Journal Européen des Systèmes Automatisés
Vol. 57, No. 6, December, 2024, pp. 1677-1685
Journal homepage: http://iieta.org/journals/jesa

Design, Modeling and PID Control of a Cable-Driven Wall Climbing Robot Using Genetic Algorithms

Aws Jona*, Ahmed Alkamachi

Department of Mechatronics Engineering, Al-Khwarizmi College of Engineering, University of Baghdad, Baghdad 10071, Iraq

Corresponding Author Email: aws.ilias2202m@kecbu.uobaghdad.edu.iq

https://doi.org/10.18280/jesa.570615

Received: 18 October 2024
Revised: 9 December 2024
Accepted: 17 December 2024
Available online: 31 December 2024

Keywords:
cable, genetic algorithms, climbing robots, PID controller

ABSTRACT

Climbing robots are emerged to meet the needs required by high-rise buildings (maintenance, cleaning, etc.). Human labor is responsible for these needs for a long time. With the increase in accidents and deaths, more reliance is placed on robots, especially since some buildings are not suitable for human presence (e.g.: nuclear reactors). Despite the expansion of climbing robots design, cable robots remain the most effective due to their high capacity to carry and control weights even for small distances. These features are accompanied by inaccuracy due to many factors that affect the cable, such as temperature and length changes (due to flexibility) through adding weight to the cable. Unavoidable factors can be addressed using control tools and artificial intelligence. In this research, a PID controller is used to control the cable length to correct the error by adding or decreasing the number of cables turns and thus the cables length. Artificial intelligence and genetic algorithms are used to determine the appropriate values for the PID controller. The final results that are observed by identifying several paths (circular and rectangular) and comparing them in different cases (with or without the controller) are shown. The matching percentage between the required path and the real path is found, which appears to be highly efficient with very low error rate, even after adding the error factor.

1. INTRODUCTION

With development and rise in population, architectures have been turned to vertical designs, which have given rise to skyscrapers. There is a need to have a way to climb it for its sustainability. It may need cleaning, maintenance, decorating, and other needs. Many designs appeared, each of which is distinguished in a specific field. Some have extreme accuracy and others have speed, but the most important need remains for high lifting weight capability at low cost. The optimal solution for these needs is to design a climbing robot that meet the requirements in terms of weight and cost.

Climbing robot research is not a relatively new, but it has flourished in the last two decades, as many designs and research have appeared in this field. Some researchers studied cable driven planar robots [1, 2]. Other pieces of literature suggested another design that help robots to climb such as suction cup, magnetic adhesion, bio-inspired adhesion, and ducted Fans [3-6].

Despite the appearance of drones, cable driven robots are still used in the field of surveillance, and as cable-moving cameras in football stadiums [7, 8].

This research proposes a new design, inspired by all of the above, to make the robot climb the wall relying on cables and with a two dimensional working area. It also discusses the most important advantages and disadvantages of using cable robots, and propose a new design in which the motors are set inside the robot, making it easy to use and install. Proportional integral derivative (PID) controller is used to regulate the robot operations. The controller parameters are tuned with the aid of genetic algorithm (GA).

Section 2 explores the most important characteristics of cables. In Section 3, the mathematical equations that govern the system are derived. Several simulation cases are performed in Section 4 using the MATLAB platform. Finally, the paper is concluded in Section 5.

2. CABLE CHARACTERISTICS

Lifting cables is one of the oldest lifting methods known to mankind, but with the invention of motors and control, it gives more accurate results in addition to ease of use. As a result, cables have been used in many applications such as performing surgical or industrial operations for the need to move heavy weights or dangerous objects, photographic cameras in stadiums, etc. These are known as cable array robots, which are intended to replace arms. Mechanism for the purpose of carrying the end effector using cables has many advantages and many disadvantages as well [2, 9]. Among these are:

2.1 Cable advantages

With the development of chemical and industrial engineering, the rope and cable industry have its own field.

Cable improvements aim to reduce its weight and increase rigidity, as there are a large number of types of cables used in the industrial field and other fields. Replacing joints with cables gives greater freedom to the robot and motors in particular. For example, the weight of the Puma 560 robot exceeds 54 kg to achieve a working area of no more than 1.5 meters, while other cable driven robot with only 5 kg and can achieve a larger working area. Light weight cables reduce the load on the motors, making them an energy and cost saver. In addition to their flexibility that exceeds the joints, increasing the load on the joints may lead to their breakage [10, 11]. Over the above, the most important advantage of using this type of design is its wide working area. The robot can be fully controlled and moved quickly and freely beyond any other design.

2.2 Cable defects

Despite their many advantages, cables are not without some disadvantages that may be annoying. Some of these are the need to change measurements and re-assemble when the working location changes. It may also have an effect on the accuracy of the work. The flexibility of cables and their length changes with load, temperature, and time. The percentage of elongation that may occur as a result of the load alone may exceed 2% and can be more if the temperature effect is added. This elongation cannot be neglected especially in case of long cables. Another factor that affects the accuracy is the change in diameter of the winding spools which leads to the generation of an accumulated inaccuracy that appears clearly at the final location. In addition, vibration due to the rotation of motors during cable wrapping can cause errors. Cables also need to maintain tension at the same capacity over all operation time [12]. All of these errors become a real obstacle in the absence of good control system [13]. Then designing a controller that can cope with all of these errors is a challenge.

2.3 Types of cable connections

It is known that cables cannot be used to push an object, but only to pull it. To ensure that something is pulled efficiently, the rope must be in a state of constant tension, because when the rope is slack, it will lead to creep if the location changes. Here, care must be taken when choosing the type and number of cables based on the existing conditions and the number of desired degrees of freedom. Cable robots are classified in terms of the number of cables and freedom into two categories. Fully restricted and unrestricted. Unrestrained includes those objects that are tied with one or two cables, which gives the body freedom to rotate and move vertically, as shown in Figure 1(a) and Figure 1(b). Completely restricted cable robot, which require changing the lengths of all the cables to move the body, is divided into two types: Completely restricted and does not give the controller freedom of movement Figure 1(c), or it can be completely restricted and gives the controller freedom of movement (3 degrees of freedom) Figure 1(d).

Choosing appropriate number of cables depends on the type of application, accuracy and required cost. Increasing the number of cables increases control ability and the number of degrees of freedom. It will also help distribute the load over several cables, allowing smaller drives to be selected. However, some applications may not require a large number of degrees of freedom or high loads. Increasing the number of cables merely increases complexity and cost [1, 14, 15]. In this study, only three cables are sufficient where two of them are used to control the x-y coordinates of the robot while the third is to tighten the robot and prevent its swing. Adding another cable leads to an increase in cost and weight and has no improvement of the robot efficiency.

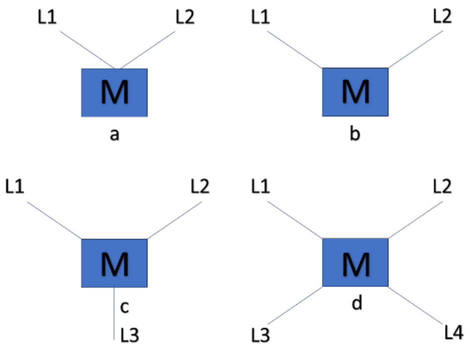


Figure 1. Types of cable connections

3. CLIMBING ROBOT WITH INTERNAL MOTOR

To study cable climbing robots more accurately and objectively, a model design suitable for movement on straight wall – like surfaces is proposed. The design should be as simple as possible, so we will choose a rectangular shape (imitating a box). The motors are installed inside the robot itself. This makes the design easy to fit and install on any buildings by anchoring the cables on the wall. To avoid excess weight and facilitate the control of the robot, the movement will be limited to only three motors connected to three cables. Two cables are pivoted in the upper corners of the wall and one at the bottom of the wall in the middle. Spherical wheels are added to the robot to avoid the friction with the wall surface.

3.1 Design parameters and workspace

To begin with the robot design, it is required to define some parameters, assumptions, and measurements. Parameters values can be adjusted to fit specific location and/or application. These parameters include the robot's dimensions, weight, and the workspace in which it will move.

Robot height a=15 cm
Robot width b= 30cm
Robot length c=15 cm
Robot weight 5 kg (FT=5*9.81=49.05N)
Horizontal working space in (x-axis) B=200 cm
Vertical working space in (y-axis) A=100 cm
Maximum cable length (for the proposed work space) LT=223.6 cm
where,

$$LT = \sqrt{100^2 + 200^2} = 223.6\text{cm} \tag{1}$$

$$\text{Single turn diameter} = \frac{223.6}{3.14} = 71.2\text{cm} \tag{2}$$

Pulley diameter: It is the diameter of the pulley connected to the motor and around which the cable is wrapped.

The pulley is in the form of serrations containing 10 protrusions as shown in Figure 2. Each prong is enough to wrap the cable twice.

$$\text{Pully diamter } (D) = \frac{71.2}{10 \times 2} = 3.56\text{cm} \qquad (3)$$

Assuming the ideal number for each turn requires 200 steps:

$$si = Wn \times 200 = \left(\frac{Li}{3.56 \times 3.14}\right) \times 200 \qquad (4)$$

where, *si*: number of step to *Li*; *Li*: the length of the cable to be moved; *Li=L0-L1; Wn:* number of turns.

The total number of cable rolls $Wt = \frac{LT}{3.56 \times 3.14} = 20$.

All angles are in degrees.

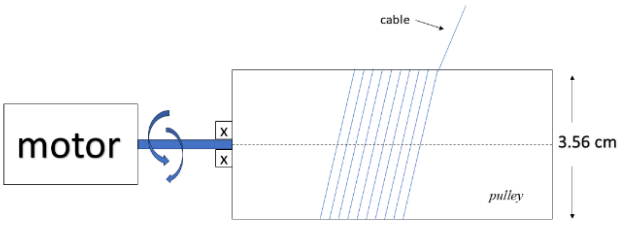


Figure 2. Multi-tooth pulley

Depending on the given information which we often obtain while designing the robot to suit the location in which it is supposed to work, the work space can be divided into a set of points (*x*, *y*). These points can be used to move from one location to another, but to achieve this a set of equations must be derived. These equations represent the system dynamics. Robot parameters used through this work are selected based on the dimensions of the workspace as in Figure 3.

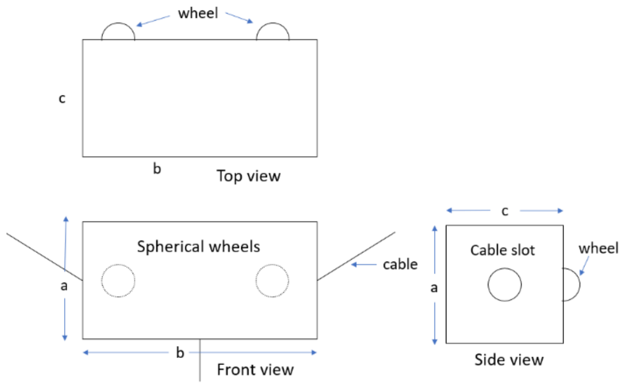


Figure 3. A drawing of the proposed robot

3.2 The length of the cable at any location

To facilitate calculations and design operations, one must rely on a specific variable and make it the main one in most operations. This will reduce the error rate that is small at first, but it accumulates with every calculation or conversion operation due to system inaccuracy factors. In this research, the lengths of the ropes will be adopted for each coordinate. Figure 4 shows the ropes positions, and the robot workspace.

First, the cables are represented as lines where L1 is the first cable and L2 is the second cable and L3 is for the third one. They start from the corner of its installation on the wall and ending with the robot and parallel to the wall (Figure 5). Then these lines are resolved into X and Y components to get (L1X, L1Y, L2X, L2Y, L3X, L3Y). After that, the length of each of these lines is calculated separately based on its distance from the origin (0,0). Then taking into account the robot length (c), the actual length of the cables (L1T, L2T) is calculated. The third dimension will be ignored in the third cable to simplify the use of equations. We assume that the cable is connected to the ground (perpendicular to the robot).

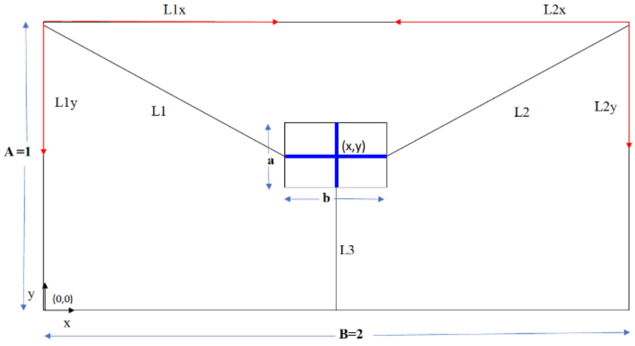


Figure 4. The space within which the robot can move

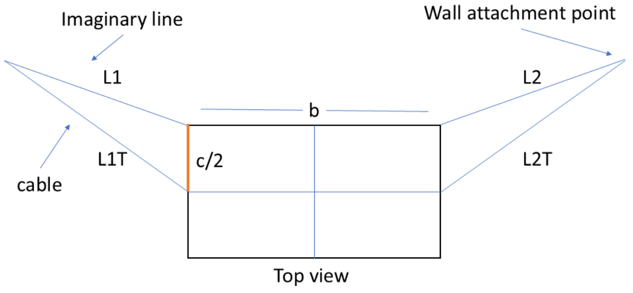


Figure 5. The upper section for connecting cables

$$L1x = x - \frac{b}{2} \qquad (5)$$

$$L1y = A - y \qquad (6)$$

$$L2x = B - x - \frac{b}{2} \qquad (7)$$

$$L2y = A - y \qquad (8)$$

$$L3x = \left| x - \frac{B}{2} \right| \qquad (9)$$

$$L3y = A - (L1y) = y \qquad (10)$$

$$L1 = \sqrt[2]{L1x^2 + L1y^2} \qquad (11)$$

$$L2 = \sqrt[2]{L2x^2 + L2y^2} \qquad (12)$$

$$L3 = \sqrt[2]{L3x^2 + L3y^2} \qquad (13)$$

$$L1T = \sqrt{\left(x - \frac{b}{2}\right)^2 + (A - y)^2 + (\frac{c}{2})^2} \qquad (14)$$

Length of the second cable:

$$L2T = \sqrt{\left(B - x - \frac{b}{2}\right)^2 + (A - y)^2 + (\frac{c}{2})^2} \tag{15}$$

The length of the third cable:

$$L3T = \sqrt{\left(x - \frac{B}{2}\right)^2 + (y)^2} \tag{16}$$

By applying the predefined numerical values, the following equations are obtained:

$$L1T = \sqrt{x^2 - 30x - 200y + y^2 + 10281.25} \tag{17}$$

$$L2T = \sqrt{x^2 - 370x - 200y + y^2 + 44281.25} \tag{18}$$

$$L3T = \sqrt{x^2 - 200x + y^2 + 10000} \tag{19}$$

3.3 Finding angles and strength in every position

As the weight of the robot remains constant, the actual load on each cable will change from one location to another. This change results from the change in the length of the cable and the change in the angle that the cable makes with the robot, and their calculation is an important part of the design to ensure that the rope is in a safe condition throughout the workspace. It will also help in choosing the appropriate type of cable that can be determined by taking into account the worst case.

The angles that the cables made with the x-axis are used through the following calculations where R1 belongs to the first cable and R2 is for the second cable (Figure 6).

$$R1 = 180 - \tan^{-1}\left(\frac{A - y}{x - \frac{b}{2}}\right) \tag{20}$$

$$R2 = \tan^{-1}\left(\frac{A - y}{B - x - \frac{b}{2}}\right) \tag{21}$$

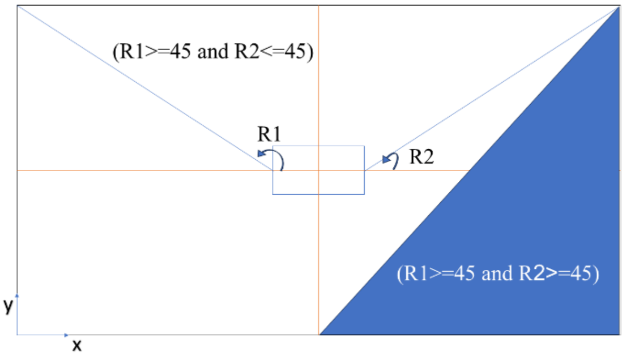


Figure 6. Front clip for connecting cables

During the movement of the robot, the angles of the cables are constantly change, causing it to create an angle that changes over a wider range of 90 degrees. Therefore, there should be a relation between the cables angle and the robot position. According to the range of angles, the workspace is divided into four quarters. In this way, Sine and Cosine functions can be calculated properly.

If (R1>=45 and R2>=45):

$$F1 = \frac{FT}{\sin R1} \tag{22}$$

$$F2 = \frac{FT}{\sin R2} \tag{23}$$

If (R1<=45 and R2>=45):

$$F1 = \frac{FT}{\cos R1} \tag{24}$$

$$F2 = \frac{FT}{\sin R2} \tag{25}$$

If (R1>=45 and R2<=45):

$$F1 = \frac{FT}{\sin R1} \tag{26}$$

$$F2 = \frac{FT}{\cos R2} \tag{27}$$

Else (R1<=45 and R2<=45):

$$F1 = \frac{FT}{\cos R1} \tag{28}$$

$$F2 = \frac{FT}{\cos R2} \tag{29}$$

Although four assumptions were made for the angle of the cable, there are only two cases that this design can pass through. The angle R1 will not become less than 45 degrees (Eq. (20)) unless we can push the robot with the cable (which is illogical). As for angle R2, we notice that it exceeds 45 in a certain area of the workspace (blue triangle).

3.4 Find the number of turns for each cable

Changing the cable lengths will be done by spinning them or releasing them from pulleys connected to motors. It must be ensured that winding the cable will not change the diameter of the pulley that could occur as a result of layering one cable turn on another. Therefore, the pulley was divided into 10 gears, and the number of turns in each location will be calculated.

$$W1 = \frac{LT - L1T}{3.56 \times \pi} \tag{30}$$

$$W2 = \frac{LT - L2T}{3.56 \times \pi} \tag{31}$$

$$W3 = \frac{LT - L3T}{3.56 \times \pi} \tag{32}$$

where, W: Number of turns of the cable.

3.5 Finding the location as a function of the number of turns

After giving the coordinates and giving the order to move, we will need a way to calculate the exact amount of error in the location. The error must be clear to the user and the robot at the same time and must be read directly. Therefore, there is a need to read the current location (x, y) with a sensor and compare it with the input which is the location we want to reach. To do this, we need an equation that links the number of cable turns (cable length) to the robot coordinates.

In the following, x and y positions are calculated based on the number of turns of each motor, W:

$$W1 = Wt - \frac{L}{3.56 \times 3.14} \tag{33}$$

$$Wt - W1 = \frac{L}{3.56 \times 3.14} \tag{34}$$

Assume that (Wt-W1=Wv) and substitute the equation for L then by squaring both sides:

$$124.76W^2v1 = x^2 - 30x - 200y + y^2 + 10281.25 \tag{35}$$

$$124.76W^2v2 = x^2 - 370x - 200y + y^2 + 44281.25 \tag{36}$$

$$124.76W^2v3 = x^2 - 200x + y^2 + 10000 \tag{37}$$

Making (37) equal to y^2, Eq. (36) equals -200y:

$$y^2 = 124.76W^2v3 - x^2 + 200x - 10000 \tag{38}$$

$$-200y = 124.76W^2v2 - x^2 + 370x - y^2 - 44281.25 \tag{39}$$

Substitute Eq. (37) into Eq. (36):

$$-200y = 124.76W^2v2 - x^2 + 370x - (124.76W^2v3 - x^2 + 200x - 10000) - 44281.25 \tag{40}$$

$$-200y = 124.76W^2v2 - x^2 + 370x - 124.76W^2v3 + x^2 - 200x + 10000 - 44281.25 \tag{41}$$

Substitute Eq. (37) and (36) in Eq. (35):

$$124.76W^2v1 = x^2 - 20x + 124.76W^2v2 - x^2 + 370x - 124.76W^2v3 + x^2 - 200x + 10000 - 44281.25 + 124.76W^2v3 - x^2 + 200x - 10000 + 10281.25 \tag{42}$$

After some further simplification, the robot coordinates are know related with the number of turns of the cables:

$$124.76W^2v1 = 350x + 124.76W^2v2 - 34000 \tag{43}$$

$$x = \frac{124.76W^2v1 - 124.76W^2v2 + 34000}{350} \tag{44}$$

$$x = \frac{124.76(Wt - W1)^2 - 124.76(Wt - W2)^2 + 34000}{350} \tag{45}$$

$$y = \sqrt{124.76(Wt - W3)^2 - x^2 + 200x - 10000} \tag{46}$$

3.6 The amount of error in the design

It is possible to note a large number of factors that can accumulate to form an actual error in the location coordinates. The most important of these factors are:

(1) The flexibility of the cable and the length of the elongation resulting from adding weight, which also changes with the change in the length of the cable

(2) The temperature and conditions in which the cable operates and the resulting expansion and contraction

(3) The diameter of the pulley that wraps the cable changes with each cycle

(4) other errors that would be difficult to calculate separately. due to changes in conditions and changes in the pieces used

For simplicity, the Simulink block diagram shown in Figure 7 is be divided into a group of sections and they will be mentioned in order of priority.

To eliminate these errors and ensure the greatest degree of accuracy, a controller is used. It should limit the amount of error and reduce it, even with the emergence of influential factors that may not be taken into account.

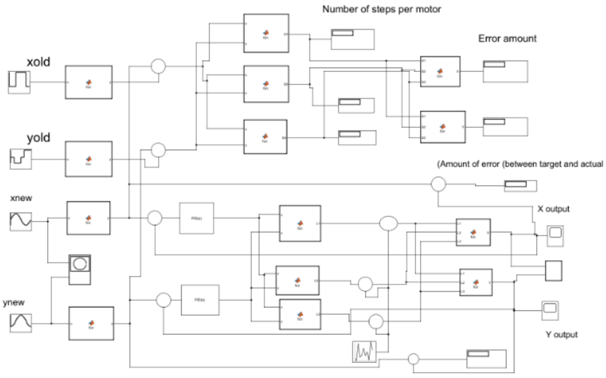

Number of steps per motor
Error amount
xold
yold
(Amount of error (between target and actual
xnew
X output
ynew
Y output

Figure 7. MATLAB diagram for robot control

4. SIMULATED DESIGN

To apply the equations mentioned in this research and determine their accuracy, MATLAB simulation is used. The system is created that is as close to reality as possible, where the input is the required (x, y) coordinates and the output is the real coordinates and the number of steps that each motor should revolve. Figure 8 represents the blocks for these requirements.

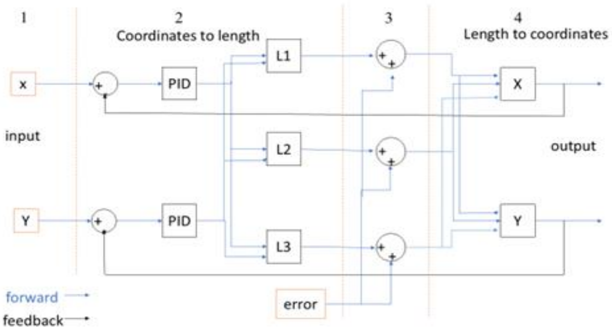


Figure 8. Block diagram for controlling the robot (coordinate calculation part)

1. In the first part, the entries signals are defined, in the form of a set of x and y coordinates. They are mentioned according to the sequence or path we want the robot to follow. As for the function, it is to ensure that the x and y coordinates will work at the same time (in parallel).

2. The error signals then enter the PID controller. The output signals are converted from coordinates to length. Each output from these three boxes represents the length of one of the three cables.

3. A white noise is then added to the control signal to imitate the unexpected error due to the cable tension and sag.

4. In the last part, a set of results are presented after converting the values from cable lengths to coordinates. The displayed results include the numerical value of x and y, the final graph (the actual path), the difference between the given values and the true values, and the final response for each signal.

Next, Figure 9 shows the block diagram that calculate the number of steps each motor spins during each movement.

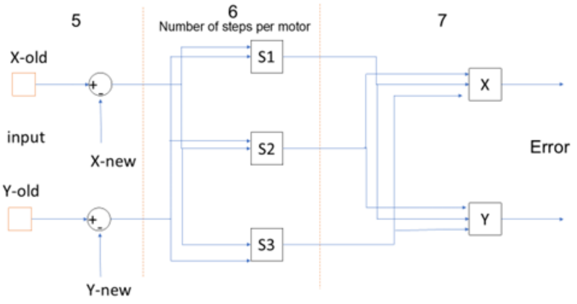


Figure 9. Block diagram for robot control (step count part)

5. To calculate the number of steps required to move from one location to another, one should know the current location and the next location to calculate the distance between them. It is the difference between the x-coordinates and the y-coordinates.

6. The coordinate differences are translated into motors turns that can be calculated based on the equations given previously.

4.1 PID controller

With the mechanical development of robots and the need for smooth paths, good and ideal control remains the most important element you may encounter when designing any robot, especially for non-linear systems [16]. The proportional-integral-derivative PID is an algorithm and controller that was created at the end of the last century. The controller is a closed circuit (it gives values and then receives errors to improve the output). Because of its ease of use and good results even with very small errors, it has become one of the most widely used controllers.

The following equation represents the signal when using a PID controller. It contains three parts (proportional–integral–derivative controller):

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau)d\tau + K_d \frac{de}{dt} e(t) \tag{47}$$

where, $u(t)$: Output signal (control it); $e(t)$: error; t: time (the present); τ: variable of integration; K_p: proportional signal; K_i: integral signal; K_d: derivative signal.

The controller consists of three sections, each of which has the ability to correct a specific factor in the system (or control it as needed). Each section also has the ability to control the entire system, meaning that raising the value of a factor may improve the system in a certain direction, but it will worsen in the other direction [17, 18].

Therefore, it is necessary to ensure that the correct and appropriate values are set, which, although they may have a negative impact. Table 1 shows the negative and positive impact of each element, the positive impact is clearer. It is difficult to choose these values randomly, so there are some methods that help in choosing the most appropriate values. In this research, artificial intelligence, specifically the genetic algorithm, will be used to determine the appropriate values [19].

Table 1. Effects of independent P, I, and D tuning on closed-loop response [19]

	Rise Time	Overshoot	Settling Time	Steady-State Error	Stability
Increasing KP	Decrease	Increase	Small Increase	Decrease	Degrade
Increasing KI	Small Decrease	Increase	Increase	Large Decrease	Degrade
Increasing KD	Small Decrease	Decrease	Decrease	Minor Change	Improve

4.2 Genetic algorithm

The genetic algorithm is one of the artificial intelligence tools that borrows from Darwin's laws of evolution and natural selection. In its work, the algorithm depends on proposing a set of values and finding the result for each value. Then it chooses the best of these values and combines them with each other to find new and hybrid values that give better results. It continues this process until it obtains the best result [20, 21]. In our case, there is a special library for the genetic algorithm in the MATLAB program that can be used, but the search cannot continue until it reaches the best values for the PID, but rather we must choose a certain number that may give good and reasonable values for the PID (we choose 10).

After the genetic algorithm performed ten cycles of hybridization Figure 10, it gave us the following values (Table 2), note that there are six values, three of which are for the x coordinate and three for the y coordinate.

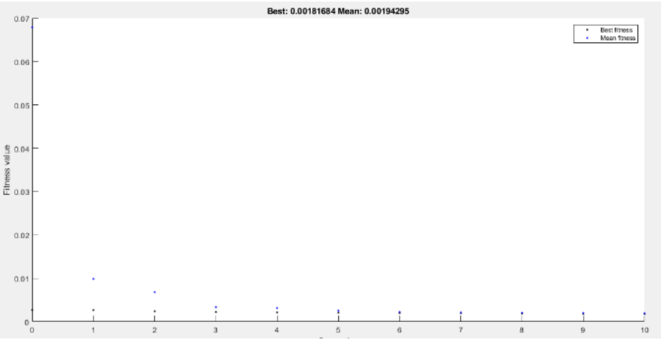
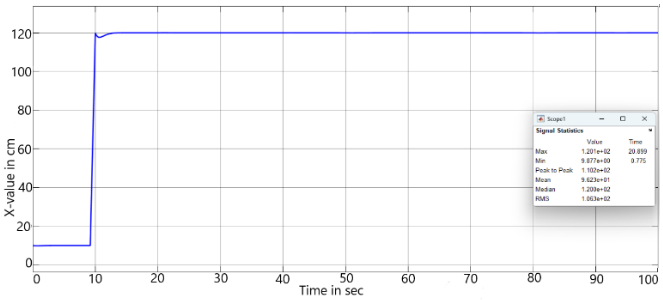


Figure 10. Generating PID values

Table 2. PID values generated by the genetic algorithm

	KP	KI	KD
X	32.5067	31.5158	11.9141
Y	33.5067	32.2787	31.8569

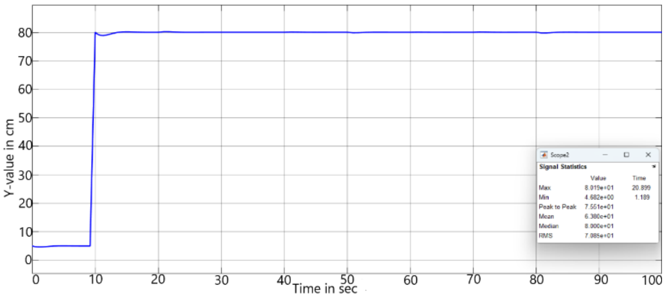
After applying the values given by the genetic algorithm, will see the response diagram to find out how stable the system is and whether its outputs are within the required range. Figure 11 and Figure 12 show the step response for the x and y coordinates respectively.



Signal Statistics

	Value	Time
Max	1.201e+02	20.899
Min	9.877e+00	0.775
Peak to Peak	1.102e+02	
Mean	9.623e+01	
Median	1.200e+02	
RMS	1.063e+02	

Figure 11. X-axis response diagram



Signal Statistics

	Value	Time
Max	8.019e+01	20.899
Min	4.682e+00	1.189
Peak to Peak	7.551e+01	
Mean	6.380e+01	
Median	8.000e+01	
RMS	7.085e+01	

Figure 12. Y-axis response diagram

4.3 Final accuracy

The motion path is one of the important elements in the design of any robot, and the accuracy of this path, its flexibility, and its commitment to the specified speed and time are among the most important results that we expect in any robot. This is because any defect or delay, even if it is simple, in these elements (speed and location with time) may lead to the accumulation of error, so that the result is disastrous. In addition, some applications have the path as part of the goal (i.e. there are a group of goals and not one goal, and these goals are along the path), so it is necessary for the robot to be in a specific location at a specific time. In this design, the input is defined as the x – y coordinates through which the robot will pass. Our ultimate goal is for the input coordinates to match the output as closely as possible.

To investigate the effectiveness of the GA tunned PID controller, several cases are studied. Rectangular and circular geometric shapes are considered as a desired path for the robot in two different cases. First, a rectangular path is drawn that the robot should runs along the edges of the workspace, to know the maximum point that the robot can reach smoothly. The path is shown in Figure 13.

In the circular path, a large number of close points were added. In this way, the circle is composed of connected arcs. The more points we add, the more curved the shape becomes as in Figure 14. However, the appropriate number of points must be used to generate an acceptable curvature, and the number of points and the distance between them must be proportional to the engine's characteristics.

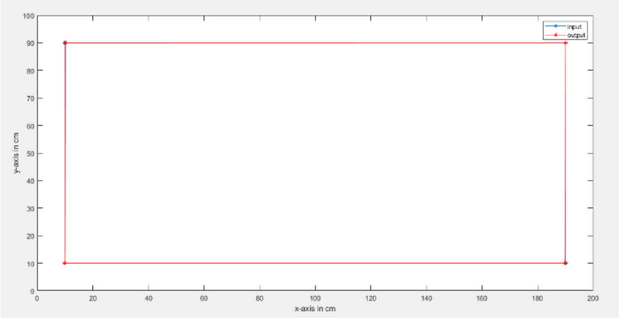


Figure 13. A rectangular path using PID and GA

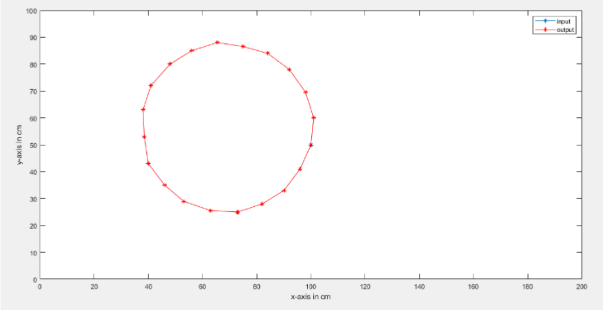


Figure 14. A circular path using PID and GA

When comparing the images above (input form with output form), we notice that the robot follows the coordinates given to it accurately with an error rate close to zero. The calculations and equations were able to proceed in the form of a rectangle and a circle, which contain the basics of any geometric shape (line and curve). That is, they can be relied upon in any path.

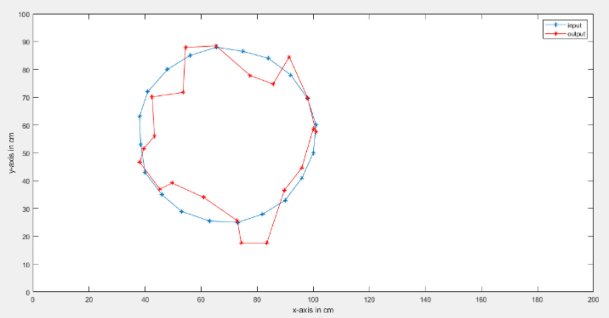


Figure 15. Circular path without PID or GA

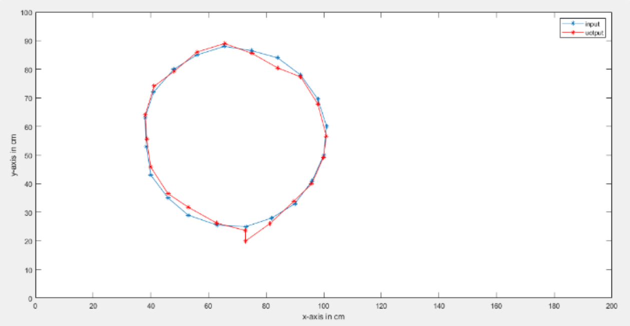


Figure 16. Circular path using PID only

Table 3. The effect of PID controller and GA

X-axis (output)			
Control method	PID+GA	PID	Without control
Error rate %	0.003587	0.022768	1.006933
Y-axis (output)			
Control method	PID+GA	PID	Without control
Error rate %	0.001242	0.655525	0.744278

In order to investigate the controller effectiveness, a comparison is made between the response with and without controller. Figure 15 shows the open loop robot path, while Figure 16 represents the path with manually tunned PID controller.

Although the difference between the three paths (using (PID +GA), using only PID, and without both) can be clearly observed with the naked eye, a calculation is made to prove it. Table 3 shows the error percentage for the three cases.

Error rates show the accuracy that the PID controller adds to handle errors and expected and unexpected factors. Adding a controller with appropriate values can make the design more flexible in working in different environments.

5. CONCLUSIONS

By the appearance of vertical buildings, the need for climbing robots for maintenance and cleaning purposes has been showed up. This type of robot is in the stage of continuous study and development (with the development of building design and other sciences). It has become part of the architectural designs of the building due to the importance of this type of robots in vertical buildings (where the robot parts were added to the building foundations). In the case of old buildings where it is difficult to change or add anything, cable robots are available that can be easily installed on any building. The existence of motors of these robots inside the robot facilitates their use, so there is no need to add any parts except for the set of cables that are connected to the required work area. However, this design is accompanied by a set of obstacles that appear in the form of changes that occur to the cables as a result of a number of factors such as weight, temperature, and mechanical design. To avoid this change which generates inaccuracy in motion and speed, genetic algorithms and PID controller have been utilized. They show high efficiency in dealing with any change in coordinates which includes errors due to expected or unexpected factors, and provide a real and effective solution to many of the mentioned problems. It can also be noted that adding (PID + GA) will reduce the error rate from (1.006933) to (0.003587) in the x-axis and from (0.744278) to (0.001242) in the y-axis, which is important to some amount in some applications.

This work mainly focuses on mathematical modeling and control of a new proposed climbing robot. Several adaptation and improvement have been left for future works which may include finding more accurate algorithm for calculating PID parameters, or moving forward to find better materials for cables. This design, which can be considered simple (and fairly effective), opens up a large number of ideas for future development research.

REFERENCES

[1] Castrillón-Restrepo, J.P., Jiménez-Gómez, S., Saldarriaga-Ángel, F. (2018). Development of a parametric cable driven planar robot. Dyna, 85(206): 49-58. https://doi.org/10.15446/dyna.v85n206.69233

[2] Bayani, H., Masouleh, M.T., Kalhor, A. (2016). An experimental study on the vision-based control and identification of planar cable-driven parallel robots. Robotics and Autonomous Systems, 75: 187-202. https://doi.org/10.1016/j.robot.2015.10.002

[3] Albagul, A., Asseni, A., Khalifa, O. (2011). Wall climbing robot: Mechanical design and implementation. Recent Advances in Circuits, Systems, Signal and Telecommunications, 1(1).

[4] Liu, R., Hong, Q., He, N.S.B. (2008). A miniature multi-joint wall-climbing robot based on new vibration suction robotic foot. In 2008 IEEE International Conference on Automation and Logistics, Qingdao, China, pp. 1160-1165. https://doi.org/10.1109/ICAL.2008.4636327

[5] Chattopadhyay, P., Majumder, A., Dikshit, H., Ghoshal, S.K., Maity, A. (2018). A bio-inspired climbing robot: Design, simulation, and experiments. IOP Conference Series: Materials Science and Engineering, 377(1): 012105. https://doi.org/10.1088/1757-899X/377/1/012105

[6] Murphy, M.P., Tso, W., Tanzini, M., Sitti, M. (2006). Waalbot: An agile small-scale wall climbing robot utilizing pressure sensitive adhesives. In 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, Beijing, China, pp. 3411-3416. https://doi.org/10.1109/IROS.2006.282578

[7] Gosselin, C. (2014). Cable-driven parallel mechanisms: State of the art and perspectives. Mechanical Engineering Reviews, 1(1): DSM0004. https://doi.org/10.1299/mer.2014dsm0004

[8] Bartoš, M., Kuric, I., Bulej, V. (2019). The usage possibility analysis of parallel robots for vision-based monitoring. In Designing, Researches and Exploitation'2019 vol. 1, Wydawnictwo Naukowe Akademii Techniczno-Humanistycznej w Bielsku-Białej, pp. 15-22.

[9] Riechel, A.T., Ebert-Uphoff, I. (2004). Force-feasible workspace analysis for underconstrained, point-mass cable robots. In IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA'04. 2004, New Orleans, LA, USA, pp. 4956-4962. https://doi.org/10.1109/ROBOT.2004.1302503

[10] Khosravi, M.A., Taghirad, H.D. (2011). Dynamic analysis and control of cable driven robots with elastic cables. Transactions of the Canadian Society for Mechanical Engineering, 35(4): 543-557. https://doi.org/10.1139/tcsme-2011-0033

[11] Bosscher, P., Williams II, R.L., Bryson, L.S., Castro-Lacouture, D. (2007). Cable-suspended robotic contour crafting system. Automation in Construction, 17(1): 45-55. https://doi.org/10.1016/j.autcon.2007.02.011

[12] Hiller, M., Fang, S., Mielczarek, S., Verhoeven, R., Franitza, D. (2005). Design, analysis and realization of tendon-based parallel manipulators. Mechanism and Machine Theory, 40(4): 429-445. https://doi.org/10.1016/j.mechmachtheory.2004.08.002

[13] Roberts, R.G., Graham, T., Lippitt, T. (1998). On the inverse kinematics, statics, and fault tolerance of cable-suspended robots. Journal of Robotic Systems, 15(10): 581-597. https://doi.org/10.1002/(SICI)1097-4563(199810)15:10%3C581::AID-ROB4%3E3.0.CO;2-

P

[14] Gorman, J.J., Jablokow, K.W., Cannon, D.J. (2001). The cable array robot: Theory and experiment. In Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No. 01CH37164), Seoul, Korea (South), pp. 2804-2810. https://doi.org/10.1109/ROBOT.2001.933047

[15] Tang, X. (2014). An overview of the development for cable-driven parallel manipulator. Advances in Mechanical Engineering, 6: 823028. https://doi.org/10.1155/2014/823028

[16] Ahmed, M.S., Mary, A.H.M., Jasim, H.H. (2021). Robust computed torque control for uncertain robotic manipulators. Al-Khwarizmi Engineering Journal, 17(3): 22-28. https://doi.org/10.22153/kej.2021.09.002

[17] Alkamachi, A., Erçelebi, E. (2017). Modelling and genetic algorithm based-PID control of H-shaped racing quadcopter. Arabian Journal for Science and Engineering, 42: 2777-2786. https://doi.org/10.1007/s13369-017-2433-2

[18] Liu, G.F., Li, H.W. (2016). Design of stepper motor position control system based on DSP. In Proceedings of the 2017 2nd International Conference on Machinery, Electronics and Control Simulation (MECS 2017), Taiyuan, China. https://doi.org/10.2991/mecs-17.2017.38

[19] Li, Y., Ang, K.H., Chong, G.C. (2006). PID control system analysis and design. IEEE Control Systems Magazine, 26(1): 32-41. https://doi.org/10.1109/MCS.2006.1580152

[20] Dagher, K.A., Al-Araji, A.S. (2013). Design of an adaptive PID neural controller for continuous stirred tank reactor based on particle swarm optimization. Al-Khwarizmi Engineering Journal, 9(4): 46-53.

[21] Forrest, S. (1996). Genetic algorithms. ACM Computing Surveys (CSUR), 28(1): 77-80.