

A minimum angle of 40 degrees between the attachment points was stablished in order to be possible to fit the actuators.

The design parameters are ϕ_1, ϕ_2, γ, β, α_{1_1}, α_{1_2}, α_{1_3}, α_{2_1}, α_{2_2}, α_{2_3}, δ_1, δ_2 and δ_3. The optimization problem is defined as

$$min \quad g_1,\ g_2,\ g_3 \tag{34}$$

$$subject\ to\ h_1 \geq 40,\ h_2 \geq 40,\ h_3 \geq 40 \tag{35}$$

CENSGA algorithm was applied using a random population of 100 individuals and a reduction ratio of 0.5. The maximum number of iterations was 200. For the genetic operators, it was used a linear crossover and a non-uniform mutation with a mutation probability of 0.5%.

5 Results

After finishing the optimization, the Pareto front for the eight-different assembly solutions was obtained. The function g_2 (Maximum torque of the motors) was plotted vs g_1(amplitude of the defined trajectory), Series 1 in Figure 8. g_2 vs g_3 (dexterity), Series 2 in Figure 8. Series 1 and 2 are in the same plot with two vertical axes in order to visualize the three objective values of a possible solution. For the g_2 vs g_3 plot, only solutions that completely satisfy the motion were plotted (g_1= 24) in order to reduce the data plotted and as the main interest is only in solutions that would satisfy the stablished trajectory. The scale in all the plots was the same in order to simplify the comparison. In Figure 8a the values of the objective functions for the solutions of the first possible assembly are plotted. It can be seen that for this configuration one the most suitable solution has an objective value of g_1=24, g_2=3.5 and g_3= 1.4. There are other solutions that require a lower torque but the value of the dexterity increases to g_1=24, g_2=3.3 and g_3= 2.9; or some other solutions with a better dexterity but the torque increases to g_1=24, g_2=3.9 and g_3= 1.2. Figure 8b shows objective functions for the second configuration. In this configuration, the most suitable solution has an objective value of g_1=24, g_2=5.9 and g_3= 2.5, so clearly is not a better one as compared with that for configuration 1.

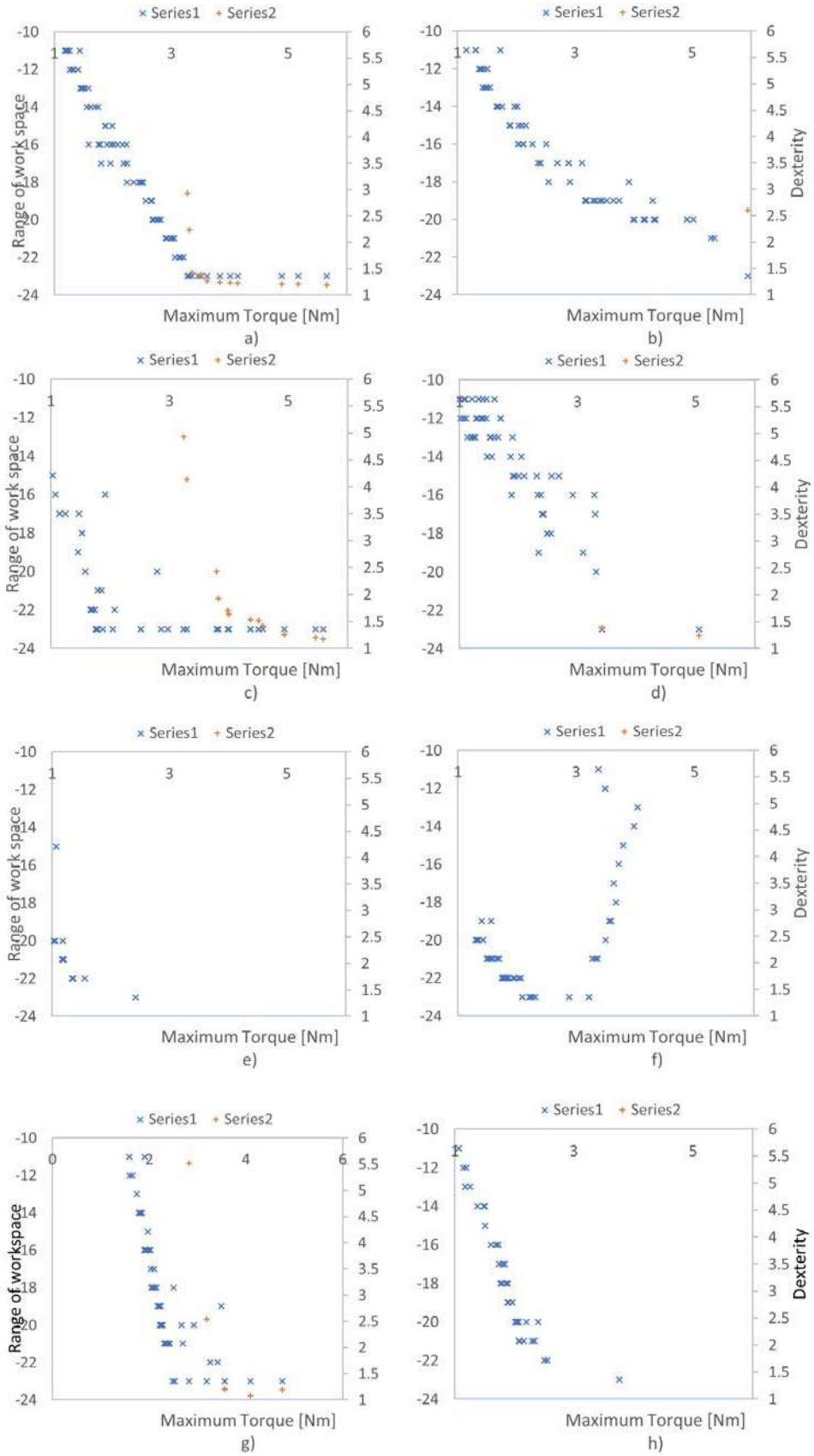


Figure 8: *Values of the three objective functions for the different solutions: a) Assembly solution 1; b) Assembly solution 2; c) Assembly solution 3; d) Assembly solution 4; e) Assembly solution 5; f) Assembly solution 6; g) Assembly solution 7; h) Assembly solution 8.*

For configuration 3 (Figure 8c), the solution with the smallest torque has values of g_1=24, g_2=3.2 and g_3= 4.9 and as the dexterity decreases, the torque increases so it is not better as the solution of configuration 1. In Figure 8d it can be seen that the most suitable solution has the values of the objective function g_1=24, g_2=3.4 and g_3= 1.4. This solution is slightly better than configuration 1. For configuration 5 and 6 and 8(Figure 8 e, Figure 8f and Figure 8h), there is no solution in the used scale, so it can be concluded that there is not a better solution than the one proposed in the other configuration. In Figure 8g it can be seen a set of suitable solutions like the one with values for objective function g_1=24, g_2=3.18 and g_3= 2.5. and g_1=24, g_2=3.5 and g_3= 1.2.

After comparing the solutions, it was chosen the one with objective function g_1=24, g_2=3.18 and g_3= 2.5 despite it has a slightly worst dexterity than the solution of configuration 1 because it is the one with the lowest torque, and a lower torque represents smaller actuators and lower overall weight. The value of the parameters associated to this solution are shown in Table 1.

Table 1: *Parameters used for the synthesis of the mechanism.*

Parameter	Value [°]	Parameter	Value [°]
ϕ_1	48	α_{2_1}	76
ϕ_2	224	α_{2_2}	52
Γ	77	α_{2_3}	53
B	76	δ_1	0
α_{1_1}	39	δ_2	76
α_{1_2}	39	δ_3	200
α_{1_3}	47		

With the calculated values, a CAD design of the manipulator was done. Two dynamic simulation were performed in order to evaluate the performance of the manipulator. The first simulation aimed to perform a flexion-extension movement with a load of 5 N at the location of the hand, Figure 9. The movement begins with 20° of extension and finishes at 90° of flexion. The time required to perform the movement was stablished to be 2 s. The material of the prosthetic device was ABS except for the links of the parallel manipulator that were model as aluminum.

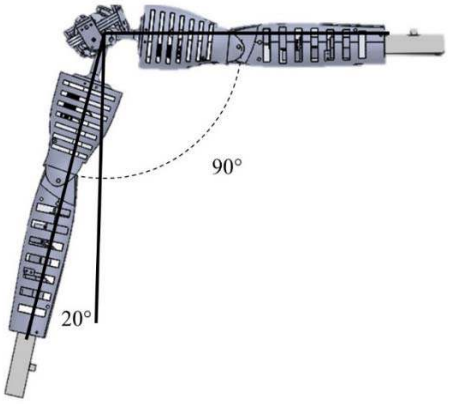


Figure 9*: Representation of the movement performed during the dynamic simulation.*

The results of the simulation show that during the movement, the actuators perform smooth movements. The manipulator does not approach any singularity, and this is because the optimization of the dexterity. The maximum torque that the actuators require is 2.98 Nm, Figure 10.

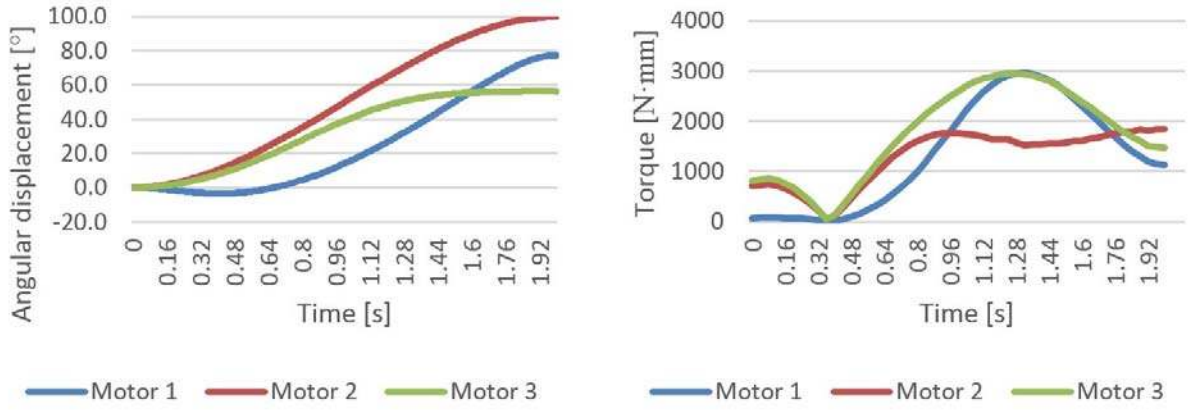


Figure 10: *Results of the humeral flexion-extension simulation: a) angular displacement of the actuators; b) torque of the actuators.*

The second simulation consisted in the same movement as in the previous analysis but with the elbow in a flexed position, Figure 11.

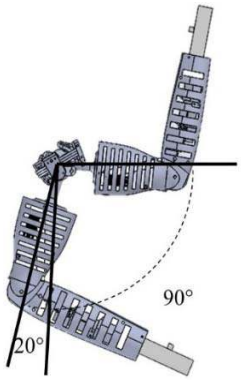


Figure 11: *Representation of the movement performed during the second dynamic simulation.*

The results show that the maximum required torque to perform this movement is 2.9 Nm, Figure 12.

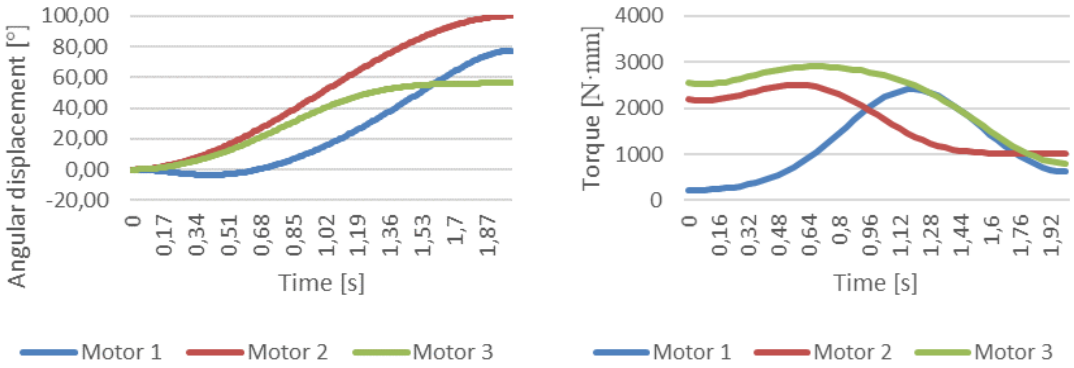


Figure 12: *Results of the humeral flexion-extension simulation: a) angular displacement of the actuators; b) torque of the actuators.*

A third simulation was performed in order to compare with the previous design. The characteristics of the simulation are the same as simulation one but the payload was increased to 10 N and showed that when the payload is increased to 1 kg, the maximum required torque is 4 Nm.

6 Conclusions

In this work, a multi-objective optimization is used to design of a spherical parallel manipulator as the shoulder of a prosthetic device. The results show that the required torque to perform the defined movement was reduced in

comparison with the previous design (4 Nm vs 6 Nm). In comparison to the previous configuration of the mechanism used, the range of motion was increased and an improvement in the controllability of the mechanism was achieved in terms of the dexterity. From the results, it is evident that the optimization of the three objectives needs to be carried out in a simultaneous way because they are dependent to each other and an improvement in one characteristic could yield to an undesired repercussion of the others.

Acknowledge

The first author acknowledges Consejo Nacional de Ciencia y Tecnología (CONACyT) for supporting his PhD study and research with a period of study in 2016 at LARM in Cassino within a double PhD degree program.

References

Blum, C., Chiong, R., Clerc, M., Jong, K. D., Michalewicz, Z., Neri, F. and Weise, T. (2012). Evolutionary optimization. Variants of Evolutionary Algorithms for Real-World Applications, Springer Berlin Heidelberg: 1-29.

Boudrear, R. and Gosselin, C. (1999). The systhesis of planar parallel manipulators with a genetic algorithm, Journal of mecanical design 121: 533-537.

Boudreau, R. and Turkkan, N. (1996). Solving the forward kinematics of parallel manipulators with a genetic algorithm, Journal of robotic systems 13: 111-125.

Cabrera, J. A., Simon, A. and Prado, M. (2002). Optimal synthesis of mechanisms with genetic algorithms, Mechanism and machine theory 37: 1165-1177.

Castejón, C., Carbone, G., Prada, J. C. G. and Ceccarelli, M. (2010). A multi-objective optimization of a robotic arm for service tasks, Strojniški vestnik - Journal of Mechanical Engineering 56: 316-329.

Ceccarelli, M. (2004). Fundamentals of mechanics of robotic manipulation, Springer Science & Business Media.

Ceccarelli, M. and Lanni, C. (2004). A multi-objective optimum design of general 3R manipulators for prescribed worspace limits, Mechanism and machine theory 39: 113-132.

Coello, C. A. C. (2015). Multi-objective evolutionary algorithms in real-world applications: Some recent results and current chalenges. Advances in Evolutionary and Deterministic Methods for Design, Optimization and Control in Engineering and Sciences, Springer International Publishing: 3-18.

Coello, C. A. C., Lamont, G. B. and Veldhuizen, D. A. V. (2007). Evolutionary algorithms for solving multi-objective problems, Springer.

Chaker, A., Mlika, A., Laribi, M. A., Romdhane, L. and Zeghloul, S. (2012). Synthesis of spherical parallel manipulator for dexterous medical task, Front. Mech. Eng. 7(2): 150-162.

Davis, L. (1991). Handbook of genetic algorithms.

Deb, K. (2014). Multi-objective optimization. Search methodologies, Springer US: 403-449.

Deb, K. and Goel, T. (2001). Controlled elitist non-dominated sorting genetic algorithms for better convergence. Evolutionary Multi-Criterion Optimization. EMO 2001. Lecture Notes in Computer Science, vol 1993., Springer, Berlin, Heidelberg: 67-81.

Deb, K. and Kumar, A. (1995). Real-coded genetic algorithms with simulated binary crossover: studies on multimodal and multiobjective problems, Complex systems 9: 431-454.

Deb, K., Prapat, A., Agarwal, S. and Meyarivan, T. (2002). A fast elitist multiobjective genetic algorithm: NSGA-II, IEEE Transactions on evolutionary computation 6(2): 182-197.

Dillingham, T. R., Pezzin, L. E. and MacKenzie, E. J. (2002). Limb amputation and limb deficiency: epidemiology and recent trends in the United States, South Med J 95(8): 875-883.

Elarbi, M., Bechikh, S., Said, L. B. and Datta, R. (2017). Multi-objective optimization: Classical and evolutionary approaches. Recent advances in evolutionary multi-objective optimization. Springer International Publishing: 1-30.

Essomba, T., Laribit, M., Zeghloul, S. and Poisson, G. (2016). Optimal synthesis of a spherical parallel mechanism for medical application, Robotica 34(3): 671-686.

Fogel, D. (1994). An introduction to simulated evolutionary optimization, IEEE Transactions on neural networks 5(1): 3-14.

Gosselin, C. and Angeles, J. (1987). The optimum kinematic design of a spherical three degree of freedom parallel manipulator, Journal of mechanism, transmissions, and automation in design 111: 202-207.

He, L., Xiong, C. and Zhang, K. (2014). Mechatronic Design of an Upper Limb Prosthesis with a Hand. Intelligent Robotics and Applications: 7th International Conference, ICIRA 2014, Guangzhou, China, December 17-20, 2014, Proceedings, Part I. X. Zhang, H. Liu, Z. Chen and N. Wang. Cham, Springer International Publishing: 56-66.

Hernandez, E. E., Valdez, S. I., Ceccarelli, M., Hernandez, A. and Botello, S. (2015). Design optimization of a cable-based parallel tracking system by using evolutionary algorithms, Robotica 33(3): 599-610.

Herrera, F., Lozano, M. and Verdegay, J. L. (1998). Tackling real-coded genetic algorithms: Operators and tools for behavioural analysis, Artificial Intelligence Review 12(4): 265-319.

Holland, J. (1992). Adaptation in natural and artificial systems, MIT University press.

Johannes, M. S., Bigelow, J. D., Burck, J. M., Harshbarger, S. D., Kozlowski, M. V. and Van Doren, T. (2011). An overview of the developmental process for the modular prosthetic limb, Johns Hopkins APL Technical Digest 30(3): 207-216.

Leal-Naranjo, J. A., Ceccarelli, M. and Torres-San Miguel, C. R. (2016). Mechanical Design of a Prosthetic Human Arm and its Dynamic Simulation. International Conference on Robotics in Alpe-Adria Danube Region, Springer: 482-490.

Li, T. and Payandeh, S. (2002). Design of spherical parallel mechanisms for application to laparoscopic surgery, Robotica 20(2): 133-138.

Michalewicz, Z. (1996). Genetic Algorithms + Data Structures = Evolution Programs. Springer-Verlag Berlin Heidelberg.

Miettinen, K. (2008). Introduction to multiobjective optimization: Noninteractive approaches. Multiobjective Optimization. Interactive and evolutionary approaches, Springer-Verlag Berlin Heidelberg: 1-26.

Ramana, B., Ramachandra, R. and Ramji, K. (2016). Design optimization of 3PRS parallel manipulator using global perfomance indices, Jorunal of Mechanical Science and Technology 30(9): 4325-4335.

Resnik, L., Klinger, S. L. and Etter, K. (2014). The DEKA Arm: its features, functionality, and evolution during the Veterans Affairs Study to optimize the DEKA Arm, Prosthet Orthot Int 38(6): 492-504.

Sivanandam, S. and Deepa, S. (2007). Introduction to genetic algorithms, Springer Science & Business Media.

Tan, K. C., Khor, E. F. and Lee, T. H. (2006). Multiobjective evolutionary algorithms and applications, Springer Science & Business Media.

Tsai, L.-W. (1999). Robot analysis: the mechanics of serial and parallel manipulators, John Wiley & Sons.

Wang, Z., Wong, Y. and Rahman, M. (2004). Optimisation of multi-pass milling using genetic algorithm and genetic simulated annealing, The International Journal of Advanced Manufacturing Technology 24(9-10): 727-732.

Wu, G. (2012). Multiobjective optimum design of a 3-RRR spherical paralle manipulator with kinematic and dynamic dexterities, Modeling, Identification and Control 33(3): 111-122.

Zhen, G., Dan, Z. and Yunjian, G. (2010). Design optimization of a spatial six degree-of-freedom parallel manipulator based on artificial intelligence approaches, Robotics and Computer-Integrated Manufacturing 26: 180-189.