Intelligent Automation And Soft Computing, 2018
Copyright © 2018, TSI® Press
Vol. 24,no.2,443–457

Active Control of a Piezoelectric Actuated Four-Bar Mechanism Deployed in Robotics Applications

Qais A. Khasawneh[1, 3], Mohammad Abdel Kareem Jaradat[1,2], Mohammad Al-Shabi[4], and Hala Khalaf[1]

Mechanical Engineering Department, Jordan University of Science and Technology, P.O.Box 3030 Irbid, Jordan [1].
Mechanical Engineering Department, American University of Sharjah, P.O.Box (2666) Sharjah, UAE.[2]
Electromechanical Engineering Department, Institute of Applied Technology, Abu Dhabi, UAE [3]
Mechanical Engineering Department, University of Sharjah, P.O.Box (27272), Sharjah, UAE [4]

ABSTRACT
This work presents a new micro-positioning system that is implemented in an inchworm robot to move into desired locations. The system consists of four-bar mechanism; one link is fixed, and each one of the remaining links carries a piezoelectric actuator (PZT). PZTs are specifically chosen since they provide fast response and small displacements; up to ±30 µm for ±100 Volts. The system's mathematical model is derived and is numerically simulated by MATLAB. Three fuzzy PI controllers, which are tuned automatically by genetic algorithm, are designed to control the system. Results indicate an error of less than 1% although disturbances present.

KEY WORDS: Fuzzy, Genetic Algorithm, Micro-positioning, PI controller, Piezoelectric Actuators.

1 INTRODUCTION

FOUR bar mechanism is a closed chain linkage that consists of four rigid links connected together by joints. Due to their structural simplicity, flexibility, low cost and ease manufacturing, it could be found in many applications such as; bicycles, train suspensions, windshield wipers, motors, robot arms, hydraulic actuators and many more. Depending on the links' lengths and angle, many useful mechanisms could be formed. For instance, if the joints between the four bars move in parallel planes, the mechanism is called planar four bar mechanism. Despite the diversity of the applications, the function of the linkage could be classified into function generation, path generation and motion generation. Therefore, by using position analysis, the location of every link during motion can be specified and used to generate various output motions and track certain paths. To activate the links movements, active actuators; e.g. motors, smart materials such as piezoelectric actuators and shape memory alloys, and passive actuators; which have internal source of energy such as springs and dampers or a combination, are used [Mermertas, 2004; Von Albrichsfeld and Tolle, 2002; Liaw and Shirinzadeh, 2008; Liaw and Shirinzadeh, 2008; Liaw, Shirinzadeh, and Smith, 2008].

Piezoelectric actuators (PEAs) are widely used due to their characteristics. They have fast response, simple structure, high resolution, high bandwidth, high stiffness and high natural frequency. Piezoelectric actuators are easy to control and need low current for operation thus, eliminating heating problems. Moreover, they are able to manage extremely small displacements in the range of 10 nm to 100 µm. However, they have some major drawbacks due to hysteresis behavior and high driving voltage which can be overcome using controllers [Adriaens, de Koning, and Banning, 2000].

PEA four bar mechanisms have been developed lately for ultra-precision movements and micro manipulation in biomedical and surgical applications such as the inchworm [Lu, Zhu, Lin, and Guo, 2009; Oh, Choi, Nam, Bu, and Kim, 2010; Mrad, Abhari, and Zu, 2003; Lim, et al., 2008; Lianzhi and Weichong, 2012; Wu and Zhou, 2004; Eigoli and Vossoughi, 2012; Liu and Li, 2010; Qiao, Shang, and Goldenberg, 2012; Rincón and Castro, 2003; Kotay and Rus, 2000; Ghanbari, Rostami, Noorani, and Fakhrabadi, 2008; Lobontiu, Goldfarb, and Garcia, 2001]. The latter allows flexibility and inspection in

CONTACT Qais A. Khasawneh qakhasawneh@just.edu.jo>

narrow environments where high precision and high accuracy are needed and enables the development of smart, multifunctional, adaptive and environment friendly robots. This motion can adapt with the surrounding environment and pass obstacles and rough surfaces [Mermertas, 2004; Von Albrichsfeld and Tolle, 2002; Liaw and Shirinzadeh, 2008; Liaw and Shirinzadeh, 2008; Liaw, Shirinzadeh, and Smith, 2008].

Piezoelectric actuators were mathematically modeled by many researchers [Liaw and Shirinzadeh, 2008; Liaw and Shirinzadeh, 2008; Liaw, Shirinzadeh, and Smith, 2008]. More advanced structures were implemented later. Those included the work of Yao et al in [Yao, Dong, and Ferreira, 2007] and, Pozzi and King in [Pozzi and King, 2003]. The former developed a parallel kinematic micro positioning XY stage which had two chains of two serially-connected PEA parallelogram four-bar mechanism with two degrees of freedom; in the x and y axes. On the other hand, Pozzi and King developed a model for PEA consisted of multilayers of piezoelectric materials; the layers were connected in series but electrically were connected in parallel to reduce the voltage used, and amplify the displacements by summation of small steps.

As discussed earlier, PEA has some major drawbacks including hysteresis behavior and the need to high driving voltage. Those issues were addressed and solutions were proposed in [Adriaens, de Koning, and Banning, 2000]. Several works tried to minimize the hysteresis behavior including Lin and Yang work of [Lin and Yang, 2006], and Bashash and Jalili of [Bashash and Jalili, 2007]. The former proposed an accurate model of a piezo-actuated positioning stage which was incorporated nonlinear non-symmetric hysteresis effect resulted by the PEA. To compensate uncertainties and external disturbances in the proposed model, a PI feedback control associated with the feedforward compensating based on the hysteresis observer was designed. On the other hand, Bashash and Jalili addressed the problem of hysteresis behavior in piezoelectrically driven Micro and Nano positioning systems. For modeling the hysteresis effect in their system, the original PI hysteresis operator based on Preisach model was modified for compensation. A feedforward controller was implemented on the system for tracking multi frequency trajectories, and a feedback controller was added which formed a robust controller to stabilize the closed loop system. Later, several control algorithms have been applied to reduce the hysteresis. Those included the work of classical control [Von Albrichsfeld and Tolle, 2002; Yao, Dong, and Ferreira, 2007], proportional position feedback and force feedforward controller implemented [Shim, Cho, and Kim, 1997], PID [Yan and Yan, 2009], adaptive robust control implemented [Liaw and Shirinzadeh, 2008; Liaw and Shirinzadeh, 2008; Liaw, Shirinzadeh, and Smith, 2008], and some intelligent controllers such as; neural network in [Liaw and Shirinzadeh, 2009; Aguilar, Tapia, Valderrabano, and Rivas, 2015; Kumarakulasingam and Agah, 2013; Godjevac and Steele, 1998], fuzzy theory in [Gundogdu and Erenturk, 2005; Wongsoontorn and and Zhuang, 2013; Chang and Lilly, 2003; Pin and Watanabe, 1995; Civelek, Lüy, Çam, and Barışçı, 2016], H∞ controller [Xianmin, Changjian, and Erdman, 2002], LQR [Zhang, 2004], and the Sliding mode controller presented by Hwee Choo Liaw et al [Liaw, Shirinzadeh, and Smith, 2007].

This work introduces a PZT actuated mechanism which tracks a given command and provides the required length. As proposed in [Lin and Yang, 2006], the classical PI controller was selected to reduce the steady state error and improve the tracking performance to get the required length from the PEA. However, due to modeling uncertainties, the unacceptable system performance under the influence of external disturbances and noise, the Fuzzy logic PI controller is proposed and implemented; and the controller parameter is tuned using GA to get optimal performance. The tuned controller performance was compared to the classical approach. The fuzzy logic PI shows good improvement in the system performance compared to the classical one.

The PEAs were used in several applications especially the manipulators. For example, the flying insect wing based on two piezoelectrically actuated four-bar mechanism for stroke amplification proposed in [Sitti, 2003], and the biomimetic fish robot fin which was actuated by two piezoelectric ceramics with a four-bar mechanism in [Heo, Wiguna, Park, and Goo, 2007] to mimic the tail's motion. Manipulators performance has been increased through several work including Mermertas's optimum design that maximized the performance index which depends on the input link location, Yans' integrated optimum design for a servo motor actuated four-bar mechanism in [Yan and Yan, 2009], and the work of Li and Xu proposed in [Li and Xu, 2006], which was proposed as a two-dimensional compliant parallel micromanipulator using two piezoelectric actuators. Those were modeled using the pseudo-rigid body model and were analyzed to get the forward, inverse kinematics and velocity equations. Their model was used to get the maximum workspace area depending on the piezoelectric actuators and the hinges.

The rest of this work is divided as follows: the Mathematical models of the proposed PEAs and the overall four bar mechanism are introduced in section 2. The control system including classical and fuzzy controllers is presented in section 3. Applying the proposed PI fuzzy controller to the proposed system is discussed in section 4, while section 5 provides with a real application for the proposed method. Section 6 contains the conclusions of this work.

2 THE PROPOSED PIEZOELECTRICALLY-ACTUATED FOUR-BAR MECHANISM

THE piezo-actuated positioning mechanism for precise motion and tracking systems rely mainly on two parts; piezoelectric actuator and movement mechanism which is here the four bar mechanism. Each part will be analyzed and equations of motion will be derived in this section to come up with the final system that represents the final motion of the joints.

2.1 Modeling Piezoelectric Actuators

Several models were built to compensate for the hysteresis effect in PEAs. The Bouc-Wen model was used for modeling PEAs [Lin and Yang, 2006], and this was the starting point of this work. The model of the PEA with one bar is shown in Figure 1. It consists of a mass-spring-damper mechanical system represented by second order differential equations. The motion of the PEA is represented using the following differential equations [Lin and Yang, 2006];

$$m\ddot{x} + b\dot{x} + kx = u_h = k(du - h) + \rho \quad (1)$$

$$\dot{h} = \alpha_0 d\dot{u} - \beta_0|\dot{u}|h - \gamma_0\dot{u}|h| \quad (2)$$

where h is the hysteresis nonlinear term, x is the displacement, $\dot{x}$ and $\dot{h}$ are the derivatives of x and h with respect to time, respectively, u is the applied voltage, and d is the ratio of displacement for the applied voltage. The parameters m, b and k are the mass, stiffness factor and the damper coefficient, respectively. $\rho=kx_o$ where x_o is the initial displacement as u=0. The parameters α_0, β_0, γ_0 are designing parameters that determine the hysteresis loop's magnitude and shape. Their optimal values were listed in Table 1.

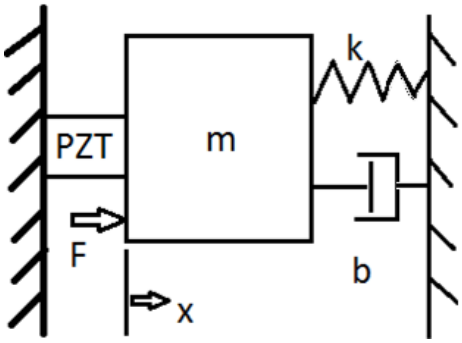


Figure 1. Model of PEA stage, [Lin and Yang, 2006].

Table 1. The optimal modeling parameters of PEA.

m_o	0.148 *kg*	d	$1.59\text{x}10^{-8}$
B_o	129.5 *N s/m*	α_0	0.385
K_o	$2.95\text{x}10^6$ *N/m*	β_0	0.0235
P	0.05 *N*	γ_0	0.0495

In this research, the PEA is considered ideal with no hysteresis effect; therefore, equation (1) becomes linear as follows;

$$m\ddot{x} + b\dot{x} + kx = kdu \quad (3)$$

By transforming equation (3) to the Laplace form it becomes;

$$ms^2X + bsX + kX = kdU \quad (4)$$

After separating the output X and dividing by the input U in equation (4), the transfer function for the PEA is obtained as follows;

$$\frac{X}{U} = \frac{kd}{ms^2 + bs + k} = \frac{kd/m}{s^2 + \frac{b}{m}s + \frac{k}{m}} \quad (5)$$

Substituting the values of the parameters in table (1) into equation (5) and multiplying by a linearization gain gives;

$$\frac{X}{U} = \frac{5.964058}{s^2 + 875s + 1.9932\times10^7} \quad (6)$$

The proposed PEA operates in the range of (-100 *V*) to (100 *V*) for input voltage and (-30 μm) to (30 μm) for output displacement. The linear relationship between the voltage and the displacement is shown in Figure 2.

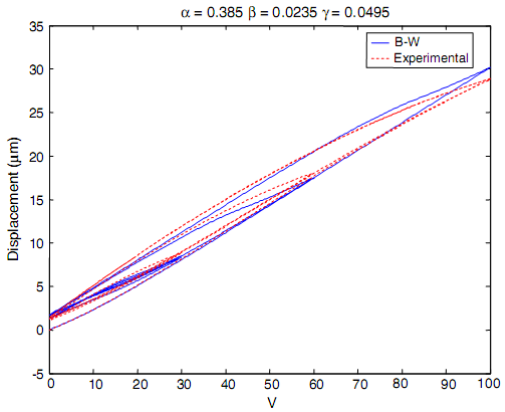


Figure 2. A comparison between theoretical and experimental curve, [Lin and Yang, 2006].

2.3 Modeling of the Four-Bar Mechanism

The four bar linkage used here consists of four bars which have equal lengths initially connected by four joints. The joints move freely in two directions, but since the motion range specified by the PEAs is very small, the joints movement is limited to micro meters. In the linkage, three links can move while the ground link is fixed parallel to the ground with an angle equal to zero. The combination of PEAs with four bar mechanism started gradually, first using one PEA on one bar and moving it as explained in the previous sub-section. Second, using two PEAs on two bars and finally using three PEAs on the three bars. Each case will be demonstrated next with the model derivation. It should be noted that only the kinematics of the model will be analysed.

2.3.1 Modeling Two Piezoelectrically Actuated Four-Bar Mechanism

The first scenario analysed is the use of two PEAs in two links, where the length of these links is changed according to the input voltage supplied to the PEAs, while the third link has fixed length during the process as well as the ground link. The links start initially with length L. According to the desired location, the joint between the two moving links reach that location while the links move and extend or contract simultaneously. This scenario is described by Figure 3 with the used variables on the figure. Assuming that θ_3=90º and θ_4=0º, the derivation of the expressions for ΔL_1 and ΔL_2 in terms of x and y are done as follows:

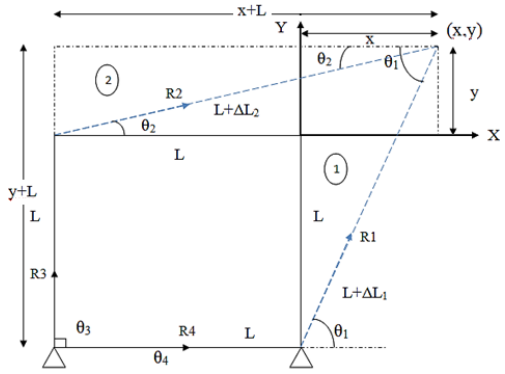


Figure 3. Two piezoelectrically actuated four bar mechanism.

The vector equation is defined as:

$$\boldsymbol{R_1 + R_4 = R_2 + R_3} \quad (7)$$

The x-component equation is;

$$\boldsymbol{(L+\Delta L_1)cos\theta_1 + L = (L+\Delta L_2)cos\theta_2} \quad (8)$$

The y-component equation is;

$$\boldsymbol{(L+\Delta L_1)sin\theta_1 = (L+\Delta L_2)sin\theta_2 + L} \quad (9)$$

Now eliminating θ_2 by squaring equations (8) and (9) and adding them;

$$\boldsymbol{(L+\Delta L_2)^2 = \begin{bmatrix} 2L^2 + (L+\Delta L_1)^2 + \\ 2L(L+\Delta L_1)[cos\theta_1 - sin\theta_1] \end{bmatrix}} \quad (10)$$

Rearranging equation (10) as follows;

$$\boldsymbol{cos\theta_1 - sin\theta_1 = \lambda} \quad (11)$$

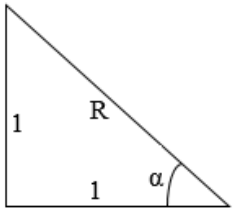
where $\lambda = \frac{(L+\Delta L_2)^2-(L+\Delta L_1)^2-2L^2}{2L(L+\Delta L_1)}$. Assuming a triangle with the following measures;

$\boldsymbol{R = \sqrt{1^2+1^2} = \sqrt{2}}$

$\boldsymbol{sin\alpha = \frac{1}{\sqrt{2}}}$,

$\boldsymbol{cos\alpha = \frac{1}{\sqrt{2}}}$

$\boldsymbol{\alpha = \tan^{-1}(1) = 45^o}$



Divide equation (11) by $\sqrt{2}$;

$$\boldsymbol{\frac{1}{\sqrt{2}}cos\theta_1 - \frac{1}{\sqrt{2}}sin\theta_1 = \frac{\lambda}{\sqrt{2}}} \quad (12)$$

Or $\boldsymbol{sin\alpha \times cos\theta_1 - cos\alpha \times sin\theta_1 = \frac{\lambda}{\sqrt{2}}}$ (13)

The left hand side of equation (12) is an identity for substracting the sin of two angles;

$$\boldsymbol{\sin(\alpha - \theta_1) = \sin(45^o - \theta_1) = \frac{\lambda}{\sqrt{2}}} \quad (14)$$

$$\text{Or } \boldsymbol{\theta_1 = 45^o - \sin^{-1}\left(\frac{\lambda}{\sqrt{2}}\right)} \quad (15)$$

From equation (8), θ_2 can be found in terms of θ_1 as follows;

$$\boldsymbol{cos\theta_2 = \frac{(L+\Delta L_1)cos\theta_1 + L}{(L+\Delta L_2)}} \quad (16)$$

$$\text{OR } \boldsymbol{\theta_2 = \cos^{-1}\left[\frac{(L+\Delta L_1)cos\theta_1+L}{(L+\Delta L_2)}\right]} \quad (17)$$

From triangle (1), the following triangular relationships are obtained;

$$\boldsymbol{(L+\Delta L_1)^2 = (L+y)^2 + x^2} \quad (18)$$

$$\boldsymbol{\tan\theta_1 = \frac{y+L}{x} \rightarrow \theta_1 = \tan^{-1}(\frac{y+L}{x})} \quad (19)$$

From triangle (2), the following triangular relationships are obtained;

$$\boldsymbol{(L+\Delta L_2)^2 = y^2 + (x+L)^2} \quad (20)$$

$$\boldsymbol{\tan\theta_2 = \frac{y}{x+L} \rightarrow \theta_2 = \tan^{-1}(\frac{y}{x+L})} \quad (21)$$

From equations (15), (18), (19) and (20), the following equation is obtained;

$$\boldsymbol{\Delta L_1 = \frac{y^2+(x+L)^2-(L+y)^2-x^2-2L^2}{2\sqrt{2}L\sin\left[45^o - \tan^{-1}(\frac{y+L}{x})\right]} - L} \quad (22)$$

From equations (17) and (21), the following equation is obtained;

$$\boldsymbol{\Delta L_2 = \frac{(L+\Delta L_1)cos\left[\tan^{-1}(\frac{y+L}{x})\right]+L}{\cos\left[\tan^{-1}(\frac{y}{x+L})\right]} - L} \quad (23)$$

2.3.2 Modeling Three Piezoelectrically Actuated Four-Bar Mechanism

The second scenario is to implement three PEAs in the four bar linkage to give more flexibility to the joint movement within the specified limits of the PEAs. The three links can move freely when voltage is supplied to the PEAs according to the desired point location. The ground link is fixed with length L while the other links start with length L then increase or decrease to adopt the movement. The proposed scenario for analysis is shown in Figure 4.

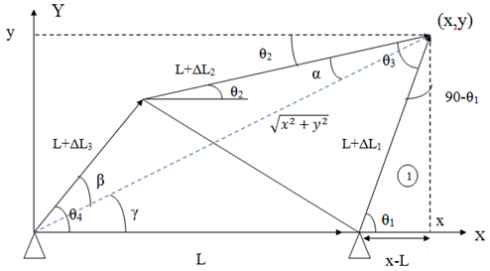


Figure 4. Three piezoelectrically actuated four bar mechanism.

A direct relationship between the ΔL's and point (x, y) can't be found in the case, so a numerical analysis will be presented in this sub-section. Six equations with six unknowns (ΔL_1, ΔL_2, ΔL_3, θ_1, θ_2, θ_4) will be derived and solved numerically using MATLAB software.

From vector analysis, the first two equations for the x and y components are;

$$\begin{bmatrix}(L+\Delta L_1)cos\theta_1 + \\ L\end{bmatrix} = \begin{bmatrix}(L+\Delta L_2)cos\theta_2 + \\ (L+\Delta L_3)cos\theta_4\end{bmatrix} \quad (24)$$

$$(L+\Delta L_1)sin\theta_1 = \begin{bmatrix}(L+\Delta L_2)sin\theta_2 - \\ (L+\Delta L_3)sin\theta_4\end{bmatrix} \quad (25)$$

From triangle (1), the third equation is obtained as:

$$y^2 + (x-L)^2 = (L+\Delta L_1)^2 \quad (26)$$

The line facing θ_3 and θ_4, could be obtained using the cosine rule as follows;

$$\begin{bmatrix}L^2 + \\ (L+\Delta L_3)^2 - \\ 2L(L+\Delta L_3)cos\theta_4\end{bmatrix} = \begin{bmatrix}(L+\Delta L_1)^2 + \\ (L+\Delta L_2)^2 - \\ 2(L+\Delta L_1)(L+\Delta L_2)\cos(\theta_3)\end{bmatrix} \quad (27)$$

But $\theta_3 = \theta_1 - \theta_2$ which can be substituted in the previous equation to find the fourth equation as follows;

$$\begin{bmatrix}L^2 + \\ (L+\Delta L_3)^2 - \\ 2L(L+\Delta L_3)cos\theta_4\end{bmatrix} = \begin{bmatrix}(L+\Delta L_1)^2 + \\ (L+\Delta L_2)^2 - \\ 2(L+\Delta L_1)(L+\Delta L_2)\cos(\theta_1-\theta_2)\end{bmatrix} \quad (28)$$

Knowing that $\theta_4 = \gamma + \beta$, $\gamma = \tan^{-1}(\frac{y}{x})$ and β can be found from the cosine rule as follows;

$$\begin{bmatrix}(L+\Delta L_3)^2 + \\ (x^2+y^2) - \\ 2(L+\Delta L_3)\sqrt{x^2+y^2}\cos(\beta)\end{bmatrix} = (L+\Delta L_2)^2 \quad (29)$$

Getting the value of β as follows;

$$\beta = cos^{-1}\left(\frac{(L+\Delta L_3)^2 + (x^2+y^2) - (L+\Delta L_2)^2}{2(L+\Delta L_3)\sqrt{x^2+y^2}}\right) \quad (30)$$

Substituting the values of γ and β to get the fifth equation;

$$\theta_4 = \tan^{-1}(\frac{y}{x}) + cos^{-1}\left(\frac{\begin{bmatrix}(L+\Delta L_3)^2 + \\ (x^2+y^2) - \\ (L+\Delta L_2)^2\end{bmatrix}}{2(L+\Delta L_3)\sqrt{x^2+y^2}}\right) \quad (31)$$

Applying the cosine rule on angle α to get the sixth equation;

$$(L+\Delta L_3)^2 = \begin{bmatrix}(L+\Delta L_2)^2 + \\ (x^2+y^2) - \\ 2(L+\Delta L_2)\sqrt{x^2+y^2}\cos(\alpha)\end{bmatrix} \quad (32)$$

But $\alpha = \gamma - \theta_2$, then;

$$(L+\Delta L_3)^2 = \begin{bmatrix}(L+\Delta L_2)^2 + \\ (x^2+y^2) - \\ 2(L+\Delta L_2)\sqrt{x^2+y^2}\cos(\gamma-\theta_2)\end{bmatrix} \quad (33)$$

$$(L+\Delta L_3)^2 = \begin{bmatrix}(L+\Delta L_2)^2 + \\ (x^2+y^2) - \\ 2(L+\Delta L_2)\sqrt{x^2+y^2}\cos\left(\tan^{-1}(\frac{y}{x}) - \theta_2\right)\end{bmatrix} \quad (34)$$

Equations (24, 25, 26, 28, 31, and 34) are the six equations with the six unknowns. These equations are solved numerically using MATLAB to get the values of the needed ΔL's when provided with the point x and y.

3 CONTROL SYSTEM DESIGN

THE next step in the proposed system is to pass the needed voltage to the PEAs in order to change their length and reach the desired location as defined by the user. In order to achieve this goal, a control system is needed to manipulate the input and get the desired effect on the output. Each PEA, plant model, is controlled separately with its own controller. As a result, three controllers are built for the three PEAs; plant models 1, 2 and 3, where they work in parallel and simultaneously as shown in Figure 5. Basically, the main idea of the whole system including the controllers is to enter the desired point or path for the joint movement, and then the MATLAB code finds the solution of the changes of lengths (ΔL's) for every PEA and passes these values as inputs for the three controllers. The controllers afterwards produce the suitable output voltage accordingly to the PEAs to make the right movement. Since the three controllers are similar, one controller will be presented and analysed then the whole system will be combined later.

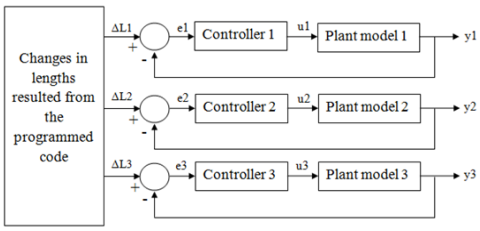


Figure 5. **PEAs overall controllers.**

3.1 Closed Loop PI Controller

PID (Proportional-Integral-derivative) controllers are one of the earliest feedback control strategies used in industries. It has a simple control structure and proved satisfactory in wide range of applications. A typical structure of a PID control system is shown in Figure 6. The error signal e(t) is used to generate the proportional (P), integral (I) and derivative (D) actions and the resulting signals weighted and summed to form the control signal u(t) applied to the plant model. In order to determine the weights, the three parameters K_p, K_i and K_d are tuned to get the proper effect. Combinations such as PI and PD controllers are very often used in practical systems.

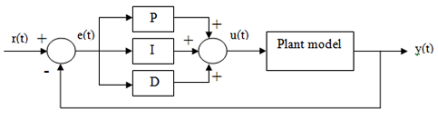


Figure 6. **PID controller structure.**

A traditional closed loop control system is built to improve the system response. Proportional- integral controller (PI controller) is specifically chosen for the system since it can minimize steady state error which is an essential property in the proposed design. The controller output equation is given as;

$$u(t) = k_p e + k_i \int e\, dt \qquad (35)$$

The designed PI controller in Simulink is illustrated in Figure 7. For anti- wind up, a limiter is used in the closed loop implementation as shown in the figure below

It was found that the PI gains are k_p= 0.001 and k_i= 8.5549. The controller output is stable with no overshoot and settling time of 0.0527 *sec* as shown in Figure 8.

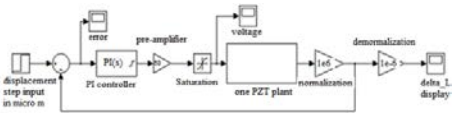


Figure 7. **Closed loop PI controller system.**

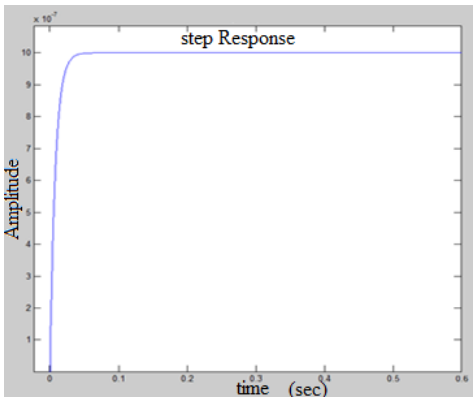


Figure 8. **PI Controller's System Performance.**

3.1.1 Closed Loop PI Controller with White Noise and Disturbance

In practical life there is no such thing as ideal system without any interferes from the outside world and this applies to every system including the proposed design in this research. White noise and disturbance are the major problems facing any existing system and must be dealt with. Otherwise, they can alter the output signal of the system. Therefore, the chosen control system must be able to eliminate noise and disturbance introduced at any time to a certain limit. As a result, the presented piezo-actuated four bar mechanism with the PI controller must be tested when introducing noise and disturbance to the system and check its response.

Using Simulink, noise and disturbance were introduced; the PI block diagram is demonstrated in Figure 9. Band-limited white noise signal is added to the system whose value is 10% of the original signal. As for the disturbance, a signal of amplitude equal to 0.5 *V*, period equal to 0.1 *sec* and 50% pulse width is used in the system. The PI gains are tuned, using MATLAB tuning toolbox, to have values of k_p= 0.001 and k_i= 8.5549.

The output of the system after introducing noise and disturbance is shown in Figure 10. It is clear that response was affected severely by the introduced interferences even after tuning the controller gains. The system response reaches peaks of 1.25 μm and 0.75 μm. This means that the designed PI controller is not able to eliminate the noise and disturbance properly.

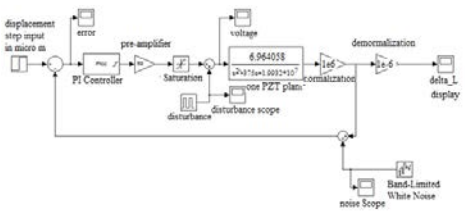


Figure 9. **PI controller system with noise and disturbance.**

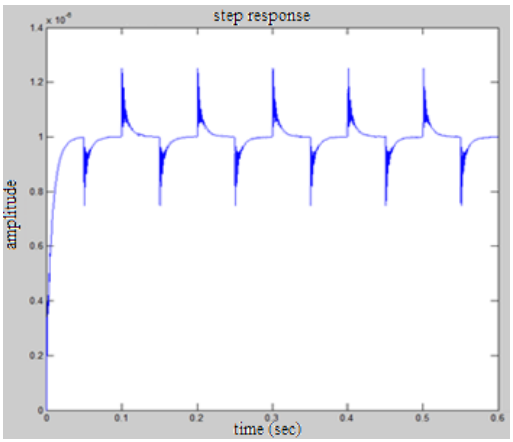


Figure 10. System response with noise and disturbance.

3.1.2 PI Fuzzy Logic Controller

Since traditional control algorithms did not provide the system with the desired results, other control techniques must be studied in order to investigate their ability to overcome the obstacles faced in the previous controller such as PI fuzzy logic controllers.

The PI fuzzy logic controller model, Figure 11, used for the system was built in MATLAB. The model had two inputs; the error (which is the difference between the desired change in length and the actual change in length) and change in error. The inference method used is Mamdani method.

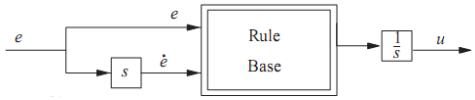


Figure 11. PI fuzzy logic controller configuration, [Aracil and Gordillo, 2004].

As for the first input, the error, it has universe of discourse between (-60) and (60) with seven triangular membership functions; negative big (NB), negative medium (NM), negative small (NS), zero (Z), positive small (PS), positive medium (PM) and positive big (PB). The triangular membership function equations are:

$$f(x;a,b,c)=\begin{cases}0, & x\le a\\ \dfrac{x-a}{b-a}, & a\le x\le b\\ \dfrac{c-x}{c-b}, & b\le x\le c\\ 0, & c\le x\end{cases} \quad (36)$$

The parameters a and c are the feet of the triangle and the parameter b is the peak. The second input, the change in error, has universe of discourse between (-120) and (120) and seven triangular membership functions; namely NB, NM, NS, Z, PS, PM and PB. The output voltage of the fuzzy controller consists of seven triangular membership functions with universe of discourse between (-100 *V*) and (-100 *V*). The fuzzy sets are divided into NB, NM, NS, Z, PS, PM and PB. The 49 fuzzy rules derived for the model are in table 2.

The defuzzification method used in the fuzzy model of the controller is the centroid method. The equation for finding the centroid is [Lee, 2005];

$$z_o=\frac{\sum_{j=1}^{n}\mu_c(z_j).z_j}{\sum_{j=1}^{n}\mu_c} \quad (37)$$

The resulted surface based on the stated rules is illustrated in Figure 12. It is noticed that the surface is smooth.

The structure of the PI fuzzy controller implemented in Simulink is shown in Figure 13.

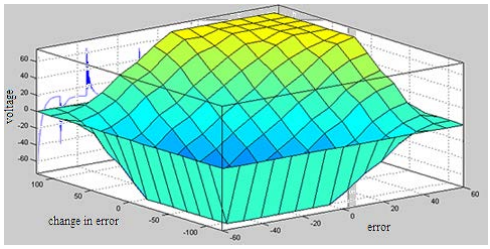


Figure 12. Fuzzy surface for the proposed system.

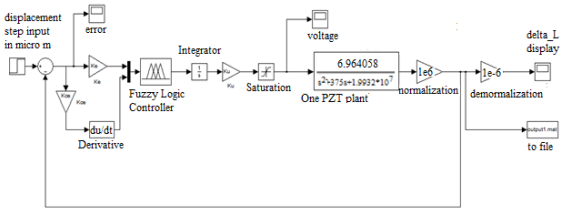


Figure 13. PI fuzzy controller.

3.2 PI-Fuzzy Tuning

The PI fuzzy controller needs to be tuned in order to get better performance while maintaining the designed system requirements and hardware limits. The most important requirements that the PEA system seeks to achieve is minimum error to get accurate results along with ensured stability. The system contains three parameter gains that affect the system which are K_e, K_{ce} and K_u. Two methods of tuning these parameters will be presented in this section; manual tuning and auto tuning.

3.2.1 Manual Tuning

Manual tuning of the gains includes trial and error where initial values of the gains K_e, K_{ce} and K_u are suggested based on personal experience and the specified range of the gain values. Then, these values are adjusted while observing the output response until reaching the best results. Since the main goal is to minimize the root mean square error in the output response, it is the main criteria for comparing between

the responses when tuning the gains. The selected range for the gains is between (0) and (100). The best obtained system response after manual tuning is when K_e=10, K_{ce}=0.2 and K_u=100. This response is in Figure 14 with zero overshoot and steady state error and settling time of 0.07788 *sec*.

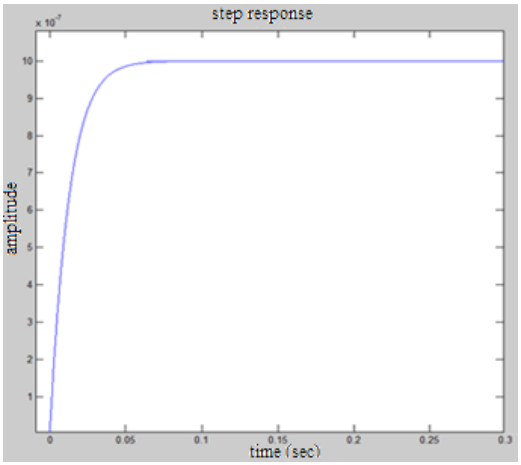


Figure 14. **PI fuzzy output response with manual tuning.**

3.2.2 Auto Tuning

Auto tuning of the controller parameters includes using certain software for loop optimization where it gathers data from the system, processes them and suggest optimal tuning of the values. Genetic algorithm will be used for automatic tuning of the PI fuzzy controller. The designed flowchart of the genetic algorithm used for tuning the PI fuzzy parameters is shown in Figure 15. Using MATLAB, the genetic algorithm code for finding the optimum values of the controller gains is built. The initial population range is chosen between (0) and (100), the number of generations used is 10 and the population size is 7. These values are picked based on trial and error. Mean squared error is the fitness function used in the algorithm where the goal is to minimize it. Any controller with unstable performance will be ignored and not selected in the next generation. The mean squared error can be calculated as follows [gepsoft, n.d.];

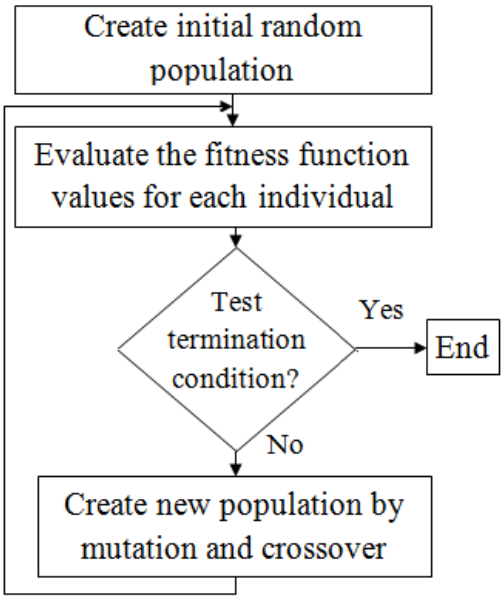


Figure 15. **Genetic algorithm flowchart.**

$$MSE = \frac{1}{n}\sum_{m=1}^{n}(X_t - X_a(m))^2 \qquad (38)$$

where MSE is the mean squared error, n is the sample size, X_t is the target value and X_a is the actual value. The resulted optimum gains from the genetic algorithm code which gives the minimum mean square error are K_e= 46.69966, K_{ce}= 1.5916 and K_u= 35.14008. The system output after simulation is shown in Figure 16. It has zero overshoot and steady state error and settling time of 0.1194 *sec*.

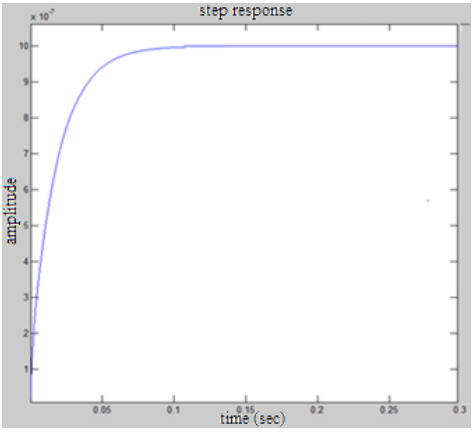


Figure 16. **PI fuzzy output response with auto tuning.**

When comparing the performance of the two proposed controllers without the influence of noise and disturbance, it can be seen that the both the classical PI controller and the fuzzy controller are satisfactory and give zero overshoot and steady state error. The only difference between them is the time to

reach the steady state value, the settling time. It takes the classical PI controller 0.0527 *sec* to reach the steady state value while for the fuzzy controller it takes 0.07788 *sec* for manual tuning and 0.1194 *sec* for auto tuning. Since the classical PI controller gives faster response, it is suitable and sufficient to give the desired response.

3.3 PI Fuzzy Logic Controller with White Noise and Disturbance

The system with the new PI fuzzy controller is now tested by introducing noise and disturbance to check the resulted responses. The controller structure built in Simulink is shown in Figure 17. The same band-limited white noise and disturbance signals used in the previous controller are used here. Both manual tuning and auto tuning techniques are used here also to find the optimum controller parameters.

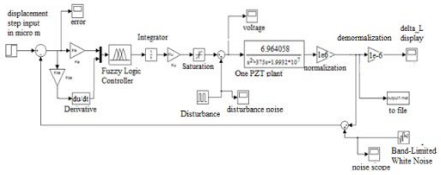


Figure 17. PI fuzzy controller with noise and disturbance.

3.3.1 Manual Tuning

The resulted parameters are K_e=10, K_{ce}=0.2 and K_u=60. The system output obtained after simulation is in Figure 18, where it has zero overshoot and settling time of 0.0861 *sec*.

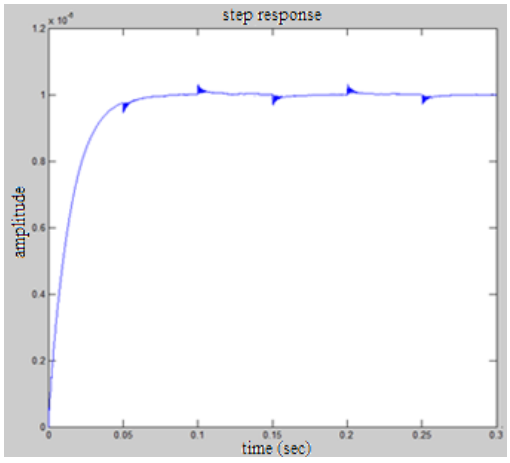


Figure 18. PI fuzzy system output with noise and disturbance and manual tuning.

3.3.2 Auto Tuning

Auto tuning is then performed using the same genetic algorithm used before for evaluating the optimum gains that have the minimum mean squared error. The initial population range is chosen between (0) and (100), the number of generations used is 10 and the population size is 6. These values are picked based on trial and error. The code gives the following optimum results K_e= 45.147, K_{ce}= 1.952 and K_u= 63.1386. The system output obtained after simulation is in Figure 19, the controller output is in Figure 20. The system response has zero overshoot and settling time of 0.1075 *sec*.

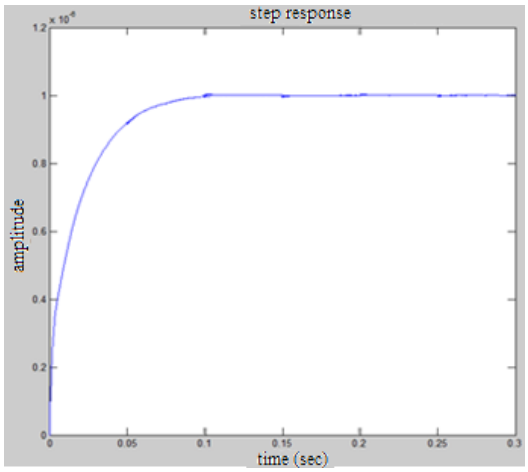


Figure 19. PI fuzzy system output with noise and disturbance and auto tuning.

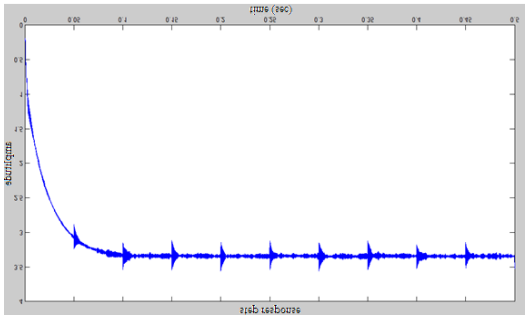


Figure 20. PI fuzzy controller output with noise and disturbance and manual tuning.

After introducing noise and disturbance to the system, the classical PI controller and the fuzzy controller differ in their responses. The classical PI controller shows a stable response, but there are oscillations and incapability of eliminating noise and disturbance. On the contrary, the fuzzy controller reduces the influence of noise and disturbance successfully for both manual tuning and auto tuning. Both responses give zero overshoot, however, the manual tuning response reaches the steady state value quicker than the auto tuning response. The settling time for the manual tuning and the auto tuning responses are 0.0861 *sec* and 0.1075 *sec* respectively.

4 SIMULATION OF THREE PIEZOELECTRICALLY ACTUATED FOUR-BAR MECHANISM

THE four bar mechanism driven by the piezoelectric actuators designed in this research is simulated to see how the system works and to investigate its behaviour. Two cases are considered for the movements, going to a location on the upper side of the original location (positive direction) and going to a location on the lower side of the original location (negative direction). For each case, an example will be demonstrated and the error will be calculated in details.

4.1 Positive Direction Simulation

The length of each link is chosen to be 100 μm to make the changes in lengths noticeable in the micro scale, so the starting point is (100 μm, 100 μm). If the final point to be reached is (113 μm, 116 μm), then the inputs for the x and y points are (13 μm) and (16 μm), respectively. The linear path of the motion from the starting point to the final location is illustrated in Figure 21.

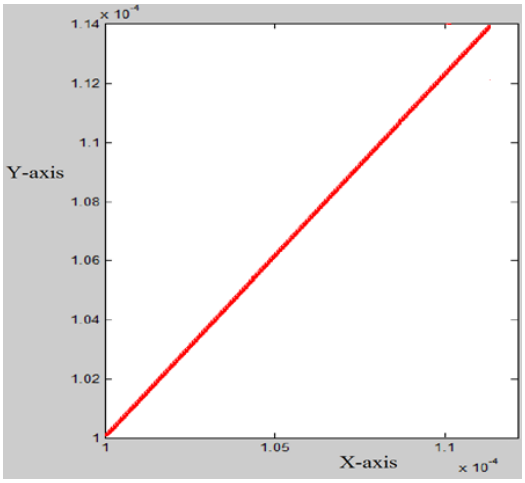


Figure 21. The linear path of the positive direction example.

Taking a closer look at the final position of the joint, the actual values are (112.989 μm, 115.96 μm). The error for the x-component is 0.00973% and for the y-component is 0.0345%. Other points were tested in the positive direction to illustrate the resulting error of the mechanism. Table 3 shows randomly selected coordinates with the resulted error. As seen from the previous table, most error values for the x and y components are less than 1% except for cases where the change in length exceeds (30 μm) the error is greater than 1%.

Table 2. The fuzzy rules.

E / CE	NB	NM	NS	Z	PS	PM	PB
NB	NB	NB	NB	NB	NM	NS	Z
NM	NB	NB	NB	NM	NS	Z	PS
NS	NB	NB	NM	NS	Z	PS	PM
Z	NB	NM	NS	Z	PS	PM	PB
PS	NM	NS	Z	PS	PM	PB	PB
PM	NS	Z	PS	PM	PB	PB	PB
PB	Z	PS	PM	PB	PB	PB	PB

4.2 Negative Direction Simulation

Assuming the final point to be reached is (92 μm, 95 μm), then the inputs for the x and y points are (-8 μm) and (-5 μm), respectively. The linear path of the motion from the starting point to the final location is illustrated in Figure 22. The errors in the x and y-components have values of 0.010868% and 0.021048%, respectively. Other points were tested in the negative direction also to illustrate the resulting error of the mechanism. Table 4 shows randomly selected coordinates with the resulted error. From the previous table, most error values for the x and y components are less than 1% except for cases where the change in length exceeds (-30 μm) the error is greater than 1%.

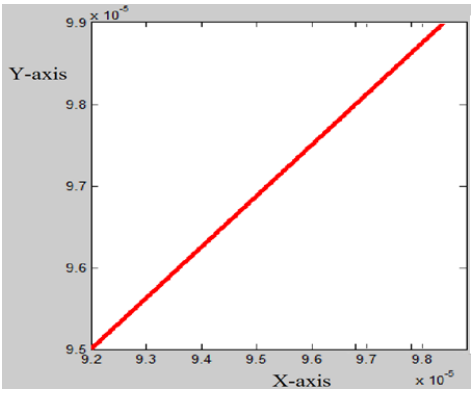


Figure 22. The linear path of the negative direction example.

5 INCHWORM ROBOT APPLICATION

INCHWORMS have been a subject of interest to researchers since they exhibit extremely flexible and robust mobility. The inchworm is capable of manoeuvring in small spaces, moving on different surfaces, smooth or rough, and crossing obstacles. The inchworm strategy of movement is simple; it depends on its both body ends and elongating body. First, an inchworm attaches its rear end to a surface, elongates its body forward, attaches its front end to the surface and then shortens its body to meet the front end forward. This motion is demonstrated in figure 23. The simplicity and reliability of the inchworm motion makes it suitable for applications involving unmapped, unpredictable, very narrow and even hostile environments.

Table 3. Positive direction examples and computed error.

	Desired X (μm)	Desired Y (μm)	Actual X (μm)	Actual Y (μm)	Error in X %	Error in Y %
1	100.4	100.15	100.4	100.1	0.0179	0.0654
2	100.8	101.3	100.8	101.3	0.0099	0.0158
3	102	102	102	102	0.0196	0.0588
4	104.5	103.6	104.5	103.6	0.0096	0.0251
5	106	102.1	106	102.1	0.0198	0.0171
6	105	109	105	108.9	0.0181	0.0836
7	108.7	110	108.7	109.9	0.0190	0.0664
8	112.4	112	112.4	111.9	0.0107	0.0482
9	116	111	116	111	0.0198	0.0371
10	120.2	110.6	119.7	110.3	0.406	0.2803
11	118	118	118	118	0.0093	0.0186
12	121.9	125.1	118.6	121.4	2.674	2.989
13	117	104	117	103.9	0.0300	0.0769
14	108	100	108	100.1	0.0241	0.0797
15	101	107	101	106.8	0.0178	0.1963
16	128	130	119.1	120.4	6.984	7.37

Table 4. Negative direction examples and computed error.

	Desired X (μm)	Desired Y (μm)	Actual X (μm)	Actual Y (μm)	Error in X %	Error in Y %
1	99.7	99.4	99.7	99.6	0.015	0.153
2	99	98.3	99	98.4	0.0157	0.1322
3	97	97	97	97.1	0.0066	0.074
4	95.8	96.2	95.8	96.3	0.0167	0.0727
5	94.5	97.9	94.5	98	0.0106	0.0613
6	95	92	95	92.1	0.0147	0.149
7	92.2	89	92.2	89.1	0.0033	0.12
8	86.9	88	86.9	88	0.0081	0.0295
9	85	88.5	85	88.6	0.0059	0.101
10	79.6	83.3	82.7	86	3.904	3.193
11	82	82	82.3	82.3	0.4146	0.4146
12	77.8	74	83.1	80.2	6.774	8.34
13	83	86	83	86.1	0.0084	0.087
14	99.5	98	99.5	98.3	0.0101	0.2857
15	98	93	98	93.1	0.0173	0.15
16	72	70	82.5	81.3	14.63	16.11

An inchworm robot imitates the locomotion of a natural inchworm. Generally, it consists of an elongation component and a grapping component to realize the inchworm motion. A commonly used elongation actuator in inchworm robots is the piezoelectric actuator due to their quick responses in expansion and contraction by supplying the required energy. The PEAs used in the inchworm robot are controlled in a way to generate a series of micro steps each corresponding to one cycle. These cycles are repeated to reach the desired location.

The approach pursued here adopts a design of the inchworm robot as illustrated in Figure 24 in two-dimensional view (2D) and in Figure 25 in three-dimensional view (3D). It consists of the robot body that holds two four bar linkages each with three PEAs as designed earlier and four moving legs, two on the front and two on the back. Every front and rear legs are attached to and controlled by one four bar linkage. The left and the right side of the robot containing the four bar mechanism are symmetrical regarding the design, equations of motion and control system. The four bar mechanism works in two modes, moving all three links with the PEAs and moving only two links with two PEAs while fixing the third link. The front and rear legs are attached to the upper two joints of the four bar linkage so they provide contact with the ground for attachment. The movement technique is organized by a controller to produce a series of steps.

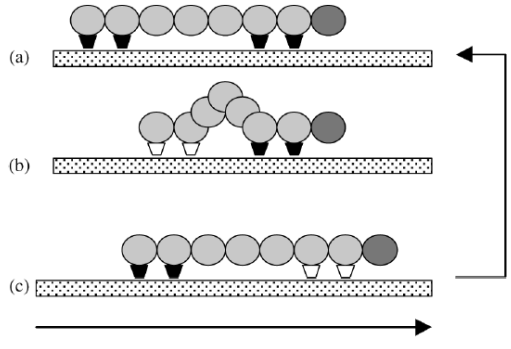


Figure 23. Inchworm motion, [Ferworn and Stacey , n.d.].

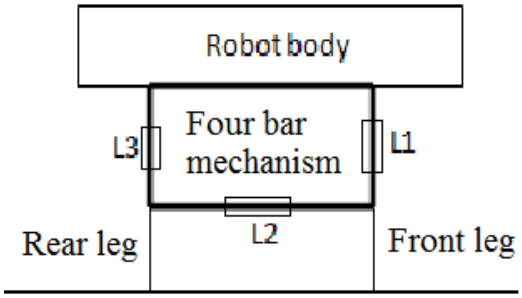


Figure 24. 2D schematic of the proposed inchworm robot.

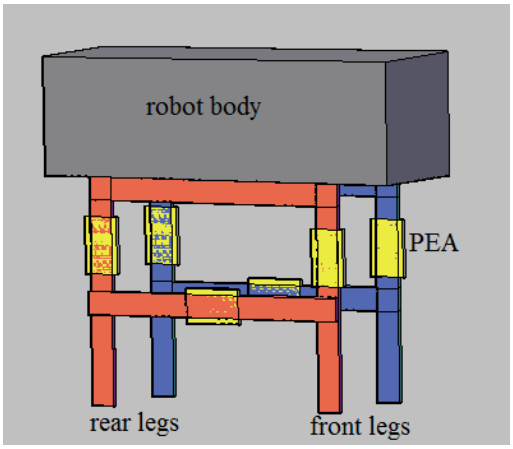


Figure 25. 3D schematic of the proposed inchworm robot.

In order to move the robot, repetitive motion cycles must take place to reach a certain location. Each cycle is composed of two phases, the two-link mode and the three-link mode. These two phases are demonstrated as follows:

5.1 Two-link mode:

In this phase, two links can move and extend (or shorten) while the third link maintains its length and is fixed in position. Only two PEAs are provided with the suitable voltages and the third PEA is neglected. The phase objective is to move the front leg forward while attaching the rear leg to the ground starting from the initial state where all links have the same lengths.

The mechanism suggested for this phase movement is shown in details in figure 26. The two moving links are L_1 and L_2. First, link L_1 is shortened to a calculated location to lift the front leg off the ground (see figure 26-a). Then, link L_2 is extended to another calculated location such that the extension of the front leg to the ground is the final location to be reached (see figure 26-b).

Finally, link L_1 is extended to a point such that the front leg reaches the ground in the final destination (see figure 26-c). To reach every point in the previous steps, both links L_1 and L_2 change their lengths but only the major link is mentioned for demonstration.

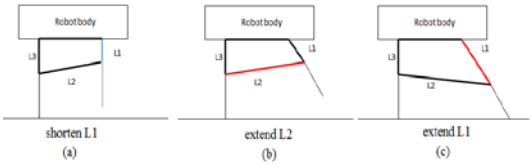


Figure 26. Two-link movement mode.

5.2 Three-link mode:

In this phase, the three links with the three PEAs move and change their lengths according to the incoming voltages in order to move forward and reach the final destination. The phase objective is to move the rear leg forward while attaching the front leg to the ground and returning to the initial state where all links have the same lengths.

The mechanism suggested for this phase movement is shown in details in Figure 27. The moving links are L_1 and L_2 and L_3. First, Link L_3 is shortened to a calculated point in order to lift the rear leg off the ground (see figure 27-a). Then, link L_2 is shortened such that the front leg becomes perpendicular to the ground (see figure 27-b). Finally, link L_3 is extended until the rear leg reaches the ground and at the same time link L_3 returns to its initial length (see figure 27-c). To reach every point in the previous steps, all links L_1, L_2 and L_3 change their lengths and move but only the major link is mentioned for demonstration.

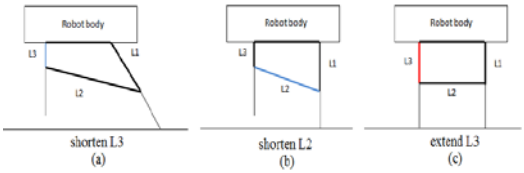


Figure 27. Three-link movement mode.

The previously described motion is for each side of the robot, it is controlled in a way to move the robot legs in harmony. Three scenarios are considered for movement based on the front legs:

1) Moving forward and backward: if both the front legs walk the same distance, the robot will either move forward or backward according to the specified direction.
2) Turning to the right: if the left front leg outstrips the right front leg, the robot will turn to the right side.
3) Turning to the left: if the right front leg outstrips the left front leg, the robot will turn to the left side.

A special designed control system is needed to organize the desired movement of the robot as defined by the user according to the previously proposed scenarios.

6 CONCLUSION

A modified design of three piezoelectrically actuated four bar mechanism had been developed in this paper. The design is capable of moving joints by actuating three PEAs simultaneously mounted on the three bars to reach a desired location. The range of movement was limited between (-30 μm) to (30 μm) by applying voltage to the PEAs, also limited between (-100 V) and (100 V). The mathematical model for the design describing the equations of motion for one joint was derived and solved numerically using a programmed code in MATLAB. The solution included changes in lengths of the three bars and angles of the links for a given location calculated iteratively.

Three parallel control systems were built to manage the movement of the three links separately using two types of controllers, classical PI controller and fuzzy PI controller. The PI controller was chosen to reduce the steady state error and get an accurate result which was the main requirement for the design. In the absence of noise and disturbance, the classical PI controller gave faster performance than the fuzzy PI controller but both gave zero overshoot and steady state error. In the presence of noise and disturbance, the fuzzy PI controller was more robust and eliminated the introduced noise and disturbance unlike the classical PI controller which failed to handle those signals. The fuzzy controller parameters were tuned using two methods, manual tuning and auto tuning. The auto tuning method was performed using genetic algorithm which gave the optimum solution for the controller gains. By comparing the resulted responses of the two methods in case of noise and disturbance, auto tuning method gave less error and more accurate response than the manual tuning method. So it can be concluded that the fuzzy PI controller with genetic algorithm auto tuning is more efficient in the overall performance.

The design was simulated using MATLAB GUI to demonstrate its reliability and the yielded error. Simulation of the movement was illustrated in the positive and negative directions with error less than 1% except for cases where the change in length exceeded the designed limits. These tests proved that the system is capable of moving within the limited range with satisfactory results. The error values can be reduced by decreasing the step size used thus increasing the number of points in the proposed linear path and increasing the solution time.

Finally, a design of an inchworm robot employing two four bar linkages each with three PEAs was presented as an application. The movement technique and the initial vision of the robot were described in figures for later development.

7 REFERENCES

H. Adriaens, W. de Koning, and R. Banning. (2000). Modeling Piezoelectric Actuators. IEEE/Asme Transcations on Mechatronics, 331-341.

O. Aguilar, R. Tapia, R., A. Valderrabano, and I. Rivas, (2015). Adaptive Neural Network Control of Chaos in Permanent Magnet Synchronous Motor. Intelligent Automation and Soft Computing, 22, 1-9.

J. Aracil and F. Gordillo, (2004). Describing function method for stability analysis of PD and PI fuzzy controllers. Fuzzy sets and systems, 233-249.

S. Bashash and N. Jalili, (2007). Robust Multiple Frequency Trajectory Tracking Control of Piezoelectrically Driven Micro/Nanopositioning Systems. IEEE Transactions on Control Systems Technology, 867-878.

X. Chang and J. Lilly, (2003). Fuzzy Control for Pneumatic Muscle Tracking via Evolutionary Tuning. Intelligent Automation and Soft Computing, 9(4), 227-244.

Z. Civelek, M. Lüy, E. Çam, and N. Barışçı, (2016). Control of Pitch Angle of Wind Turbine by Fuzzy Pid Controller. Intelligent Automation and Soft Computing, 22(3), 463-471.

A. Eigoli and G. Vossoughi. (2012). Locomotion modes of a novel piezo-driven microrobot: Analytical modeling and performance evaluation. Mechanism and Machine Theory, 248-266.

A. Ferworn and D. Stacey (n.d.). Inchworm Mobility--Stable, Reliable and Inexpensive. Retrieved from http://www.scs.ryerson.ca/aferworn/research/inchworm.html

gepsoft. (n.d.). Retrieved from http://www.gepsoft.com/gxpt4kb/Chapter09/Section1/SS02/SSS3.htm

A. Ghanbari, A. Rostami, S. Noorani, and M. Fakhrabadi. (2008). Modeling and Simulation of Inchworm Mode Locomotion. Springer ICIRA, 617-624.

J. Godjevac and N. Steele (1998). Fuzzy Systems and Neural Networks. Intelligent Automation and Soft Computing, 4(1), 27-37.

O. Gundogdu and K. Erenturk. (2005). Fuzzy control of a dc motor driven four-bar mechanism. Mechatronics, 423-438.

S. Heo, T. Wiguna, H. Park, and N. Goo, (2007). Effect of an Artificial Caudal Fin on the Performance of a Biomimetic Fish Robot Propelled by Piezoelectric Actuators. Journal of Bionic Engineering, 151-158.

K. Kotay and D. Rus. (2000). The Inchworm Robot: A Multi-Functional System. Autonomous Robots, 53-69.

P. Kumarakulasingam and A. Agah. (2013). Neural Network-Based Single Sensor Sound Localization Using A Mobile Robot. Intelligent Automation and Soft Computing, 14(1), 89-103.

K. Lee. (2005). First course on fuzzy theory and applications. Korea Advanced Institute of Science and Technology, Springer.

Y. Li and Q. Xu. (2006). A Novel Design and Analysis of a 2-DOF Compliant Parallel Micromanipulator for Nanomanipulation. IEEE Transactions on Automation Science and Engineering, 248-254.

Y. Lianzhi and L. Weichong. (2012). Cascade Hopfield Neural Network Model and Application in Robot Moving Process. Procedia Engineering, 814-818.

H. Liaw and B. Shirinzadeh. (2008). Enhanced adaptive motion tracking control of piezo-actuated flexure-based four-bar mechanisms for micro/nano manipulation. Sensors and Actuators A, 254-262.

H. Liaw and B. Shirinzadeh. (2008). Robust generalised impedance control of piezo-actuated flexure-based four-bar mechanisms for micro/nano manipulation. Sensors and Actuators A, 443-453.

H. Liaw and B. Shirinzadeh. (2009). Neural Network Motion Tracking Control of Piezo-Actuated Flexure-Based Mechanisms for Micro /Nanomanipulation. IEEE/Asme Transcations on Mechatronics, 517-527.

H. Liaw, B. Shirinzadeh, and J. Smith. (2007). Enhanced sliding mode motion tracking control of piezoelectric actuators. Sensors and Actuators A, 194-202.

H. Liaw, R. Shirinzadeh, and J. Smith. (2008). Robust motion tracking control of piezo-driven flexure-based four-bar mechanism for micro/nano manipulation. Mechatronics, 111-120.

Lim, J., Park, H., An, J., Hong, Y., Kim, B., and Yi, B. (2008). One pneumatic line based inchworm-like micro robot for half-inch pipe inspection. Mechatronics, 315-322.

Lin, C., and Yang, S. (2006). Precise positioning of piezo-actuated stages using hysteresis-observer based control. Mechatronics, 417-426.

Y. Liu and B. Li. (2010). Precision positioning device using the combined piezo-VCM actuator with frictional constraint. Precision Engineering, 534-545.

N. Lobontiu, M. Goldfarb, and E. Garcia. (2001). A piezoelectric-driven inchworm locomotion device. Mechanism and Machine Theory, 425-443.

H. Lu, J. Zhu, Z. Lin, and Y. Guo. (2009). An inchworm mobile robot using electromagnetic linear actuator. Mechatronics, 1116-1125.

V. Mermertas. (2004). Optimal design of manipulator with four-bar mechanism. Mechanism and Machine Theory, 545-554.

R. Mrad, A. Abhari, and J. Zu. (2003). A Control Methodology for an Inchworm Piezomotor. Mechanical Systems and Signal Processing, 457-471.

C. Oh, J. Choi, H. Nam, J. Bu, and S. Kim. (2010). Ultra-compact, zero-power magnetic latching piezoelectric inchworm motor with integrated position sensor. Sensors and Actuators A, 306-312.

F. Pin and Y. Watanabe. (1995). Automatic Generation of Fuzzy Rules Using the Fuzzy Behaviorist Approach: The Case of Sensor-Based Robot Navigation. Intelligent Automation and Soft Computing, 1(2), 161-178.

M. Pozzi and T. King. (2003). Piezoelectric modelling for an impact actuator. Mechatronics, 553-570.

J. Qiao, J. Shang, and A. Goldenberg. (2012). Development of Inchworm In-Pipe Robot Based on Self-Locking Mechanism. IEEE/Asme Transcations on Mechatronics, 1-8.

D. Rincón and J. Castro. (2003). Dynamic and Experimental Analysis for Inchwormlike Biomimetic Robots. IEEE Robotics and Automation Magazine, 53-57.

J. Shim, H. Cho, and S. Kim. (1997). An Actively Compliable Probing System. IEEE Control Systems, 14-21.

M. Sitti. (2003). Piezoelectrically Actuated Four-Bar Mechanism with Two Flexible Links for Micromechanical Flying Insect Thorax. IEEE/Asme Transcations on Mechatronics, 26-36.

C. Von Albrichsfeld and H. Tolle. (2002). A self-adjusting active compliance controller for multiple robots handling an object. Control Engineering Practice, 165-173.

S. Wongsoontorn and H. Zhuang. (2013). Design and Tuning of Fuzzy Control Surfaces with Bezier Functions. Intelligent Automation and Soft Computing, 14(1).

Y. Wu and Z. Zhou. (2004). An XY θ mechanism actuated by one actuator. Mechanism and Machine Theory, 1101-1110.

Z. Xianmin, S. Changjian, and A. Erdman. (2002). Active vibration controller design and comparison study of flexible linkage mechanism systems. Mechanism and Machine Theory, 985-997.

H. Yan and G. Yan. (2009). Integrated control and mechanism design for the variable input-speed servo four-bar linkages. Mechatronics, 274-285.

Q. Yao, J. Dong, and P. Ferreira. (2007). Design, analysis, fabrication and testing of a parallel-kinematic micropositioning XY stage. International Journal of Machine Tools and Manufacture, 946-961.

X. Zhang. (2004). Integrated optimal design of flexible mechanism and vibration control. International Journal of Mechanical Sciences, 1607-1620.

8 DISCLOSURE STATEMENT

No potential conflict of interest was reported by the authors.

9 NOTES ON CONTRIBUTORS

Qais Khasawneh received his BSc (1999) From JUST and received his MSc (2002) and PhD (2008) from the University of Akron, USA. He has been serving as an assistant professor in the mechanical engineering department/mechatronics since 2009. His research activities involve MEMS, Nanotechnology, renewable energy, and control.

Mohammad A. Jaradat received his B.Sc. from JUST, Jordan, and M.Sc. and Ph.D. in mechanical engineering from Texas A and M University, TX, USA. Currently he is an Associate Professor at the American University of Sharjah, and with JUST. His research areas are: robotics, artificial intelligent systems, intelligent control, and embedded control systems.

Mohammad Al-Shabi is currently serving as an assistant professor in the Mechanical Engineering Department at University of Sharjah/UAE. He obtained his BSc and MSc in Mechanical Engineering from Jordan University for Science and Technology/Jordan. He obtained his PhD in Mechanical Engineering/Mechatronics from McMaster University/Canada in 2011.