




# Automatic Vision Based Calibration System for Planar Cable-Driven Parallel Robots

Andrés García-Vanegas<sup>1(✉)</sup>, Brhayan Liberato-Tafur<sup>1,2</sup>,  
Manuel Guillermo Forero<sup>2(✉)</sup> , Antonio Gonzalez-Rodríguez<sup>3</sup>,  
and Fernando Castillo-García<sup>3(✉)</sup>

<sup>1</sup> Semillero MEC-AUTRONIC, Facultad de Ingeniería, Universidad de Ibagué,  
Ibagué, Tolima, Colombia

[jorge.garcia@unibague.edu.co](mailto:jorge.garcia@unibague.edu.co)

<sup>2</sup> Semillero Lún, Facultad de Ingeniería, Universidad de Ibagué,  
Ibagué, Tolima, Colombia

[manuel.forero@unibague.edu.co](mailto:manuel.forero@unibague.edu.co)

<sup>3</sup> Universidad de Castilla-La Mancha, Toledo, Spain

[Fernando.Castillo@uclm.es](mailto:Fernando.Castillo@uclm.es)

**Abstract.** Techniques commonly employed to calibrate cable-driven parallel robots are manual or use sensors to measure the end-effector position indirectly. Therefore, in this paper, a cable-driven robot calibration system based on artificial vision techniques is introduced, which takes advantage of the available information, which allows calibrating the system directly from the observed video. The proposed method was validated by calibrating a planar cable-driven parallel robot prototype with 2 degrees of freedom. The measured positioning error was lower than 1 mm.

**Keywords:** Cable-driven parallel robot · Parallel robotic · Computer vision · Vision-based control · Robot calibration

## 1 Introduction

A cable-driven parallel robot (CDPR) is a kind of robot, composed of a mobile platform (end-effector), connected by cables to a fixed one (fixed frame). These robots have a cable collector system composed of: an actuator, a reduction box and a collector mechanism. There are two kinds of CDPRs: planar, with 2 degrees-of-freedom, and spatial, with 3 or more degrees-of-freedom. They can be suspended, controlled and over-controlled. To control the location of the end-effector in the 2D space, the cables length is modified according to a mathematical model, that relates these lengths to the parameters of the robot. These robots can handle high payloads and are characterized by their fast response and improved accuracy [1]. The end-effector is placed on the mobile platform. It can be a clamp for pick-and-place applications, a welding or painting tool, a

plasma or laser cutting gun, a camera for the transmission of videos (Skycam), an extruder for 3D printing [2], among others applications. These robots are employed in applications such as production engineering, construction, motion simulation and entertainment.

Techniques commonly employed to calibrate cable-driven parallel robots are manual or use sensors to measure the end-effector position indirectly [3–6]. Although several vision-based control systems have been developed [7–9], there is not in the literature any automatic method developed to calibrate the end-effector robotic system. Therefore, in this paper, a cable-driven robot calibration system based on artificial vision techniques is introduced, which takes advantage of the available information, allowing calibrating the system directly from the observed video.

This paper is organized as follows: Sect. 2 details the materials used in this works. Section 3 introduces the calibration and tracking system methods of the CDPR. Section 4 presents the experimental results. The paper concludes with some remarks and suggestions for future works.

## 2 Materials

In this study, a two-degrees-of-freedom planar cable-driven parallel robot prototype was developed, shown in Fig. 1. The prototype was designed with Solidworks and built with aluminum profiles and 3D printed parts (PLA). The prototype specifications are given in Table 1. As can be seen, the positioning system consists of three red reference squares; two of 80 mm located on the corners of the pulley system, and the other 111 mm located in the end-effector. The robot video input signal is acquired using a Logitech HD Pro C920 webcam with 1080p resolution, located at 1.2 m from the end-effector.

This study used a HP-440 G5 computer with 8 GB of memory, an Intel Core i5 processor (R) CPU @ 1.8, and a Windows 10 operating system. The calibration algorithms were implemented in Python 3.7 using the OpenCV and PyQt5

**Table 1.** Overview of the CDPR specification.

Parameter	Value/Component	Unit
Stepper motors (NEMA 23)	2	-
Number of cable	4	-
Gear ratio	10:1	-
Drum diameter	63.5	mm
Cable type	Steel	-
Cable diameter	6.0	mm
Degrees-of-freedom	2	-
Size of the robot frame	1.0 × 1.0	m
Size of the mobile platform	15.0 × 15.0	cm

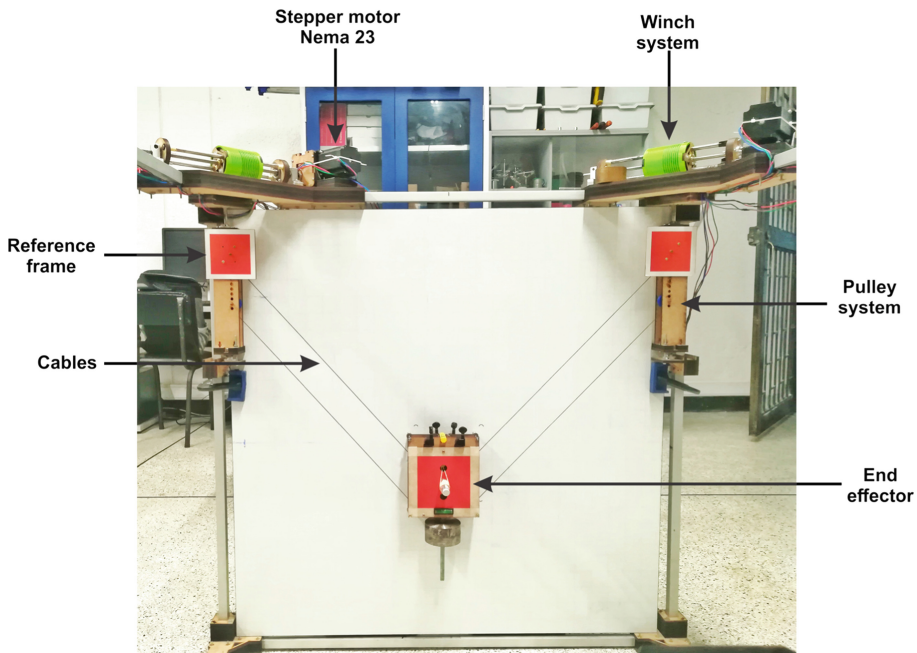


Fig. 1. Prototype of planar cable-driven parallel robot.

libraries. The image processing methods were initially developed and tested using the free access software ImageJ. This application was also employed to measure and validate the trajectories of the end-effector.

### 3 Methods

The calibration system developed here is based on computer vision techniques. First, the video input is decomposed into frame images. Then, calibration is performed using the red squares, observed in Fig. 1, as landmarks. Red squares were chosen as landmarks because the predominant color in a crop is normally green, and the red color is the opposite one or complementary to the green in the chromatic circle. In this way, a better color separation between the red squares and the background is obtained, making it easier their segmentation. The landmarks correspond to points where excess red is found [10], i.e., where the red intensity is very high and much higher than the intensities of the other two color channels. To find the landmark areas, the red end areas are searched for in images. To do this, a monochromatic image is obtained using the standard BT.601. Then, this image is subtracted from the red channel [11]. Mathematically, this operation is expressed in (1) and the process of identifying objects with excess red color is illustrated in Fig. 2.

As shown in Fig. 4b, most of the image corresponding to the crop remains in the background and areas of extreme red are detected as objects of interest.

Then, the next step is to binarize the image to isolate the pixels whose extreme red values are above a given value. It was empirically found that a value of 38 resulted in the best segmentation between the background and the landmarks.

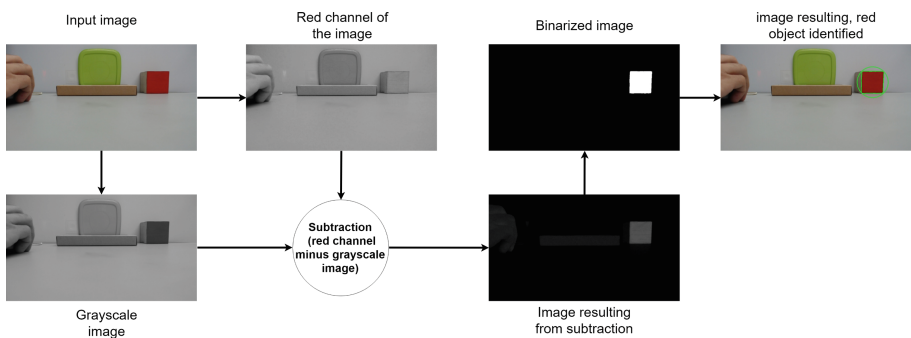
As seen in Fig. 4c, some isolated pixels are also selected as part of the object; these pixels are produced by noise in the image. To eliminate them, a morphological erosion is performed, so that the final image contains only the regions of interest. The resulting image is shown in Fig. 4d. The results obtained using this method were compared to those of the techniques listed in Table 1, and the precision provided by the algorithm was calculated, comparing the pixels of the images that were manually segmented and those that were chosen via the algorithms.

In this way, a better contrast between the red squares and the background is obtained, making it easier their segmentation. Given that these squares present a very high red value, the detection of these regions was carried out by finding pixels with an excessive red, i.e., pixels where the red value is very high and far from the values of the other two channels. Hence, square pixels are obtained from the red channel by subtracting the original grayscale image [10] obtained by using the BT.601 standard, which generates a monochromatic gray image based on the human vision model [11]. Mathematically, this operation is expressed in (1) and the process of identifying objects with excess red color is illustrated in Fig. 2.

$$Y = 0,701R - 0,5876G - 0,114B \quad (1)$$

where  $Y$  is the resulting image and  $R$ ,  $G$  and  $B$  are the Red, Green and Blue components of the input image.

Once the excessive red image is obtained, it must be binarized to find the red squares, as it is shown in Fig. 2. A threshold value of 38, found empirically, was employed, as shown in Fig. 3.



**Fig. 2.** Identification of excess red color objects. (Color figure online)

The two smallest squares, located on the pulley system (concentric to the upper pulleys), are employed as reference points required to calculate the robot



**Fig. 3.** Identification of red landmarks. Left. Excess red image. Right. Binarized image, threshold = 38. (Color figure online)

distances. A larger red square is located in it to find the end-effector position. The location of these squares allows calculating the cables length at any point in the workspace.

The landmark on the upper left is employed to measure the sizes of the squares. The square width is computed once the landmarks' four corners are detected. Then, the Euclidean distance in pixels between the landmark centroids is computed in (2). To convert distances from pixels to millimeters, the pixel to millimeter ratio is obtained by dividing the previously calculated distance between the upper left landmark width  $w$  that it is known (80 mm).

$$dE_{(p1,p2)} = \sqrt{(x2 - x1)^2 + (y2 - y1)^2} \quad (2)$$

where  $x_1, y_1, x_2, y_2$  are the coordinates of the top corners  $p_1, p_2$  of the upper left landmark.

$$pixels/millimeters = \frac{dE_{(p1,p2)}}{w} \quad (3)$$

Squares must meet two conditions for their correct identification and measurement. First, they must have four corners and, second, the relationship between the midpoints corner distances must be equal to 1. If that is true, edges are drawn, and the dimensions of the square are printed in the developed software, as shown in Fig. 4. If any landmark is tilted with respect to the camera plane, it is not identified as square, because the ratio between its sides will not be equal to 1. As shown in Fig. 5, the distance between the centroid of the upper left landmark ( $A$ ) and the upper left corner of the end-effector square ( $B$ ) is calculated by using Eq. 2. The actual length of the cables is calculated and employed to know the localition of the end-effector. As can be observed in Fig. 4, the robot has reflective pulleys located in the end-effector, employed to get a point-to-point cable length measures. "The equations of the kinematics and the dynamics of the real system are equivalent to those of the point-to-point model and, therefore, this simplified method can be used without inherent errors" [12].

As mentioned above, the calculation of the cable length is done by the point-to-point model, based on the inverse kinematics of the robot. This means that, both, the steps of the motors and the cable lengths are calculated from the

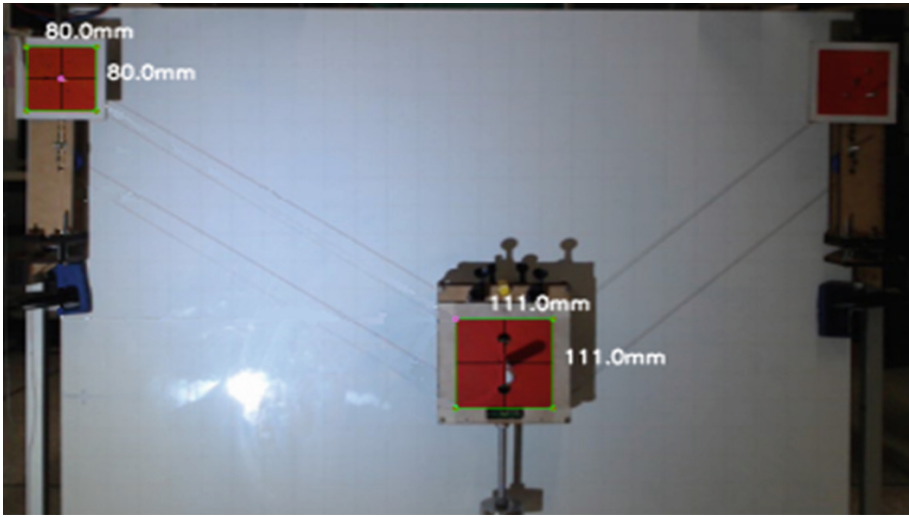


Fig. 4. Measurement of red squares. (Color figure online)

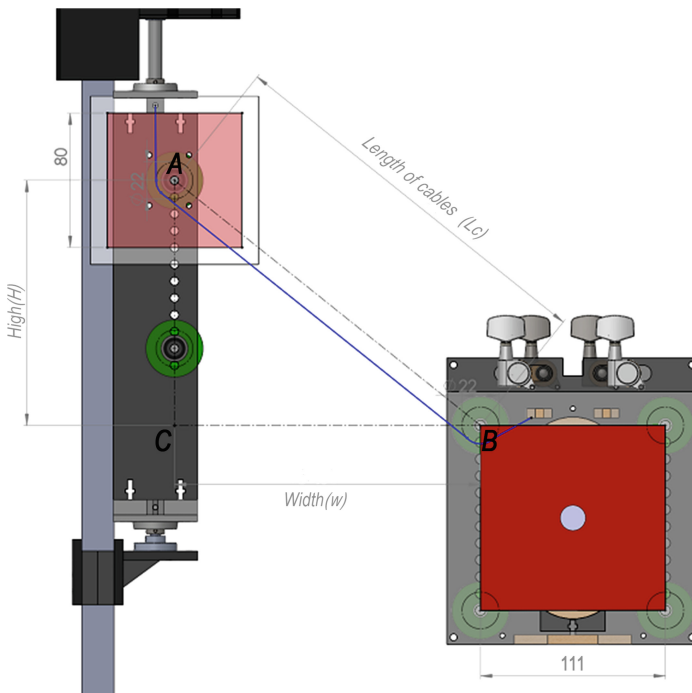


Fig. 5. Calculation of cables length.

location of the end-effector. The control of the robot is based mainly on the aforementioned model and, therefore, the controller input is given by the  $x, y$  coordinates of the end-effector.

To generate trajectories using the mathematical model, the movement of the robot must start from the reference point  $(x_0, y_0)$ , used as the home location. If the robot does not initiate a trajectory from home it will be incorrect or wrong. To determine the robot home, the coordinate in  $x$  is half the width between the centers of the pulleys, and the coordinate in  $y$  is half the height of the frame, see Fig. 6. This point will be employed as reference point of the calibration system.

Each time the robot is used and its end-effector is not at home, it must store its current position and calculate the distance between that current point and the reference point. Therefore, this will be the distance that the effector must travel to start a trajectory. It must be borne in mind that when this movement is made, it is not starting from the point of reference, hence, this will not travel exactly the calculated distance and must be calibrated several times until the effector is at home. Figure 6 represents the length that the end-effector must move to get home.

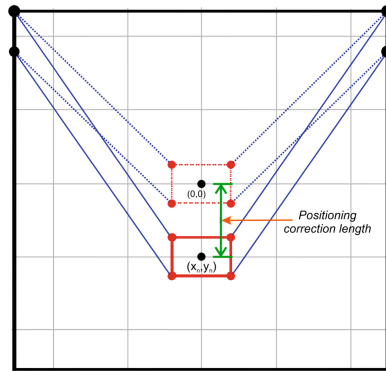


Fig. 6. Positioning correction length.

## 4 Results

The calibration system was tested by making trajectories of squares and circles, with a repetition of 10 times each. Before performing these trajectories, the end effector was calibrated about 5 times in order to reach the point of reference or home. as shown in the Table 2, the trajectories performed can be seen in Figs. 7 and 8.

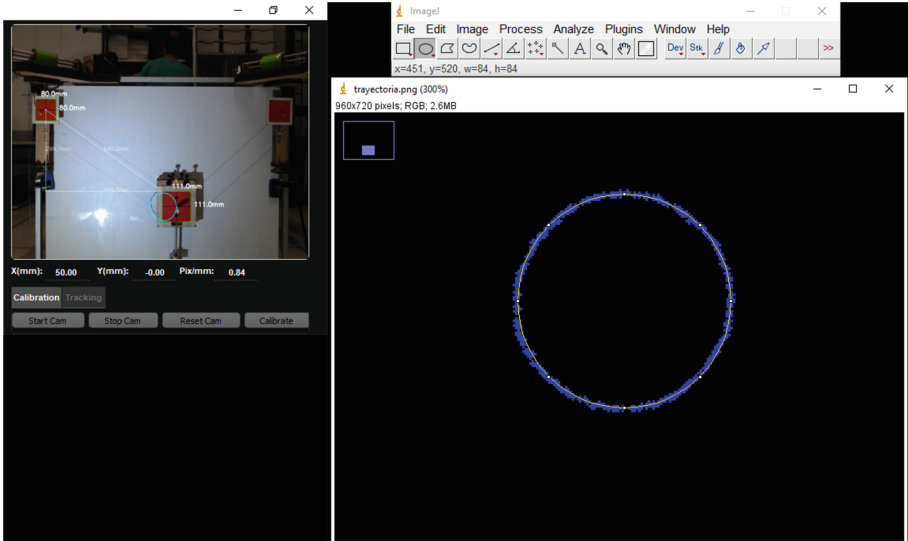
As is observed in Table 2, the maximum error was of 0.22 mm, much lower than the error obtained by published methods until now [3,6]. In addition, the performed trajectories are very accurate, despite the vibrations presented by the mechanical system (Table 3).

**Table 2.** Results of the calibration.

Number of iteration	Home position (mm)	Measured position (mm)	Error (mm)
1	0.00, 0.00	0.22, 0.22	0.22, 0.22
2	0.00, 0.00	0.00, 0.00	0.00, 0.00
3	0.00, 0.00	0.00, 0.22	0.00, 0.22
4	0.00, 0.00	0.00, 0.22	0.00, 0.22
5	0.00, 0.00	0.22, 0.00	0.22, 0.00

**Table 3.** Comparison of real and desired paths to circle of 100 mm, Square of 100 mm.

Resolution	Size (px)	Software measure (mm)	Real measure (mm)
Test trajectory: Circle of 100 mm			
0.84 px/mm	84 px (Width)	100	100
0.84 px/mm	84 px (High)	100	100
Test trajectory: Square of 100 mm			
0.84 px/mm	84 px (Width)	100	100
0.84 px/mm	84 px (High)	100	100

**Fig. 7.** Circular path measurement of 100 mm diameter.



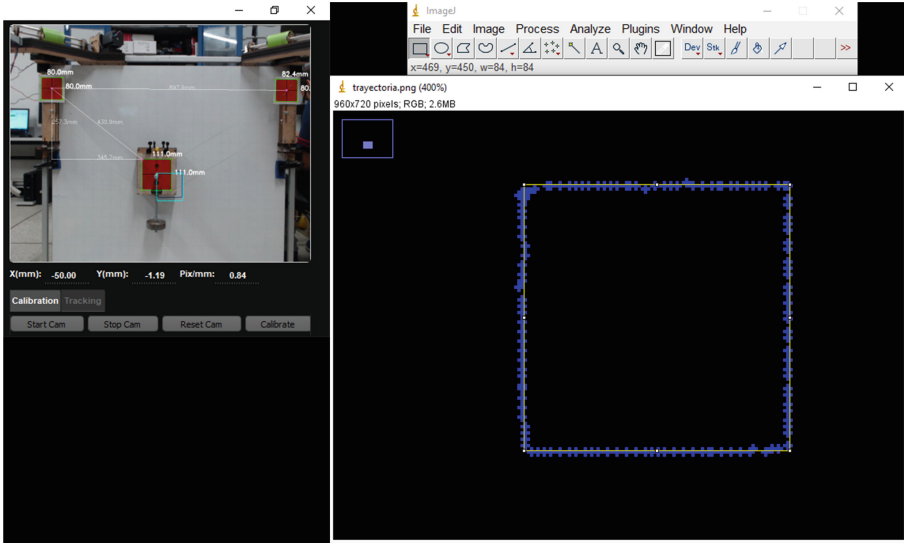


Fig. 8. Square path measurement of 100 mm.

## 5 Conclusions

In this paper a new computer vision-based system for the calibration of planar cable-driven parallel robot was introduced. It is determined that being able to visualize both the dimensions of the squares and the different distances between the corners, allows to determine the correct positioning and orientation of the camera.

It is essential that the corners or centroids of the squares coincide with the centers of the pulleys, either those of the frame or those of the end-effector, otherwise the measurement of the lengths of the cables will be erroneous.

Since the reference point is located in the middle of the robot's workspace, it is possible to calibrate the robot only with the measurement of one of its cables, since at this point all the cables have the same length.

The proposed method of vision-based calibration along with the reflective pulleys, which eliminate the position error presented in several cable robots, has an error much lower than one millimeter.

**Acknowledgment.** This work was supported by project 17-462-INT Universidad de Ibagué.

## References

1. Pott, A.: Cable-Driven Parallel Robots: Theory and Application. Springer, Heidelberg (2018). <https://doi.org/10.1007/978-3-319-76138-1>
2. Zi, B., Wang, N., Qian, S., Bao, K.: Design, stiffness analysis and experimental study of a cable-driven parallel 3D printer. *Mech. Mach. Theory* **132**, 207–222 (2019). <http://www.sciencedirect.com/science/article/pii/S0094114X18315568>
3. Borgstrom, P.H., et al.: NIMS-PL: a cable-driven robot with self-calibration capabilities. *IEEE Trans. Rob.* **25**(5), 1005–1015 (2009)
4. Miermeister, P., Pott, A., Verl, A.: Auto-calibration method for overconstrained cable-driven parallel robots. In: *ROBOTIK 2012, 7th German Conference on Robotics*. VDE, pp. 1–6 (2012)
5. Sandretto, J.A.D., Daney, D., Gouttefarde, M., Baradat, C.: Calibration of a fully-constrained parallel cable-driven robot. Ph.D. dissertation, Inria (2012)
6. Jin, X., Jung, J., Ko, S., Choi, E., Park, J.-O., Kim, C.-S.: Geometric parameter calibration for a cable-driven parallel robot based on a single one-dimensional laser distance sensor measurement and experimental modeling. *Sensors* **18**(7), 2392 (2018)
7. Bayani, H., Masouleh, M.T., Kalhor, A.: An experimental study on the vision-based control and identification of planar cable-driven parallel robots. *Robot. Auton. Syst.* **75**, 187–202 (2016)
8. Dallej, T., Gouttefarde, M., Andreff, N., Michelin, M., Martinet, P.: Towards vision-based control of cable-driven parallel robots. In: *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2855–2860. IEEE (2011)
9. Paccot, F., Andreff, N., Martinet, P.: A review on the dynamic control of parallel kinematic machines: theory and experiments. *Int. J. Robot. Res.* **28**(3), 395–416 (2009). <https://doi.org/10.1177/0278364908096236>
10. Bose, A.: How to detect and track red objects in live video in MATLAB, Octubre 2013. <http://arindambose.com/?p=587>
11. Forero, M.G., Herrera-Rivera, S., Ávila-Navarro, J., Franco, C.A., Rasmussen, J., Nielsen, J.: Color classification methods for perennial weed detection in cereal crops. In: Vera-Rodriguez, R., Fierrez, J., Morales, A. (eds.) *CIARP 2018*. LNCS, vol. 11401, pp. 117–123. Springer, Cham (2019). [https://doi.org/10.1007/978-3-030-13469-3\\_14](https://doi.org/10.1007/978-3-030-13469-3_14)
12. Gonzalez-Rodriguez, A., Castillo-Garcia, F., Ottaviano, E., Rea, P., Gonzalez-Rodriguez, A.: On the effects of the design of cable-driven robots on kinematics and dynamics models accuracy. *Mechatronics* **43**, 18–27 (2017). <http://www.sciencedirect.com/science/article/pii/S0957415817300156>