

# Compensating the End-Effector Position of Semi-Flexible Robot Using Smart Structure Technique

Dr. Hussein M. Al-Khafaji

*Department of Machines & Equipment Engineering  
University of Technology/ Baghdad  
College of Engineering  
E-mail: dr.hussalkhafaji@yahoo.com*

Prof. Dr. Muhsin J. Jweeg

*Department of Mechanical Engineering  
Alnahrain University/ Baghdad  
College of Engineering  
muhsinji@yahoo.com*

**Abstract-** This study proposed the using of a smart structure principle with a methodology for reducing the difference (error) between the actual position (for a semi-flexible robot) and the theoretically calculated position (for a rigid robot) on-line. The methodology depends on the interfering between the maps of the two cases; the rigid case (ideal), and the deformed case (actual) for compensation of error. According to this methodology, a class (program) was built using the visual Basic.Net; this class is called the compensation class. In this work, a two degrees of freedom articulated type lightweight semi-flexible robot was used. This robot is confined to move in a vertical plane. The smart structure system was represented by; the sensors for measuring the error deformation variables were mounted on the two links of the robot, Data acquisition (DAQ) system and the actuators of the joints. The smart structure robot systems were designed and built in this work. Also, to control the smart structure robot's systems, software was built using Visual Basic.Net. Compensation tests have been achieved on the complete system to check the performance and results of the compensation system. This system showed a good improvement in the performance of robot for compensation and reduction in the error between the ideal position (rigid robot) and the practical position (measured position). The average error after the compensation reduced to 12.32 times in the x-direction and 21.76 times in the y-direction.

**Keywords:** Semi- Flexible robot, Smart Structure, Compensation of Error.

## 1. Introduction

Most of the existing robotic manipulators are bulky designed and built from heavy materials to maximize the stiffness in an attempt to minimize the vibration of the end-effector and to achieve a good position accuracy. The existing heavy rigid manipulators are shown to be inefficient in terms of power consumption or speed with respect to the operating payload. In order to improve the industrial productivity, it is required to reduce the weight of the arms and/or to increase their speed of operation. For these purposes, it is very desirable to build light weight robot manipulators. Compared to the conventional heavy and bulky robots, light weight link manipulators have the potential advantage of lower cost, larger work volume, higher operational speed, greater payload-to-manipulator-weight ratio, smaller actuators, lower energy consumption, better maneuverability, better transportation costs and safer operation due to reduced inertia. But, the greatest disadvantages of these manipulators are the deflection due to low stiffness, so this factor reduces the accuracy of the robot [1 and 2].

The existence of the flexural effect of the robot links complicates the problem for finding the end effector mapping. The reason is that the end-effector position of the light weight robot is not only related to joint displacements

but also to link elastic displacements. As the result, the end-effector position of a light weight robot cannot be obtained based on the measured joint angles and its kinematics as a rigid robot could. So, the need for a directly measuring the error parameters (deflection, elongation and rotation of arms) and joint displacement is to compute the actual position of the end-effector and then compensate the deviation of the end-effector from the ideal position (rigid robot). From this combination of measuring and compensating of error, the idea of using a smart structure robot was proposed.

The smart structure may be defined as a structure that has the capability to sense, measure, process and diagnose at critical locations any change in selected variables, and to command appropriate action to preserve structural integrity and continue to perform the intended functions. The variables may include deformation, temperature, pressure and changes in state and phase and may be optical, electrical, magnetic, chemical or biological [3 and 4].

Due to important of reducing and compensation of error in the position of end-effector of robot many researches are engaged in the investigation in this topic.

Wang Et. al. [5], Xu et. al. [6] and [7] used a laser beam as a slop sensor to measure the deviation (deflection) in position of link. Then each one of them has his own methodology for compensation and control position. Sarkar et. al. [8] developed and demonstrated through simulation results two simple numerical algorithms to minimize the end-point error for static case and tracking error for dynamic case. The proposed algorithms computed the input torques in an off-line computation for a two links flexible manipulator under gravity. Stieber et. al. [9] developed Photogrammetric measurement and robot control techniques for rapid and precise positioning of payloads with flexible space manipulators. Gong et.al. [10] studied the effect of non-geometric errors, specifically compliance errors and thermal errors, on the robot performance. Based on this, a general methodology was proposed to calibrate these errors by an inverse calibration method.

Zhang et. al. [11] presented the critical issues and methodologies to improve the robotic machining performance with flexible industrial robots. Compared with CNC machines, the stiffness of industrial robots was significantly lower, this methodology consisted of stiffness modeling and real-time deformation compensation. Olabi et. al. [12] proposed a feed rate planning method adapted for a continuous machining with an industrial robot. Starting from a parametric representation of the tool-paths, this method generated a smooth jerk limited law of motion for the tool, respecting the robot joints constraints.

There are many research deal with using smart structure in the robot, here it will be reviewed some of them:

Samanta [13] used the concept of intelligent structures as the compensator for elastic vibrations. This idea was illustrated by a two link flexible manipulator. Gong et. al. [14] investigated the controller design for a single link smart material robot, which combined both the advantages of flexible robots and piezoelectric materials. Shin and Choi [15] and Bottega et. al. [16 and 17] built a control system consisted of piezoelectric actuators attached to the surfaces of the flexible links to control undesirable oscillations and vibration of the links. Molter et. al. [18] introduced a technique for optimization of placement and size of a piezoelectric material on link for the optimal vibration control of flexible robot links.

From the above literature review, it is clear that a substantial amount of work has been carried out on the correction and compensation of error for robot. Thus, many works have been carried out concerning the error inherent during the design or by calibration after exposing the robot to specified circumstances, like specified loads, and a few researches have studied the correction of error on line during the working of robot in service, even if they carried out, but also for a specified and known conditions. The correction and compensation of error on-line are difficult, especially for the light and flexible robot because the presence of the parameters of deformation which complicate the direct and inverse kinematic equation of the robot, also, its need to measure these parameters on time (on line) to compensate the error, for instance references [5,6 and 7] where they used a laser beam technique to measure the deflection of link, this technique has some restriction to use in many robot's application. The using smart structure in flexible robot has been studied in many researches, as mentioned above, but most of these studies considered with correction and decreasing the effects of vibration.

In this work: a smart structure robot was proposed to work with a suggested methodology of compensation to improve the ability of light weight robot to correct the position on-line.

The kinematic equations of robot

The kinematic analysis of a mechanical system means the deformation of the position, velocity and acceleration of the various mechanical elements forming the mechanism under consideration. The combination of position velocity and acceleration of an element at a certain time is referred to henceforth as the state of this elements [19]. The direct kinematic equation (forward kinematic equation) provides functional relationship between the displacements of all the joints and the last link position and orientation involved in the open kinematic chain [20].

In robot's end-effector positioning, it is required to find the generalized parameters that lead the end-effector to the specified position and orientation. This is done by solving the forward kinematic equation for a set of joint displacements for each end-effector's position. This problem called the inverse problem, and the result equation called the inverse kinematic equation.

#### Forward kinematic equation of the rigid robot (without error)

The description of the position and orientation of the end-effector with reference to the base frame as function of joint's displacement can be done by using the Denavit-Hartenberg notation. The degree of freedom of the robot

arm manipulator which is used in this work, is two degrees of freedom, and confined to move within the vertical plane with the kinematic parameters listed in the Table (1). The kinematic equations for end-effector in x ( $p_x$ ) and y ( $p_y$ ) directions respectively, are [21]:

$$p_x = a_2 c_1 c_2 - a_2 s_1 s_2 + a_1 c_1 \quad (1)$$

$$p_y = a_2 s_1 c_2 + a_2 s_2 c_1 + a_1 s_1 \quad (2)$$

Because of the robot confined to move in the plane, the movement in z- direction will be:

$$p_z = 0 \quad (3)$$

Where:

$$c_i = \cos \theta_i, \quad s_i = \sin \theta_i$$

$a_i$  is the length of robot's arms.

Equations (1) and (2) can be expressed as  $p = f(\theta_i)$ , where  $\theta_i$  is joint space.

#### Forward kinematic equations of a semi-flexible robot (with error)

The link flexibility can cause elastic deformations of the structural members of the manipulator, resulting in large end-effector errors, especially in long reach manipulator systems. Hence as a result, the frames defined at the manipulator joints are displaced from their expected locations. So that, the deflection parameters (generalized error parameters) should be taken in consideration in the forward kinematic equation and this equation will be as follow [22 and 23]:

$$p_x = c_1 c_2 [cd_1(\delta x_2 + a_2) - sd_1 \delta y_2] - c_1 s_2 [cd_1 \delta y_2 + sd_1(\delta x_2 + a_2)] - s_1 c_2 [sd_1(\delta x_2 + a_2) + cd_1 \delta y_2] + s_1 s_2 [sd_1 \delta y_2 - cd_1(\delta x_2 + a_2)] + c_1(\delta x_1 + a_1) - s_1 \delta y_1 \quad (4)$$

$$p_y = s_1 c_2 [cd_1(\delta x_2 + a_2) - sd_1 \delta y_2] - s_1 s_2 [sd_1(\delta x_2 + a_2) + cd_1 \delta y_2] + c_1 c_2 [sd_1(\delta x_2 + a_2) + cd_1 \delta y_2] + c_1 s_2 [cd_1(\delta x_2 + a_2) - sd_1 \delta y_2] + c_1 \delta y_1 + s_1 [\delta x_1 + a_1] \quad (5)$$

$$p_z = 0 \quad (6)$$

Where:

$p_x, p_y$  and  $p_z$  are the translation part of transformation matrix of robot, and it will be called  $p_{defl} = f(\theta_i, \epsilon)$ ,

Where  $\theta_i$  and  $\epsilon$  are joint space and deflection parameters, respectively.  $\delta x, \delta y, \delta z, d\psi, d\beta$  and  $d\phi$  are the deflection parameters of links as shown in figure (1).

$$c_i = \cos \theta_i, \quad cd_i = \cos d\phi_i, \quad ca_i = \cos \alpha_i$$

$$s_i = \sin \theta_i, \quad sd_i = \sin d\phi_i, \quad sa_i = \sin \alpha_i$$

As mentioned above, a robot arm manipulator with two degrees of freedom confined to move within the vertical plane was used; figure (1) shows the parameters of deformation. The other kinematic parameters of this robot are listed in the table (1).

#### Inverse kinematic equations of rigid robot

From equations (1) and (2) the value of joint displacement were found as follow [24]:

$$\theta_1 = \tan^{-1} \frac{p_y}{p_x} - \sin^{-1} \frac{a_2}{\sqrt{p_x^2 + p_y^2}} \sqrt{1 - \left( \frac{p_x^2 + p_y^2 - a_1^2 - a_2^2}{2 a_1 a_2} \right)^2} \quad (7)$$

$$\theta_2 = \tan^{-1}(c_2 \pm \sqrt{1 - c_2^2}) \quad (8)$$

Inverse kinematic equations of a semi-flexible robot (with error)

The presence of flexibility of robot's arm complicates the inverse problem because of the deformation parameters exist in the forward kinematic equation, these parameters are not depend on the position and orientation of robot's arms only but also, on the external and inertial forces. So, in this work, it was used the solution of inverse kinematic equation which is in reference [23]. This solution presented by using the neural network (as function approximation) to solve the inverse kinematic equation of semi-flexible robot. The result neural network was used to build a class for compensation called (neural class), this class implemented within the compensation class as the part for inverse kinematic equation calculations.

### Compensations of error

In order to cause the desired motion of the end-effector of rigid serial robot manipulators, it can be directly control the motion of the joints, using a control law based on a kinematic equation which describes a mapping relations between the workspace and the joint space. For robot arms with the presentation of deflections, the problem becomes very difficult, because of the motions of end-effector are not only determined by the joints behavior but also by the link flexural behavior.

In order to control  $P_{defl}$  (equations 4 to 6), the errors of  $P_{defl}$  due to  $\epsilon$  must be compensated by  $\theta$ . This compensation is not always possible, because  $\theta$  does not make all possible motions of  $P_{defl}$  due to the kinematic concentrate.

In this work, a strategy and method was proposed for solving the problem of compensation which was based on the interfering between the maps of the paths of the two cases; the rigid case and with deformation case. Depending on this method, a class was built by using visual Basic.Net to compensate the position's error of the end-effector. This error methodologies and computer programs have been integrated within the main program as class for controlling the motion of the manipulator in the compensation mode. The control strategy can be specified as:

- 1- The displacement of the end-effector due to the link flexural behavior will be considered as "error".
- 2- The error must be compensated by the joint variables ( $\theta_1$ ) and ( $\theta_2$ ).

### Methodology for compensation

The end- effector position and orientation error  $\Delta X$  is defined as the vector that represents the difference between the real position (measured positions) and orientation of the end -effector and the ideal or desired one (computed position) [25] and [26]:

$$\Delta X = X_{tip}^a - X_{tip}^I \quad (9)$$

$$\Delta X = \begin{cases} x_{tip}^a - x_{tip}^I \\ y_{tip}^a - y_{tip}^I \end{cases} \quad (10)$$

Here  $X_{tip}^a$  and  $X_{tip}^I$  are the vectors that represent the position and orientation of the end-effector for the real and ideal case equations (4 to 6) and (2 to 3) respectively.

If the robot in real world is perfect like in CAD model, the error  $\Delta X$  would be zero.

### Smart structure

A smart structure is basically a distributed parameter system that employs sensors and actuators at different finite locations on the beam, uses of one or more microprocessors that analyze the responses obtained from the sensors, uses different control logics to command the actuators to respond in a desired fashion, and brings the system to the desired state [27].

Two beam models are in common use in the structural mechanics namely the Euler-Bernoulli beam model and the Timoshenko beam model [4]. In this work for modeling the beam, the Euler-Bernoulli (classical method) was used.

Lateral loads acting on a beam will cause the beam to deflect, thereby deforming the longitudinal axis of the beam into a curved line. In engineering practice, it is often necessary to determine the deflections at various points along the axis of the beam and the slope of the curve (angle of rotation).

Before the load applied, the longitudinal axis of the beam is straight. After bending, the axis of the beam becomes a curve, as represented by the line AB in the figure (2). The curve AB, called the deflection curve of the beam.

In this work, the experimental relations between the loads and deflection  $\delta y$  were found by applying different values of load and measuring the deflection at the tip for each load, and then by using the curve fitting, the relations were found. To find the rotation of the tip of the beam  $d\phi$  experimentally, the measuring of the deflections at two points along the beam was done, as shown in figure (2), at the points where the strain gauge No.3 and at the tip, and by applying these values of the deflections into the following equation [28]:

$$\tan d\phi = \Delta\delta y / \Delta x \quad (11)$$

Where:

$\Delta\delta y$  represents the difference between the two deflections.  $\Delta x$  represents the distance between the two points where the deflections measured.

$\delta x$  was experimentally found by extracting the strain from two sensors at the upper and lower surface at the same position of the strain gauge No.1 and No. 2 as shown in figure (2). The difference value represents the extension of the beam in the longitudinal direction, as represented by the relation:

$$\delta x = \epsilon \cdot L \quad (12)$$

These values of  $\delta x$ ,  $\delta y$  and  $d\phi$  will be compensated in the kinematic equations (4) and (5) to be the deflection parameters.

## Experimental Work

### Part one: The Rig and experimental relations

In this work, the test rig was represented by two degrees manipulator (two semi-flexible links) confined to move in a vertical plane. The two links were designed to have properties gathering the lightness approximate to the flexible beam and stiffness approximate the rigid beam; the two links are hollow rectangular beams were made from the aluminum type (6061-T6), as shown in figures ( 3 ) and ( 4 ), where table (2) lists their weight and length. Each link has its own motor and control system for movement. The motors which are used in the manipulator are of low inertia AC servo motors, for the first joint 100 watt type (ASMT01L250AK) and for the second joint 400 watt type (ASMT04L250AK), these motor from Delta Electronic Company. To control these AC servomotors, a two drivers type ASDA-A series were used (ASDA-A04 and ASDA-A01 for the first and second joints, respectively) . Also, the rig has measurement systems and data acquisition system (DAQ) which consist of the following components:

- **Sensors (strain gauges).**

The system of sensors, consist of three strain gauges bonded to each link. The experimental analysis of deflection is accomplished by measuring the strain of a part under load and inferring the existing state of deflection from the measured strain. In this work, the voltage measured from the sensor was calibrated to represent the value of deflection without needing for calculating the strain and then converted to deflection. The distribution of the strain gauges and dial gauges on the two links and the application of the load are shown in figures (5) and (6). The distribution of the strain gauges was taken by dividing the whole length of link to three regions. Three strain gauges were used for each link, two on the upper surface of the link numbered 1 and 3, and the third one on the lower surface exactly under the strain gauge No.1 to use it to recognize the direction of bending and to find the value of the link's elongation, this strain gauge numbered 2. The dial gauges distribution was based upon two things; the first dial gauge was put at the position of the strain gauges No.3, while the second one was put at a position where the joint of the next link effects. The two links of the manipulator were separately tested under different loads and angles. Each link was assembled to its joint to change the angle during the test; these joints were rigidly fixed to the frame. For each angle, a series of different values of load were applied by hanging masses to the tip of the link under test, measuring the deflections by using two dial gauges at two points on the link, reading the voltage of the strain gauges through the DAQ system, and then obtaining the relations between the voltage of strain gauges and the deflections as shown in figures (5) and (6). The relation between the deflection and the voltages of strain gauges were estimated as follow:

#### For link -1:

The relation between the tip deflection and the voltage of the strain gauge No.1 is:

$$\delta y_{tip} = 0.0026 V + 8 \times 10^{-5} \quad (13)$$

The relation between the tip deflection and the voltage of the strain gauge No.2 is:

$$\delta y_{tip} = 0.001 V + 3 \times 10^{-5} \quad (14)$$

The relation between the deflection at the point 31cm from the center of the shaft of the first joint and the voltage of the strain gauge No.3 is:

$$\delta y_{at\ 31\ cm} = 0.0008 V - 2 \times 10^{-5} \quad (15)$$

#### The Link-2:

The relation between the tip deflection and the voltage of the strain gauge No.1 is:

$$\delta y_{tip} = 0.0016 V + 6 \times 10^{-5} \quad (16)$$

The relation between the tip deflection and the voltage of the strain gauge No.2 is:

$$\delta y_{tip} = 0.0022 V + 3 \times 10^{-5} \quad (17)$$

The relation between the deflection at the point 36 cm from the center of the shaft of the second joint and the voltage of the strain gauge No.3 is:

$$\delta y_{at\ 36\ cm} = 0.0022 V - 3 \times 10^{-5} \quad (18)$$

The purpose of measuring the deflection at the strain gauge No.3 is to find the value of tip rotation through the equations (15) as follow:

$$\tan d\phi = \frac{\delta y_{tip} - \delta y_{at\ strain\ gauge\ NOs}}{\Delta x} \quad (19)$$

Where :

$\Delta x$  is the distance between the tip and the position of strain gauge No.3.

The  $\delta x$  was measured by using the two strain gauges 1 and 2, by subtracting their voltages to calculate the difference which represents the elongation (The difference value represents the extension of the beam in the longitudinal direction, as represented by equation (16), these values of  $\delta x$ ,  $\delta y$  and  $d\phi$  will be used in the kinematic equations (4) and (5) to be the deflection parameters. Also, the experimental relation between the deflection of the link- 2 and the load at the tip is:

$$load * \cos \theta_{2x} = \frac{\delta y_{tip} + 0.0211}{0.1143} \quad (20)$$

Where:

$\theta_{2x}$  Is the angle of link-2 with x- axis.

- **Data Acquisition system hardware (DAQ)**

The type of DAQ hardware (interface card) was (Mini LAB 1008 from Measurement Computing Company).

- **DAQ Software**

In addition to the control class, the main program has a class which was built in order to control data acquisition card and collect the data to be analyzed before giving the required control signal to the motor drivers. This class was built by visual Basic.Net.

### Part two: Compensation of error

- **The class of compensation**

Figure (7) shows the schematic diagram of the compensation class. This class was built by using the Visual Basic.Net. the steps of execution of the compensation class were explained as follow:

1. Input target position  $x_2$  and  $y_2$ .
2. Read the configuration variables  $\theta_1$  and  $\theta_2$  at position 1.
3. Read deformation data from DAQ system.
4. Compute the load from equation (20) for over load protection.

5. Compute  $\theta_1$  and  $\theta_2$  at position 2 using the rigid inverse kinematic equations (7) and (8).
6. Go to the position 2.
7. Read the sensors voltages at position 2 and compute the deformation variables using equations (13) to (18).
8. Compute the actual position at 2  $x_2^a$  and  $y_2^a$  using equations (4) and (5).
9. Find the errors using relations:

$$\Delta X = x_2^a - x_2$$

$$\Delta Y = y_2^a - y_2$$

10. Compare with small value (0.01 mm), if the  $\Delta X$  and  $\Delta Y$  are equal or less than this value, the system not need to compensate. If  $\Delta X$  and  $\Delta Y$  larger than this value, the required angles  $\theta_1^a$ ,  $\theta_2^a$  for compensation will compute by feeding the target  $x_2$  and  $y_2$  with error variables to inverse kinematic equation with error (neural class).
11. Calculate the difference of the angles

$$\Delta\theta_1 = \theta_1 - \theta_1^a$$

$$\Delta\theta_2 = \theta_2 - \theta_2^a$$

12. Give the order to the motors driver to move to specified position.

#### • Compensation of error

The proposed method of compensation that was explained previously and the program of the compensation class, were based on the interfering between the trajectory maps of the rigid robot equations (1) and (2) and the error kinematic equations (4) and (5), as shown in figures (8) and (9). This interference and crossing points between the two trajectory paths represent the way to compensate the error. Though, the crossing point represents a sharing point between the two paths (for the two equations). Due to the contribution of the generalized parameters in the flexible kinematic equation, it is not conditionally; these two equations have the same configuration variables ( $\theta_1$  and  $\theta_2$ ) at these crossing points. So, it can get a benefit from the presence of error to try for compacting the rigid trajectory with trajectory with error by moving the robot to the point, where the combination of the configuration variables and the generalize parameters lead or close to the intended location.

The aim of the compensation is to reach the values of px and py that resulted from the equation of rigid robot [equations (1) and (2)].

The results of positioning compensation were taken for different points, using three different mass loads (0, 1 and 2 kg) tested with the range of angles ( $\theta_1 = 40$  to  $290$ ) and ( $\theta_2 = 350$  to  $600$ ). Each test for one value of load was experimented three times, and the average of the results was taken.

The measuring of the coordinates of the end effector was experimentally done by fixing a pen on the end of the second link to draw the path of end effector on a scalar board (with grid resolution of 1mm) which was installed beside the manipulator as shown in figures (10) and (11), and by taking photos to the drawn path on board at each stations to analyze these photos by using the software Bytescout Graph Digitizer Scout to get the end effector coordinates. This software is a tool that helps to digitize oscillograms, photos and different graphs from the scanned

pictures and other images. Data can be exported to Excel, CSV format, and other programs. Figures (12) and (13) show the difference between the end-effector positions calculated by using the measured generalized errors parameters in the equations (4) and (5) and the end-effector positions measured by analyzing the photo by using Bytescout Graph Digitizer Scout software.

The procedure of this work encompasses two stages:

The first stage presented by moving the robot to specific points according to the range of angles by using the single mode of the main program to give enough time for stopping at each point (static case), to take photo for the analysis by the software Bytescout Graph Digitizer Scout, and measuring the generalized parameters. This procedure was done without the activation of the compensation class.

The px and py were calculated by three methods: the first by using the equations (4) and (56), the second by equations (1) and (2), and the third by using the analysis of photo by a Graph Digitizer software. According to the above three methods, the px and py were called (deflection px, py value), (rigid px, py value) and (photo px, py value), respectively.

Figures (14) to (15) show the difference between the configurations of the above three methods of calculation, these differences represent the errors. The error values were calculated by subtracting the deflection value px py from the photo px py value, and also by subtracting the rigid px py value from the deflection px py value and from the photo px py value, respectively. The average values of error are listed in the Table (4).

In the second stage, the same procedure that carried out in the first stage was followed but with activation of compensation class to do the compensation process. In this stage, also the same process of measuring in the first stage was done. Figures (18) and (19) reveal the errors between the rigid px py and the photo px py after the compensation. Also, these figures show how the compensation was done, and the positioning accuracies are much improved compared to the results without compensation.

## 2. Discussion of results

It's difficult to achieve the desired aim precisely (position of the rigid robot) due to different parameters that affect the position beside the deflection parameters, like the back lash of the joint, noise that comes from the electric power source affects the measurement system, and the drawing of the path may have error due to movement of the pen on the scalar board. The approximation of neural and the kinematic constraint for the robot spatially at the configurations were  $\theta_2$  be zero (the two links become straight).

Before compensation, the average error between the rigid px value and photo px value was  $(2.04 \cdot 10^{-3})$  m, and after compensation it was enhanced to  $(1.656 \cdot 10^{-4})$  m, the maximum value of errors before the compensation was  $(6.87 \cdot 10^{-3})$  m with a compensation decrease to  $(2.461 \cdot 10^{-4})$  m, while after the compensation, the maximum value of error became  $(4.397 \cdot 10^{-4})$  m which corresponded to  $(5.33 \cdot 10^{-3})$  m before the compensation. Also, the average value of error for py before the compensation was  $(4.856 \cdot 10^{-3})$  m, and after compensation became  $(2.231 \cdot 10^{-4})$  m.

<sup>4</sup>) m, the maximum value reduced from  $(1.091 \times 10^{-2})$  m to  $(9.667 \times 10^{-4})$  m which also represents the maximum value after the compensation.

So, from the above results of correction, it could be seen that it is not conditionally the maximum value of error before the compensation is still the maximum value of error after the correction, because, as said before, the operation of compensation is based on finding the points of interfering between the two paths ( the path from the rigid kinematic equation with the path from the kinematic equation with error), so that it may be found a point belongs to the trajectory path of the kinematic equation with error more closely to the point, where the maximum error is. While one may not find a close point to the other point on the rigid path, where the error was less than the maximum (before compensation), this is because of two main reasons, the kinematic constraint for the robot configuration and the error that presents between the path of kinematic equation with error that used in the compensation class and the actual path. So, after the completing of compensation, it may be found another set of errors with another maximum error but with values less than the set before the compensation.

### 3. Conclusions

From the results have been obtained from this work, the following remarks can be listed:

- The method of compensation indicated a good improvement of robot's accuracy for positioning. The reduction of the average of error between  $p_x$  and  $p_y$  from the rigid kinematic equation and the measured  $p_x$  and  $p_y$  by the photo analysis are 12.32 times for  $p_x$  and 21.76 times for  $p_y$ . These averages represent the average of data were taken for different values of load.
- The kinematic constraints of the used robot affect the correction of the error due to presence of some points out of the trajectory map of the kinematic equation with error.
- The first and second arms were tightened to move in the limited ranges of angles. This reduces the probability of interfering of maps for rigid and deformed paths. So, it might be with extending the range of angles, the ability for compensation increases.
- With this system, the robot will have a good ratio payload weight to arms weight, and improved accuracy.
- The method of the analyzing of photos, using the software Bytescout Graph Digitizer Scout to get the end effector coordinates, shows a very well accuracy of results, reaching to (0.001 mm).
- To enhance the system of reading and to reduce the effect of noise an average of 100 samples were taken for a single read, and it was given a very well reduction of to the offset point from the linear fitting relation (the root mean square of the results is between (0.997) to (0.999).

### 4. References

[1] **Hiroshi Ueno, Yangsheng Xu and Tetsuji Yoshida**, "Modeling and Control Strategy of A 3-D Flexible Space Robot", IEEE, RSJ International Workshop on Intelligent Robots and Systems IROS, 1991: pp. 978-983.  
 [2] **Masaru Uchiyama and Zhao Hui Jiang**, "Compensability of End-Effector Position Errors for

Flexible Robot Manipulators" , American Control Conference, IEEE, 1991: Volume 2, pp.1873-1878.  
 [3] **A. V. Srinivasan and D. Michael Mcfarland**, "Smart Structure: Analysis and Design", Book, Vol. fifth, United State of America: The press Syndicate of the Unviersity of Cambridge, 2001.  
 [4] **J. Khanna**, "Modeling and Control of Smart Cantilever Beam", MSc. Thesis, Bombay: I.I.T., 2004.16.  
 [5] **Wen-Jieh Wang, Shui-Shong Lu and Chen-Fa Hsu**, "Experiments on the Position Control of a One Link Flexible Robot Arm", Transactions on Robotics and Automation, 1989: Vol5, No.3, pp. 373-377.  
 [6] **W. L. Xu and S. K. Tso**, "Conceptual Design of an Integrated Laser-Optical Measuring System for Flexible Manipulator" , IEEE, 1996: pp. 1247-1252.  
 [7] **W. L. Xu, S. K. TSO and X.S.WANG**, "SensorBasedDeflection Modeling and Compensation Control of Flexible Robotic Manipulator", Mech. Mach. Theory, Elsevier Science, 1998: Vol. 33, No.7, pp. 909-924.  
 [8] **Pitman Kumar Sarkar, Motoji Yamamoto and Arika Mohri**, "On the Trajectory Planning of a Planar Elastic Manipulator Under Gravity", Transactions on Robotics and Automation, 1999: Vol. 15, No.2, pp. 357-362.  
 [9] **Michael E. Stieber, Michael McKay and George Vukovich** , "Vision Based Sensing and Control for Space Robotics Applications", IEEE Transactions on Instrumentation and Measurement, 1999: pp. 807-812.  
 [10] **C. Gong, J. Yuan and J. Ni** , "Nongeometric Error Identification and Compensation for Robotic system by Inverse Calibration" ,International Journal of Machine Tools & Manufacture, 2000: pp. 2119-2137.  
 [11] **Hui Zhang, Jianjun Wang, George Zhang, Zhongxue Gan, Zengxi Pan, Hongliang Cui and Zhenqi Zhu**, "Machining with Flexible Manipulator: Toward Improving Machining Performance" , International Conference on Advanced Intelligent Mechatronics, California-USA: IEEE/ASME, 2005: pp. 1127-1132.  
 [12] **A. Olabi, R.Bearee, O. Gibaru and M. Damak**, "Feed Rate Planning for manching with Industrial Six-Axis Robots", Control Engineering Practice, 2010: pp. 471-482.  
 [13] **Biswanath Samanta**, "Active Control of Flexible Robot Arms Using the Concept of Intelligent Structures", IEEE International Workshop on Intelligent Robot and System, 1990: pp. 941-947.  
 [14] **J. Q. Gong, S. S. Ge and T. H. Lee**, "Model Free Controllers of a Single Link Smart Material Robot " , American Control Conference Philadelphia. Pennsylvania: AACC, 1998: pp. 3065-3069.  
 [15] **Ho cheol Shin and Seung Bok Choi**, "Position Control of a Two-Link Flexible Manipulator Featuring Piezoelectric Actuators and Sensors" , Mechatronics (Elsevier Science Ltd.), 2001: pp. 707-729.  
 [16] **Valdecir Bottega, Alexandre Molter and Jun S. O. Fonseca**, "Vibration Control of Manipulators with Flexible Non Prismatic Links Using Piezoelectric Actuators and Sensors", Mathematical Problems in Engineering, April 2009: pp. 1-16.  
 [17] **Valdecir Bottega, Rejane Pergher and Jun S.O. Fonseca**, "Simultaneous Control and Piezoelectric Insert Optimization for Manipulators with Flexible Link", Brazil

Society of Mechanical Scientifics and Engineer XXXI, No. 2, June 2009: pp.105-116.

[18] **Alexandre Molter, Otavio A Alves da Silveira, Jun S. Ono Fonseca and Valdecir Bottega**, "Simultaneous Piezoelectric Actuator and Sensor Placement Optimization and Control Design of Manipulators with Flexible Links Using SDRE Method", *Mathematical Problems in Engineering*, 2010: pp. 1-23.

[19] **Yoram Koren**, "Robotics for Engineers", United States of America: McGraw-Hill, Inc., 1985.

[20] **H. Asada and J. J. E. Slotine**, "Robot Analysis and Control", Book, New York: John Wiley and Sons, 1985.

[21] **Richard P. Paul**, "Robot Manipulators: Mathematics, Programming and Control the Computer Control of Robot Manipulators", Book, Printed in the United States of America: The Massachusetts Institute of Technology, 1983

[22] **Alkhafaji, Hussein Mohammed**. "Kinematic Analysis of Smart Structure Tow Links Semi- Flexible Robot". Baghdad : PhD Thesis, University of Technology-Mechanical Department, 2011.

[23] **Muhsin J. Jweeg, Hussein M AlKhafaji**. "Kinematic Analysis of Semi-Flexible Robot". *Journal of Engineering and Technology*, University of Technology, Vol. 30, No. 13., Baghdad , Iraq, 2012.

[24] **Rachid Manseur**, "Robot Modeling and Kinematics", Book, India: Firewall Media, 2007.

[25] **C. Mavroidis, S. Dubowsky, P. Drouet, J. Hintersteiner and J. Flanz**, "A Systematic Error Analysis of Robotic Manipulators: Application to a High Performance Medical Robot", *International Conference in Robotic and Automation*. Albuquerque, 1997.

[26] **H. K. Lim, D. H. Kim, S. R. Kim and H. J. Kang**, "A Practical Approach to Enhance Positioning Accuracy for Industrial Robots" , ICROS-SICE International Joint Conference, Fukuoka, Japan: Fukuoka International Congress Center, Japan, 2009: pp. 2268-2273.

[27] **G. Dreyfus**, "Neural Networks Methodology and Applications", Book, Berlin-Germany: Springer- Verlag Berlin Heidelberg, 2005.

[28] **Stephen P. Timoshenko and James M. Gere**, "Mechanics of Materials", Book, London-Britain: Van Nostrand Reinhold Company Ltd., 1978.

Table 3 Network training parameters for (Kinematic inverse neural network)

Training parameters	Value
Epochs	5000
Goal	$1 \times 10^{-8}$
Minimum gradient	$1 \times 10^{-10}$
Learning rate	$1 \times 10^{-5}$
Learning increment	0.9

Table 4 The average of Errors

	Error average Rigid and Deflection	Error average Rigid and Photo	Error average Deflection and photo
px m	$2.006 \times 10^{-3}$	$2.04 \times 10^{-3}$	$4.2667 \times 10^{-5}$
py m	$4.546 \times 10^{-3}$	$4.856 \times 10^{-3}$	$1.398 \times 10^{-4}$

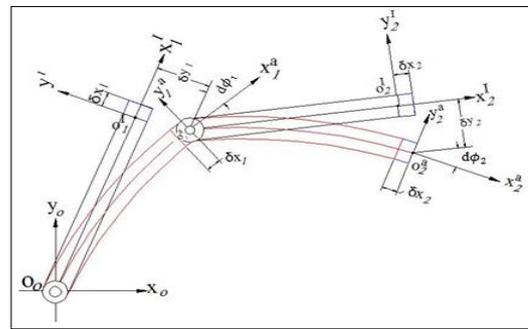


Fig. 1 The coordinates systems of two links of semi-flexible robot arm

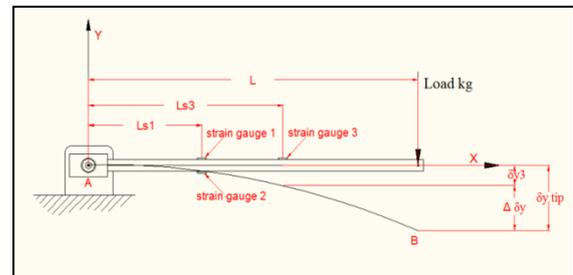


Fig. 2 Deflection curve of the beam

Table 1 The kinematic parameters of robot

Joints	$\theta_i$	$d_i$	$\alpha_i$	$a_i$
1	$\theta_1$	0	0	$a_1$
2	$\theta_2$	0	0	$a_2$

Table 2 The parameters of two links

Link	Length (cm)	Weight (kg)
One	57	0.167104
Two	62	0.18284

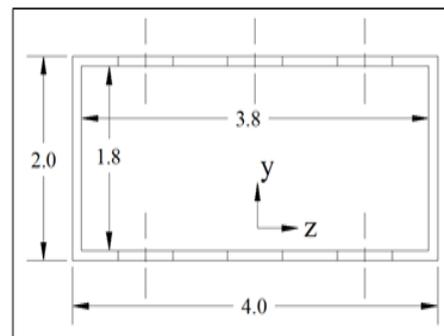


Fig. 3 The section of the two links section of two links

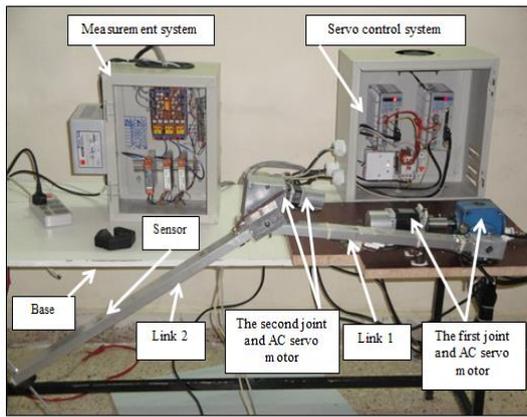


Fig. 4 The two degrees robot arm manipulator

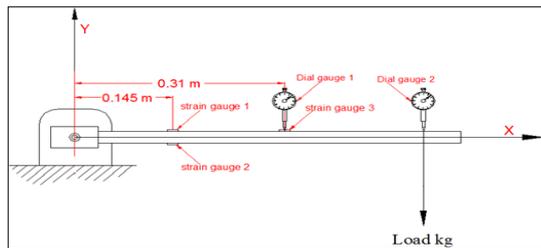


Fig. 5 Distribution of strain gauges for link-1

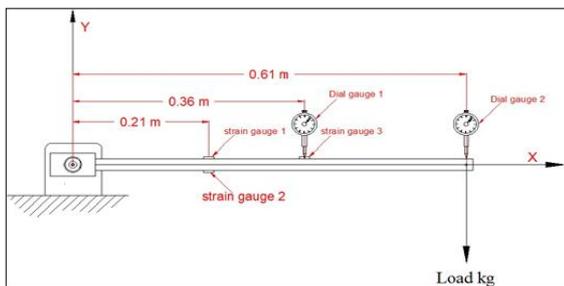


Fig. 6 Distribution of strain gauges for link-2

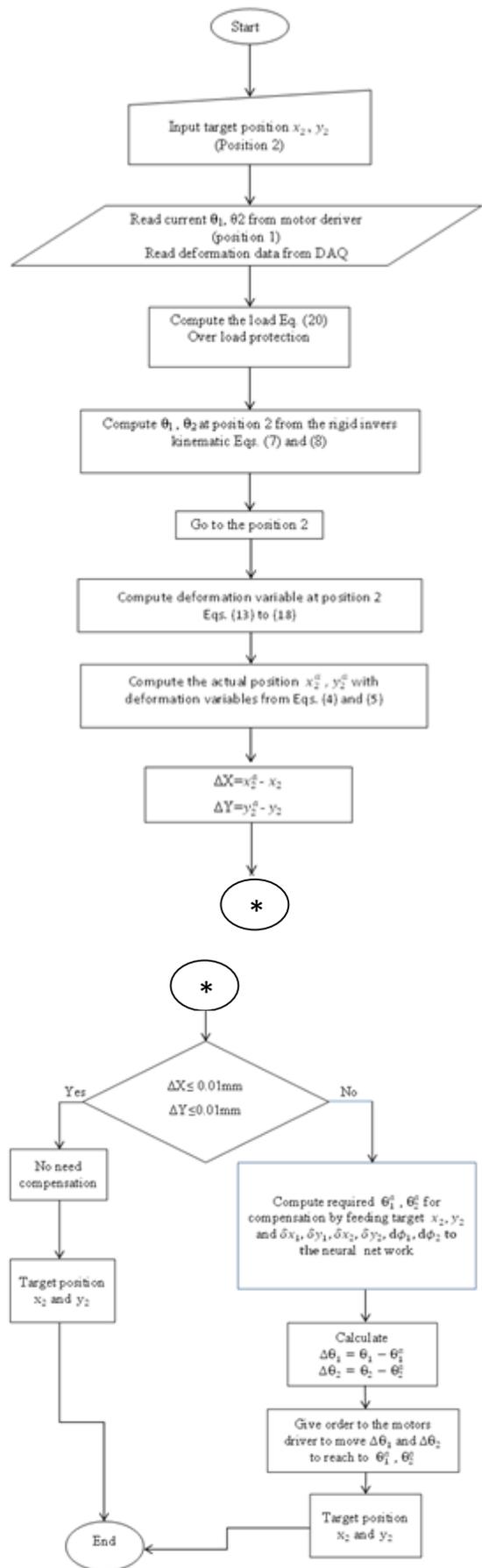


Fig. 7 Schematic diagram of the

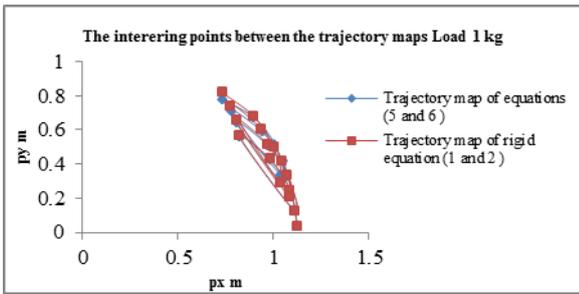


Fig. 8 The interfering between the maps of the rigid equations (1 and 2) and the error equations (4) and (5)  
Load 1kg

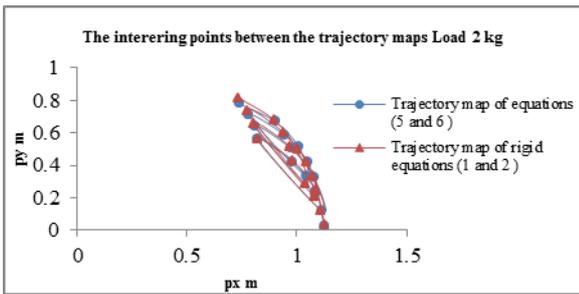


Fig. 9 The interfering between the trajectory maps of the rigid equations (1) and (2) and the error equations (4) and (5)  
Load 2kg



Fig. 10 The scalar board and the path of end effector

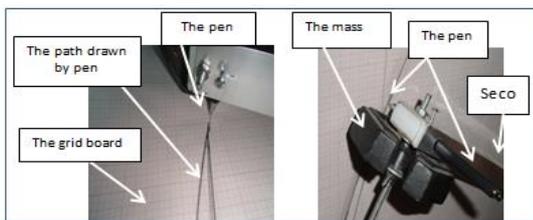


Fig. 11 The pen at the end of the second link and the grid of the scalar board

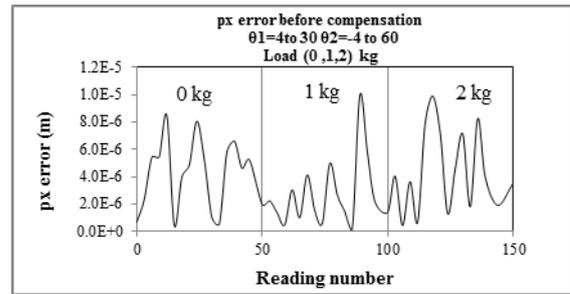


Fig. 12 The error between Deflection px and photo px Before compensation

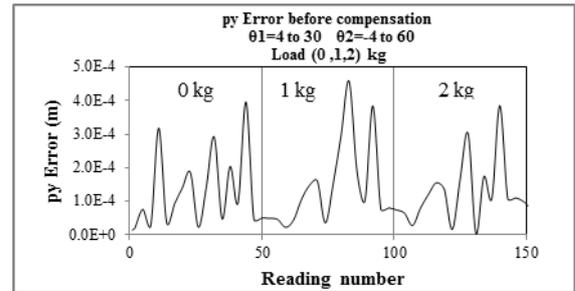


Fig. 13 The error between deflection px and photo px Before compensation

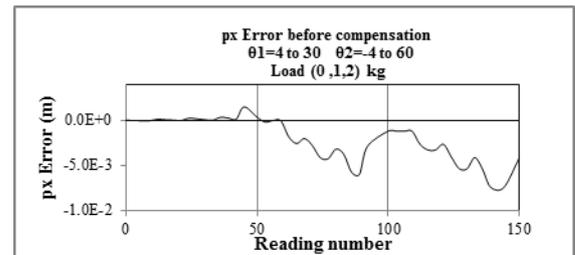


Fig. 14 The error between rigid px and deflection px Before compensation

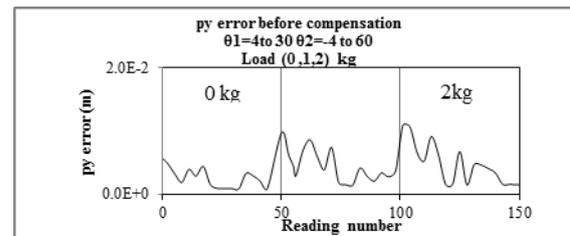


Fig. 15 The error between rigid py and deflection py Before compensation

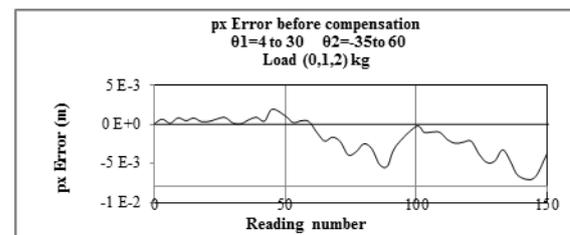


Fig. 16 The error between rigid px and photo px Before compensation

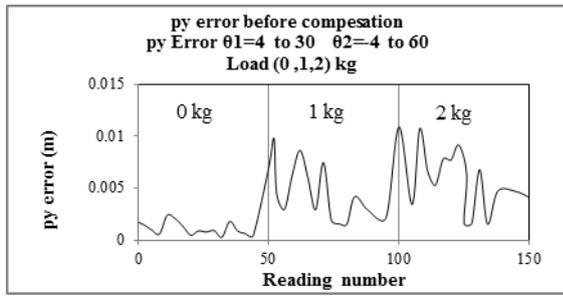


Fig. 17 The error between rigid py and photo py Before compensation

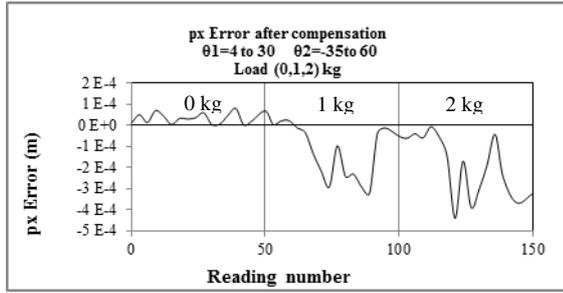


Fig. 18 The error between rigid px and photo px After compensation

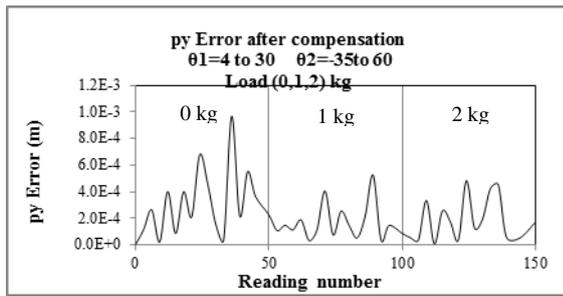


Fig. 19 The error between rigid py and photo py After compensation