Chapter 13

Construction and Control of Parallel Robots

This chapter presents parallel manipulators, especially cable robots, as a new generation of cranes for which, all of six spatial degrees of freedom (DoFs) can be supported. The robot, its design and its manufacturing process are introduced from a mechatronics points of view. Actuating systems, required sensors, employed data transfer protocol and its designed graphical user interface (GUI) are discussed in this chapter for a parallel manipulator-like cable robot. The efficiency of the mentioned hardware setup and software installation, its advantages and drawbacks are focused and analyzed.

13.1. Introduction

Parallel robots, which are called kinematic machines, are a kind of closed-loop mechanism that have a high load carrying capability with an acceptable accuracy. Thus, a lot of applications can be mentioned for them in industrial environments like many kinds of simulators, cranes and automation lines. In a parallel mechanism, all of the actuators are connected to the end-effector and control it simultaneously. Moreover, for a parallel manipulator, there is a fixed platform by which the actuators are mounted and there is a moving platform, which is the end-effector of the robot, and the tool or load

---

Chapter written by Moharam HABIBNEJAD KORAYEM, Soleiman MANTEGHI and Hami TOURAJIZADEH.

is supposed to be installed on it and the manipulation is supposed to be performed through it.

There are different kinds of parallel robots depending on the type of actuators that are used by the robot to move the end-effector. The conventional type of parallel robots are made up of hydraulic actuators, and are called Stewart. The actuators can be also pneumatic or any combination of hydraulic, pneumatic and even electronic motors. A new generation of parallel robots that has been developed recently, is the cable robot in which the hydraulic jacks are replaced by cables. In these robots, the movement of the end-effector is provided by the aid of tensional force and the elongation of the cables, which are fulfilled using motors and pulleys. This idea was first proposed by National Institute of Standards and Technology (NIST) research center [ALB 93] and was developed rapidly in many academic and industrial centers. These robots have a good ratio of load to weight, high stiffness and perfect capability of assembling and de-assembling.

There are a number of robotic apparatus that use cables to position a platform. One of the most applicable types of underconstrained cable robots is cranes, which are widely used in industries. This idea is related to cable-suspended robots; the literature in this area started with the NIST RoboCrane [ALB 93]. Inspired by RoboCrane, many researchers have been involved with cable-suspended robots. A few of these have focused on cable-suspended crane devices. The typical cranes have the problem of stability due to their lack of actuating cables and cannot control the orientation of the suspended object. In the application of cranes, the designed cable robot is more trustworthy, though. This kind of underconstrained parallel mechanism, which is considered in this chapter, provides a practical mechanism for handling and positioning heavy loads while a larger useful workspace can be saved.

Aria *et al*. developed a seven-DoF, three-cable-suspended crane-type robot for an automobile assembly line, intended for heavy product assembly [OSU 93] and one mechanism with seven cables was also developed by Kawamura *et al*. [KAM 95]. Yamamoto *et al*. proposed a crane-type parallel mechanism with three active cables for handling heavy objects [YAM 99]. Shiang *et al*. presented a parallel four-cable positioning crane for offshore loading and unloading of cargo vessels under high seas [SHI 99]. Bostelman *et al*. changed the design from the symmetric RoboCrane structure to improve pose measurement and human interaction [BOS 95]. An experiment facility was developed at the University of Delaware that consists of a six-DoF cable-suspended robot with a reference governor [OH 05].

Another cable robot, which was designed to develop the cranes' applications, is the one that was manufactured at Pennsylvania University. This crane in contrast to the simple previous cranes has more cables to control the end-effector, which reduces the amount of end-effector's oscillations. There is no framework for this robot and all of the motors are mounted on the end-effector [CHA 04]. Two cable robots with an application in rehabilitation have been developed at Padua University. NeReBot is a three-DoF cable-based robot, designed for the treatment of patients with stroke-related paralysis or upper limb paretic during the acute phase [ROS 07]. Although the first clinical tests showed encouraging results in functional outcome, the robot presented some limitations. Therefore, a new cable-based robot was designed. The MariBot is a five-DoF serial–parallel robot. Like the NeReBot, three cables allow the movement of a patient's arm in the thee-dimensional space by means of a splint. However, an additional two-DoF serial overhead structure is used to adapt a cable configuration to the current position of the patient's arm. This solution results in a significant increment of the workspace volume [ROS 06].

The kinematic calibration of a cable-actuated parallel robot is also discussed in [VAR 06]. Available redundant sensing of joint displacements makes it feasible to consider central linkage and cable actuators of the robot one at a time for calibration purposes. Dynamic characteristics of this type of mechanism such as the manipulation problem of load by multiple cables have been well studied [ALP 01, KOR 09].

In this chapter, formulation, design and manufacturing of these new type of parallel robots, that is cable suspended robots, are presented as a complicated sample of a mechatronic mechanism. First, the overall scheme of its mechanical structure is presented. Afterwards, its formulation including kinematics, kinetics and control is derived. Then, its corresponding hardware setup and installation including actuators, sensors and data transfer protocol is explained. The required electrical boards and derivers and their corresponding calibrations are presented. Finally, its related GUI and simulator, which are programmed in LabView, are described as the required software implementation.

13.2. A parallel robot mechanism

An underconstrained cable robot is designed and manufactured at Iran's University of Science and Technology (ICaSbot), which supports six DoFs including three translational and three rotational movements of the

end-effector with the aid of six active cables and six DC motors. A setup of the above-mentioned robot can be seen in Figure 13.1 [KOR 11a]. The experimental tests are conducted on this robot in order to verify the efficiency of the proposed force measurement sensors and used them in the mentioned applications. A hardware setup and supporting software are designed and programmed for evaluating the required parameters of load cells.

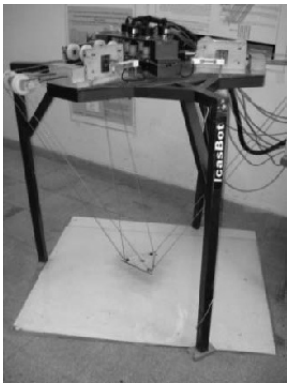


Figure 13.1. *The ICaSbot robot*

Six DC motors are responsible for supporting six suspended cables. The change in length of these cables controls three translational and three rotational movements of a triangular shape end-effector. The cable, which transfers the weight load, is passed over the pulley and is wrapped around the shaft that is controlled by the aid of motors. Other specifications of the robot are listed in Table 13.1. The assembled system of encoder, shaft and motor is shown in Figure 13.2 [KOR 11a].

13.2.1. *Mechanical structure*

The main part of the robot mechanical features consists of an end-effector supported by six actuating cables, which can be retracted and extracted through their related pulleys. These pulleys are also connected to six drums and corresponding motors. The encoders are attached to the couples of the drums and motors. They are responsible for counting the rotation of each motor. Figure 13.1 shows the scheme of the Iran University of Science and Technology (IUST) built cable robot and Table 13.1 shows its main geometrical specifications. One end of the cables is attached to the pulleys

whereby the platform is suspended in the workspace. The other end is anchored to the end-effector and the cables are made taut by the actuator system. Each of them is adapted for generating a position signal that is responsible for retraction and extension of the cables, and then the central control system is able to provide commands for translating and orienting the platform with the aid of combining these individual signals related to each cable [KOR 11a].

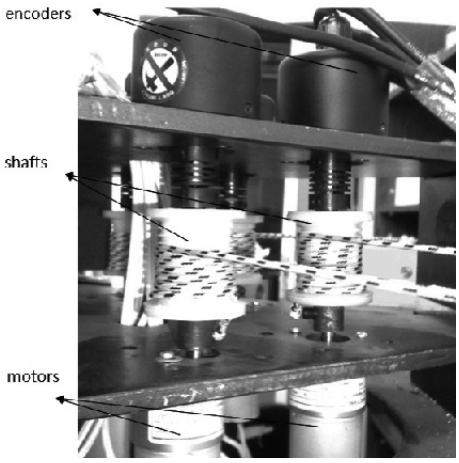


Figure 13.2. *The assembled system, from top to bottom: (1)encoder, (2) shaft, (3) motor*

Robots Specifications	Value
Height	120 cm
Variable side length of base triangle	100–120 cm
Weight	350 kg
End-effector	
Side length of base	17 cm
Thickness	8 mm
Weight	1,100 g

Table 13.1. *The specifications of the robot [KOR 11a]*

This section also presents a system of Cartesian pose measurements using a camera installed on the robot structure. The proposed system is relatively simple and economical [KOR 11a].

The actuator system is located on the base platform and includes three separate important parts, that is: motors, encoders and drums. Therefore, the cables can be retracted or extended by this actuator system in order to suspend, translate and orient the platform as desired [KOR 11a].

On the other hand, the control system is designed so that the platform could be translated and rotated in the workspace by controlling the length of the cables extending from their respective pulleys. A designed electronic board controls motion of each cable by calculating and imposing its desired length. Using at least six cables, the platform has six DoFs: translation in three axes and rotation about each of the foregoing three axes. Numerous types of end-effectors can be mounted on the platform, thus enabling the robot to perform a wide variety of tasks. The robot as described here does not require a separate rotational device on the platform to orient the end-effector in the workspace. Figure 13.3 illustrates the sequential procedure of power transmission of the manufactured robot and Figure 13.4 is a preview of the used end-effector with its lasers connected to it, which illustrates the position of the moving platform with the aid of a vision system [KOR 11a].

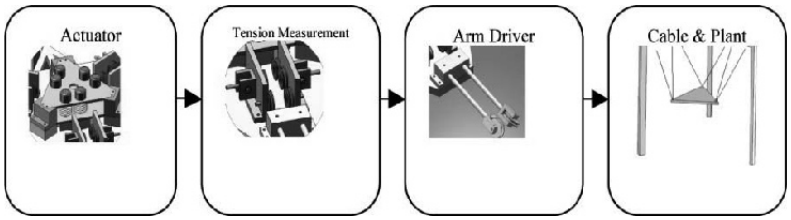


Figure 13.3. *Sequential procedure of power transmission [KOR 11a]*

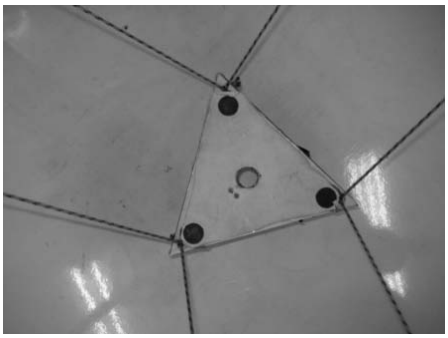


Figure 13.4. *Used end-effector*

13.2.2. *Modeling and Formulation*

13.2.2.1. *Kinematics*

The overall scheme of a ICaSbot cable-suspended robot is shown in Figure 13.5 [ALP 01].

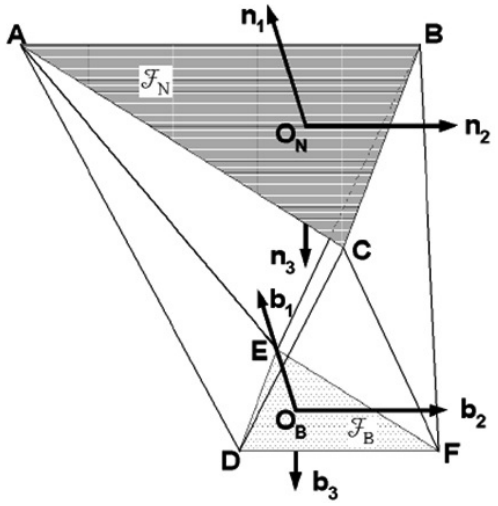


Figure 13.5. *Schematic of six cable robots [ALP 01]*

where x_m, y_m,$z_m(m)$ are translational coordinates of the end-effector and $\psi,\theta,\phi(\text{rad})$ are its rotational coordinates. Presenting the cables' elongation with L, the following relation can be established between the elongation of the cables and end-effector's velocity [IRA 09]:

$$\dot{L} = J\begin{bmatrix}\dot{X}\\ \omega\end{bmatrix} = J\left(\dot{x}_m,\dot{y}_m,\dot{z}_m,\dot{\psi},\dot{\theta},\dot{\phi}\right)^T \quad [13.1]$$

$$J = \left[\frac{\partial L_i}{\partial X}\right] \quad [13.2]$$

where J is the Jacobian matrix. On the other hand, the following equation can be expressed between the cables' elongation L and pulleys' angular rotation β:

$$\beta = \begin{Bmatrix} \beta_1(X) \\ \cdots \\ \beta_n(X) \end{Bmatrix} = \frac{1}{r} \begin{Bmatrix} L_{01} - L_1(X) \\ \vdots \\ L_{0n} - L_n(X) \end{Bmatrix} \quad [13.3]$$

where r is the pulley radius. As is shown in Figure 13.6, calculating pulleys' angular velocity from a path is called inverse kinematics and obtaining the path from a pulleys' angular velocity is called direct kinematics [KOR 11d].

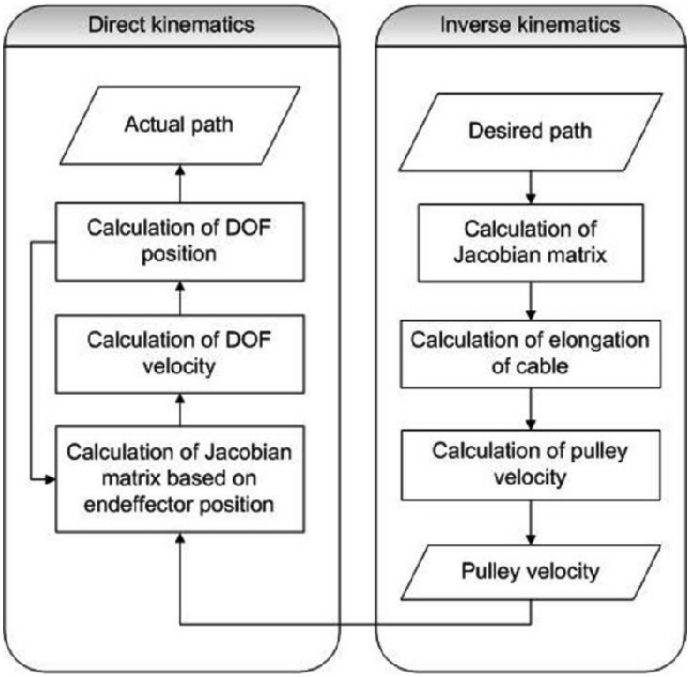


Figure 13.6. *Kinematics chart [KOR 11d]*

13.2.2.2. *Dynamics*

The use of a Newton–Euler equation leads the system to have the following dynamic equation [IRA 09]:

$$D(X)\ddot{X} + C\left(X, \dot{X}\right)\dot{X} + g(X) = -J^T(X)T \quad [13.4]$$

where:

$$D=\begin{bmatrix} mI_3 & 0 \\ 0 & P^T IP \end{bmatrix}; C=\begin{bmatrix} 0_3 \\ P^T\{I\dot{P}\dot{o}+(P\dot{o})\times I(Po)\} \end{bmatrix}; g=\begin{bmatrix} 0 \\ 0 \\ -mg \\ 0_3 \end{bmatrix};$$
$$J=\left[\frac{\partial q_i}{\partial x_j}\right]_{i\times j}; P=\begin{bmatrix} 1 & 0 & -\sin\theta \\ 0 & \cos\psi & \sin\psi\cos\theta \\ 0 & -\sin\psi & \cos\psi\cos\theta \end{bmatrix}; \dot{o}=\begin{bmatrix} \dot{\Psi} \\ \dot{\theta} \\ \dot{\varphi} \end{bmatrix} \quad [13.5]$$

T is the vector of the cables tension, D(x) is the inertia matrix, C(x,ẋ) is the vector of velocity terms, g(x) is the gravity vector, J is the conventional parallel manipulator Jacobian, X is the vector of DoFs of the system, *m* is the mass of the end-effector, *I* is the moment of inertia of the end-effector and *q* is the length of the cables. Also, the dynamics of the motor is as follows:

$$T=\frac{1}{r}\left[\tau-j\left(\frac{d}{dt}\left(\frac{\partial\beta}{\partial X}\right)\dot{X}+\frac{\partial\beta}{\partial X}\ddot{X}\right)-c\frac{\partial\beta}{\partial X}\dot{X}\right] \quad [13.6]$$

where j is the matrix of rotary inertia of the motors, c is the viscous friction matrix of the motors, $\dot{\beta}$ is the vector of angular velocity of the motors and τ is the vector of motors torque. By coupling these two dynamics, we have:

$$D(X)\ddot{X}+C\left(X,\dot{X}\right)\dot{X}+g(X)=-J^T(X)\frac{1}{r}$$
$$\left[\tau-j\left(\frac{d}{dt}\left(\frac{\partial\beta}{\partial X}\right)\dot{X}+\frac{\partial\beta}{\partial X}\ddot{X}\right)-c\frac{\partial\beta}{\partial X}\dot{X}\right] \quad [13.7]$$

As is shown in Figure 13.7, calculating motor's torque from the given path is called inverse dynamics and obtaining the actual path from the applied torque is called direct dynamics.

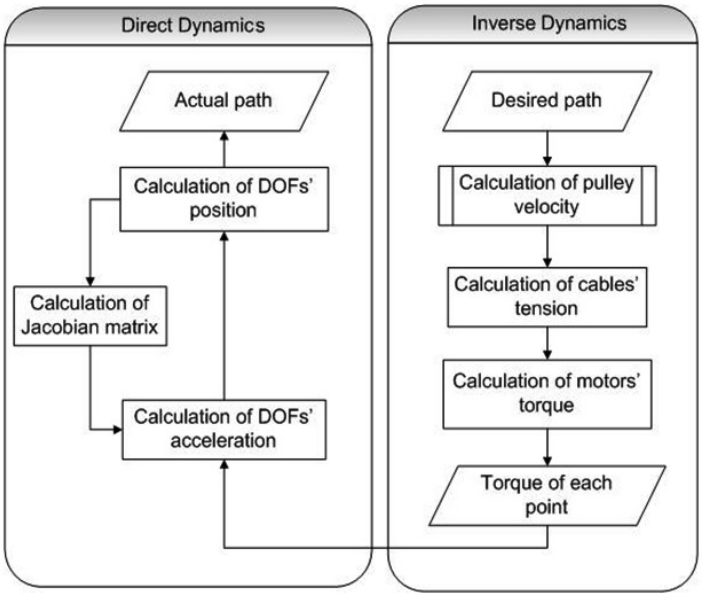


Figure 13.7. *Dynamics chart [KOR 11d]*

13.2.2.3. *Control Scheme*

[KOR 12c] using the feedback linearization method results in the following required cables' tension:

$$T_i = \{ J^{-1}(D(X)\nu + C(X, \dot{X}) + g(X))\}_i; \quad i = 1, \ldots, 6 \qquad [13.8]$$

ν is the control input of the feedback linearization and can be substituted as below to create a controllable error equation:

$$\nu_i = \ddot{X}_d + K_{iD}(\dot{X}_d - \dot{X}_a) + K_{iP}(X_d - X_a); \quad i = 1, \ldots, 6 \qquad [13.9]$$

where K_{iD} and K_{iP} are controlling gains of derivative and proportional errors, respectively, X_d is the desired trajectory of the robot and X_a is its actual value.

So, the required cable tension is calculated here using inverse kinetics of the robot together with the feedback linearization method and the required

angular velocity of the motor is evaluated using inverse kinematics of the robot. It is now possible to study the dynamics of the motor in order to evaluate its required Pulse Width Modulation (PWM). Therefore, the dynamic of the motor can be stated as [KOR 12a]:

$$\tau_i = rT + J\ddot{\beta} + c\dot{\beta}; \quad i = 1,\ldots,6 \qquad [13.10]$$

Here, the desired motor torque and angular velocity are calculated as in the previous section using the inverse dynamic of the motor, which results in the required torque of the motors (computed torque method + feedback linearization). So, substituting equation [13.8] in equation [13.10] results in:

$$\tau_i = rJ^{-1}(D\nu + C + g) + J\ddot{\beta} + c\dot{\beta}; \quad i = 1,\ldots,6 \qquad [13.11]$$

Generally, the overall scheme of controlling the robot in a closed-loop way is shown in Figure 13.7. It can be seen that two sequential closed-loop controllers are involved: motor controller and end-effector controller. The angular velocity of the motors and the torque of the motor are controlled using encoder and load cell sensors for each 0.01 s in the inner loop of robot control while the position of the end-effector is controlled for each 0.1 s using the position feedback of lasers and camera sensors.

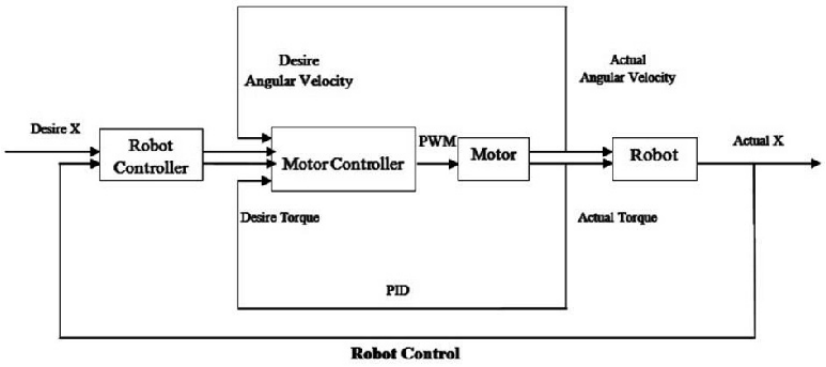


Figure 13.8. *Diagram of the proposed control system for ICaSbot*

13.3. Actuators

13.3.1. *Motor*

Six motors were selected for the robot whose specifications are listed in Table 13.2. Each motor is connected to an encoder, which has a high resolution precision, by a 3 cm-diameter circular shaft. The cable that transfers the weight load is passed over the pulley and wrapped around the shaft. As Table 13.2 shows, this 12 V 17 W geared motor runs at 150 rpm (in free running condition).

Motor's specifications	Value
Mark	Retarding gear motor
Model	1.61.070.304 Buhler
Nominal voltage	12 V
No load speed	150 rpm
Stall torque	1.7 N·m
No load current	0.3 A
Stall current	1.2 A
Reduction ratio	1:175
Weight	220 g

Table 13.2. *DC motor's specifications*

The dynamic of the motor can be stated [13.11], where the applied torque of the motor is τ, J is the rotary inertia of the motor, r is the radius of the drum, $\dot{\beta}$ is its angular velocity and c is the viscose damping of the motor. Here, the desired motor torque and angular velocity are calculated as in the previous section using the inverse dynamic of the motor, which results in the required torque of the motors (computed torque method + feedback linearization).

Here, DC motors are used to provide the desired calculated torque and angular velocity of the motors. The goal of DC motor modeling is to extract a proper formula presenting a relation between the armature voltage and the relative produced torque. Figure 13.10 shows the basic circuit diagram of a DC motor.

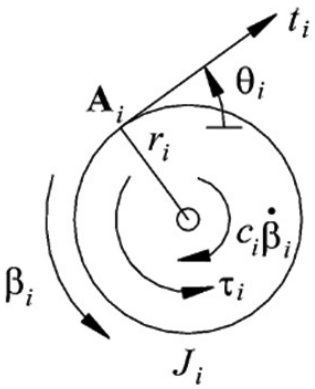


Figure 13.9. *Scheme of the motor [ALP 01]*

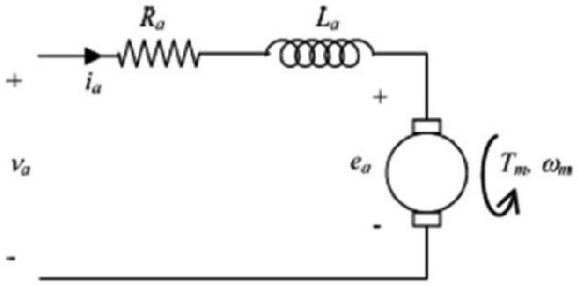


Figure 13.10. *Circuit diagram of a DC motor*

The general equations of Direct Current Permanent Magnet (DCPM) motor are:

$$v_a = R_a i_a + L_a(\frac{di_a}{dt}) + e_a \qquad [13.12]$$

$$e_a = K_m \omega_m \qquad [13.13]$$

$$\tau_m = K_m i_a \qquad [13.14]$$

where v_a is armature voltage, R_a is the resistance of armature wire, L_a is armature inductance, i_a is armature current, e_a is emf (induced voltage of

electromotor force), K_m is constant of torque motor, τ_m is produced torque and ω_m is angular velocity of the DC motor [KOR 12a]. So the required voltage of the motor can be computed based on desired torque and speed of the motor using equation [13.12], where e_a is substituted by equation [13.13] and i_a is substituted by equation [13.14]. Finally, this voltage can be implemented on the motor using a PWM method that has the least loss of energy.

$$\text{PWM} = v_a / v_{\max} \qquad [13.15]$$

where $v_{\max}$ is the maximum voltage of the motor. However, this is possible if and only if the exact model and parameters of the motor are available and it is obvious that in many conditions these parameters are not available. Finally, PWM is the required pulse width modulation of the DC motor.

The required calculated torque of the motors is provided here by the aid of the PWM method. The feed-forward term of PWM is calculated using the inverse dynamics while the errors are compensated using a PD controller on the velocity of the motors:

$$\begin{aligned} PWM = PWM_f + K_d \times (\dot{\beta}_d - \dot{\beta}_a) + K_p \times (\beta_d - \beta_a) \\ + K'_d \times (\dot{\tau}_d - \dot{\tau}_a) + K'_p \times (\tau_d - \tau_a) \end{aligned} \qquad [13.16]$$

where d indicates the desired output and a is the actual output. The coefficients K_d and K_p are controlling gains of the proportional and the derivative error of the motor angular velocity and K'_d and K'_p are the controlling gains of motor torque. Our primary goal is to show the actual amount of torque and tension of the cables to be used as a feedback in controlling the torque of the motors in order to increase the accuracy. Therefore, not only can the PWM be improved using the encoder feedback of kinematic response of the motor speed (which helps the system to achieve the desired speed), but also its desired torque can be provided using the feedback of the actual motor torque.

In this test, the PWM method is used for regulating the input voltage. In the PWM method, the desirable pulse width is amplified by a motor driver IC, L298. The electronic circuit amplifies the PWM signal generated from the data acquisition board. The lower bound of the PWM is related to 0 V and the

upper bound is related to 12 V. In general, the larger command signal produces the wider pulse. The motor-driven circuit is shown in Figure 13.11.

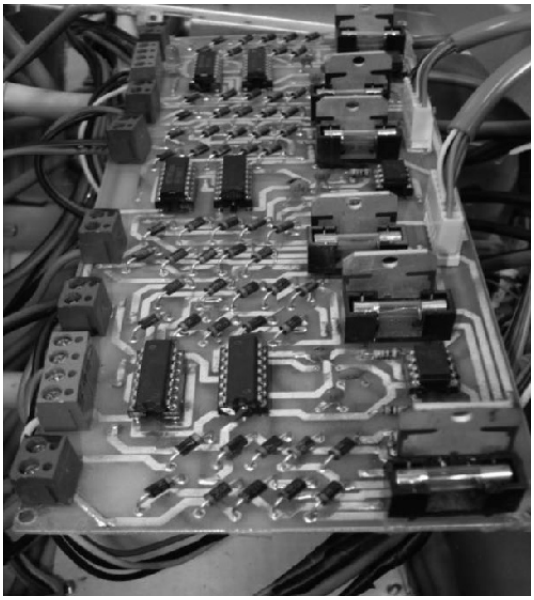


Figure 13.11. *Motor driver circuit*

13.3.2. *Cables*

Moreover, the used cables are not completely solid and can be easily vibrated in both longitudinal and lateral dimensions. It is obvious that these kinds of vibrations can enormously affect the desired position and rotation of the end-effector. Configuration of cables and their connection points to the end-effector are shown in Figure 13.4 while their specifications are explained in Table 13.3.

Cable's specification	Value
Diameter	3 mm
Elasticity module	5.6×10^6 N/m^2

Table 13.3. *Cable specifications*

During a robot operation for carrying a payload in a workspace, the end-effector position measurement can be considered in some ways. Using encoders for the motors on the active cables can result in some errors related to cable sag and stretch. The ICaSbot design allows the moving cable to rotate more than one time around a same line of the drum, resulting in further measurement error. The motors actuate the active driven cables and the control system commands the motors.

13.4. Sensors

In order to control the robot, there are three different kinds of sensors, which include encoders, load cells and also the combination of a camera and optical sensors.

13.4.1. *Motor speed sensor*

The incremental encoders used in this hardware setup have resolution precision, 4 × 600 pulses per revolution; two output signals of encoders have 90° phase shift. The encoder's specifications are shown in Table 13.1.

Encoder's specifications	Value
Mark	Autonics
Model	E50S8-600-3-T-24
Resolution	600
Output phase	A, B, Z
Control output	Totem pole
Power supply	12–24 V
Maximum allowable revolution	5,000 rpm

Table 13.4. *The specifications of the selected encoder*

There is a possibility of connecting the encoders to the data acquisition board directly without using the designed intermediate printed circuit board (PCB) and estimating the angular velocity and the motion direction using the phase shift between two output signals; however, we prefer to use the

designed intermediate PCB in order to increase the precision and reduce the amount of mathematical calculation.

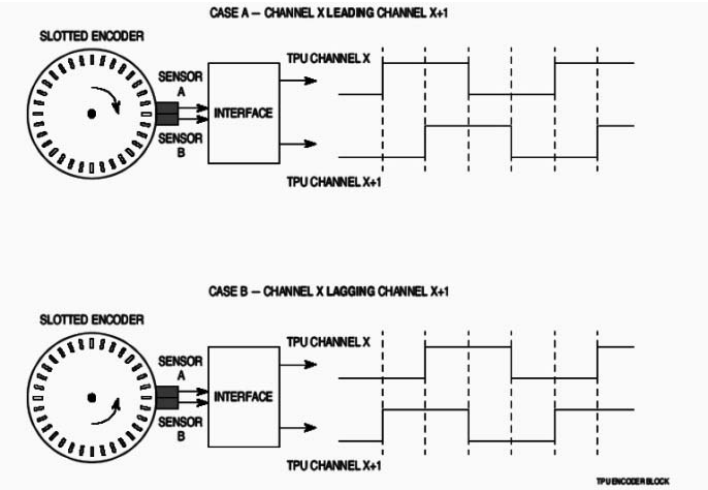


Figure 13.12. *Output of encoder pulse*

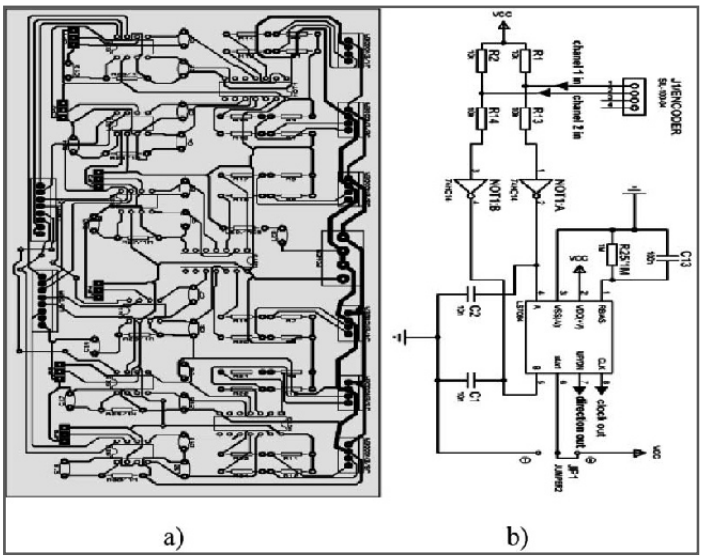


Figure 13.13. *a) Circuit diagram of the PCB reads rotary encoder's output and b) its layout*

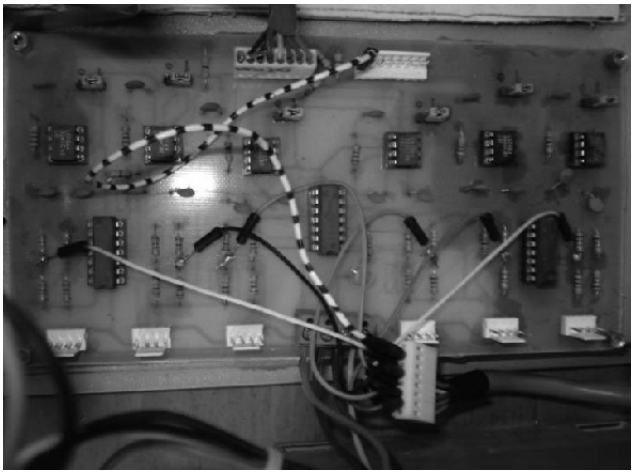


Figure 13.14. *Rotary encoder interface circuit*

The designed PCB increases the reading precision from 600 to 2,400 pulses per revolution and the output signal received by the data acquisition board is a pulse that varies between 0 and 1 V. By using the software setup, the received signal can be read. The electronic circuit read rotary encoder's output and its layout is shown in Figure 13.13. The PCBs, data acquisition board's ports and the power supply are installed in a hardware setup box. Figure 13.13 shows a schematic circuit and PCB read rotary encoder's output.

13.4.2. *Force sensor*

The torque measurement mechanism is designed in such a way that the force applied to the pulley is transferred to a load cell by a shaft. Cable tension feedback can be evaluated accurately for precise movements. As is shown in Figure 13.15, the mechanism consists of three pulleys that are installed on the structure by means of three shafts. The two lateral pulleys are fixed and they can only rotate to transfer the cables, whereas the central pulley can not only rotate but can also move vertically up to 5 cm. This pulley is installed on the force sensor by bolts. When two lateral pulleys transit the cables, the central pulley can move. The shaft connected to the force sensor can transfer the applied force to the sensor. The more the force, the greater the pulley movement and the greater the force sensor output.

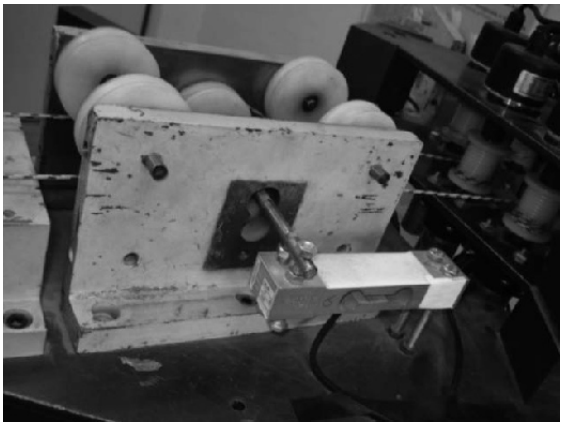


Figure 13.15. *Load cell as the used tension sensing device*

The used force sensor (load cell) type is a single-point aluminum alloy. As is shown in Figures 13.15 and 13.16, one side is connected to a pulley's shaft while the other side is connected to the end-effector by means of a bolt. This sensor consists of four strain gauges that convert the vertically applied force to voltage. Each resistance bridge (Figure 13.17), including strain gauges, magnifies the voltage up to 5–10 V DC. The capacity of each sensor is up to 5 kg and the accuracy is 0.1 V + 0.2 mV/V [TAJ 11]. Specification of each load cell is listed in Table 13.2.

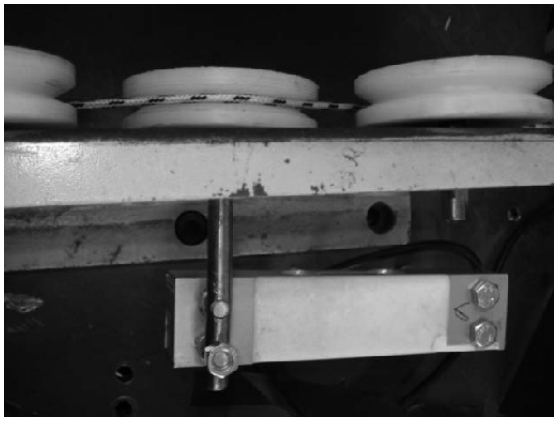


Figure 13.16. *Force sensor that is connected to torque measurement mechanism*

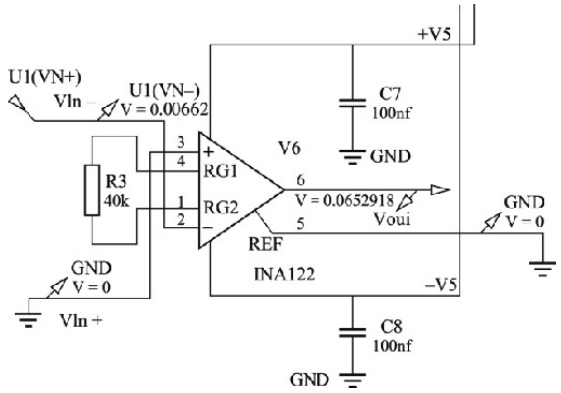


Figure 13.17. *The schematic circuit of the force sensor's output amplifier*

Model	Single Point 1661
Material	Aluminum Alloy
Surface	anodized treatment
Size	250×350 mm^2
Capacity	5 kg
Excitation Voltage	5–10 V dc
Output sensitivity	2.0 ± 0.1 mv/v

Table 13.5. *Specifications of the force sensor*

An "Advantech-PCI-1780" board is applied for reading the data of the motor and the encoder sensor (Figure 13.8). This board has eight channels and uses an external electronic board (PCLD-8710-AE) and it has specific cables (PCI-10168) to reduce the noises.

A proper data transfer system is required to manage the data transfer rate. Hence, an A/D sampling card is used. The output voltage of the load cell is approximately 0.2 mV/V. In order to measure this voltage, noise reduction and voltage amplification are necessary. To overcome this problem, an amplifier and noise reducer electronic board is constructed. This board consists of an amplifier IC (INA114) possessing a negative feedback to reduce noises. Besides, the negative feedback INA114 has some advantages including a high signal-to-noise ratio (SNR), high and variable gain and low

cost. The simplified schematic circuit of the electronic board and its PCB is illustrated in Figure 13.18. Here, *U* are the amplifier ICs of INA114 and the reducer of the noises, *Cs* are installed capacitors, *J*3 is the input voltage of the board, *J*1 and *J*2 are output channels to the data card and, eventually, *R*1–*R*6 are the employed regulator resistance for the used ICs.

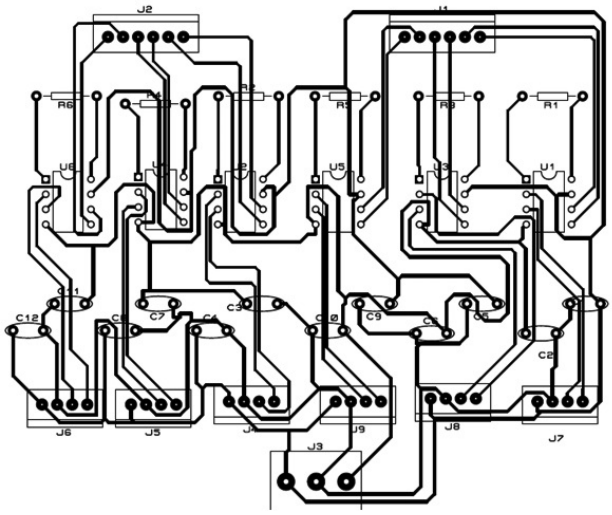


Figure 13.18. *The PCB of force sensor's output amplifier circuit*

Equations [13.17] and [13.18] calculate the INA114's output voltage and gain, respectively. The gain value depends on the resistance gain (RG) and it should be regulated from the excitation table in the INA114 datasheet. Vo is the force sensor output voltage; V_{IN}^{+} and V_{IN}^{-} are force sensor excitation voltage and G is the force sensor amplification value that varies with variation in resistance on board.

$$V_O = G \times (V_{IN}^{+} - V_{IN}^{-}) \qquad [13.17]$$

$$G = 1 + \frac{50\ \text{k}\Omega}{R_G} \qquad [13.18]$$

The final PCB board used for recording the load cell's output is shown in Figure 13.19.

Figure 13.19. *The final PCB board for recording the loadcell output*

To control and record data, a 100 kHz analog sampling card (Advantech PCI 1711L) is used, which transfers the force sensor output to a PC. This card is an interface between software and amplifier board [TAJ 11].

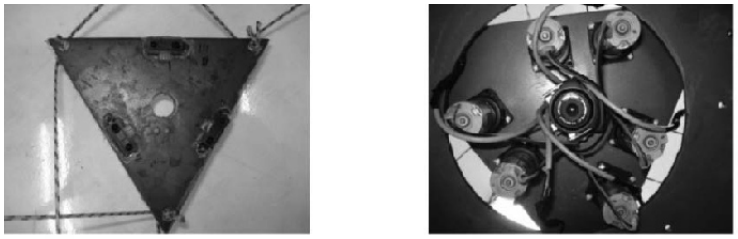
13.4.3. *Position sensor*

The third type of sensors is the combination of a camera and optical sensors.

A camera is installed at the top of the end-effector, which is able to record the virtual (raw) data of the planar coordinate of the end-effector. The specifications of the used camera are listed in Table 13.6.

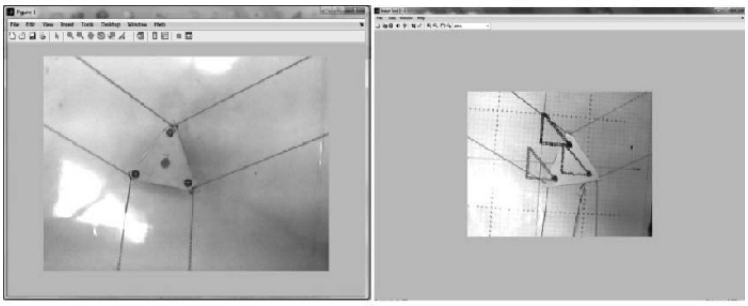
Camera parameters	Value
Category	Webcam
Video capture (4:3)	640 × 480
Image capture	640 × 480
Diagonal field of view (FOV)	68°
Connection type	USB 2.0
Width	40.4 mm (1.59 in)
Depth/length	68.5 mm (2.69 in)
Weight	88 g (3.12 oz)

Table 13.6. *Specification of the used camera*



a) b)

Figure 13.20. *a) Optical sensors. b) Camera*



a) b)

Figure 13.21. *a) Image preprocessing and b) image processing*

Two series of calibration are required for the recorded data of the camera to achieve the real planar coordinate of the end-effector. The first one is a planar calibration of the camera in which the origin of the coordinate system is determined for a specific altitude of the end-effector. The second one is perspective calibration that needs the instantaneous altitude and orientation of the end-effector. These data are fed by the aid of installed lasers that have been discussed earlier. A chess-shaped plate is used to perform the planar calibration of the camera. Two predefined points are specified in this plate for which their real planar coordinates are known. By running the program, the positions of three markers are first selected manually by clicking on the screenplay of the software. Then, the software starts recording the position of the three markers that have been selected.

Next, the program converts the raw data achieved in image processing to real data by using a perspective equation of the installed camera. For

converting raw data, the height of the markers are required, which are obtained by three laser sensors installed on the bottom face of the end-effector.

$$L_{\text{real}} = \frac{L_{\text{virtual}} \times (h + f)}{f} \qquad [13.19]$$

Figure 13.22 is used for converting the virtual or raw coordinate data to real data. Here, h is the distance between the camera and the point, and f is canonical distance of camera, which can be obtained by an experiment as discussed below.

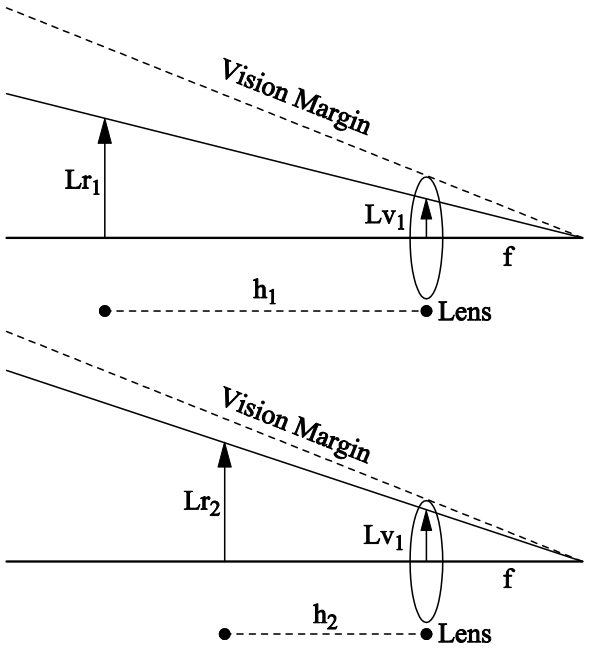


Figure 13.22. *The scheme of the reflection of the object on the lens of the camera (perspective effect)*

For obtaining the f value of the camera, a specific object should be placed at two predefined distances h_1 and h_2 from the camera lens. Using [13.20], the physical parameter of the camera, f can be obtained. By substituting the obtained f in [13.20], the perspective equation is calibrated.

$$\begin{cases} \dfrac{L_{V1}}{f} = \dfrac{L_{r1}}{h_1 + f} \\ \dfrac{L_{V2}}{f} = \dfrac{L_{r2}}{h_2 + f} \end{cases} \quad [13.20]$$

Three laser sensors of type GP2D12 are installed on the each vertex of the end-effector triangle. The specifications of the used optical laser sensor are listed in the Table 13.7.

Optical laser sensor parameters	Value
Output	Analog
Effective range	10–80 cm
LED pulse cycle duration	32 ms
Typical response time	39 ms
Typical start up delay	44 ms
Average current consumption	33 mA
Detection area diameter at 80 cm	6 cm
Supply voltage	–0.3 to +7.0 V
Output terminal voltage	–0.3 to Voltage at the Common Collector (VCC + 0.3) V
Operating temperature	–10 to +60°C
Storage temperature	–40 to +70°C

Table 13.7. *Specification of the used optical laser sensor*

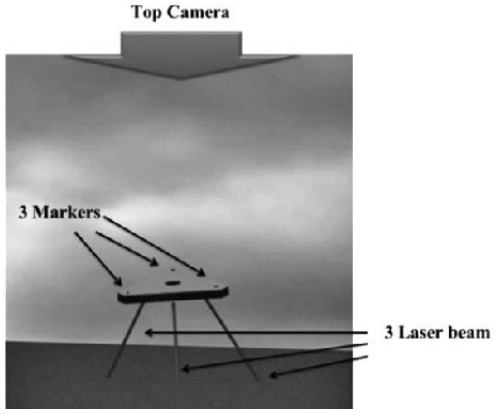


Figure 13.23. *Symbolic scheme of laser beam*

This sensor has one light emitter and one light receiver, which are placed on the base. The light emitter unit emits a light beam with specific frequency toward the object and receives that specific light reflection by light receiver unit. Three laser sensors are place under the end-effector aiming at the ground.

Sensors are calibrated manually, since it was observed that the sensors do not confirm the calibration function of their catalog (Figures 13.24 and 13.25).

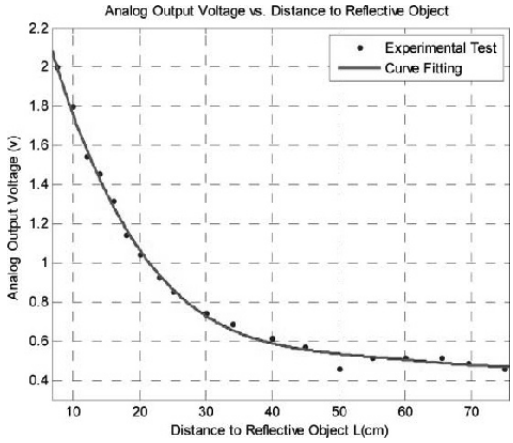


Figure 13.24. *Calibration test related to laser sensor experiment data*

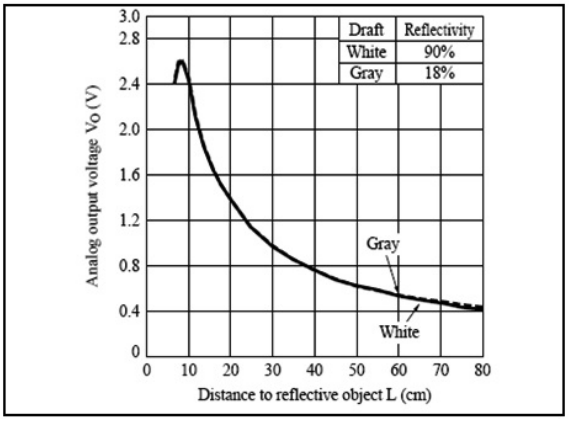


Figure 13.25. *Calibration test related to laser sensor catalog data*

In this step, three lasers are installed at three tips of the end-effector with a specific angle by which their laser beams eventually configure a half pyramid. It is required first to evaluate the translational position of the vertexes of the end-effector, and then the rotational movement can be determined in an online way. Since the length of the edges of the pyramid can be determined by the aid of lasers data, the geometry of the pyramid will be known completely by which the altitude of three tips can be calculated easily by the aid of following formula.

$$
\begin{aligned}
&X_\alpha = sin(\zeta)L_\alpha\, cos(30) + d/2\\
&Y_\alpha = -sin(\zeta)L_\alpha\, sin(30) - d/(2\sqrt{3})\\
&Z_\alpha = -cos(\zeta)L_\alpha\\
&X_\beta = -sin(\zeta)L_\beta\, cos(30) - d/2\\
&Y_\beta = -sin(\zeta)L_\beta\, sin(30) - d/(2\sqrt{3})\\
&Z_\beta = -cos(\zeta)L_\beta\\
&\\
&X_\delta = 0\\
&Y_\delta = sin(\zeta)L_\delta + d/(\sqrt{3})\\
&Z_\delta = -cos(\zeta)L_\delta
\end{aligned}
\qquad [13.21]
$$

where X,Y,Z are in local coordinate system, d is the length of the end-effector triangle edge (Figure 13.27) and ζ, which is the angle between the direction of the laser beams and local Z axis, is equal to 30° for the mentioned installation of the ICaSbot. Equation [13.11] is presented in a local coordinate system. By calculating the local coordinate of three mentioned intersection points, the equation of the plane (Figure 13.28), which includes these three points, can be obtained:

$$a(X - X_0) + b(Y - Y_0) + c(Z - Z_0) = 0 \qquad [13.22]$$

where a, b and c are the resultant vectors of the cross products of $\overrightarrow{\alpha\beta} \times \overrightarrow{\alpha\delta}$ and X_0,Y_0,Z_0 are the local coordinates of one of the tips (e.g. α point). Now by substituting the local X,Y of each marker point (R, L and U points) in this plane formula, the vertical distance of each tip with respect to the end-effector angle can be easily calculated, which is called Z. Finally, the vertical altitude of each tip from the ground (h) can be determined by multiplying this distance by $cos\,\gamma$.

$$h = z\cos(\gamma) \quad \cos\gamma = \frac{c}{\sqrt{a^2+b^2+c^2}} \qquad [13.23]$$

where γ is the angle of the normal vector of the end-effector compared to vertical axis. Now the scaling of the image processing can be completed by having the altitude of each tip of the end-effector. The real coordinate of each tip can be obtained by the aid of the raw data of the camera together with the resultant altitude of each tip. Finally, the coordinate of the end-effector centroid can be evaluated by the aid of the real calculated coordinate of three tips of the end-effector:

$$\hat{X}_G = \frac{\hat{X}_L + \hat{X}_R + \hat{X}_U}{3} \quad \hat{Y}_G = \frac{\hat{Y}_L + \hat{Y}_R + \hat{Y}_U}{3} \quad \hat{Z}_G = \frac{\hat{Z}_L + \hat{Z}_R + \hat{Z}_U}{3} \qquad [13.24]$$

where $\hat{X},\hat{Y},\hat{Z}$ are in the global coordinate system and the index G denotes the centroid coordinate of the end-effector. Alternatively, the global angles of the end-effector can be calculated using [13.24] and the following formulations:

$$A = \frac{a}{\sqrt{a^2+b^2+c^2}}$$
$$B = \frac{b}{\sqrt{a^2+b^2+c^2}} \qquad [13.25]$$
$$C = \frac{c}{\sqrt{a^2+b^2+c^2}}$$

Considering a 30° angle for each laser with respect to vertical axis (Figure 13.27), the length of edges $L_\alpha; L_\beta; L_\delta$ is obtained (Figure 13.26). A local coordinate is set in the centroid of the end-effector and the local *X*,*Y*,*Z* of the points of intersection of the lasers with the ground can be calculated by measuring the angle of the lasers with respect to the vertical axis.

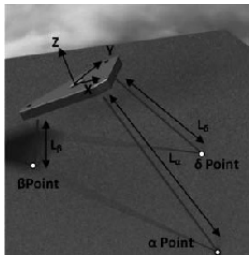


Figure 13.26. *Length of laser beams and their related intersection points with ground*

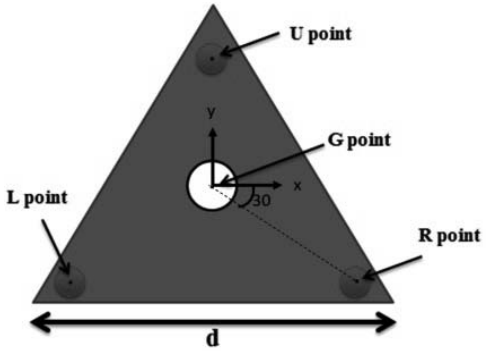


Figure 13.27. *Markers' points and center point of the end-effector*

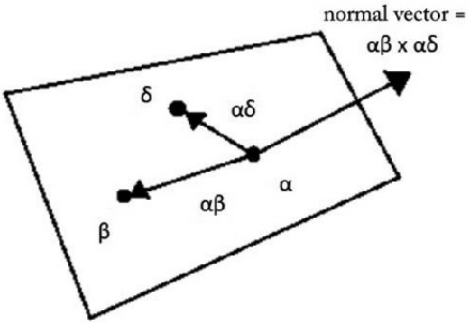


Figure 13.28. *The equation of a plane passing through three points*

Now, as the exact translational position of the three vertexes of the end-effector is determined, it is possible to calculate the instantaneous rotational condition of the end-effector, which is done in an online way. The angles of the end-effector can be calculated as:

$$\text{Yaw} = \tan^{-1}\left(\frac{A}{-B}\right) \qquad [13.26]$$

$$\text{Pitch} = \frac{\sqrt{A^2 + B^2}}{C} \qquad [13.27]$$

Moreover, the roll can be obtained by the aid of vision data as follows:

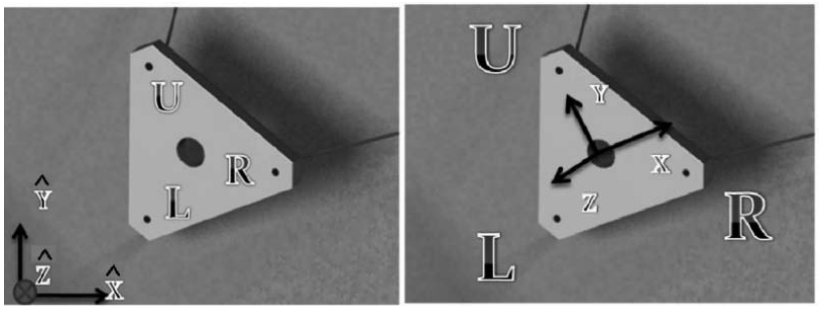


Figure 13.29. *Global coordinate and local coordinate*

Since the global $\hat{X}$, $\hat{Y}$ coordinates of three markers are known by the use of the camera (the global coordination of U-marker: $(\hat{X}_U, \hat{Y}_U)$ and also for R-marker and L-marker: $(\hat{X}_R, \hat{Y}_R)$ and $(\hat{X}_L, \hat{Y}_L)$), we can evaluate the roll by the aid of following equation:

$$\text{Roll} = \tan^{-1}\left(\frac{\hat{Y}_R - \hat{Y}_L}{\hat{X}_R - \hat{X}_L}\right) \qquad [13.28]$$

While the robot is moving, the data of $\hat{X}$, $\hat{Y}$ and $\hat{Z}$ position and also three axial orientations are recorded in a matrix in Matlab®'s workspace to be used as feedback control for each time step. The whole explained formulation can also be developed for the environmental situations in which the ground is not flat. This purpose is easily possible by either replacing the ground with a new flat surface or extracting the predefined surface formulation of the ground. Also, this limitation that the end-effector needs to be always parallel to the ground is canceled here since the procedure is done for all of the three vertexes of the end-effector and is analyzed in the mentioned formulations.

13.5. Data transfer protocol

Also, two kinds of data card of Advantech Company are used to control the robot.

Figure 13.30. *PCI 1711 Advantech data acquisition board*

Figure 13.31. *PCI 1780 Advantech data acquisition board*

The GUI software setup that reads the encoders' output signals using the data acquisition board is programmed in LabVIEW environment. A panel of the GUI software setup is shown in Figure 13.32.

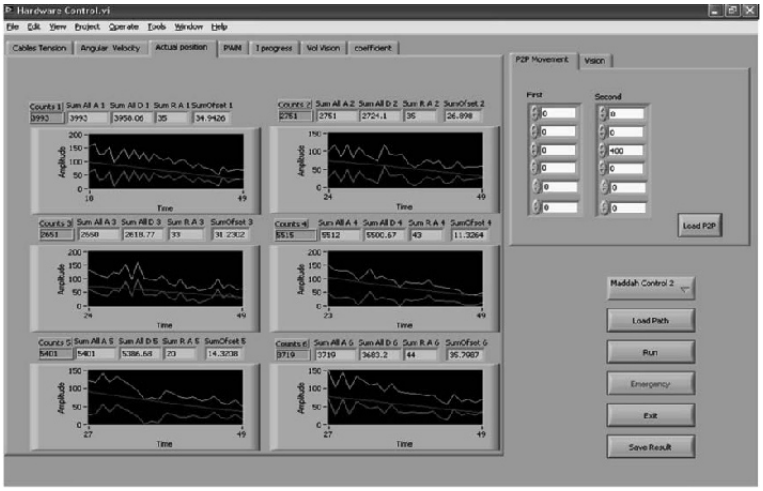


Figure 13.32. *The graphical user interface (GUI) software setup programmed in LabVIEW environment*

By using this software setup, a user is able to generate the plan of the robot's trajectory and run the robot to move in the related desired path. It is also possible to run each motor with a desired speed separately by producing its related PWM. The software setup sends the command signal to each motor by determining the pulse width of PWM signal by a data acquisition board. It also reads the output signals of encoders and calculates the position, orientation, linear velocity and angular velocity of the end-effector. It also plots the motion diagram and the position and angular velocity diagrams of each motor in a real-time way. In Figure 13.32, a panel of software is shown in which the desirable and the real positions of each six motors are being plotted simultaneously in a real-time way.

Since the whole GUI and simulator of the cable robot are designed in LabVIEW environment, it was required to transfer the analyzed data of imaging process and also lasers, which have been processed in Matlab® environment into the LabVIEW in an online way. Providing the controlling and timing consistency of these two softwares (Matlab® and LabVIEW) is not possible with the aid of a single PC. Therefore, two PCs are used here: the second PC analyzes the image processing of the camera data (Matlab®) while the first PC is responsible for processing the data of the lasers and also the process related to the dynamics and control of the robot in an online way.

An online timing was required to provide the consistency of these two PCs, which is done with the aid of methodology shown in Figure 13.33.

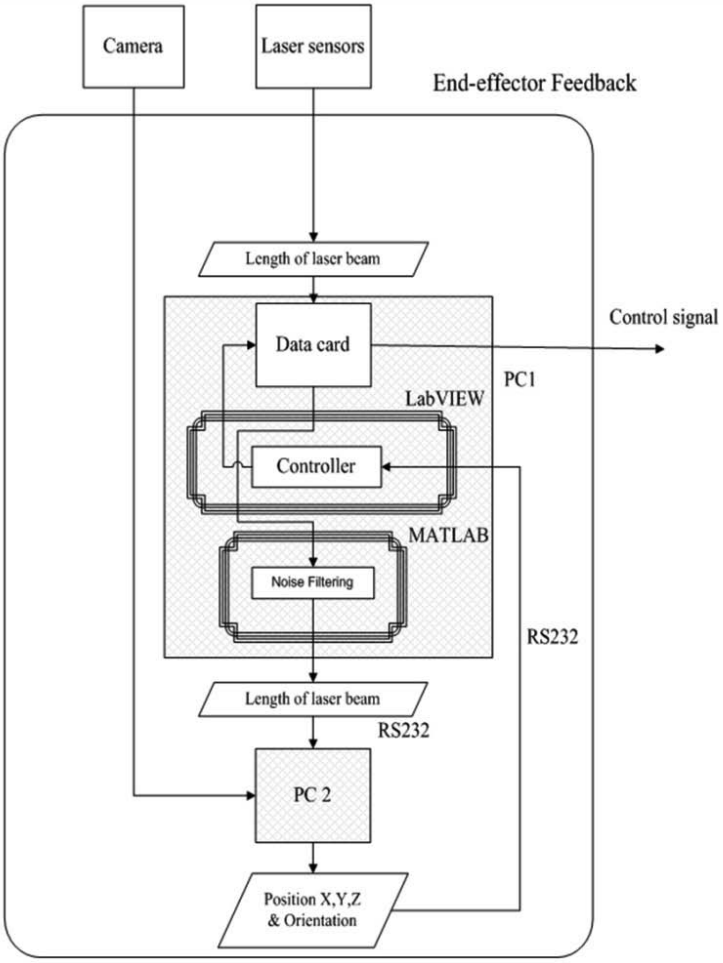


Figure 13.33. *Flowchart of information architecture*

First, the data related to the lasers are received by PC1 that is responsible for controlling the robot through the employed digital data card (A/D1780). Then, the data are transferred to the second PC with the aid of serial port and RS232 protocol that is responsible for analyzing the camera. Using the raw (virtual) data of image processing together with the altitude of the

end-effector, which is provided by the aid of lasers, it is possible to evaluate the actual position of the end-effector with the aid of perspective formulation in PC2. Finally, the obtained data will then be transferred into PC1 with the aid of the same serial protocol to be used as the end-effector feedback in controlling the robot.

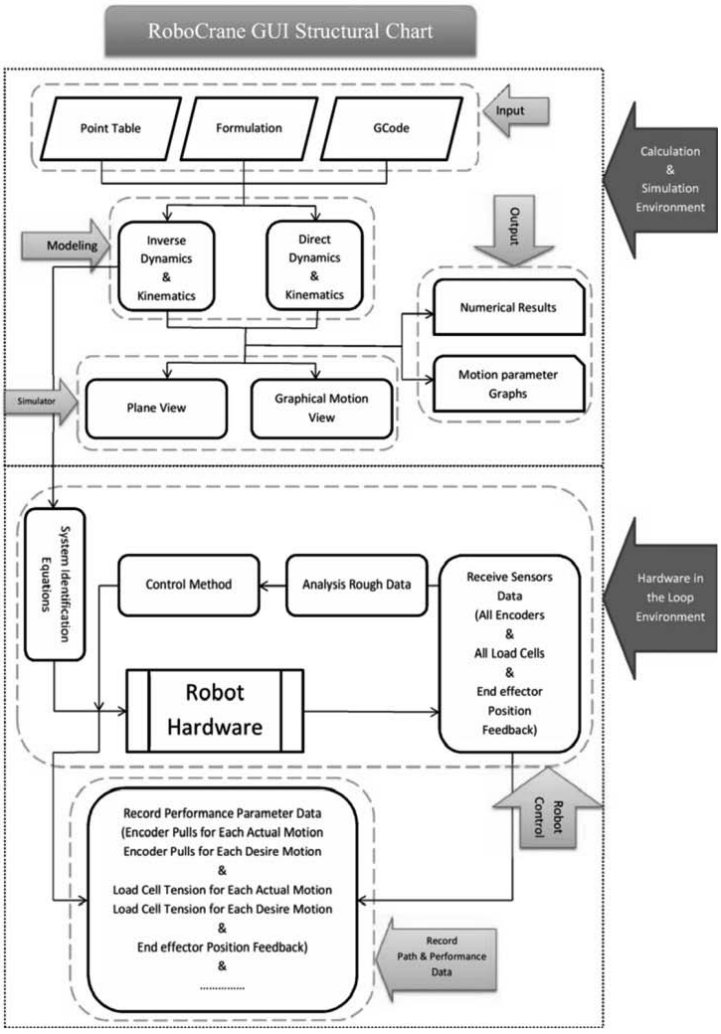


Figure 13.34. *Flowchart of the structure of the designed GUI*

13.6. Graphical user interface (GUI) [KOR 11d]

The interface is designed in a way to support these modules: taking the input; modeling, simulating and controlling the actuators; recording the sensors and saving the output. The following flowchart displays the overall scheme of the interface structure.

13.6.1. *Simulator and virtual laboratory [KOR 11d]*

The simulator interface of the ICaSbot is prepared in the LabVIEW, which is called RoboCrane, and is designed to be compatible with all versions of Windows. As soon as the mentioned interface is called, the main menu of this program appears in which the following options are available (Figure 13.35): Direct Kinematics, Inverse Kinematics, Direct Dynamics, Inverse Dynamics and Experiments.

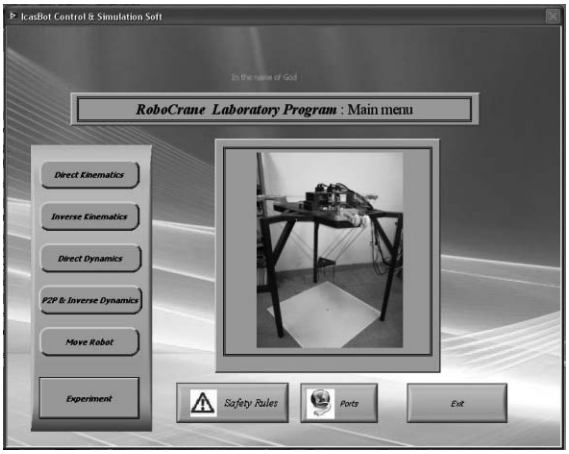


Figure 13.35. *Main page of RoboCrane*

Moreover, two selections of ports and safety keys are also positioned in the main menu. The ports key is prepared in order to make a connection to the robot using related interface electrical circuits. In this case, the robot can be easily controlled by the user through the designed graphical interface while all of the kinematics and dynamics results are available to be observed and interfered with. Finally, the safety rules key includes a series of warnings and primitive data that are necessary for the users to be considered.

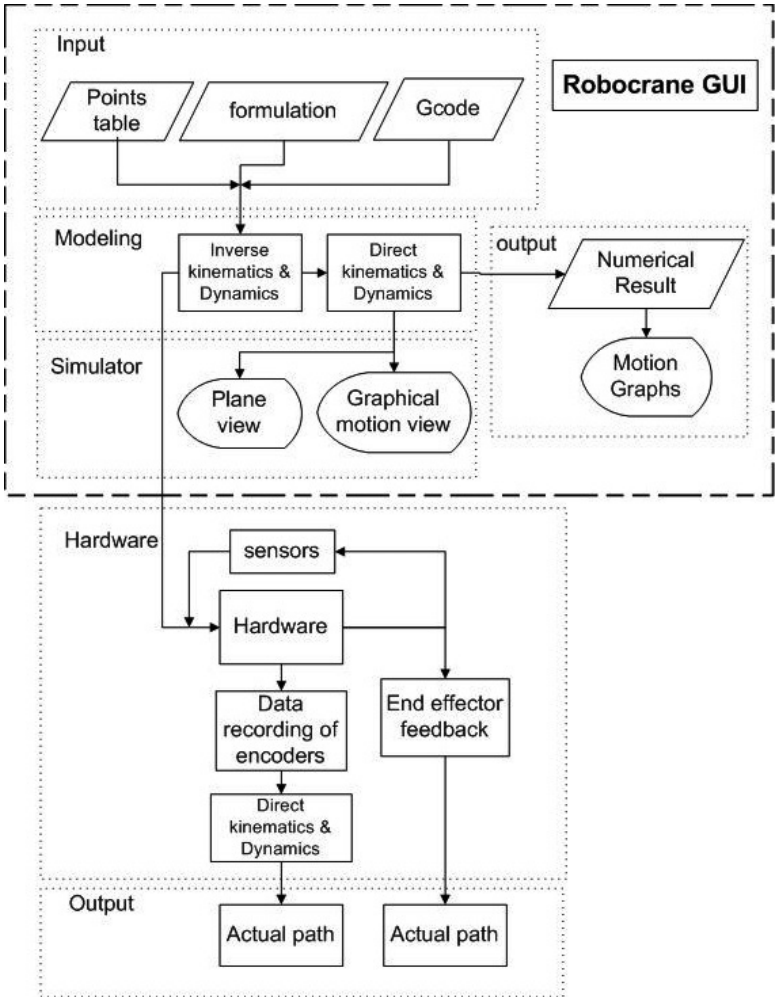


Figure 13.36. *Flowchart of the structure of the designed GUI*

In the direct kinematics window (Figure 13.37), the user is able to obtain the final position of the end-effector by specifying the amount of angular rotation of the motors together with the initial position of the end-effector. Also, the direct kinetics window (Figure 13.38) provides a virtual environment in which the user is able to estimate the actual position of the end-effector as the result of inserting a particular value of torques of the motors.

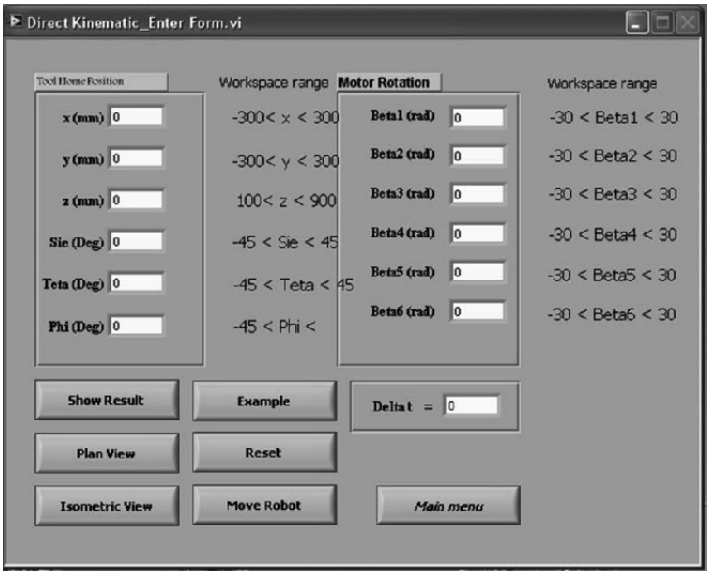


Figure 13.37. *Direct kinematics window*

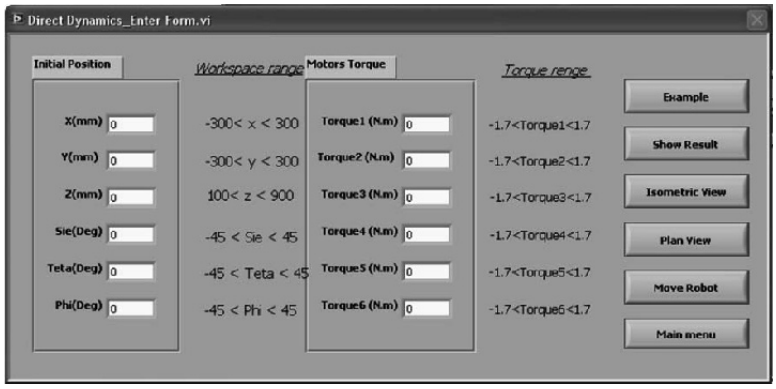


Figure 13.38. *Direct dynamic window*

The required torque of each motor for a desired motion of the end-effector can be evaluated in the inverse kinetics (Figure 13.39) of this software by providing the desired path of the end-effector. The user is free to determine the desired path of the end-effector by providing a table of desired points or determining the equation of the desired path.

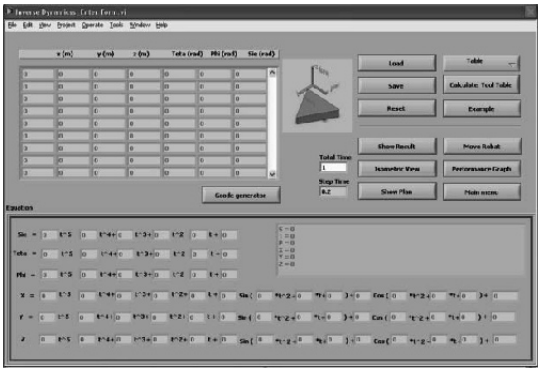


Figure 13.39. *Inverse dynamic window*

Finally, the traditional way of implementing the desired tracking of the end-effector, which is G-code, is also considered in the designed GUI, which is selected in the experiments window (Figure 13.40). The software is also able to provide the equivalent G-code of every motion of the end-effector from the input data of inverse kinetic window [KOR 11d].

Figure 13.40. *G-code section*

One of the most considerable facilities, which are designed in the proposed GUI, is the simulator section of the interface in which the user is able to observe the isometric motion of the robot in a virtual environment based on the input data.

Also the DoFs profile of the end-effector motion can be seen in another environment from different point of views in an online way for which the rate of simulating steps can be determined freely by the user (Figures 13.41 and 13.42).

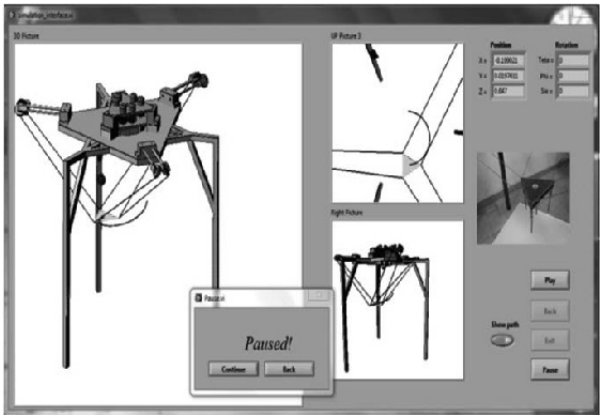


Figure 13.41. *Isometric view window*

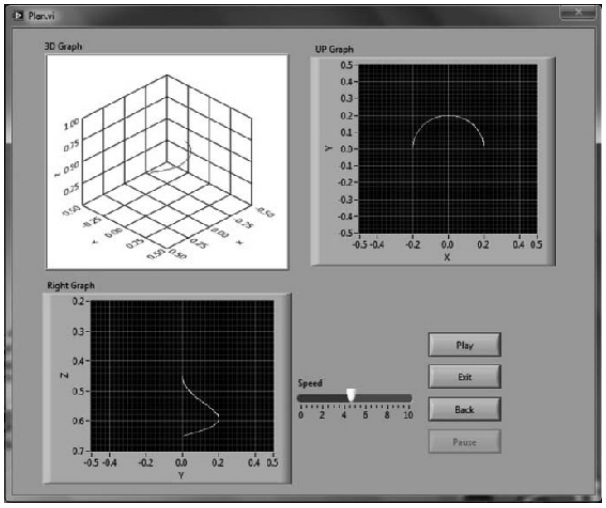


Figure 13.42. *Plane view window*

The user is also able to view the resultant profiles of the motors' torque, DoFs velocity and acceleration and cables' elongation as a function of time in this window (Figure 13.43).

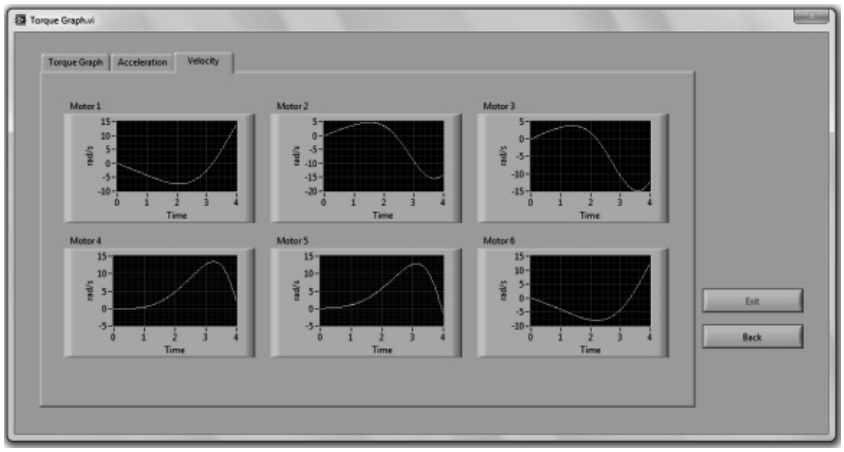


Figure 13.43. *Motion graphs*

13.6.2. *Hardware control [KOR 11d]*

Finally, the real robot can be controlled in the "hardware in the loop result" window. This part is the main control section of the software, which connects the robot to the electrical boards. The controlling commands that are driving the motors and also monitoring the actual data that are received from the sensors, can be managed in this part of GUI. Besides the control of the end-effector, the user is able to view and save the data related to the end-effector's motion including angular velocities of the motors, which are recorded by the encoders, cables' tension, which is recorded by the aid of load cells, and actual position of the end-effector, which is recorded by the position sensors.

In the main input window of the robot control section, by loading a predefined path, or by inputting the initial and final points (in point-to-point movement), the user commands the robot to move (Figure 13.44). Also, all the control methods explained in section 13.6.1 can be selected in this window. In addition, the communication parameters for the control of

hardware by computer, image processing and also the calibration coefficients of load cells and optical sensors are adjusted in this window. After the path is loaded and the control method is selected in the hardware control program, the angular velocity of motors and the required motors' torque are calculated from the inverse dynamic and kinematic equations and used as initial values for the adjustment of movement parameters.

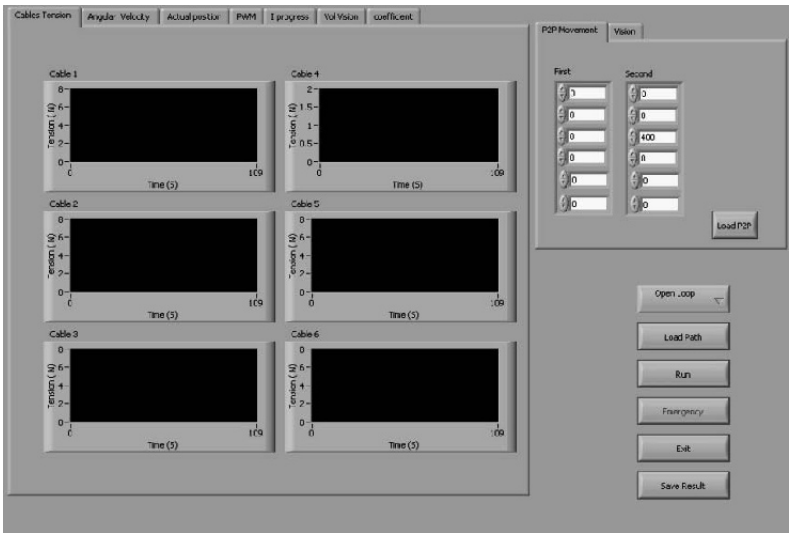


Figure 13.44. *Main control window [KOR 11d]*

After applying the considered inputs to the robot, all the received data from sensors like the encoder and load cell can be observed by going to the output window's tab. Also, the processing of images along with the robot movement can be observed and analyzed by referring to separate tab.

In the tab related to the load cell, the amount of voltage received from the load cell, which is detected through the analog-to-digital ports, can be observed.

In another tab titled "the ON time duration of motor control pulses", the user can observe the ON time duration of pulses associated with all the motors and can evaluate the performance of control coefficients (Figure 13.46).

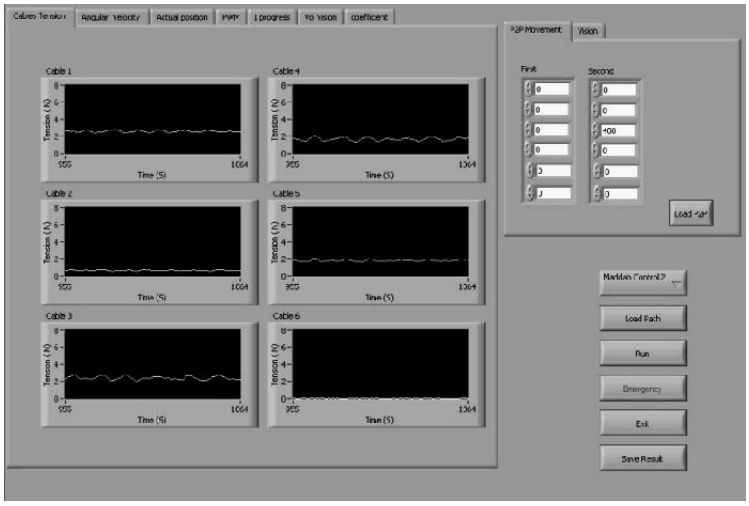


Figure 13.45. *Load cells' tab [KOR 11d]*

Figure 13.46. *ON time duration of motor control pulses' tab [KOR 11d]*

In the next tab, the data related to real pulses received and the ideal pulses that are needed and also the differences between these two are displayed. A larger difference means a larger deviation in path tracking. Also in this window, the total quantity of desired pulses that should be received during

tracking and also the total quantity of real pulses received during tracking are illustrated, which is indicative of the error in the end point (Figure 13.47).

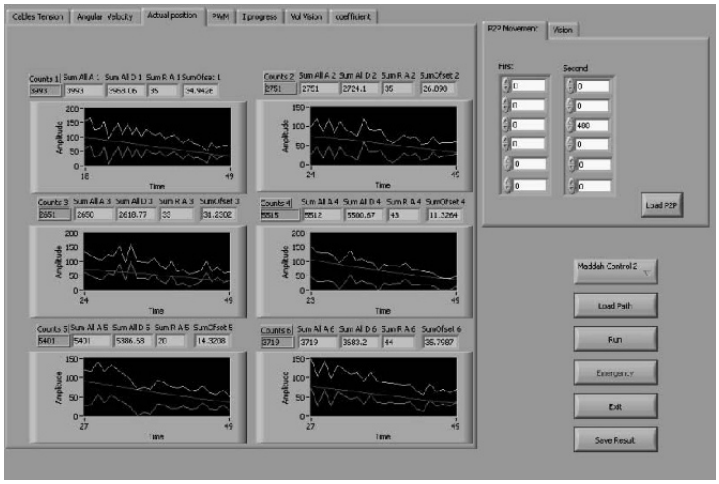


Figure 13.47. *Received pulses' tab [KOR 11d]*

In the tab related to the end-effector, the user can observe the data of the optical sensors, which are simultaneously being executed by an image processing program on a second computer (Figure 13.48).

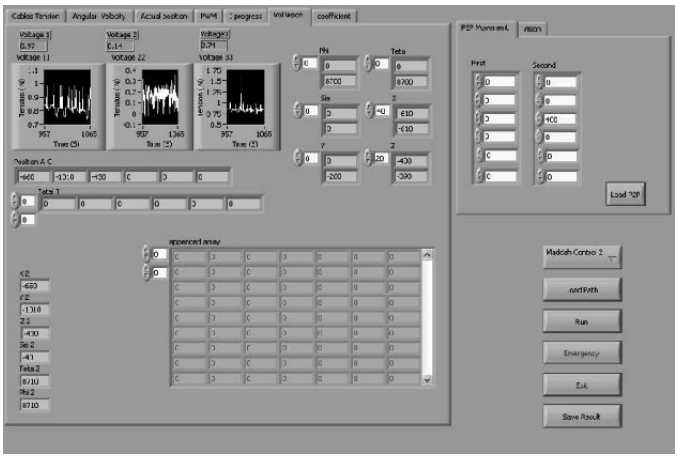


Figure 13.48. *The end-effector data's tab [KOR 11d]*

Moreover, the user can save the path-related data. These data include the load cell data, encoder data and the data received from the end-effector position.

The result of trajectory tracking in virtual simulator and also the camera data are shown in Figures 13.49 and 13.50. It can be seen from the experimental result that an acceptable compatibility can be observed between the simulation results and experimental test.

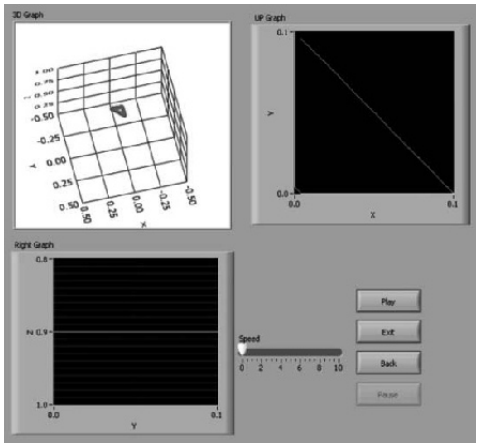


Figure 13.49. *Isometric view and plan view of triangle tracking in GUI simulator [KOR 11d]*

Figure 13.50. *Result of trajectory tracking in virtual simulator [KOR 11d]*

13.7. Result and verifications

There are some ISO tests defined in the references, which evaluate the performance of the robot maneuverability. One of the most powerful tests of this type is ISO 9283 for which a trajectory should be tracked by the robot on a plane inclined by a 45° angle and its norm of deviation error shows the performance of the robot [KOR 11c]. The ISO test that is considered here for verifying the precision of the robot is a 45° inclined circle with the following equation:

$$\begin{aligned} x &= 0.05\cos(0.14\pi(5.37-t)^2) \\ y &= -0.05\cos\left(\frac{\pi}{4}\right)\sin(0.14\pi(5.37-t)^2) \\ z &= -0.05\sin(\pi/4)\sin(0.14\pi(5.37-t)\uparrow 2+0.8\cos(\pi/4)) \end{aligned} \qquad [13.29]$$

where t is time. Through this test, the tracking performance of the ICaSbot, which is equipped with the mentioned closed-loop controller, is verified and we can compare the efficiency of the two mentioned controlling strategies from the dynamic error point of view.

For the whole simulation results and profiles, comparison is done between the controlling strategy based on the new proposed feedback recording (Cartesian space control) and the previous strategy that has been used so far for all of the cable robots, which is using joint space control or control based on the feedback of the encoders to show the efficiency of the proposed method. First, it can be observed that there is an acceptable compatibility between the experimental tests and simulation results in which the error is less than 1 cm. As can be seen in Figure 13.51, the tracking accuracy is considerably better for the robot that is equipped with the end-effector feedback controller rather than the robot that is controlled by the aid of joint space controller. For the end-effector controller, there is a little deviation at the first stage of movement, which is compensated over the time with the aid of end-effector controller as was expected. The elliptic shape of the x–y profile is due to altitude variations of the end-effector whilst tracking an inclined circle. A comparison of angular velocity of the motors among simulation, experimental results of joint space controller and experimental results of Cartesian space controller are shown in Figure 13.52.

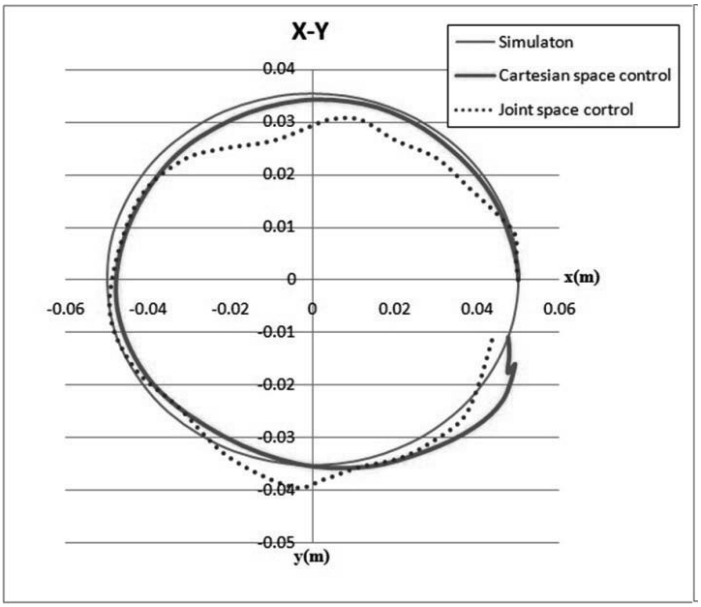


Figure 13.51. *Comparing the result of simulation path and experiment path for inclined circle*

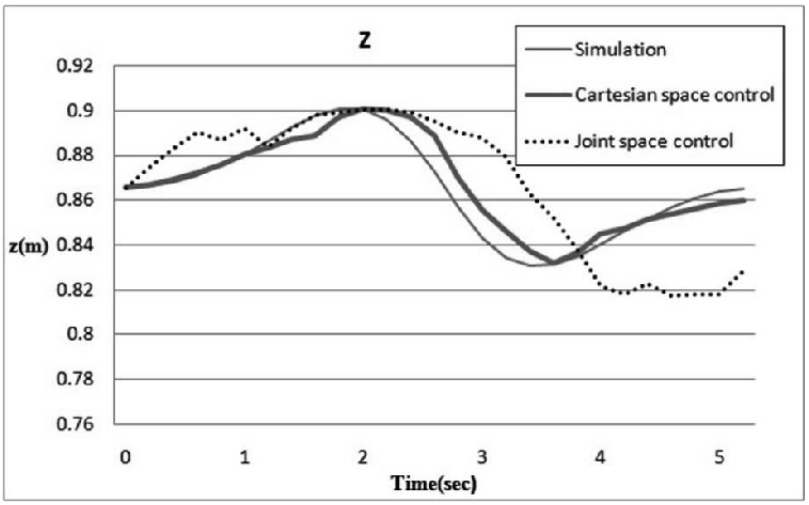


Figure 13.52. *Comparing the result of simulation path and experiment path for inclined circle*

It can be seen that as a result of the high inertia of the motors, the experimental profiles have a delay compared to simulation profiles at the initial moments of movements; however, the compatibility increases afterwards, thanks to implementing the designed controller. As a result, the overall area under the angular velocity curve of the motor, which specifies the overall rotation of the motors during the time, is roughly equal for simulation and experimental results. Again, it can be observed that this compensation and the compatibility of angular velocity profiles are extremely better in the case of the Cartesian space controller compared to the joint space controller.

In order to have a more understandable comparison of the results, the norm of error function, which can be calculated by the following equation, is plotted for two controlling cases i.e. joint space control and Cartesian space control:

$$E = \sqrt[2]{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2} \qquad [13.30]$$

In this equation, $x_1,\ y_1,\ z_1$ is the desired position and x_2, y_2, z_2 is the actual position of the end-effector in trajectory tracking.

As we expected, the normal error of the Cartesian space control method is less than the joint space control method and has the average error of approximately 1 cm versus 2 cm in joint space control mode. Cable tension results and its comparison with simulation are shown in Figure 13.54.

As can be seen from diagrams, at some points cable tension value is deviated from the simulation value, which is due to the existence of the extra load of the end-effecter and its effect on the friction and inertia of the system that causes the experimental results to be higher than simulation values. Moreover, the deviations are compensated quickly by the designed control system. As was mentioned for the previous test, it can be seen from the obtained results of most load sensors that the robot endures higher tension outputs compared to simulation at the initial stages of the motion, which is due to the high applied force on cables to overcome friction and to initiate the motion in the beginning. This phenomenon is more serious in this test compared to the previous test since the robot moves with its maximum load capacity. Afterwards, the robot tolerates lower cable tension in the next few seconds by a quick compensation of deviations. However, again here, the mean value of cable tensions of experimental motion is roughly equal to the

simulation value, which validates the correctness of the experiment. Moreover, like the previous observations, there are some oscillations in the practical diagram compared to the simulator diagram in all of the obtained results. The reasons can be contributed to the following facts: because of using PD control along trajectory in motors, the creation of oscillation in robotic motion is inevitable. This oscillation has effects on cable tensions. Flexibility of the joints, cables and structure are other sources of this vibrating response. These flexibilities of the cables and also the structure of the robots affect the vibrating response of the robot in reality. Clearance, especially in the joints, motors, drum, pulleys, flip flop, cable tension measurement mechanism and cable conducting pulley, is also highly effective in the creation of oscillation in the cable tensions' diagrams during the motion.

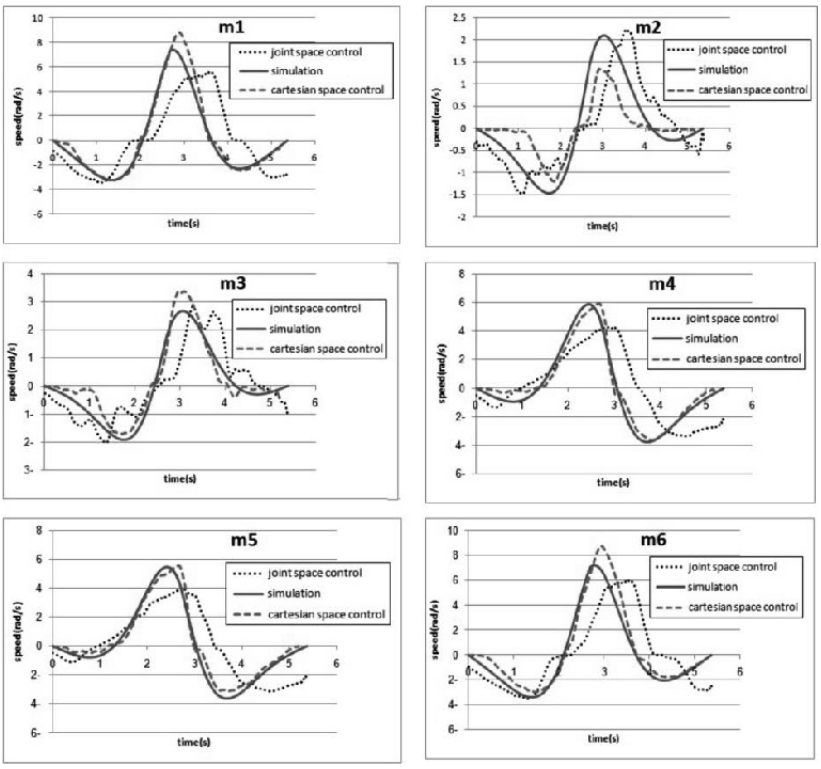


Figure 13.53. *Comparing the simulation and experiment angular velocity of motors for inclined circle*

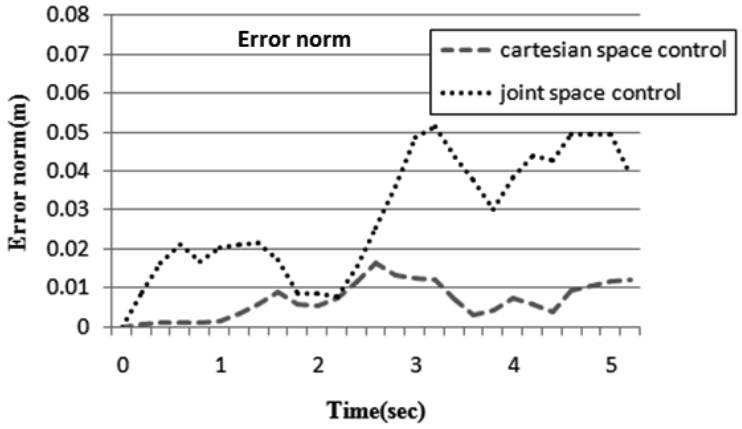


Figure 13.54. *Error norm in inclined circle path*

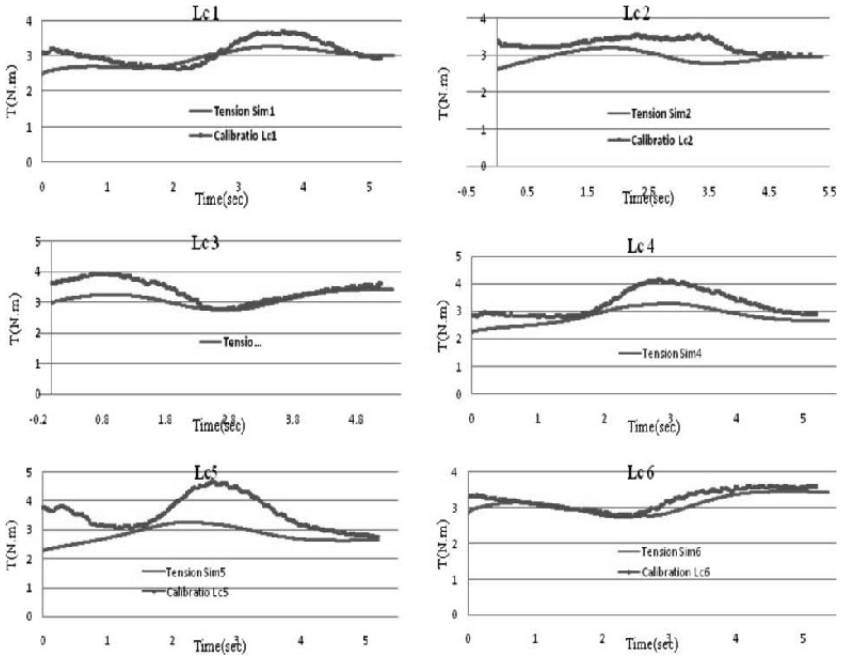


Figure 13.55. *Comparing the cables' tension between simulation results and experimental data*

13.8. Conclusion

As has been explained, parallel robots are widely used in industrial environments and on automation lines as a result of their high stiffness, load capability and acceptable accuracy. The parallel nature of these robots and necessity of providing a powerful real-time and online control for them converts these robots to a complicated mechatronics mechanism. It was stated that one of the most applicable kinds of these robots, which has been developed recently, is cable robots in which the pneumatic and hydraulic jacks are replaced by cable and motor actuators. High ratio of load to weight, large workspace and perfect capability of assembling and de-assembling are the positive consequences of this strategic replacement. On the other hand, its parallel dynamic formulation and the tensional force limitation of these kinds of robots make their control more challenging.

In this chapter, after a brief introduction of the overall structure of these robots and their applications, required dynamic and controlling formulation of these robots have been represented. It was seen that a proper nonlinear control strategy is required to control the end-effector on a predefined trajectory. This control consists of a feed-forward term in which the whole inverse kinematics and kinetic formulations are employed. This feed-forward term is then improved using feedback signals with the aid of two sequential closed-loop controllers. The inner controller controls the angular velocity of the motor for each 0.01 s while the end-effector feedback is used in the outer closed loop for each 0.1 s.

It was stated that this complicated controlling strategy needs high-tech mechatronics equipment and hardware installation including powerful actuators, high-resolution sensors, compatible data transfer protocol and a user-friendly simulator with GUI [HAB 06, HAB 07, HAB 08]. Six DC motors together with six cables rolled up with the aid of six pulleys are used as the required actuator. It was seen that with the aid of providing a suitable lookup table, it is possible to determine the required torque and angular velocity using the employed DC motors that can be used to control the end-effector based on inverse dynamics. Its related electrical boards and drivers were designed, and required calibration and a lookup table were provided. Six encoders are used to feed back the actual angular velocity of the motors. In order to estimate the actual tension of the cables, six load cells are installed. Again, required electrical interface boards and their related calibration were presented. It was seen that in order to sense and feed back all of the translational and rotational movements of the end-effector, it is required to use a combination of lasers and a camera in which the lasers help to evaluate the altitude of the end-effector together with its angles while the camera helps us

to estimate the planar coordinates of the end-effector. Data acquisition cards were used to provide a high-speed transfer protocol to determine the required real-time and online heavy calculations of the parallel robot. Digital cards are a help to manage the motor and encoder data while the analog card is used to handle the load cell data. Finally, in order to train the operator and also control the robot, a simulator and GUI are designed in the LabVIEW environment by which the desired commands can be implemented on the robot using different ways and all of the resultant actual kinematics and kinetics data can be monitored. In the end, in order to validate the mentioned dynamics and control strategies and also to verify the efficiency of all the mentioned mechatronics implementation of the robot including its hardware and software setup, a comparison was performed between the simulation result of Matlab® and experimental data. Not only was a good compatibility shown, but also complete analyses were delivered from dynamic and control points of view.

It can be seen that all of the requirements of a complete mechatronics mechanism are involved in this parallel robot and the importance of mechatronics knowledge is shown in this chapter regards design, manufacture and running the above-mentioned robot with good accuracy.

Definition	Symbol
Translational coordinates of the end-effector	$x_m, y_m, z_m (m)$
Rotational coordinates of the end-effector	ψ, θ, ϕ (rad)
Jacobian matrix	J
Cables' elongation	L
Pulleys' angular rotation	β
Pulley radius	R
Vector of cables tension	T
Inertia matrix	D(x)
Vector of velocity terms	C(x,ẋ)
Gravity vector	g(x)
Vector of DoFs of the system	X
Mass of the end-effector	M
Moment of inertia of the end-effector	I
Length of the cables	Q
Matrix of rotary inertia of the motors	J
Viscous friction matrix of the motors	C
Vector of angular velocity	β̇

Table 13.8. *Specification of the used optical laser sensor*

Vector of motors torque	τ
Control input of the feedback linearization	ν
Controlling gains of derivative errors	K_{iD}
Controlling gains of proportional errors	K_{iP}
Desired trajectory of the robot	X_d
Actual trajectory of the robot	X_a
Armature voltage of the DC motor	v_a
Resistance of armature wire of the DC motor	R_a
Armature inductance of the DC motor	L_a
Armature current of the DC motor	i_a
Emf (induced voltage of electromotor force)	e_a
Constant of torque of the DC motor	K_m
Produced torque of the DC motor	τ_m
Angular velocity of the DC motor	ω_m
Maximum voltage of the motor	v_{max}
Controlling gains of proportional error of motor angular velocity	K_d
Controlling gains of derivative error of motor angular velocity	K_p
Controlling gains of proportional error of motor torque	K'_d
Controlling gains of derivative error of motor torque	K'_p
Distance between the camera and the point	H
Canonical distance of camera	F
Local coordinate system	X,Y,Z
Angle between the direction of the laser beams and local Z axis	ζ
Angle of normal vector of the end-effector	γ
Global coordinate system	$\hat{X}, \hat{Y}, \hat{Z}$

Table 13.8. *(Continued) Specification of the used optical laser sensor*

13.9. Bibliography

[ALB 93] ALBUS J., BOSTELMAN R., DAGALAKIS N., "The NIST ROBOCRANE", *Journal of Robotics Systems*, vol. 10, no. 5, pp. 709–724, 1993.

[ALP 01] ALP A.B, Cable-suspended parallel robots, MS Thesis University of Delaware, 2001.

[BOS 95] BOSTELMAN R., ALBUS J., DAGALAKIS N., *et al.*, *RoboCrane Project: An Advanced Concept for Large Scale Manufacturing*, Intelligent Systems Division National Institute of Standards and Technology Gaithersburg, MD, 1995.

[CHA 04] CHANG A.H., Experimental development of the mobile vestibular platform, SUNFEST Technical Report, Center for Sensor Technologies, University of Pennsylvania, Philadelphia, PA, 2004.

[HAB 06] HABIB M.K., "Mechatronics engineering the evolution, the needs and the challenges", *Proceedings of the 32nd Annual Conference of IEEE Industrial Electronics Society*, IECON 2006, Paris, pp. 4510–4515, 2006.

[HAB 07] HABIB M.K., "Mechatronics: a unifying interdisciplinary and intelligent engineering paradigm", *IEEE Industrial Electronics Magazine*, vol. 1, no. 2, pp. 12–24, Summer 2007.

[HAB 08] HABIB M.K., "Interdisciplinary mechatronics: problem solving, creative thinking and concurrent design synergy", *International Journal of Mechatronics and Manufacturing Systems (IJMMS)*, vol.1, no.1, pp. 264–269, 2008.

[IRA 09] IRANPOUR M., Modeling and simulation of kinematics and dynamics of spatial cable-actuated-robots, MSc Thesis, Iran University of Science and Technology, 2009.

[KAM 95] KAMAMURA S., CHOE W., TANAKA S., *et al.*, "Development of an ultrahigh speed robot FALCON using wire drive system", *IEEE International Conference on Robotics and Automation*, Japan, 21–27 May 1995.

[KOR 09] KORAYEM M.H., BAMDAD M., "Dynamic load carrying capacity of cable-suspended parallel manipulators", *International Journal of Advanced Manufacturing Technology*, vol. 44, pp. 829–840, 2009.

[KOR 10a] KORAYEM M.H., TOURAJIZADEH H., "Maximum DLCC of spatial cable robot for a predefined trajectory within the workspace using closed loop optimal control approach", *Journal of Intelligent & Robotic Systems*, 2010.

[KOR 10b] KORAYEM M.H., TOURAJIZADEH H., BAMDAD M., "Dynamic load carrying capacity of flexible cable suspended robot: robust feedback linearization control approach", *Journal of Intelligent and Robotic Systems*, vol. 60, no. 3–4, pp. 341–363, 2010.

[KOR 11a] KORAYEM M.H., BAMDAD M., TOURAJIZADEH H., *et al.*, "Development of ICaSbot: a cable-suspended robot's with six DOF", *Arabian Journal for Science and Engineering*, 2011.

[KOR 11b] KORAYEM M.H., JALALI M., TOURAJIZADEH H., "Optimal path planning of spatial cable-suspended robot subject to maximum load carrying capacity using optimal sliding mode control approach", *3rd International Conference on Manufacturing Engineering ICME2011*, Tehran, Iran, 2011.

[KOR 11c] KORAYEM M.H., KHAYATZADEH S., SAFFARI A.H., *et al.*, "Sensing the position of the end-effector of a cable suspended robot for on-line control", *3rd International and 12th National Conference on Manufacturing Engineering (ICME)*, University of Tehran, Iran, 2011.

[KOR 11d] KORAYEM M.H., TAHERIFAR M., MADDAH S.M., *et al.*, "Design and programming a graphical user interface for the ICaSbot Robot using LabVIEW", *International Conference on Computational Intelligence Applications (ICCIA)*, Shiraz, Iran, 2011.

[KOR 12a] KORAYEM M.H., NAJAFABADI A.I., TOURAJIZADEH H., "Control of ICaSbot cable robot using simultaneous control of the speed and torque of the motors", *International Conference on Experimental Solid Mechanics and Dynamics (X-MECH)*, Iran University of Science and Technology, Tehran, Iran, 2012.

[KOR 12b] KORAYEM M.H., TAJIK A., NAJAFABADI A.I., *et al.*, "Calculation of the end-effecter position of the cable robot using cable tension and encoders outputs", *3rd International and 12th National Conference on Manufacturing Engineering (ICME)*, University of Tehran, Iran, 2012.

[KOR 12c] KORAYEM M.H., TOURAJIZADEH H., JALALI M., *et al.*, "Optimal path planning of spatial cable robot using optimal sliding mode control", *International Journal of Advanced Robotic Systems*, INTECH, 2012.

[OH 05] OH S.R., AGRAWAL S.K., "A reference governor-based controller for a cable robot under input constraints", *IEEE Transactions on Control Systems Technology*, vol. 13, pp. 639–645, 2005.

[OSU 93] OSUMI H., ARAI T., ASAMA H., "Development of a seven degrees of freedom crane with three wires (1st Report) – concept, design and control", *The Japan Society* of *Precision Engineering*, vol. 59, no. 5, pp. 767–772, 1993.

[ROS 06] ROSATI G., GALLINA P., ROSSI A., *et al.*, "Wire-based robots for upper-limb rehabilitation", *International Journal of Assistive Robotics and Mechatronics*, vol. 7, pp. 3–10, 2006.

[ROS 07] ROSATI G., GALLINA P., MASIERO S., "Design, Implementation and clinical tests of a wire-based robot for neurorehabilitation", *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 15, no. 4, pp. 560–569, 2007.

[SHI 99] SHIANG W.J., CANNON D., GORMAN J., "Dynamic analysis of the cable array robotic crane", *IEEE International Conference on Robotics and Automation*, Detroit, MI, pp. 495–500, 1999.

[TAJ 11] TAJIK A., KORAYEM M.H., TOURAJIZADEH H., "Developing a force measurement system for cable suspended robot and study its applications", *3rd International and 12th National Conference on Manufacturing Engineering (ICME)*, University of Tehran, Iran, 2011.

[VAR 06] VARZIRI M.S., NOTASH L., "Kinematic calibration of a wire-actuated parallel robot", *Mechanism and Machine Theory*, vol. 42, no. 8, pp. 960–976, August 2007.

[YAM 99] YAMAMOTO M., YANAI N., MOHRI A., "Inverse dynamics and control of crane-type manipulator", *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Kyongju, South Korea, pp. 1228–1233, 1999.