Memorias del Congreso Nacional de Control Automático
ISSN: 2594-2492

# A spherical wrist prototype for the development of haptics applications with the Novint Falcon robot.

**Enrique Martínez-Fernández** * **José Daniel Castro-Díaz** ** **Marco A. Arteaga-Pérez** **

* *Facultad de Ingeniería, UNAM, Ciudad de México, México. C.P. 04510 (email: enrique.mfz@outlook.com)*
** *Departamento de Control y Robótica, Facultad de Ingeniería, UNAM, Ciudad de México, México C.P. 04510 (email: jsdnlcstr@gmail.com)*

**Abstract:** The Novint Falcon parallel robot is an inexpensive device originally created as a joystick for the video game industry. However, its closed structure and low price make it an ideal platform for the development of science-oriented virtual reality applications with force rendering. Although its three Degrees of Freedom (DOF) are adequate for a wide variety of situations where the user feels feedback forces by modifying the end-effector's position, more complex haptic applications require three additional DOF so feedback torques related to the end effector's orientation can be generated. This paper reports a first approximation for the creation of a spherical wrist prototype that increases the Novint Falcon's dexterity. A virtual reality application was developed in order to showcase the mechanism's new functionality.

Keywords: parallel robot, virtual reality, spherical wrist, force rendering.

## 1. INTRODUCTION

The delta robot created by Raymond Clavel [Clavel, 1990] is a three degree of freedom (DOF) parallel mechanism consisting of a fixed base, a series of kinematic parallelograms and an end-effector that can perform translational motion along three axes while remaining parallel to the base. This design offers several advantages over a serial manipulator: lower end-effector inertia, higher payload-carrying capability, greater speed and acceleration, better position accuracy, among others. The delta architecture is widely used in the industry for pick-and-place tasks, but due to the previously described characteristics, this robot can be extremely useful in other areas. Based on the delta mechanism, a company called Novint created the Falcon robot, which was focused to create video games and haptics applications. This device offers a suitable platform for the development and research in the haptic and virtual reality areas at a low cost. However, despite its many advantages, the three DOF limit the range of motions the human operator can perform, reducing the robot's overall applicability. Adding three DOF to the mechanism through a spherical wrist would make it a more versatile device than it already is.

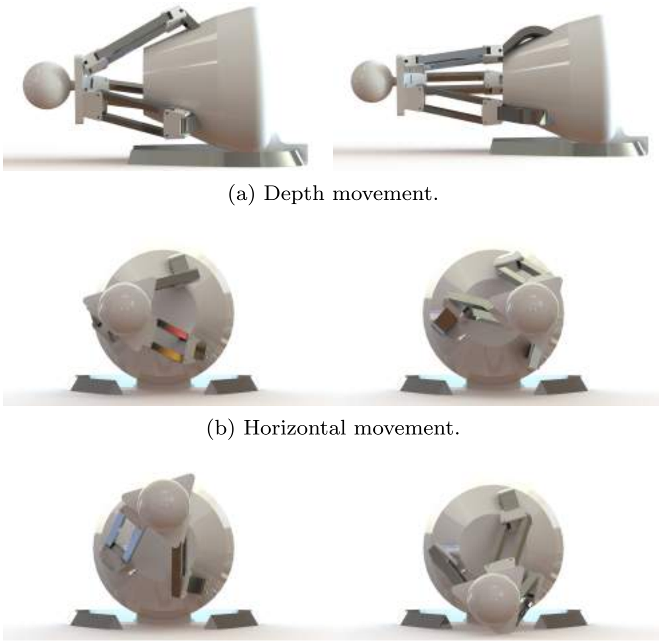
There are robots in the market with similar functionality, like the Omega 3, 6 and 7, Sigma 7 from Force Dimension. All of these are designed for haptics applications but each of them has different features. The Omega series of robots have a parallel structure and have 3, 6 or 7 DOF. The seventh DOF is used to simulate the use of a tool. The Sigma 7 differentiates from the Omega series because all of the former's DOF are actuated, while in the latter only the first three DOF are active. The biggest downside of these devices is their high price, while on the other hand the Novint Falcon can do similar tasks at a fraction of the cost.

The motions the Novint Falcon can perform are shown in Figure 1. The end-effector can translate along the $x, y$ and $z$ axis while keeping a fixed orientation. With these DOF, this device has proven to be versatile enough to be used in a wide variety of areas, such as teleoperation, virtual reality and medicine. However, advanced haptics applications such as surgery simulators require a dexterous mechanism capable of positioning and rotating its end-effector.

This paper presents the design and attachment of a spherical wrist prototype to the robot, its control and a simple virtual reality application demonstrating its new functionality.

The article is organized as follows: the mechanical structure of the spherical wrist mechanism and its kinematic structure is presented in Section 2. The development of a virtual reality application showcasing the Falcon's new capabilities is shown in Section 3. Section 4 gives some general control aspects used in this work. Concluding remarks are given in Section 5.



(a) Depth movement.

(b) Horizontal movement.

(c) Vertical movement.

Fig. 1. Range of motion of the Novint Falcon along the three axes.

## 2. SPHERICAL WRIST DESIGN

A spherical wrist is a mechanism that satisfies the following criteria [Spong et al., 2006]:

- It has three DOF.
- The three axes of rotation are perpendicular to each other.
- The three axes of rotation intersect at a common point called the *wrist center*.

in robotics the main advantage of using a spherical wrist is to confer a mechanism the extra 3 DOF that enable decoupling of position and orientation, i.e. they can be computed separately and thereafter joined to find the total solution. This property is what made a spherical wrist an ideal candidate for modifying an already existing device: it allows to attach the mechanism to the robot as in a modular design. Considering all of the above characteristics of a spherical wrist, the goals for designing the mechanism to be used in conjunction with the Novint Falcon are:

- The cost should be low in order to achieve similar functionality to the robots that already exist in the market at a lower price.
- The wrist should be light enough so the robot can still easily move around its end-effector.
- All three DOF must be actuated.

With these criteria in mind, the wrist shown in Figure 2 was designed. Each link's geometry is such that the next axis of rotation will be perpendicular to the previous one. The wrist has three DC motors, one for each joint, controlled via Pulse-Width Modulation (PWM).

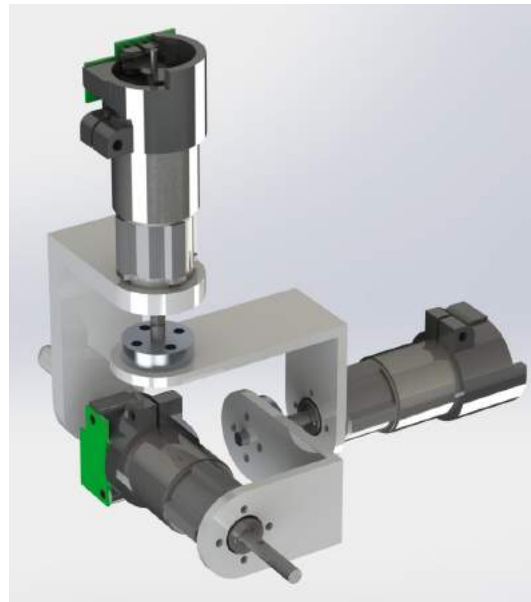


Fig. 2. Render of the designed spherical wrist mechanism.

Figure 3 shows a picture of the robot with the attached mechanism, where a pen-like end-effector was added.

Through the Falcon's grip interchange feature, the mechanism can be attached/detached at will, effectively making the wrist an upgradeable module. This means that future improvements can be prototyped and tested with ease without the need of a physical modification of the Falcon's kinematic chain.

The wrist adds a weight of approximately 4.3[N], which is less than the maximum force the robot can exert, thus it can be compensated through the control algorithm. This is important because a heavy end-effector can bring discomfort to the user's wrist or shoulder, especially after an extended usage period. This also limits the forces the robot can display and lighter motors may be used in a future iteration of the wrist design.

The electronics required to control the mechanism were also developed. A Printed Circuit Board (PCB) was designed to fulfill the following tasks:

- Rotation speed and direction control of each motor with PWM through an H bridge and a microcontroller.
- Provide optical isolation between the logic and power circuits.
- Use of a single 12[V] power supply.

The microcontroller interfaces to a PC through a USB port, enabling the wrist to be used in a virtual reality (VR) application.

### *2.1 Kinematic analysis*

The direct kinematics of the wrist prototype was calculated using the Denavit-Hartenberg convention. Figure 4 shows a diagram representation of the wrist where the reference systems were assigned by following the algorithm.

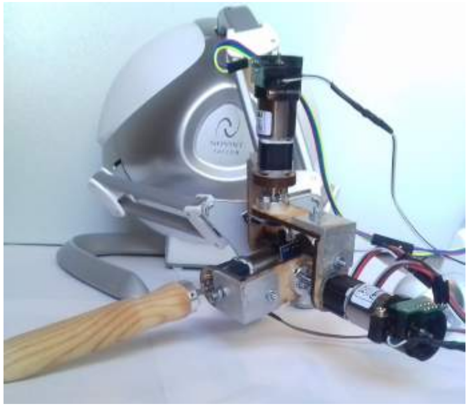


Fig. 3. Falcon robot with the spherical wrist prototype attached.

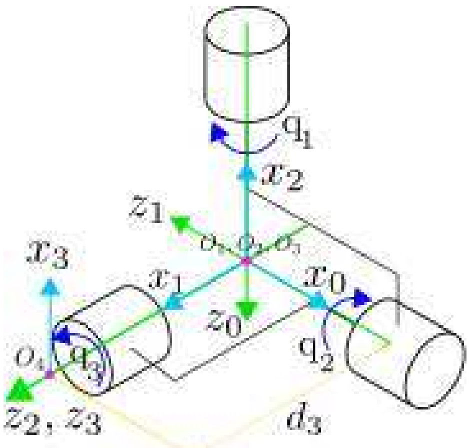



Fig. 4. Denavit-Hartenberg algorithm applied to the spherical wrist.

| Link | $a_i[mm]$ | $d_i[mm]$ | $\alpha_i[°]$ | $\theta_i[°]$ |
|---|---|---|---|---|
| 1 | 0 | 0 | -90 | $\theta_1$ |
| 2 | 0 | 0 | 90 | $\theta_2$ |
| 3 | 0 | 54 | 0 | $\theta_2$ |

Table 1. Denavit-Hartenberg parameters.

Table 1 shows the Denavit-Hartenberg parameter obtained from the mechanism's dimensions and geometry. By using this algorithm the homogeneous transformation matrices $\boldsymbol{A_1, A_2, A_3}$ can be systematically computed. For the designed mechanism, these matrices are

$$\boldsymbol{A_1} = \begin{bmatrix} \cos(\theta_1) & 0 & -\sin(\theta_1) & 0 \\ \sin(\theta_1) & 0 & \cos(\theta_1) & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

$$\boldsymbol{A_2} = \begin{bmatrix} \cos(\theta_2) & 0 & \sin(\theta_2) & 0 \\ \sin(\theta_{21}) & 0 & -\cos(\theta_2) & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

$$\boldsymbol{A_3} = \begin{bmatrix} \cos(\theta_3) & -\sin(\theta_3) & 0 & 0 \\ \sin(\theta_3) & \cos(\theta_3) & 0 & 0 \\ 0 & 0 & 1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3)$$

and the total transformation is $\boldsymbol{{}^0T_3 = A_1 A_2 A_3}$

$${}^0\boldsymbol{T_3} = \begin{bmatrix} c\theta_1 c\theta_2 c\theta_3 - s\theta_1 s\theta_3 & -c\theta_3 s\theta_1 - c\theta_1 c\theta_2 s\theta_3 & c\theta_1 s\theta_2 & d_3 c\theta_1 s\theta_2 \\ c\theta_2 c\theta_3 s\theta_1 + c\theta_1 s\theta_3 & c\theta_1 c\theta_3 - c\theta_2 s\theta_1 s\theta_3 & s\theta_1 s\theta_2 & d_3 s\theta_1 s\theta_2 \\ -c\theta_3 s\theta_{21} & s\theta_3 s\theta_{21} & c\theta_2 & d_3 c\theta_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4)$$

With this matrix the mechanism's orientation referred to the Falcon's end-effector reference system can be calculated.

## 3. VIRTUAL REALITY APPLICATION DESIGN

A virtual reality application was developed to show the Falcon's new functionality. The objective was to create an environment with which the user can haptically interact using a keyboard and the robot. This was achieved through the game engine Unity. Figure 5 shows a diagram of how the VR application works. The user interacts with the virtual world through the robot, which communicates with Unity using a Dynamic Link Library (DLL) provided by Novint. This DLL is only available on a 32 bits architecture and so Unity 5.6.6 was used because it's the last version that supported said architecture. The DLL has the necessary functions to establish communication, read the encoders and move the actuators. The wrist mechanism is controlled by an Arduino Due, and it performs the same functions as the DLL does for the Novint Falcon. Unity responds to the user's actions by updating the camera's rotation and position in the virtual world and provides visual feedback by displaying an image on the screen. Finally, if the user interacts with an object in the virtual environment, Unity communicates with the robot to render the required forces.

The user is placed in a room that can be explored and interacted with from a first person perspective. There are two states in which the user can be: *exploration mode* and *valve scene mode*.

### *3.1 Exploration mode*

The goal of this first state is for the user to explore the virtual world. This is achieved by translating and rotating the camera from which the environment is seen. By using the keys *w* and *s*, the user can move forward and backwards respectively. The same is achieved in the left or right directions by using the keys *a* and *d*. In order to look around, the user has to move the robot's end-effector horizontally or vertically as in Figure 1(b)-(c), which will rotate the camera in those directions. By combining all of these actions the user can explore the virtual world in a similar way to a first person video game.

### *3.2 Valve scene mode*

The second state of the virtual reality application involves interaction with a valve located on one of the room walls. If the user points a crosshair located at the middle of the screen directly to the valve as shown in Figure 6 and presses the *e* key, a series of actions are performed:

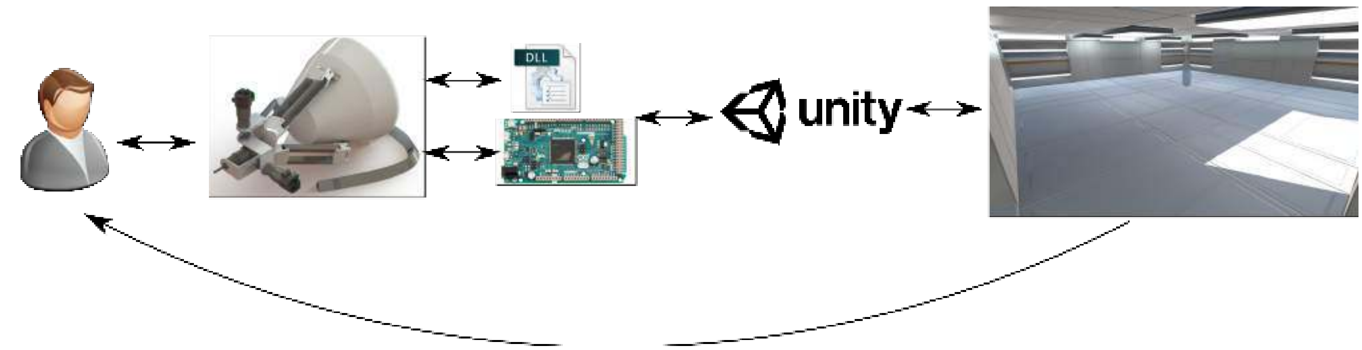



Fig. 5. Diagram of the VR application.

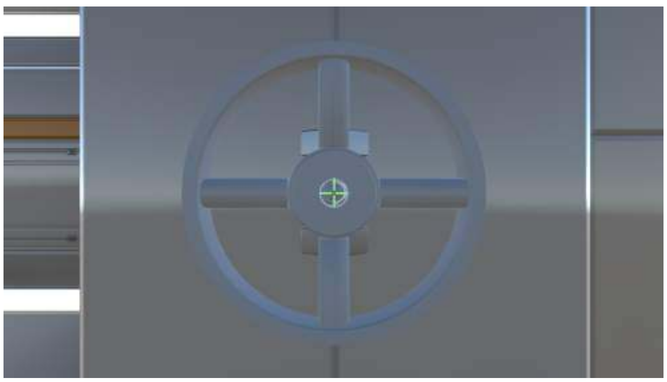


Fig. 6. The valve scene.

- The camera view is placed in front of the valve and the user is locked in place.
- A position controller, whose goal is to regulate the first five robot joints to the home position, is started.
- Haptic feedback is provided by simulating viscous friction on the last joint of the robot when the user rotates the end-effector about its rotation axis.

The goal of the above actions is to temporarily "lose" five DOF on the robot so the user can interact with the virtual world exclusively by rotating the end-effector about its own axis, making the virtual valve's angular position change at a maximum rate of 60[Hz], which is the monitor's refresh rate.

Finally, the user can return to the exploration mode by pressing the *Esc* key.

## 4. CONTROLLERS DESIGN AND PERFORMANCE

Two controllers were designed for the virtual reality application: one for the exploration mode and another for the valve scene mode.

During the exloration mode an open-loop controller is applied to compensate the wrist's weight. This avoids strain on the user's arm. The Novint Falcon creates an upwards constant force that carries the mechanism, and the user feels like the end-effector is just as light as the original robot's grip. An open-loop control is suitable because the wrist's weight is constant and the user should be able to move the end-effector freely.

During the valve scene mode a closed-loop controller is used to regulate the joints at the home position. This is done by the control law

$$\boldsymbol{u} = \boldsymbol{u}_{pos} + \boldsymbol{u}_{fv} \tag{5}$$

where

$$\boldsymbol{u}_{pos} = \boldsymbol{K}_p\tilde{\boldsymbol{q}} + \boldsymbol{K}_d\dot{\tilde{\boldsymbol{q}}} + \boldsymbol{K}_i\int_0^t \tilde{\boldsymbol{q}}\, dt \tag{6}$$

is a Proportional Integral Derivative (PID) controller that regulates the robot's position and

$$\boldsymbol{u}_{fv} = \boldsymbol{K}_{fv}\dot{\boldsymbol{q}} \tag{7}$$

simulates viscous friction. $\boldsymbol{K_p}, \boldsymbol{K_d}, \boldsymbol{K_i}$ are positive definite gain matrices that affect the first five joints of the robot. $\tilde{\boldsymbol{q}} = \boldsymbol{q}_d - \boldsymbol{q}$ is the position error and $\dot{\tilde{\boldsymbol{q}}}$ is its derivative. $\boldsymbol{K_{fv}}$ is also a gain matrix, and its only non-zero element affects the last joint exclusively. The user can modify this matrix during execution to simulate various degrees of friction.

### *4.1 Controller performance*

During exploration mode the open.loop controller takes place. Unity reads the robot's position and uses that information to move the user around in the virtual world. No graphs can be shown for this mode because the user can move the end-effector freely.

However, experimental results can be shown for the valve scene mode, where the PID controller takes place. Figures 7-11 show the graphs of each joint during the scene. The dashed line represents the reference signal, while the continuous line is the real signal. Perturbations which are shown in each graph were added during the experiment to see if the controller could compensate them. There is no position regulation at the last joint, so Figure 12 only shows its angular position.

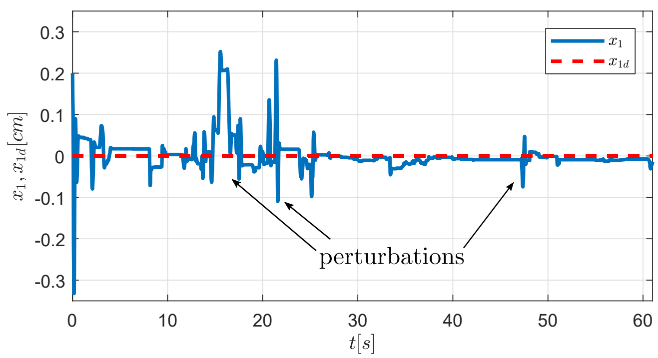



Fig. 7. Graphs of $x$ position.

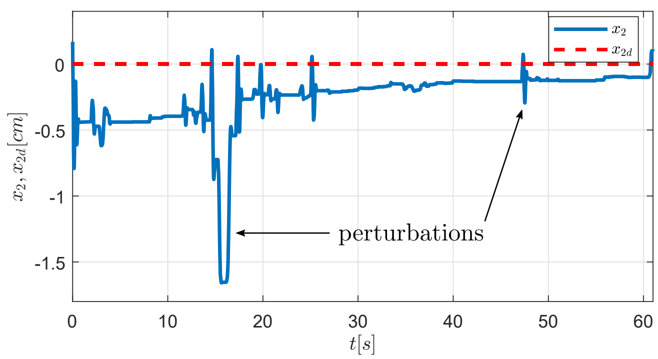



Fig. 8. Graphs of $y$ position.

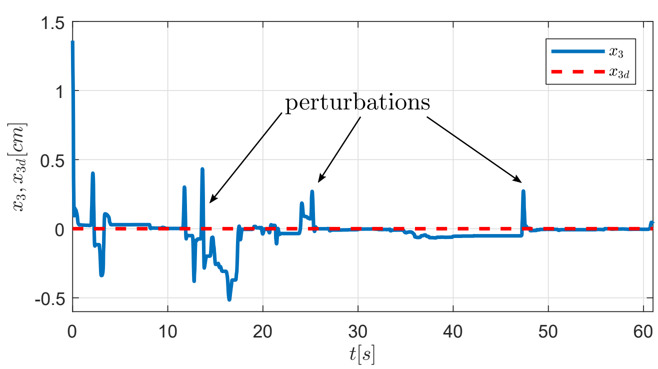



Fig. 9. Graphs of $z$ position.

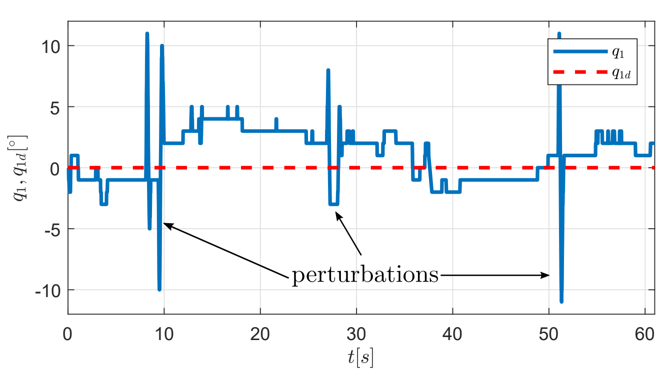



Fig. 10. Graphs of the wrist's first joint.

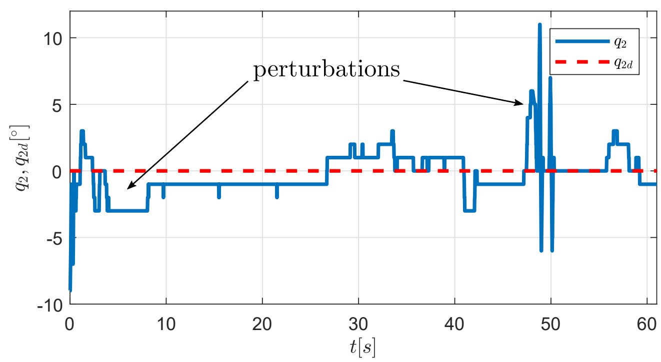



Fig. 11. Graphs of the wrist's second joint.

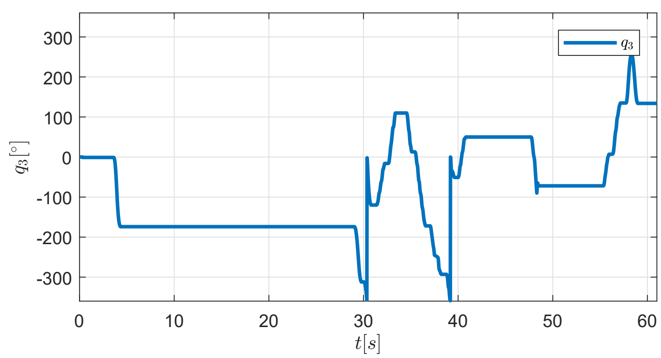



Fig. 12. Graph of the wrist's third joint.

## 5. CONCLUSIONS

An inexpensive spherical wrist prototype was developed for the Novint Falcon robot. Due to the mechanism's properties and the Falcon's grip interchange feature the design is modular, thus enabling easy upgrades and modifications. Similar functionality to the mainstream haptic robots was achieved at a significantly lower price. This could potentially make developing haptics and virtual reality applications more affordable and accessible to budget-limited projects. However, significant improvements can be made to this prototype. Cable management has to be made to prevent tangles and interference while moving the robot. Smaller motors would make the whole mechanism more compact and lighter at the cost of lower torque capability. Finally, a lighter and more resistant material is preferred for a future iteration.

The virtual reality application showed the robot's capabilities and will serve as a basis for future development of more complex haptics apps. More sophisticated control algorithms are yet to be tried.

## ACKNOWLEDGMENTS

This project was supported by **PAPIIT-UNAM** through proyect **IN114617**.

## REFERENCES

R. Clavel. *Device for the movement and positioning of an element in space*. Unites States Patent Office: 0250470, 1990.

Carla García López de Llergo. Control de fuerza sobre superficies virtuales. Master's thesis, UNAM, 2014.

Zoran Pandilov and Vladimir Dukovski. Comparison of the characteristics between serial and parallel robots. *Acta Technica Corviniensis-Bulletin of Engineering*, 7 (1):143, 2014.

Iván Torres Rodríguez. Puesta en funcionamiento del robot paralelo novint falcon. Bachelor's thesis, UNAM, 2017.

William R Sherman and Alan B Craig. *Understanding virtual reality: Interface, application, and design*. Elsevier, 2002.

B Siciliano, L Sciavicco, L Villani, and G Oriolo. Robotics–modelling, planning and control. advanced textbooks in control and signal processing series, 2009.

Mark W Spong, Seth Hutchinson, Mathukumalli Vidyasagar, et al. *Robot modeling and control*, volume 3. Wiley New York, 2006.

Lung-Wen Tsai. *Robot analysis: the mechanics of serial and parallel manipulators*. John Wiley & Sons, 1999.