

Intelligent Adaptive Gain Backstepping Technique

Sara Heidari, Ali Shahcheraghi, Kamran Heidari, Samaneh Zahmatkesh, Farzin Piltan

Institute of Advance Science and Technology, Intelligent control and Robotics Lab. IRAN SSP, Shiraz, Iran

<http://WWW.IRANSSP.COM/english>; Email: Piltan_f@iranssp.com

Abstract- In this research, intelligent adaptive backstepping control is presented as robust control for continuum robot. The first objective in this research is design a Proportional-Derivative (PD) fuzzy system to compensate the system model uncertainties. The second objective is focused on the design tuning gain adaptive methodology according to high quality partly nonlinear methodology. Conventional backstepping controller is one of the important robust controllers especially to control of continuum robot manipulator. The fuzzy controller is used in this method to system compensation. In real time to increase the system robust fuzzy logic theory is applied to backstepping controller. To approximate a time-varying nonlinear dynamic system, a fuzzy system requires a large amount of fuzzy rule base. The adaptive laws in this algorithm are designed based on the Lyapunov stability theorem. This method is applied to continuum robot manipulator to have the best performance.

Index Terms- Continuum Robot Manipulator, Robust Backstepping Controller, Fuzzy Logic System, Adaptive Methodology

I. INTRODUCTION AND BACKGROUND

In modern usage, the word of control has many meanings, this word is usually taken to mean regulate, direct or command. The word feedback plays a vital role in the advance engineering and science. There are several methods for controlling a robot manipulator, which all of them follow two common goals, namely, hardware/software implementation and acceptable performance. However, the mechanical design of robot manipulator is very important to select the best controller but in general two types schemes can be presented, namely, a joint space control schemes and an operation space control schemes[1]. Joint space and operational space control are closed loop controllers which they have been used to provide robustness and rejection of disturbance effect. The main target in joint space controller is to design a feedback controller which the actual motion ($q_a(t)$) and desired motion ($q_d(t)$) as closely as possible. This control problem is classified into two main groups. Firstly, transformation the desired motion $X_d(t)$ to joint variable $q_d(t)$ by inverse kinematics of robot manipulators[2]. This control include simple PD control, PID control, inverse dynamic control, Lyapunov-based control, and passivity based control that explained them in the following section. The main target in operational space controller is to design a feedback controller to allow the actual end-effector motion $X_a(t)$

to track the desired endeffector motion $X_d(t)$. This control methodology requires a greater algorithmic complexity and the inverse kinematics used in the feedback control loop. Direct measurement of operational space variables are very expensive that caused to limitation used of this controller in industrial robot manipulators[3]. One of the simplest ways to analysis control of multiple DOF robot manipulators are analyzed each joint separately such as SISO systems and design an independent joint controller for each joint. In this controller, inputs only depends on the velocity and displacement of the corresponding joint and the other parameters between joints such as coupling presented by disturbance input. Joint space controller has many advantages such as one type controllers design for all joints with the same formulation, low cost hardware, and simple structure.

A nonlinear methodology is used for nonlinear uncertain systems (e.g., robot manipulators) to have an acceptable performance. These controllers divided into six groups, namely, feedback linearization (computed-torque control), passivity-based control, sliding mode control (variable structure control), artificial intelligence control, backstepping control and adaptive control[4-5]. In control theory, backstepping is a technique developed circa 1990 by Petar V. Kokotovic and others for designing stabilizing controls for a special class of nonlinear dynamical systems. These systems are built from subsystems that radiate out from an irreducible subsystem that can be stabilized using some other method. Because of this recursive structure, the designer can start the design process at the known-stable system and "back out" new controllers that progressively stabilize each outer subsystem. The process terminates when the final external control is reached. Hence, this process is known as backstepping. Backstepping control is work based on cancelling decoupling and nonlinear terms of dynamics formulation of each link. This method computes the required arm torques using the nonlinear feedback control law [6-7].

In recent years, artificial intelligence theory has been used in sliding mode control systems. Neural network, fuzzy logic, and neuro-fuzzy are synergically combined with nonlinear classical controller and used in nonlinear, time variant, and uncertainty plant (e.g., robot manipulator). Fuzzy logic controller (FLC) is one of the most important applications of fuzzy logic theory. This controller can be used to control nonlinear, uncertain, and noisy systems. This method is free of some model-based

techniques as in classical controllers. As mentioned that fuzzy logic application is not only limited to the modelling of nonlinear systems [13-14] but also this method can help engineers to design easier controller. Control robot arm manipulators using classical controllers are based on manipulator dynamic model. These controllers often have many problems for modelling. Conventional controllers require accurate information of dynamic model of robot manipulator, but these models are multi-input, multi-output and non-linear and calculate accurate model can be very difficult. When the system model is unknown or when it is known but complicated, it is difficult or impossible to use classical mathematics to process this model [8-9]. The main reasons to use fuzzy logic technology are able to give approximate recommended solution for unclear and complicated systems to easy understanding and flexible. Fuzzy logic provides a method which is able to model a controller for nonlinear plant with a set of IF-THEN rules, or it can identify the control actions and describe them by using fuzzy rules. It should be mentioned that application of fuzzy logic is not limited to a system that's difficult for modeling, but it can be used in clear systems that have complicated mathematics models because most of the time it can be shortened in design but there is no high quality design just sometimes we can find design with high quality. Besides using fuzzy logic in the main controller of a control loop, it can be used to design adaptive control, tuning parameters, working in a parallel with the classical and non classical control method [9]. The applications of artificial intelligence such as neural networks and fuzzy logic in modelling and control are significantly growing especially in recent years. For instance, the applications of artificial intelligence, neural networks and fuzzy logic, on robot arm control have reported in [9].

Research about mechanical parts and control methodologies in robotic system is shown; the mechanical design, type of actuators, and type of systems drive play important roles to have the best performance controller. More over types of kinematics chain, i.e., serial Vs. parallel manipulators, and types of connection between link and joint actuators, i.e., highly geared systems Vs. direct-drive systems are played important roles to select and design the best acceptable performance controllers [10-11]. A serial link continuum robot is a sequence of joints and links which begins with a base frame and ends with an end-effector. This type of robot manipulators, comparing with the load capacitance is more weightily because each link must be supported the weights of all next links and actuators between the present link and end-effector [12-13]. Serial continuum robot manipulators have been used in medical application, and also in research laboratories. One of the most important classifications in controlling the robot manipulator is how the links have connected to the actuators. This classification divides into two main groups: highly geared (e.g., 200 to 1) and direct drive (e.g., 1 to 1). High gear ratios reduce the nonlinear coupling dynamic parameters in robot manipulator. In

this case, each joint is modeled the same as SISO systems. In high gear robot manipulators which generally are used in industry, the couplings are modeled as a disturbance for SISO systems. Direct drive increases the coupling of nonlinear dynamic parameters of robot manipulators. This effect should be considered in the design of control systems. As a result some control and robotic researchers' works on nonlinear robust controller design. Although most of continuum robot manipulator is high gear and this research focuses on design SISO controller.

In various dynamic parameters systems that need to be training on-line adaptive control methodology is used. Adaptive control methodology can be classified into two main groups, namely, traditional adaptive method and fuzzy adaptive method. Fuzzy adaptive method is used in systems which want to training parameters by expert knowledge. Traditional adaptive method is used in systems which some dynamic parameters are known. In this research in order to solve disturbance rejection and uncertainty dynamic parameter, adaptive method is applied to artificial backstepping controller based on PI like fuzzy controller. For nonlinear dynamic systems (e.g., robot manipulators) with various parameters, adaptive control technique can train the dynamic parameter to have an acceptable controller performance. Calculate several scale factors are common challenge in classical backstepping controller and fuzzy logic controller, as a result it is used to adjust and tune coefficient. Research on adaptive nonlinear control is significantly growing, for instance, different adaptive fuzzy controllers have been reported in [14].

This paper is organized as follows; second part focuses on the modeling dynamic formulation based on Lagrange methodology, design backstepping control and fuzzy logic methodology. Third part is focused on the methodology which can be used to reduce the error, increase the performance quality and increase the robustness and stability. Simulation result and discussion is illustrated in forth part which based on trajectory following and disturbance rejection. The last part focuses on the conclusion and compare between this method and the other ones.

II. THEORY

Dynamic Formulation of Continuum Robot:

The Continuum section analytical model developed here consists of three modules stacked together in series. In general, the model will be a more precise replication of the behavior of a continuum arm with a greater of modules included in series. However, we will show that three modules effectively represent the dynamic behavior of the hardware, so more complex models are not motivated. Thus, the constant curvature bend exhibited by the section is incorporated inherently within the model. The model resulting from the application of Lagrange's equations of motion obtained for this system can be represented in the form

$$\mathbf{F}_{\text{coeff}} \boldsymbol{\tau} = \mathbf{D}(\mathbf{q}) \ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}) \dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) \quad (1)$$

$$G(\mathbf{q}) = \begin{bmatrix} -m_1\mathbf{g} - m_2\mathbf{g} + k_{11}(s_1 + (1/2)\theta_1 - s_{01}) + k_{21}(s_1 - (1/2)\theta_1 - s_{01}) - m_3\mathbf{g} \\ -m_2g\cos(\theta_1) + k_{12}(s_2 + (1/2)\theta_2 - s_{02}) + k_{22}(s_2 - (1/2)\theta_2 - s_{02}) - m_3g\cos(\theta_1) \\ -m_3g\cos(\theta_1 + \theta_2) + k_{13}(s_3 + (1/2)\theta_3 - s_{03}) + k_{23}(s_3 - (1/2)\theta_3 - s_{03}) \\ m_2s_2g\sin(\theta_1) + m_3s_3g\sin(\theta_1 + \theta_2) + m_3s_2g\sin(\theta_1) + k_{11}(s_1 + (1/2)\theta_1 - s_{01})(1/2) \\ + k_{21}(s_1 - (1/2)\theta_1 - s_{01})(-1/2) \\ m_3s_3g\sin(\theta_1 + \theta_2) + k_{12}(s_2 + (1/2)\theta_2 - s_{02})(1/2) + k_{22}(s_2 - (1/2)\theta_2 - s_{02})(-1/2) \\ k_{13}(s_3 + (1/2)\theta_3 - s_{03})(1/2) + k_{23}(s_3 - (1/2)\theta_3 - s_{03})(-1/2) \end{bmatrix} \quad (5)$$

III. METHODOLOGY

The continuum robot dynamics in (1) have the appropriate structure for the so-called backstepping controller design method. With the position error defined as $Z_1 = X_d - X_a$, all joints will track the desired specified state X_d if the error dynamics are given as follows:

$$(\dot{Z}_1 + [K_p]Z_1) = 0 \quad (6)$$

where $[K_p]$ is a positive definite gain matrix. The error dynamics in (6) can be rewritten as:

$$X_2 = \dot{X}_d + [K_p]Z_1 \quad (7)$$

Substitution of (7) into (1) makes the position error dynamics go to zero. Since the state vector x_2 is not a control variable, (7) cannot be directly substituted into (1). The expression in (7) is therefore defined as a fictitious control input and is labeled expressed below as X_{2d} .

$$X_{2d} = \dot{X}_{1d} + [K_p](X_d - X_a) \quad (8)$$

The fictitious control input in (8) is selected as the specified velocity trajectory and hence the velocity error can be defined as $Z_2 = X_{2d} - X_{2a}$. With the following dynamics

$$(\dot{Z}_2 + [K_p]Z_2) = 0 \quad (9)$$

the joint position error will approach zero asymptotically, which will lead to the eventual asymptotic convergence of the joint position error. The error dynamics in (9) can be rewritten as:

$$X_2 = \dot{X}_d + [K_p]Z_2 \quad (10)$$

Substitution of (9) into (1) leads to the following expression as the desired stabilizing torque:

$$\tau = [H](\dot{X}_{2d} + [K_p]Z_2) + C(X_1, X_2) \quad (11)$$

The desired torque control input is a nonlinear compensator since it depends on the dynamics of the spherical motor. The time derivative of desired velocity vector is calculated using (9). In terms of the desired state trajectory, and its time derivatives and the position and velocity state variables, the desired torque can be rewritten in following form:

$$\tau = [H]y + C(X_1, X_2) \quad (12)$$

Where

$$y = \ddot{X}_{1d} + ([K_p] + [K_d])(\dot{X}_{1d} - \dot{X}_1) + ([K_p][K_d]X_d - X_a) \quad (13)$$

The backstepping controller developed above is very similar to inverse dynamics control algorithm developed for robotic manipulators. The backstepping controller is ideal from a control point of view as the nonlinear dynamics of the continuum robot are cancelled and replaced by linear subsystems. The drawback of the backstepping controller is that it requires perfect cancellation of the nonlinear continuum robot dynamics. Accurate real time representations of the robot dynamics resulting from imperfect knowledge of the robot mechanical parameters; existence of unmodeled dynamics and dynamic uncertainties due to payloads. The requirement for perfect dynamic cancellation raises sensitivity and robustness issues that are addressed in the design of a robust backstepping controller. Another drawback of the backstepping controller is felt during real-time implementation of the control algorithm. Implementation of the backstepping controller requires the computation of the exact robot dynamics at each sampling time.

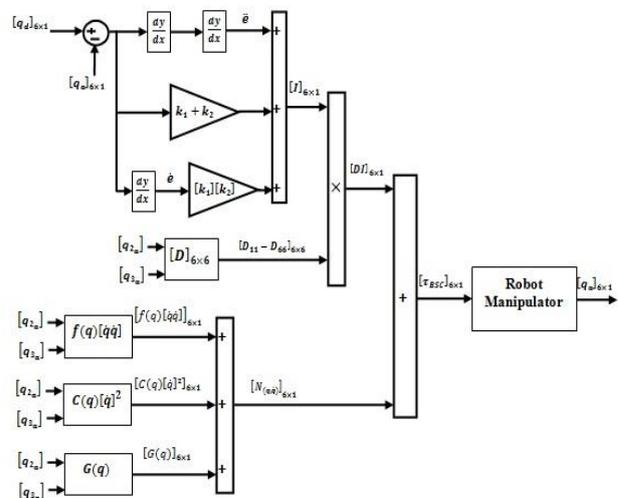


Fig. 1. Block diagram of robust backstepping controller

This computational burden has an effect on the performance of the control algorithm and imposes constraints on the hardware/software architecture of the control system. By only computing the dominant parts of the robot dynamics, this computational burden can be

reduced. These drawbacks of the backstepping controller makes it necessary to consider control algorithms that compensate for both model uncertainties and for approximations made during the on-line computation of robot dynamics. The next section provides robust modifications of the backstepping controller described in this section. Fig. 1 shows the block diagram of backstepping controller.

Robust Backstepping Control: When there are uncertainties in the spherical motor dynamics due to modeling inaccuracy and computational relaxation, robust controllers are ideal for ensuring system stability. When the system dynamics are completely known, the required torque control vector for the control of the spherical motor is given by (12) and (13). In the presence of modeling uncertainties, a reasonable approximation of the torque control input vector is given by

$$\mathbf{y} = \ddot{\mathbf{X}}_{1d} + ([\mathbf{K}_p] + [\mathbf{K}_d])(\dot{\mathbf{X}}_{1d} - \dot{\mathbf{X}}_1) + ([\mathbf{K}_p][\mathbf{K}_d]\mathbf{X}_d - \mathbf{X}_a) \quad (14)$$

$$\mathbf{T}_d = [\hat{\mathbf{H}}]\mathbf{y} + \hat{\mathbf{C}} \quad (15)$$

where $[\hat{\mathbf{H}}]$ and $\hat{\mathbf{C}}$ are estimates of the inertia and coriolis terms in the spherical motor dynamics; and \mathbf{y} is given by (13). The uncertainty on the estimates are expressed as

$$[\tilde{\mathbf{H}}] = [\hat{\mathbf{H}}] - [\mathbf{H}] \quad (16)$$

$$[\tilde{\mathbf{C}}] = [\hat{\mathbf{C}}] - [\mathbf{C}] \quad (17)$$

These uncertainties account for both imperfect modeling and intentional computational simplification. Application of the approximate control vector given by (14) leads to the following expression for the closed loop dynamics:

$$[\mathbf{H}]\dot{\mathbf{x}}_2 + \mathbf{C} = [\hat{\mathbf{H}}]\mathbf{y} + \hat{\mathbf{C}} \quad (18)$$

Since the inertia matrix $[\mathbf{H}]$ is symmetric and positive definite, the closed loop dynamics in (17) can be rewritten as

$$\ddot{\mathbf{x}}_1 = \mathbf{y} - \boldsymbol{\eta} \quad (19)$$

where

$$\boldsymbol{\eta} = ([\mathbf{I}] - [\mathbf{H}]^{-1}[\hat{\mathbf{H}}])\mathbf{y} - [\mathbf{H}]^{-1}\tilde{\mathbf{C}} \quad (20)$$

Substitution of (13) for \mathbf{y} in (18) results in the following expression for the closed loop error dynamics.

$$\ddot{\tilde{\mathbf{x}}}_1 + ([\mathbf{K}_p] + [\mathbf{K}_d])\dot{\tilde{\mathbf{x}}}_1 + [\mathbf{K}_p][\mathbf{K}_d]\tilde{\mathbf{x}}_1 = \boldsymbol{\eta} \quad (21)$$

Defining a new error state vector,

$$\boldsymbol{\xi} = \begin{bmatrix} \tilde{\mathbf{x}}_1 \\ \dot{\tilde{\mathbf{x}}}_1 \end{bmatrix} \quad (22)$$

the error dynamics in (20) can be expressed as

$$\dot{\boldsymbol{\xi}} = [\mathbf{F}]\boldsymbol{\xi} + [\mathbf{D}](\ddot{\mathbf{x}}_{1d} - \mathbf{y} + \boldsymbol{\eta}) \quad (23)$$

Where $[\mathbf{F}] = \begin{bmatrix} [\mathbf{0}] & [\mathbf{1}] \\ [\mathbf{0}] & [\mathbf{0}] \end{bmatrix}$ and $[\mathbf{D}] = \begin{bmatrix} [\mathbf{0}] \\ [\mathbf{1}] \end{bmatrix}$ are block matrices of dimensions $R^{6 \times 6}$ and $R^{6 \times 3}$ respectively. Since $\boldsymbol{\eta}$ is a nonlinear function of the position and

velocity state vectors, the system error dynamics in the above equation are nonlinear and coupled. The backstepping controller developed in the previous section cannot guarantee system stability. The Lyapunov direct method is, however, used to design an outer feedback loop on the error dynamics that compensates for the system uncertainty contributed by $\boldsymbol{\eta}$.

Supposed that U is the universe of discourse and x is the element of U , therefore, a crisp set can be defined as a set which consists of different elements (x) will all or no membership in a set. A fuzzy set is a set that each element has a membership grade, therefore it can be written by the following definition;

$$\mathbf{A} = \{\mathbf{x}, \mu_A(\mathbf{x}) | \mathbf{x} \in \mathbf{X}\}; \mathbf{A} \in \mathbf{U} \quad (24)$$

Where an element of universe of discourse is x , μ_A is the membership function (MF) of fuzzy set. The membership function ($\mu_A(x)$) of fuzzy set A must have a value between zero and one. If the membership function $\mu_A(x)$ value equal to zero or one, this set change to a crisp set but if it has a value between zero and one, it is a fuzzy set. Defining membership function for fuzzy sets has divided into two main groups; numerical and functional method, which in numerical method each number has different degrees of membership function and functional method used standard functions in fuzzy sets. The membership function which is often used in practical applications includes triangular form, trapezoidal form, bell-shaped form, and Gaussian form.

Linguistic variable can open a wide area to use of fuzzy logic theory in many applications (e.g., control and system identification). In a natural artificial language all numbers replaced by words or sentences.

If – then Rule statements are used to formulate the condition statements in fuzzy logic. A single fuzzy *If – then* rule can be written by

$$\text{If } x \text{ is } A \text{ Then } y \text{ is } B \quad (25)$$

where A and B are the Linguistic values that can be defined by fuzzy set, the *If – part* of the part of “ x is A ” is called the antecedent part and the *then – part* of the part of “ y is B ” is called the Consequent or Conclusion part. The antecedent of a fuzzy if-then rule can have multiple parts, which the following rules shows the multiple antecedent rules:

$$\text{if } e \text{ is } NB \text{ and } \dot{e} \text{ is } ML \text{ then } T \text{ is } LL \quad (26)$$

where e is error, \dot{e} is change of error, NB is Negative Big, ML is Medium Left, T is torque and LL is Large Left. *If – then* rules have three parts, namely, fuzzify inputs, apply fuzzy operator and apply implication method which in fuzzify inputs the fuzzy statements in the antecedent replaced by the degree of membership, apply fuzzy operator used when the antecedent has multiple parts and replaced by single number between 0 to 1, this part is a degree of support for the fuzzy rule, and apply implication method used in consequent of fuzzy rule to replaced by the degree of membership. The fuzzy inference engine offers a mechanism for transferring the rule base in fuzzy set which it is divided into two most

important methods, namely, Mamdani method and Sugeno method. Mamdani method is one of the common fuzzy inference systems and he designed one of the first fuzzy controllers to control of system engine. Mamdani's fuzzy inference system is divided into four major steps: fuzzification, rule evaluation, aggregation of the rule outputs and defuzzification. Michio Sugeno uses a singleton as a membership function of the rule consequent part. The following definition shows the Mamdani and Sugeno fuzzy rule base

$$\begin{aligned}
 & \text{Mamdani } F.R^1: \text{if } x \text{ is } A \text{ and } \\
 & \quad \quad \quad y \text{ is } B \text{ then } z \text{ is } C \\
 & \text{Sugeno } F.R^1: \text{if } x \text{ is } A \text{ and } \\
 & \quad \quad \quad y \text{ is } B \text{ then } f(x, y) \text{ is } C
 \end{aligned} \tag{27}$$

When x and y have crisp values fuzzification calculates the membership degrees for antecedent part. Rule evaluation focuses on fuzzy operation (AND/OR) in the antecedent of the fuzzy rules. The aggregation is used to calculate the output fuzzy set and several methodologies can be used in fuzzy logic controller aggregation, namely, Max-Min aggregation, Sum-Min aggregation, Max-bounded product, Max-drastic product, Max-bounded sum, Max-algebraic sum and Min-max. Two most common methods that used in fuzzy logic controllers are Max-min aggregation and Sum-min aggregation. Max-min aggregation defined as below

$$\begin{aligned}
 \mu_U(x_k, y_k, U) &= \mu_{\cup_{i=1}^r FR^i}(x_k, y_k, U) \\
 &= \max \left\{ \min_{i=1}^r \left[\mu_{R_{pq}}(x_k, y_k), \mu_{p_m}(U) \right] \right\}
 \end{aligned} \tag{28}$$

The Sum-min aggregation defined as below

$$\begin{aligned}
 \mu_U(x_k, y_k, U) &= \mu_{\cup_{i=1}^r FR^i}(x_k, y_k, U) \\
 &= \sum \min_{i=1}^r \left[\mu_{R_{pq}}(x_k, y_k), \mu_{p_m}(U) \right]
 \end{aligned} \tag{29}$$

where r is the number of fuzzy rules activated by x_k and y_k and also $\mu_{\cup_{i=1}^r FR^i}(x_k, y_k, U)$ is a fuzzy interpretation of i -th rule. Defuzzification is the last step in the fuzzy inference system which it is used to transform fuzzy set to crisp set. Consequently defuzzification's input is the aggregate output and the defuzzification's output is a crisp number. Centre of gravity method (COG) and Centre of area method (COA) are two most common defuzzification methods, which COG method used the following equation to calculate the defuzzification

$$COG(x_k, y_k) = \frac{\sum_i U_i \sum_{j=1}^r \mu_u(x_k, y_k, U_i)}{\sum_i \sum_{j=1}^r \mu_u(x_k, y_k, U_i)} \tag{30}$$

and COA method used the following equation to calculate the defuzzification

$$COA(x_k, y_k) = \frac{\sum_i U_i \cdot \mu_u(x_k, y_k, U_i)}{\sum_i \mu_u(x_k, y_k, U_i)} \tag{31}$$

Where $COG(x_k, y_k)$ and $COA(x_k, y_k)$ illustrates the crisp value of defuzzification output, $U_i \in U$ is discrete element of an output of the fuzzy set, $\mu_u(x_k, y_k, U_i)$ is the fuzzy set membership function, and r is the number of fuzzy rules.

Based on foundation of fuzzy logic methodology; fuzzy logic controller has played important rule to design nonlinear controller for nonlinear and uncertain systems [20-32]. However the application area for fuzzy control is really wide, the basic form for all command types of controllers consists of:

- Input fuzzification (binary-to-fuzzy[B/F]conversion)
- Fuzzy rule base (knowledge base)
- Inference engine
- Output defuzzification (fuzzy-to-binary [F/B] conversion).

Fig 2 shows the block diagram of Fuzzy inference engine.

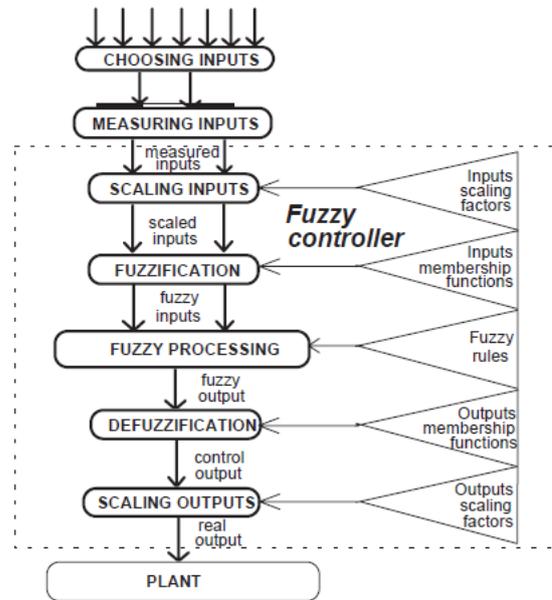


Fig. 2. Fuzzy Controller operation

The backstepping controller for continuum robot is calculated by;

$$U_{B,S} = U_{eq_{B,S}} + D \cdot I \tag{32}$$

Where $U_{B,S}$ is backstepping output function, $U_{eq_{B,S}}$ is backstepping nonlinear equivalent function which can be written as (49) and I is backstepping control law which calculated by (1)

$$U_{eq_{B,S}} = [f + C + G] \tag{33}$$

We have

$$I = [\ddot{e} + (K_1 + K_2) \times e + (K_1 \times K_2) \cdot \dot{e}] \tag{34}$$

Based on above formulation, FIS in this research has an input ($N(q, \dot{q})$) and an output (U_{fuzzy}).

$$U_{fuzzy} = \sum_{l=1}^M \theta^l \zeta(x) \tag{35}$$

To estimate the nonlinear term of backstepping controller fuzzy inference system is used in serial with nonlinear dynamic part

$$U_{eqB.S_{fuzzy}} = [(f + C + G) + \sum_{l=1}^M \theta^T \zeta(x)] \quad (36)$$

Based on above formulation the formulation of backstepping fuzzy control is;

$$U_{B.S} = U_{eqB.S_{fuzzy}} + D.I \quad (37)$$

Most robust control designs are based on the assumption that even though the uncertainty vector η is unknown, some information is available on its bound. In designing the robust backstepping controller, a robust term that compensates for the uncertainty in the system dynamics is added to the control law in (50) and (51).

Fuzzy backstepping controller is robust and stable but in presence of unlimited uncertainty gain tuning method based on fuzzy estimator is presented. In this method the fuzzy out port is used to estimate the system dynamic and also is use to gain tuning.

To reduce the error and increase the stability and robust adaptive methodology is used as a follows;

$$F(q, \dot{q}) = \Theta^{1T} \varepsilon(q, \dot{q}) = [\theta_1^{1T} \varepsilon(q, \dot{q}), \dots, \theta_m^{1T} \varepsilon(q, \dot{q})]^T \quad (38)$$

The adaptation law is given by

$$\dot{\theta}_j^1 = -\Gamma_{1j} s_j \varepsilon(q, \dot{q}) \quad (39)$$

Where $j = 1, \dots, m$ and $\Gamma_{1j} - \Gamma_{3j}$ are positive diagonal matrices.

IV. RESULT AND DISCUSSION

To validate this research backstepping controller and proposed method is compared. These controllers are tested and implemented in MATLAB/SIMULINK.

Tracking performances: In proposed controller the gain is adjusted online depending on the last values of error (e) and change of error (\dot{e}) by gain updating factor (α). Figure 3 shows tracking performance in proposed method, and backstepping controller without disturbance for step trajectory.

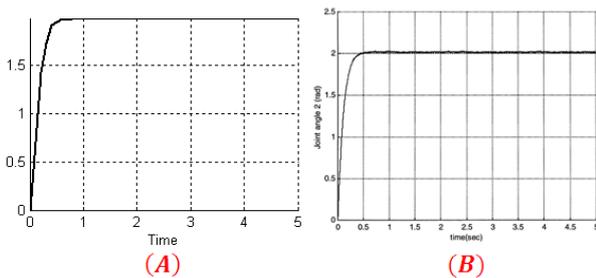


Fig. 3. Tracking performance a) proposed method b)Backstepping

Based on Figure 3 it is observed that, the overshoot in proposed method and backstepping controller are 0%, and rise time in proposed method is 0.48 seconds and in backstepping controller is about 0.6 seconds.

Disturbance rejection: Figure 4 shows the power disturbance elimination in proposed and BSC with disturbance for desired trajectory. In this part between $t=0$ and $t=3$ load moves to position 1. From $t=2$ sec to $t=3$ sec manipulator is carried load and expected to reach to position 2 at $t=5$ sec and disturbance is added to first link, Figure 5 is shown this situation.

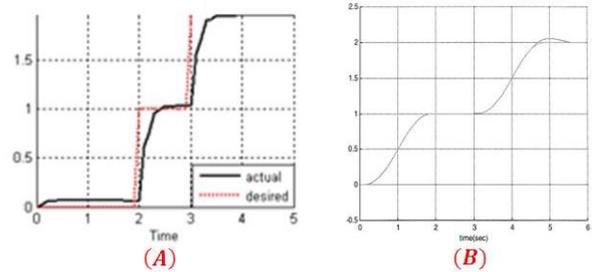


Fig. 4. Disturbance Rejection: a) proposed method b)Backstepping

The disturbance rejection is used to test the robustness comparisons of these two controllers for desired trajectory. Based on Figure 4; by comparing disturbance trajectory response in proposed method and BSC, proposed method overshoot is about (0%) and BSC's overshoot is about (0.4%). Proposed method's rise time (0.48 seconds) is lower than BSC (0.53 second).

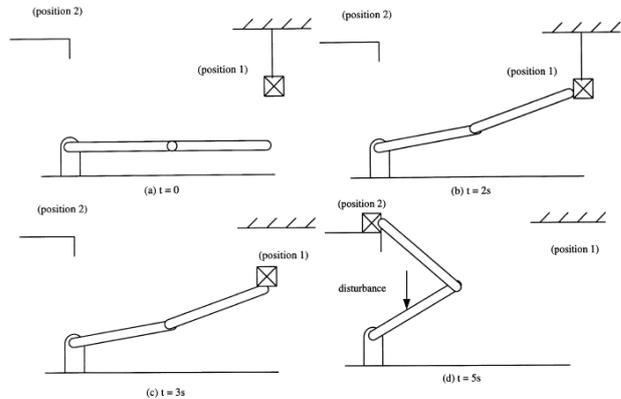


Fig. 5. Complicated situations applied into two controllers

V. CONCLUSION

According to this research, gain updating intelligent backstepping controller is introduced. This research has two objectives: design a Proportional-Derivative (PD) fuzzy system to compensate the system model uncertainties and design tuning gain adaptive methodology according to high quality partly nonlinear methodology. The fuzzy controller is used in this method to system compensation. In real time to increase the system robust fuzzy logic theory is applied to backstepping controller. To approximate a time-varying nonlinear dynamic system, a fuzzy system requires a large amount of fuzzy rule base. The adaptive laws in this algorithm are designed based on the Lyapunov stability theorem.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their careful reading of this paper and for their helpful comments. This work was supported by the SSP Institute of Advance Science and Technology Program of Iran under grant no. 2014-Persian Gulf-2A.

REFERENCES

- [1] T. R. Kurfess, Robotics and automation handbook: CRC, 2005.
- [2] J. J. E. Slotine and W. Li, Applied nonlinear control vol. 461: Prentice hall Englewood Cliffs, NJ, 1991.
- [3] L. Cheng, et al., "Multi-agent based adaptive consensus control for multiple manipulators with kinematic uncertainties," 2008, pp. 189-194.
- [4] J. J. D'Azzo, et al., Linear control system analysis and design with MATLAB: CRC, 2003.
- [5] B. Siciliano and O. Khatib, Springer handbook of robotics: Springer-Verlag New York Inc, 2008.
- [6] I. Boiko, et al., "Analysis of chattering in systems with second-order sliding modes," IEEE Transactions on Automatic Control, vol. 52, pp. 2085-2102, 2007.
- [7] J. Wang, et al., "Indirect adaptive fuzzy sliding mode control: Part I: fuzzy switching," Fuzzy Sets and Systems, vol. 122, pp. 21-30, 2001.
- [8] F. Piltan, et al., "Artificial Control of Nonlinear Second Order Systems Based on AFGSMC," Australian Journal of Basic and Applied Sciences, 5(6), pp. 509-522, 2011.
- [9] V. Utkin, "Variable structure systems with sliding modes," Automatic Control, IEEE Transactions on, vol. 22, pp. 212-222, 2002.
- [10] R. A. DeCarlo, et al., "Variable structure control of nonlinear multivariable systems: a tutorial," Proceedings of the IEEE, vol. 76, pp. 212-232, 2002.
- [11] K. D. Young, et al., "A control engineer's guide to sliding mode control," 2002, pp. 1-14.
- [12] O. Kaynak, "Guest editorial special section on computationally intelligent methodologies and sliding-mode control," IEEE Transactions on Industrial Electronics, vol. 48, pp. 2-3, 2001.
- [13] S. Zahmatkesh, Farzin Piltan, K. Heidari, M. Shamsodini, S. Heidari, "Artificial Error Tuning Based on Design a Novel SISO Fuzzy Backstepping Adaptive Variable Structure Control" International Journal of Intelligent Systems and Applications, vol.5, no.11, pp.34-46, 2013. DOI: 10.5815/ijisa.2013.11.04.
- [14] Meysam Kazeminasab, Farzin Piltan, Zahra Esmaili, Mahdi Mirshekaran, Alireza Salehi, "Design Parallel Fuzzy Partly Inverse Dynamic Method plus Gravity Control for Highly Nonlinear Continuum Robot", IJISA, vol.6, no.1, pp.112-123, 2014. DOI: 10.5815/ijisa.2014.01.12.

Authors' Profiles



Sara Heidari is a mathematical researcher of research and development company SSP. Co. She is an expert artificial intelligence and expert systems in this company. Her research activities deal with the robotic control, artificial intelligence and expert system.



Ali Shahcheraghi is currently working as a primary researcher in the laboratory of Control and Robotic, Institute of Advance Science and Technology, IRAN SSP research and development Center. His current research interests are in the area of nonlinear control, artificial control system and robotics.



Kamran Heidari is a researcher of research and development company SSP. Co. His research activities deal with the robotic control, artificial intelligence and expert system.

Samaneh Zahmatkesh is an Electrical engineer researcher of research and development company SSP. Co. She is an expert artificial intelligence and expert systems in this company. Her research activities deal with the robotic control, artificial intelligence and expert system.



Farzin Piltan was born on 1975, Shiraz, Iran. In 2004 he is joined Institute of Advance Science and Technology, Research and Development Center, IRAN SSP. Now he is a dean of Intelligent Control and Robotics Lab. He is led of team (47 researchers) to design and build of nonlinear control of industrial robot manipulator for experimental research and education and published about 54 Papers in this field since 2010 to date, team supervisor and leader (9 researchers) to design and implement intelligent tuning the rate of fuel ratio in internal combustion engine for experimental research and education and published about 17 Journal papers since 2011 to date, team leader and advisor (34 researchers) of filtering the hand tremors in flexible surgical robot for experimental research and education and published about 31 journal papers in this field since 2012 to date, led of team (21 researchers) to design high precision and fast dynamic controller for multi-degrees of freedom actuator for experimental research and education and published about 7 journal papers in this field since 2013 to date, led of team (22 researchers) to research of full digital control for nonlinear systems (e.g., Industrial Robot Manipulator, IC Engine, Continuum Robot, and Spherical Motor) for experimental research and education and published about 4 journal papers in this field since 2010 to date and finally led of team (more than 130 researchers) to implementation of Project Based-Learning project at IRAN SSP research center for experimental research and education, and published more than 110 journal papers since 2010 to date. In addition to 7 textbooks, Farzin Piltan is the main author of more than 115 scientific papers in refereed journals. He is editorial review board member for 'international journal of control and automation (IJCA)', Australia, ISSN: 2005-4297; 'International Journal of Intelligent System and Applications (IJISA)', Hong Kong, ISSN:2074-9058; 'IAES international journal of robotics and automation, Malaysia, ISSN:2089-4856; 'International Journal of Reconfigurable and Embedded Systems', Malaysia, ISSN:2089-4864. His current research interests are nonlinear control, artificial control system and applied to FPGA, robotics and artificial nonlinear control and IC engine modeling and control.