Self-Adjusting Fuzzy Logic Based Control of Robot Manipulators in Task Space

B. Melih Yilmaz, Enver Tatlicioglu, Aydogan Savran, and Musa Alci

***Abstract*—End effector tracking control of robot manipulators subject to dynamical uncertainties is the main objective of this article. Direct task space control that aims minimizing the end effector tracking error directly is preferred. In the open loop error system, the vector that depends on uncertain dynamical terms is modeled via a fuzzy logic network and a self-adjusting adaptive fuzzy logic component is designed as part of the nonlinear proportional derivative based control input torque. The stability of the closed-loop system is investigated via Lyapunov based arguments and practical tracking is proven. The viability of the proposed control strategy is shown with experimental results. Extensions to uncertain Jacobian case and kinematically redundant robots are also presented.**

***Index Terms*—Adaptive fuzzy logic (AFL), fuzzy approximation, Lyapunov methods, robot manipulators, task space control, universal fuzzy controller.**

I. Introduction

IN THE robot manipulator control literature, significant amount of the past research is devoted to designing tracking controllers in the joint space. However, in most applications, desired task is defined in the end effector space. To address this problem, one line of research prefers to transform the desired task into the joint space and then make use of the vast amount of literature on joint tracking control. The main shortcoming of this method is possible problems due to the inverse kinematics at position level that must be used when transforming the desired task from operational space to the joint space. Other line of research has focused on designing controllers that aim directly minimizing the tracking errors in the end effector space [1], where one of the early results was presented in [2]. In [3], dynamic control of robot manipulators in end effector space was aimed where accurate knowledge of both dynamic and kinematic models were required for implementation. While in [4], an experimental evaluation of control in end effector space for different end effector orientation parameterizations was provided. In [5], quaternion based resolved rate and resolved acceleration end effector space controllers were proposed. Configuration control of kinematically redundant robot manipulators was discussed in [6]. In [7], compliant motion controllers were designed for kinematically redundant robot manipulators where, similar to [3], redundancy was made use of to optimize a user-defined objective function. While not requiring inverse kinematics at position level this method needs the inverse of the Jacobian matrix, which may be problematic when the Jacobian matrix is uncertain and/or not square. Review of control methods at task space can be found in [8] and [9].

Another important research issue independent of the choice of the control method described above is the presence of uncertainties in the model of the robot manipulators. Specifically, the dynamic model of the robot manipulator can include parametric and/or unstructured uncertainties while there may be uncertain kinematic model parameters, and these uncertainties avoid them to be used as part of the control input design. When the uncertainties are linearly parameterizable adaptive methods are in order. Recently, in [10], a neural network based approach was followed to compensate for both dynamic and kinematic uncertainties. In [11] and [12], regulation of robot manipulators in task space was aimed, where the uncertainty of the Jacobian matrix was compensated with an approximate matrix. In [13], tracking control of robot manipulators in end effector space was aimed, where quaternion feedback was used. In [14], the results in [3] are extended to deal with parametric uncertainties in robot's dynamic model. The results in [14] are extended in [15] and [16] with the addition of a novel stability analysis for the subtask objectives. These results are further improved in [17], where asymptotic stability of the subtask objectives were proven. An end effector space tracking controller was designed in [18], where both kinematic and dynamic uncertainties were dealt with gradient based adaptive update laws. In [19], regulation of robot manipulators in end effector space was addressed with an amplitude limited controller, where parametric uncertainties in dynamic and kinematic models were also compensated for. In [20], a sliding mode type control strategy was proposed where dynamic parametric uncertainties was compensated adaptively. One of the shortcomings of the gradient based adaptive techniques is the lack of compensation for unstructured uncertainties which can be addressed via invoking adaptive methods with robust tools. Another shortcoming is the need to find the regressor matrix which is usually specific to the robot manipulator at hand. When there are unstructured uncertainties in the robot dynamic model robust control methods

Manuscript received October 5, 2020; revised January 16, 2021 and February 6, 2021; accepted February 22, 2021. Date of publication March 10, 2021; date of current version October 27, 2021. *(Corresponding author: B. Melih Yilmaz.)*

The authors are with the Department of Electrical and Electronics Engineering, Ege University, Izmir 35100, Turkey (e-mail: bayram.melih.yilmaz@ege.edu.tr; enver.tatlicioglu@ege.edu.tr; aydogan.savran@ege.edu.tr; musa.alci@ege.edu.tr).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TIE.2021.3063970.

Digital Object Identifier 10.1109/TIE.2021.3063970

can be made use of. In [21], a self-tuning adaptive end effector space controller was designed, where bounded uncertainties were also compensated for robustly. In [22], the results in [14] are extended by adding a robustifying component to the controller, where uniform ultimate boundedness (UUB) of the closed-loop system was proven. Tracking control of kinematically redundant robot manipulators subject to parametric uncertainties in both kinematics and dynamics along with unstructured uncertainties in dynamics was addressed in [23], where a UUB result was ensured. In [24], a saturation function based repetitive learning controller was designed that does not require dynamic model knowledge. Recently in [25], a robust signum of the integral of the end effector tracking error based controller was designed where kinematic uncertainties were dealt with adaptively. Robust techniques usually require discontinuous terms and/or rely on the upper bound of the uncertainties, which may be considered as weaknesses.

In this article, tracking control of robot manipulators in task space is aimed. Direct task space control is invoked to avoid the complications and possible problems associated with the inverse kinematics at position level. The task space tracking error is quantified as the difference of the end-effector configuration vector and the desired task space trajectory. The error system development is further pursued via the definition of an auxiliary error like vector that requires the Jacobian matrix. The control problem is complicated by the presence of uncertainties in the dynamic model. In a novel departure from the existing works on task space control, a self-adjusting adaptive fuzzy logic (AFL) based approach [26]–[28] is proposed. Specifically, to avoid finding the regressor matrix and thus to obtain a universally applicable control structure, a fuzzy logic network based modeling is used to compensate for the dynamical uncertainties adaptively. The self-adjusting property of the AFL strategy is obtained by tuning the centers and widths of the membership functions (MFs) via dynamic adaptive update laws. For the MFs, a polynomial structure is preferred to avoid the weaknesses of exponential type MFs not changing much with their spread and piecewise continuous MFs not being continuous. The stability of the closed-loop system is investigated via novel Lyapunov based arguments and practical tracking is proven. Experimental verification of the proposed AFL based robust control is demonstrated on an in-house developed robot manipulator.

In addition to the dynamical uncertainties, kinematic uncertainties are also encountered. These uncertainties usually possess the linear parameterization property, where the regressor matrix is relatively easier to obtain. To address this research issue, an uncertain Jacobian matrix extension is also aimed. The uncertainties in the Jacobian matrix avoided it to be used as part of the error system definition and thus a different auxiliary error like vector definition is introduced. Via the design of an adaptive update rule for the uncertain kinematic parameters the kinematic uncertainties are dealt with while the previously designed self-adjusting AFL-based controller is modified accordingly. To investigate the stability of the closed-loop system, a new Lyapunov function is introduced, where practical tracking is ensured. Experiments are also performed to demonstrate the viability of the proposed controller.

When the dimension of the end effector space is less than the number of degree of freedom (DOF), the robot manipulator is called kinematically redundant [29] and the Jacobian matrix is not square. This constitutes a problem as the definition of the auxiliary error like vector that ties the end effector tracking objective to the dynamic model relies on the inverse of the Jacobian matrix. To circumvent this problem, an extension to the kinematically redundant robot manipulators is also studied. The difference between the number of DOF and the dimension of the end effector space shows the level of redundancy and may be used to achieve secondary control objectives [30], [31]. Examples of secondary objectives are singularity avoidance, joint limit avoidance, bounding the impact forces, and bounding the potential energy. These secondary control objectives are required to be designed appropriately as they are achieved via the self-motion of the robot without affecting the end effector tracking objective. In this part of the study, a new definition is proposed for the error like vector to include a pseudoinverse of the Jacobian matrix and the control design is modified accordingly. The stability of the closed-loop system is investigated in two steps. In the first part, the stability of the task space tracking error is investigated in a similar manner to that of the nonredundant case and practical tracking is achieved. In the second part, the stability of the secondary objective is investigated and UUB of the secondary objective is proven.

The rest of this article is organized as follows. In Section II, kinematic and dynamic models along with model properties are presented. In Section III, error system development along with the control design are presented for the known Jacobian case, while the associated stability analysis is given in Section IV. The derivations for the uncertain Jacobian case as a whole are shown in Section V. Section VI includes the experimental results for both known Jacobian and uncertain Jacobian cases. Finally, Section VII concludes this article.

II. Kinematic and Dynamic Models

The kinematic model of an n DOF revolute joint robot manipulator operating on n dimensional space is acquired from [32]

$$x = f(q) \tag{1}$$

where $x(t) \in \mathbb{R}^n$ is the task space position, $q(t) \in \mathbb{R}^n$ represents the joint position vector and $f : \mathbb{R}^n \to \mathbb{R}^n$ denotes the forward kinematics. By differentiating (1), velocity kinematics is obtained as

$$\dot{x} = J\dot{q} \tag{2}$$

where $\dot{x}(t) \in \mathbb{R}^n$ denotes the task space velocity, $\dot{q}(t) \in \mathbb{R}^n$ is the joint velocity vector and $J(q) \in \mathbb{R}^{n\times n}$ represents the Jacobian matrix defined as

$$J = \frac{\partial f}{\partial q}. \tag{3}$$

Remark 1: In accordance with the literature on task space control, all kinematic singularities are considered to be always avoided thus the inverse of the Jacobian matrix $J^{-1}(q)$ exists for all possible $q(t)$ [32].

Remark 2: $J(q)$ and $J^{-1}(q)$ depend on $q(t)$ via trigonometric functions only and thus they are bounded $\forall q$.

The dynamical model of the robot manipulator has the following structure [32]

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) + F\dot{q} = \tau \tag{4}$$

where $\ddot{q}(t) \in \mathbb{R}^n$ represents joint acceleration vector, $M(q) \in \mathbb{R}^{n\times n}$ is the positive definite and symmetric inertia matrix, $C(q,\dot{q}) \in \mathbb{R}^{n\times n}$ models the centripetal and Coriolis terms, $G(q) \in \mathbb{R}^n$ represents the gravitational effects, $F \in \mathbb{R}^{n\times n}$ denotes viscous frictional effects, and $\tau(t) \in \mathbb{R}^n$ is the control input torque. The above dynamical terms satisfy the below properties.

Property 1: As a direct consequence of utilizing Lagrange formulation in obtaining the above dynamics, the inertia matrix $M(q)$ is positive definite and symmetric and satisfies following inequalities [32]

$$m_1\|\eta\|^2 \le \eta^T M(q)\eta \le m_2\|\eta\|^2 \quad \forall \eta \in \mathbb{R}^n \tag{5}$$

where $m_1, m_2 \in \mathbb{R}$ are positive bounding constants.

Property 2: In the Lagrange's formulation, the velocity dependent terms can be arranged by utilizing Christoffel symbols to make the centripetal Coriolis matrix and the time derivative of the inertia matrix satisfy following skew symmetry property [33]

$$\eta^T(\dot{M} - 2\,C)\eta = 0 \quad \forall \eta \in \mathbb{R}^n. \tag{6}$$

III. Error System Development and Control Design

The main objective of the controller design is to ensure tracking of a desired task space position vector $x_d(t) \in \mathbb{R}^n$ which along with its first two time derivatives are considered to be bounded functions of time. The dynamic model in (4) is uncertain and thus can not be benefited as part of the control design so the error system development must be tailored accordingly.

The task space tracking error, denoted by $e(t) \in \mathbb{R}^n$, is defined as

$$e \triangleq x_d - x. \tag{7}$$

Differentiating (7) by utilizing (2) yields

$$\dot{e} = \dot{x}_d - J\dot{q}. \tag{8}$$

To ease the presentation of the rest of the design and analysis, an auxiliary error-like term, shown with $r(t) \in \mathbb{R}^n$, is defined as

$$r \triangleq J^{-1}(\dot{x}_d + \alpha e) - \dot{q} \tag{9}$$

in which $\alpha \in \mathbb{R}^{n\times n}$ is a constant, positive definite, diagonal gain matrix. Substituting (9) into (8) yields

$$\dot{e} = -\alpha e + Jr. \tag{10}$$

The definition of $r(t)$ in (9) can also be considered as a transformation between task space in which tracking is aimed and joint space where control input torque is to be designed [14]. In an attempt to reach open loop dynamics for the auxiliary error-like term $r(t)$, its first time derivative is taken and then multiplied with the inertia matrix to deduce

$$M\dot{r} = -Cr - \tau + f \tag{11}$$

where (4) and (8) along with time derivative of (7) were utilized with $f(x,\dot{x},x_d,\dot{x}_d,\ddot{x}_d) \in \mathbb{R}^n$ being defined as

$$f \triangleq M\frac{d}{dt}[J^{-1}(\dot{x}_d + \alpha e)] + CJ^{-1}(\dot{x}_d + \alpha e) + G + F\dot{q}. \tag{12}$$

Since the dynamic modeling matrices $M(q), C(q,\dot{q}), G(q)$, and F include uncertainties, f defined above is also uncertain. By utilizing the universal approximation property of fuzzy logic network [34]–[36], the auxiliary uncertain term in (12) can be modeled as

$$f = w^T\phi + \epsilon \tag{13}$$

where $w \in \mathbb{R}^{a\times n}$ is the constant control representative value matrix, $\phi(p,\Delta,\chi) \in \mathbb{R}^{a\times 1}$ is the membership value vector with $a \triangleq 5\prod_{i=1}^{n}\Lambda_i$ where Λ_i denote the number of fuzzy rules, p, $\Delta \in \mathbb{R}^{b\times 1}$ are centers and widths with $b \triangleq 5\sum_{i=1}^{n}\Lambda_i$, $\chi(t) \triangleq [x,\dot{x},x_d,\dot{x}_d,\ddot{x}_d] \in \mathbb{R}^{5n}$ and $\epsilon(\chi) \in \mathbb{R}^n$ is the functional approximation error satisfying $\|\epsilon\| \le \bar{\epsilon}$ for known bound $\bar{\epsilon}$.

Utilizing (13) and (11) can be rewritten as

$$M\dot{r} = -Cr - \tau + w^T\phi + \epsilon. \tag{14}$$

Based on the subsequent analysis, the control input torque is designed as

$$\tau = J^Te + K_r r + \hat{w}^T\hat{\phi} \tag{15}$$

where $K_r \in \mathbb{R}^{n\times n}$ is a constant, positive definite, diagonal control gain matrix, $\hat{w}(t) \in \mathbb{R}^{a\times n}$ is the estimate of control representative value matrix, $\hat{\phi}(\hat{p},\hat{\Delta},\chi) \triangleq \phi|_{p=\hat{p},\Delta=\hat{\Delta}} \in \mathbb{R}^{a\times 1}$ is the estimate of membership value vector with $\hat{p}(t)$, $\hat{\Delta}(t) \in \mathbb{R}^{b\times 1}$ denoting estimated centers and widths, respectively. Control representative values, width and center values are updated adaptively according to

$$\dot{\hat{w}} = \Gamma_w\hat{\phi}r^T - \Gamma_w\frac{\partial\hat{\phi}}{\partial\hat{\Delta}}\hat{\Delta}r^T - \Gamma_w\frac{\partial\hat{\phi}}{\partial\hat{p}}\hat{p}r^T - k_w\|r\|\Gamma_w\hat{w} \tag{16}$$

$$\dot{\hat{\Delta}} = \Gamma_\Delta\left(\frac{\partial\hat{\phi}}{\partial\hat{\Delta}}\right)^T\hat{w}r - k_\Delta\|r\|\Gamma_\Delta\hat{\Delta} \tag{17}$$

$$\dot{\hat{p}} = \Gamma_p\left(\frac{\partial\hat{\phi}}{\partial\hat{p}}\right)^T\hat{w}r - k_p\|r\|\Gamma_p\hat{p} \tag{18}$$

where $\Gamma_w \in \mathbb{R}^{a\times a}$, Γ_Δ, $\Gamma_p \in \mathbb{R}^{b\times b}$ are constant, positive definite, diagonal adaptation gain matrices, and k_w, k_Δ, $k_p \in \mathbb{R}$ are constant, positive adaptation gains. To ensure boundedness of the estimations, a projection algorithm (such as the one in [37]) is considered to be used on the right-hand sides of (16)–(18). The control input torque design in (15) includes a nonlinear proportional derivative controller $J^Te + K_r r$ fused with an AFL component with self-adjusting MFs for uncertainty compensation. By substituting the control input torque design in (15) to the open loop error system in (14), the closed-loop error system is concluded as

$$M\dot{r} = -Cr - K_r r - J^Te + \epsilon + w^T\phi - \hat{w}^T\hat{\phi}. \tag{19}$$

IV. Stability Analysis

In this section, in view of the previously obtained error dynamics, the stability of the closed-loop system will be investigated by utilizing Lyapunov type analysis methods.

Theorem 1: The control input torque in (15) and the update rules in (16)–(18) ensures boundedness of the closed-loop system and guarantees practical tracking of a desired task space position vector.

Proof: To prove the theorem, a Lyapunov function, shown with $V_1(t) \in \mathbb{R}$, is defined as

$$V_1 \triangleq \frac{1}{2}e^T e + \frac{1}{2}r^T M r + \frac{1}{2}\text{tr}\{\tilde{w}^T \Gamma_w^{-1} \tilde{w}\} + \frac{1}{2}\tilde{\Delta}^T \Gamma_\Delta^{-1} \tilde{\Delta} + \frac{1}{2}\tilde{p}^T \Gamma_p^{-1} \tilde{p} \tag{20}$$

in which $\text{tr}\{\cdot\}$ is the trace operator and $\tilde{w}(t) \in \mathbb{R}^{a\times n}$, $\tilde{\Delta}(t)$, $\tilde{p}(t) \in \mathbb{R}^{b\times 1}$ are estimation errors defined as

$$\tilde{w} \triangleq w - \hat{w},\ \tilde{\Delta} \triangleq \Delta - \hat{\Delta},\ \tilde{p} \triangleq p - \hat{p}. \tag{21}$$

Via using projection algorithm on right-hand sides of (16)–(18), $\|\tilde{w}(t)\|_{i\infty} \le \bar{\tilde{w}}$, $\|\tilde{\Delta}(t)\| \le \bar{\tilde{\Delta}}$, $\|\tilde{p}(t)\| \le \bar{\tilde{p}}$ are guaranteed for known positive bounding constants $\bar{\tilde{w}}$, $\bar{\tilde{\Delta}}$, $\bar{\tilde{p}}$, and in view of these bounds, the expression in (20) can be lower and upper bounded as

$$\frac{1}{2}\min\{1, m_1\}\|z\|^2 \le V_1 \le \frac{1}{2}\max\{1, m_2\}\|z\|^2 + c_1 \tag{22}$$

$$c_1 \triangleq \frac{\bar{\tilde{w}}^2 \lambda_{\max}\{\Gamma_w^{-1}\}}{2} + \frac{\bar{\tilde{\Delta}}^2 \lambda_{\max}\{\Gamma_\Delta^{-1}\}}{2} + \frac{\bar{\tilde{p}}^2 \lambda_{\max}\{\Gamma_p^{-1}\}}{2} \tag{23}$$

where the notation $\lambda_{\max}\{\cdot\}$ denotes the maximum eigenvalue of a matrix, $z(t) \triangleq [\,e^T\ r^T\,]^T \in \mathbb{R}^{2n}$ is the combined error vector and $\|\cdot\|_{i\infty}$ denotes the induced infinity norm.

The time derivative of (20) is obtained as

$$\dot{V}_1 = e^T \dot{e} + r^T M \dot{r} + \frac{1}{2} r^T \dot{M} r + \text{tr}\{\tilde{w}^T \Gamma_w^{-1} \dot{\tilde{w}}\} + \tilde{\Delta}^T \Gamma_\Delta^{-1} \dot{\tilde{\Delta}} + \tilde{p}^T \Gamma_p^{-1} \dot{\tilde{p}}. \tag{24}$$

Substituting (10) and (19), and time derivatives of estimation errors in (21) into (24) yields

$$\dot{V}_1 = e^T(Jr - \alpha e) + r^T\left(-Cr - K_r r - J^T e + \epsilon + w^T \phi - \hat{w}^T \hat{\phi}\right) + \frac{1}{2} r^T \dot{M} r - \text{tr}\{\tilde{w}^T \Gamma_w^{-1} \dot{\hat{w}}\} - \tilde{\Delta}^T \Gamma_\Delta^{-1} \dot{\tilde{\Delta}} - \tilde{p}^T \Gamma_p^{-1} \dot{\tilde{p}} \tag{25}$$

and after canceling common terms, using Property 2, adding and subtracting $r^T w^T \hat{\phi}$ yields

$$\dot{V}_1 = -e^T \alpha e - r^T K_r r + r^T w^T \tilde{\phi} + r^T \tilde{w}^T \hat{\phi} + r^T \epsilon - \text{tr}\{\tilde{w}^T \Gamma_w^{-1} \dot{\hat{w}}\} - \tilde{\Delta}^T \Gamma_\Delta^{-1} \dot{\tilde{\Delta}} - \tilde{p}^T \Gamma_p^{-1} \dot{\tilde{p}} \tag{26}$$

where $\tilde{\phi}(t) = \phi - \hat{\phi} \in \mathbb{R}^{a\times 1}$ can be rewritten via using the Taylor series expansion as [38], [39]

$$\tilde{\phi} = \frac{\partial \hat{\phi}}{\partial \hat{p}}\tilde{p} + \frac{\partial \hat{\phi}}{\partial \hat{\Delta}}\tilde{\Delta} + \xi \tag{27}$$

in which $\xi(\tilde{p}, \tilde{\Delta})$ represents higher order terms. Substituting (27) and the adaptive update rules in (16)–(18) into (26) and then canceling common terms gives

$$\dot{V}_1 = -e^T \alpha e - r^T K_r r + r^T \epsilon + r^T w^T \xi + k_w \|r\| \text{tr}\{\tilde{w}^T \hat{w}\} - \Delta^T \left(\frac{\partial \hat{\phi}}{\partial \hat{\Delta}}\right)^T \tilde{w} r + k_\Delta \|r\| \tilde{\Delta}^T \hat{\Delta} - p^T \left(\frac{\partial \hat{\phi}}{\partial \hat{p}}\right)^T \tilde{w} r + k_p \|r\| \tilde{p}^T \hat{p} \tag{28}$$

where $\text{tr}\{\eta + \nu\} = \text{tr}\{\eta\} + \text{tr}\{\nu\}$ and $\text{tr}\{\eta\nu\} = \text{tr}\{\nu\eta\}$ [38] were used as well. Following bounds can be obtained [26]:

$$|r^T w^T \xi(\tilde{p}, \tilde{\Delta})| \le (a_0 + a_1\|\tilde{\Delta}\| + a_2\|\tilde{p}\|)\|r\| \tag{29}$$

$$\left|\Delta^T \left(\frac{\partial \hat{\phi}}{\partial \hat{\Delta}}\right)^T \tilde{w} r\right| \le a_3 \|\tilde{w}\|_{i\infty}\|r\| \tag{30}$$

$$\left|p^T \left(\frac{\partial \hat{\phi}}{\partial \hat{p}}\right)^T \tilde{w} r\right| \le a_4 \|\tilde{w}\|_{i\infty}\|r\| \tag{31}$$

where a_0, a_1, a_2, a_3, $a_4 \in \mathbb{R}$ are known positive constants. In obtaining the above bounds, the boundedness of ξ in [26] and [40] was used along with the boundedness of $\hat{w}$, $\hat{\Delta}$, $\hat{p}$ ensured via utilizing a projection algorithm on the right-hand sides of (16)–(18) and w, Δ, p being constant vectors. By using the upper bounds in (29)–(31), the right-hand side of (28) can further be upper bounded as

$$\dot{V}_1 \le -e^T \alpha e - r^T K_r r + a_0\|r\| + \bar{\epsilon}\|r\| + \|r\|\left[(a_3 + a_4)\|\tilde{w}\|_{i\infty} + k_w \text{tr}\{\tilde{w}^T w\} - k_w \text{tr}\{\tilde{w}^T \tilde{w}\}\right] + \|r\|\left[a_1\|\tilde{\Delta}\| + k_\Delta \tilde{\Delta}^T \Delta - k_\Delta \tilde{\Delta}^T \tilde{\Delta}\right] + \|r\|\left[a_2\|\tilde{p}\| + k_p \tilde{p}^T p - k_p \tilde{p}^T \tilde{p}\right] \tag{32}$$

and utilizing $\|w\|_{i\infty} \le \bar{w}$, $\|\Delta\| \le \bar{\Delta}$, $\|p\| \le \bar{p}$ for known positive constants $\bar{w}$, $\bar{\Delta}$, $\bar{p}$, and then completing the squares deduces to

$$\dot{V}_1 \le -e^T \alpha e - r^T K_r r + \|r\|\left[a_0 + \epsilon + \frac{(a_3 + a_4 + k_w \bar{w})^2}{4k_w} + \frac{(a_1 + k_\Delta \bar{\Delta})^2}{4k_\Delta} + \frac{(a_2 + k_p \bar{p})^2}{4k_p}\right]. \tag{33}$$

Following compact upper bound is obtained for the right-hand side of (33)

$$\dot{V}_1 \leq -\beta\|z\|^2 + \frac{\gamma^2}{4\delta} \tag{34}$$

in which γ is a positive constant bigger than the square bracketed term of (33) and $\beta \in \mathbb{R}$ is a positive constant defined as

$$\beta \triangleq \min\{\lambda_{\min}(\alpha), \lambda_{\min}(K_r) - \delta\} \tag{35}$$

where the notation $\lambda_{\min}\{\cdot\}$ denotes the minimum eigenvalue of a matrix and $\gamma\|r\| \leq \frac{\gamma^2}{4\delta} + \delta\|r\|^2$ was made use of for a positive damping constant δ.

From the structures of (22) and (34), $V(t) \in \mathcal{L}_\infty$ can be shown and thus $e(t), r(t)$ can be proven to be bounded. Standard signal chasing arguments can be used to ensure boundedness of all the signals under the closed-loop operation. Also, from (22) and (34), UUB of $e(t)$ and $r(t)$ are guaranteed with the size of the ultimate bounds depending on constants on the right-hand sides of (22) and (34), which can be adjusted via control and adaptation gains, thus practical tracking of a desired task space position vector is ensured. ■

V. Uncertain Jacobian Matrix Extension

When the kinematic parameters of the robot manipulator are not known accurately, the Jacobian matrix includes parametric uncertainties and thus can not be utilized in the definition of the auxiliary error-like vector and in the control design. Thus, the definition of r and the control design are required to be revised based on this restriction. In this section, an adaptive update rule for the uncertain kinematic parameters will be proposed to be fused with the control strategy in the previous part. The task space variables $x(t)$ and $\dot{x}(t)$ are considered to be available via alternative sensing methods.

Property 3: The velocity kinematics in (2) is linearly parameterizable in the sense that

$$J\dot{q} = W_J\Phi_J \tag{36}$$

where $W_J(q, \dot{q}) \in \mathbb{R}^{n\times s}$ is the known regression matrix and $\Phi_J \in \mathbb{R}^s$ denotes unknown constant kinematic parameter vector.

In this case, due to the unavailability of J, the interconnection between task space and joint space is provided by $r(t)$ that is defined as [25]

$$r \triangleq \hat{J}^{-1}(\dot{x}_d + \alpha e) - \dot{q} \tag{37}$$

where $\hat{J}(q) \in \mathbb{R}^{n\times n}$ is the estimated Jacobian matrix obtained by substituting the uncertain kinematic parameters Φ_J with their estimations $\hat{\Phi}_J(t) \in \mathbb{R}^s$. In view of (36), following expression is obtained

$$\hat{J}\dot{q} = W_J\hat{\Phi}_J \tag{38}$$

and subtracting (38) from (36) yields

$$\tilde{J}\dot{q} = W_J\tilde{\Phi}_J \tag{39}$$

where $\tilde{J}(q) \triangleq J - \hat{J} \in \mathbb{R}^{n\times n}$ and $\tilde{\Phi}_J(t) \triangleq \Phi_J - \hat{\Phi}_J \in \mathbb{R}^s$. In view of the new definition of $r(t)$ in (37), the time derivative of the task space tracking error is obtained as

$$\dot{e} = -\hat{J}r - \alpha e - \tilde{J}\dot{q} \tag{40}$$

which includes the extra term $\tilde{J}\dot{q}$ when compared with (8). The open loop error dynamics has the same structure as in (11) but due to the new definition of $r(t)$ the auxiliary vector $f(x, \dot{x}, x_d, \dot{x}_d, \ddot{x}_d) \in \mathbb{R}^n$ is now defined as

$$f \triangleq M\frac{d}{dt}[\hat{J}^{-1}(\dot{x}_d + \alpha e)] + C\hat{J}^{-1}(\dot{x}_d + \alpha e) + G + F\dot{q}. \tag{41}$$

Similarly, f is expressed via the fuzzy logic structure given in (13). Based on error system development and subsequent stability analysis, the control input torque is now designed as

$$\tau = \hat{J}^T e + K_r r + \hat{w}^T\hat{\phi} \tag{42}$$

and $\hat{\Phi}_J(t)$ is updated according to

$$\dot{\hat{\Phi}} = -\Gamma_J W_J^T e \tag{43}$$

where $\Gamma_J \in \mathbb{R}^{n\times n}$ is a constant, positive definite, diagonal adaptation gain matrix, and the projection algorithm in [18] is considered to be utilized on the right-hand side to ensure boundedness of both the parameter updates and its time derivative and also existence of the inverse of $\hat{J}$. The closed-loop error system is obtained as

$$M\dot{r} = -Cr - K_r r - \hat{J}^T e + \epsilon + w^T\phi - \hat{w}^T\hat{\phi}. \tag{44}$$

To investigate the stability of the closed-loop system, the Lyapunov function, shown with $V_2(t) \in \mathbb{R}$ is defined as

$$V_2 \triangleq V_1 + \frac{1}{2}\tilde{\Phi}_J^T\Gamma_J^{-1}\tilde{\Phi}_J. \tag{45}$$

The expression in (45) can be lower and upper bounded as

$$\frac{1}{2}\min\{1, m_1\}\|z\|^2 \leq V_2 \leq \frac{1}{2}\max\{1, m_2\}\|z\|^2 + c_2 \tag{46}$$

$$c_2 \triangleq \frac{\bar{\tilde{w}}^2\lambda_{\max}\{\Gamma_w^{-1}\}}{2} + \frac{\bar{\tilde{\Delta}}^2\lambda_{\max}\{\Gamma_\Delta^{-1}\}}{2} + \frac{\bar{\tilde{p}}^2\lambda_{\max}\{\Gamma_p^{-1}\}}{2} + \frac{\bar{\tilde{\Phi}}_J^2\lambda_{\max}\{\Gamma_J^{-1}\}}{2} \tag{47}$$

where $\|\tilde{\Phi}_J(t)\| \leq \bar{\tilde{\Phi}}_J$ is ensured via the use of a projection algorithm on the right-hand side of (43) for known positive bounding constant $\bar{\tilde{\Phi}}_J \in \mathbb{R}$. The time derivative of (45) can be obtained as

$$\dot{V}_2 = e^T\dot{e} + r^T M\dot{r} + \frac{1}{2}r^T\dot{M}r + \text{tr}\{\tilde{w}^T\Gamma_w^{-1}\dot{\tilde{w}}\} + \tilde{\Delta}^T\Gamma_\Delta^{-1}\dot{\tilde{\Delta}} + \tilde{p}^T\Gamma_p^{-1}\dot{\tilde{p}} + \tilde{\Phi}_J^T\Gamma_J^{-1}\dot{\tilde{\Phi}}_J. \tag{48}$$

Utilizing similar steps to the ones in (25)–(35), the same compact upper bound in (33) is obtained for the right hand side of (48) as well. Thus, practical tracking of a desired task space position vector is ensured.

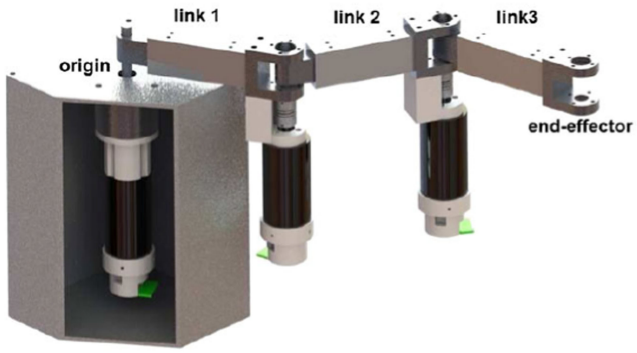


Fig. 1. Three-DOF planar robot manipulator.

TABLE I
CHARACTERISTIC OF MOTORS AND DRIVERS

E37576 Maxon Motor	
Characteristic	Value
Nominal voltage	24 VDC
Nominal speed	5530 rpm
Speed constant	263 rpm/V
Torque constant	36.4×10^{-3} Nm/A
Nominal torque	78.2×10^{-3} Nm
Maximum power of driver	72 W

VI. EXPERIMENT RESULTS

To demonstrate the performance of the proposed controllers, experiments were fulfilled with the 3 DOF planar robot manipulator presented in Fig. 1, where properties of its motors and motor drivers are presented in Table I.

To measure the joint positions AS5045 magnetic rotary encoders based on contactless magnetic sensor technology were used. Quark Q8 data acquisition card is used to realize the data transmission between computer and drivers. The control signals received from the data acqusition card as pulsewidth modulation are transmitted to motor drivers and encoder signals are received as quadrature counter inputs. The control algorithm was run on MATLAB Simulink with real time windows target with a sampling rate of 0.001 s. In the experiments, to achieve a nonredundant robot manipulator, the first joint was locked mechanically and the last two links were used. The Jacobian matrix has the following form:

$$J = \begin{bmatrix} -l_1\sin(q_1) - l_2\sin(q_1+q_2) & -l_2\sin(q_1+q_2) \\ l_1\cos(q_1) + l_2\cos(q_1+q_2) & l_2\cos(q_1+q_2) \end{bmatrix} \quad (49)$$

where q_1 and q_2 are the joint positions and l_1 and l_2 are link lengths. The manipulator was initially at rest at the joint position $q(0) = [0, \pi/2]^T$[rad]. The desired task space position vector was designed as

$$\dot{x}_d(t) = \begin{bmatrix} -0.015\sin(0.1t) \\ 0.01\cos(0.2t) \end{bmatrix}, x_d(0) = \begin{bmatrix} 0.65 \\ 0.25 \end{bmatrix}[m]. \quad (50)$$

The gains were tuned via trial and error method[1] and the controller gains were selected as $K_r = \text{diag}\{50, 20\}$, $\alpha = \text{diag}\{8, 7.6\}$ while the adaptation gains were set to $\Gamma_w = I_a$, $\Gamma_\Delta = 0.02I_b$, $\Gamma_p = 0.07I_b$, $k_w = 0.002$, $k_\Delta = 0.8$, $k_p = 0.5$. The number of fuzzy rules were set to $\Lambda_1 = \Lambda_2 = 3$ (*i.e.*, 3-rule), and the initial values of estimations of control representative values and centers of MFs were selected randomly in the range of $[-1, 1]$, while the initial values of estimated widths were selected randomly in the range of [0,1]. For the uncertain Jacobian extension, in view of (49), the regression matrix W_j is obtained as

$$W_J = \begin{bmatrix} -\sin(q_1)\dot{q}_1 & -\sin(q_1+q_2)(\dot{q}_1+\dot{q}_2) \\ \cos(q_1)\dot{q}_1 & \cos(q_1+q_2)(\dot{q}_1+\dot{q}_2) \end{bmatrix} \quad (51)$$

and the adaptive gain matrix was chosen as $\Gamma_J = I_2$ and the initial values of the adaptive updates of both link lengths were set as $\hat{l}_1(0) = \hat{l}_2(0) = 0.5$.

The experiment results are presented in Figs. 2–6. The tracking error and control input torque for known Jacobian case are presented in Figs. 2 and 3, respectively. The position tracking error, control input torque and estimates of link lengths are presented in Figs. 4–6, respectively, for the uncertain Jacobian extension. From Figs. 2 and 4, it is clear that satisfactory tracking were achieved for both known Jacobian and uncertain Jacobian cases, respectively, for reasonable control input torques

[1]The gain tuning process was initiated with conservative (i.e., big) gains and when the experiments worked smaller control gains were tried until satisfactory tracking performance was obtained.

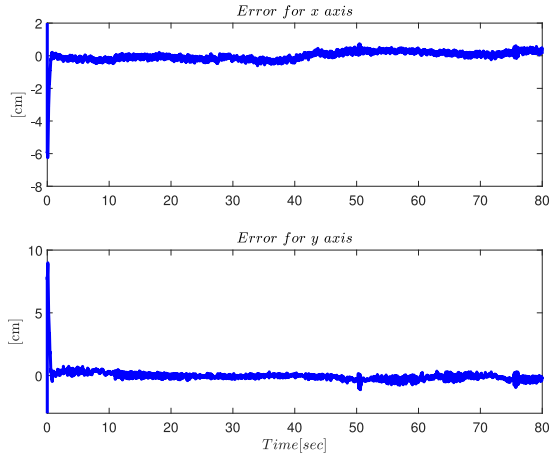


Fig. 2. Known Jacobian case: Task space position tracking error $e(t)$.

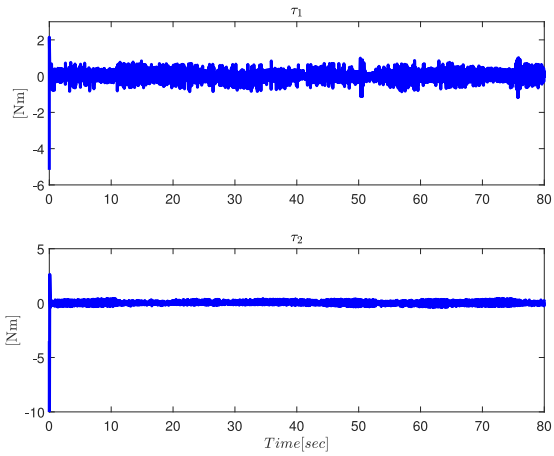


Fig. 3. Known Jacobian case: Control input torque $\tau(t)$.

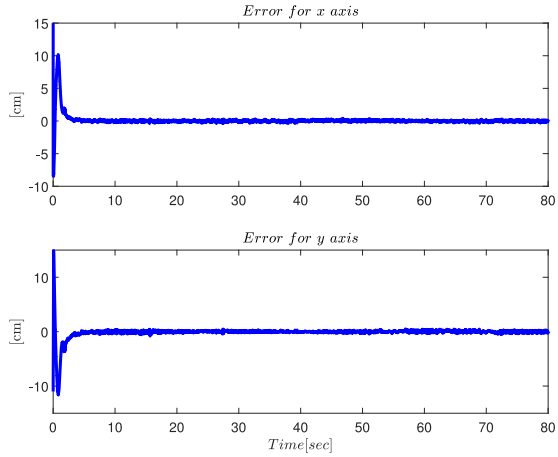


Fig. 4. Uncertain Jacobian case: Task space position tracking error $e(t)$.

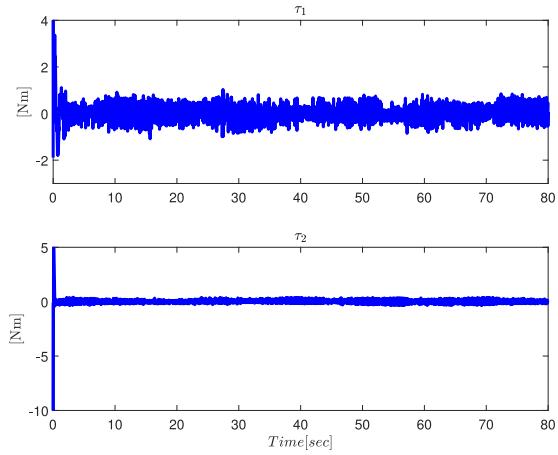


Fig. 5. Uncertain Jacobian case: Control input torque $\tau(t)$.

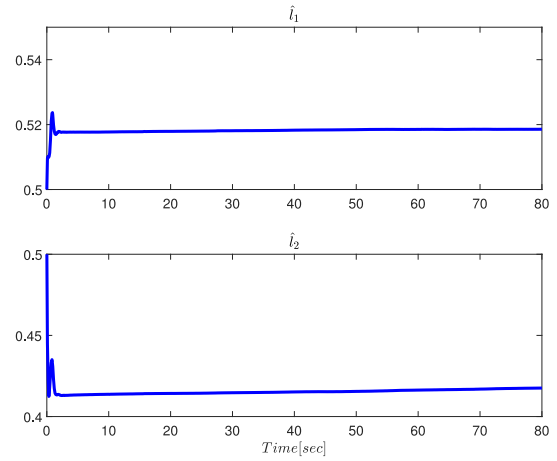


Fig. 6. Uncertain Jacobian case: Estimates of link lengths.

presented in Figs. 3 and 5. Specifically, after the transients ended, the steady-state behavior of the tracking error remained within a small neighborhood of zero. The integral of the square of the norm of tracking error and control input are presented in Table II. From the first and fourth lines of the table, it can be seen that in the known Jacobian case despite requiring higher control effort a significantly better tracking performance was achieved.

TABLE II
PERFORMANCE MEASURES

Type of the controller	$\int \|\|e(\nu)\|\|^2 d\nu$	$\int \|\|\tau(\nu)\|\|^2 d\nu$
$J^T e + K_r r + \hat{w}\hat{\phi}(\hat{p}, \hat{\Delta}, \chi)$	0.0870	5.758×10^3
$J^T e + K_a r + \hat{w}_{nn}^T \psi(\chi)$	0.3315	5.811×10^3
$J^T e + K_r r + \hat{w}^T \phi(p_c, \Delta_c, \chi)$	0.5828	1.399×10^4
$\hat{J}^T e + K_r r + \hat{w}\hat{\phi}(\hat{p}, \hat{\Delta}, \chi)$	0.5423	4.183×10^3
$\hat{J}^T e + K_b r + \hat{w}_{nn}^T \psi(\chi)$	1.2020	4.250×10^3
$\hat{J}^T e + K_r r + \hat{w}^T \phi(p_c, \Delta_c, \chi)$	0.6859	7.285×10^3

The performance of the controllers in (15) and (42) were compared by substituting their AFL component $\hat{w}^T \hat{\phi}(\hat{p}, \hat{\Delta}, \chi)$ with two other adaptive structure. For the first comparison, another AFL structure of the form $\hat{w}^T \phi(p_c, \Delta_c, \chi)$ was used in which p_c and Δ_c are constant centers and widths which are chosen as vectors with all entries being equal to 1 and $\hat{w}$ was updated adaptively via (16). For the second comparison, an adaptive neural network structure of the form $\hat{w}_{nn}^T \psi(\chi)$ was used with $\psi(\chi)$ being the activation function chosen as hyperbolic tangent function and the weight matrix is updated according to $\dot{\hat{w}}_{nn} = \Gamma_{nn} \psi(\chi) r^T$ with $\Gamma_{nn} = 20$ as given in [41, eq. (4.2.9)]. It is highlighted that both of these structures are model independent adaptive control strategies. In an attempt to quantify the performances, experiments were conducted for each case separately. When comparing the proposed controllers in (15) and (42) with the adaptive neural network controller, the control gains are adjusted as $K_a = \text{diag}\{55, 22\}$ and $K_b = \text{diag}\{65, 26\}$, respectively, so that similar amount of control efforts were obtained which is quantified by the integral of norm of the squared control input torque values. The performance values of integral of the square of the norm of tracking error and control input torque are presented in Table II. Based on the performance measures, it can be said that the proposed approach outperforms the controller when centers and widths are not updated adaptively, as expected. When compared with the one layer neural network based approach, it is wise to say that at least comparable performances are obtained.

VII. CONCLUSION

In this article, task space trajectory tracking of robot manipulators having uncertainties in their dynamic models was studied. The dynamical uncertainties were considered to be modeled with a fuzzy logic network. A nonlinear proportional derivative controller was fused with an AFL component whose control representative value matrix, centers, and widths of MFs were obtained via dynamic update rules and thus avoiding the need for offline learning phases. The stability of the designed controller was proven via Lyapunov type arguments where UUB of the task space tracking error was ensured. Experiment results obtained from an in-house developed robot manipulator demonstrated satisfactory task space tracking performance.

Another subject examined in this work was the case when the Jacobian matrix includes parametric uncertainties. An extension was presented where the parametric uncertainties in the kinematic model avoided the Jacobian matrix to be used as

part of the definition of the error like vector. The definition of $r(t)$ was modified and an adaptive update rule was designed to compensate for the lack of accurate knowledge of kinematic parameters. The control design was also modified accordingly. Boundedness of the closed-loop system and practical tracking were proven via Lyapunov based tools. Experiments were also performed where despite the lack of exact knowledge of kinematic parameters satisfactory tracking performance was achieved.

Finally, the first controller was extended to be applicable to kinematically redundant robot manipulators. Specifically, the Jacobian matrix being nonsquare avoided the auxiliary error like vector to be used thus an extension with a pseudoinverse was essential. While the control design and the stability analysis were parallel to the first controller, an extended stability analysis was performed for the secondary controller.

The main contributions along with the novelties of this study are briefly summarized as follows. A novel self-adjusting AFL-based controller was designed that does not require accurate knowledge of robot dynamic model. The AFL terms were designed as self-adjusting to improve their learning capabilities by adaptively adjusting centers and widths of the MFs in a seemingly novel departure from the existing results in the literature where usually MFs with constant centers and widths are preferred. Additionally, as presented in Appendix X, a polynomial MF structure is preferred as opposed to MFs that rely on exponential functions (such as Gaussian) or piecewise linear MFs (such as triangular or trapezoidal). The piecewise linear MFs fail to satisfy continuity which is usually required for many robot control applications, and Gaussian type MFs may not change significantly due to the spread of the exponential functions [40]. The first controller required accurate knowledge of the Jacobian matrix, where as an extension a gradient based adaptive update law was designed when Jacobian matrix includes parametric uncertainties. The first controller was extended to be applicable to kinematically redundant robot manipulators as well. Some recent works on control of robot manipulators utilized fuzzy logic based methods in designing the control input torques. In [42] and [43], control of robots in joint space were aimed while imposing constraints on the system states. Fuzzy logic based tools were used in conjunction with the controllers but centers and widths of the MFs were chosen as constants. In [44], synchronization of robot manipulators in task space were studied. However, adaptive control strategy was adopted where regressor matrices are required to be found for all robots in the network. Similarly, in [17], the dynamical uncertainties are required to satisfy the linear parametrization property. When compared with [25], our proposed approach does not require variable structure type components (*i.e.*, signum of the error) in the design of the control input torque. When compared with relevant neural network based methods, the main advantage of the proposed AFL compensation is the self-adjusting nature of the membership value vector obtained by estimating its widths and centers via the design of dynamic adaptive update laws in (17) and (18). A possible future work may aim designing output feedback versions of the controllers designed in this article that is restricting the controller design via the unavailability of the joint velocity feedback. It is important to note that since joint velocity measurements are required for $r(t)$, which is used as part of both the control input and the adaptive update laws, it is not a straightforward task to make use of a velocity observer in conjunction with the proposed controllers. Another possible future research may be taking the constraints of the states into account when designing the controller [42], [43].

APPENDIX A
EXTENSION TO KINEMATICALLY REDUNDANT ROBOT MANIPULATORS

For a kinematically redundant robot manipulator, the end effector position $x(t) \in \mathbb{R}^m$ is obtained as

$$x = f(q) \tag{52}$$

where $q(t) \in \mathbb{R}^n$ is the joint position vector for $n > m$ and $f : \mathbb{R}^n \to \mathbb{R}^m$ is the forward kinematics. The time derivative of (52) yields

$$\dot{x} = J\dot{q} \tag{53}$$

where $\dot{q}(t) \in \mathbb{R}^n$ represents the joint velocity vector, $\dot{x}(t) \in \mathbb{R}^m$ is the task space velocity vector and $J(q) \in \mathbb{R}^{m\times n}$ denotes the Jacobian matrix. Since $n > m$, $J(q)$ is not square and thus a pseudoinverse, shown with $J^+(q) \in \mathbb{R}^{n\times m}$, is introduced as

$$J^+ \triangleq J^T(JJ^T)^{-1} \tag{54}$$

which satisfies

$$JJ^+ = I_m. \tag{55}$$

The pseudoinverse of the Jacobian matrix defined in (54) ensures the Moore–Penrose conditions given in [1].

Remark 3: Similar to Remark 1, the pseudoinverse of the Jacobian is considered to be available for all possible $q(t)$ thus all kinematic singularities are avoided a priori.

For kinematically redundant robot manipulators, the auxiliary error-like term in (9) is defined as follows:

$$r \triangleq J^+(\dot{x}_d + \alpha e) + (I_n - J^+J)g - \dot{q} \tag{56}$$

where $\alpha \in \mathbb{R}^{m\times m}$ is a constant, positive definite, diagonal gain matrix, and $g(q) \in \mathbb{R}^n$ is a subtask controller that will be designed subsequently. Taking the time derivative of (56) and multiplying by inertia matrix M, the open loop error system is obtained as in (11), where the auxiliary vector $f(x, \dot{x}, x_d, \dot{x}_d, \ddot{x}_d) \in \mathbb{R}^n$ is now have the following form:

$$\begin{aligned} f \triangleq\ & M\frac{d}{dt}[J^+(\dot{x}_d + \alpha e) + (I_n - J^+J)g] \\ & + C[J^+(\dot{x}_d + \alpha e) + (I_n - J^+J)g] + G + F\dot{q} \end{aligned} \tag{57}$$

which can be expressed using the fuzzy logic network in (13). Based on error system development and subsequent stability analysis, control input torque is designed as

$$\tau = J^Te + K_r r + \hat{w}^T\hat{\phi}. \tag{58}$$

Substituting the control input torque in (58) into the open loop error system given in (11), the closed-loop error system is obtained as

$$M\dot{r} = -Cr - K_r r - J^Te + \epsilon + w^T\phi - \hat{w}^T\hat{\phi}. \tag{59}$$

The stability of the closed-loop system and UUB of the task space tracking error are both ensured after making use of the stability analysis detailed in Section IV.

To make use of the extra DOFs for possible secondary control objectives (such as singularity avoidance, joint limit avoidance, upper bounding the impact forces, upper bounding the potential energy [15]) subtask controller $g(t)$ will be designed and its stability will be also investigated via Lyapunov-based arguments. It is essential to highlight that the secondary objective must be designed while still ensuring the tracking control objective, which is the main control objective.

The subtask controller is designed as

$$g = -k_s[J_s(I_n - J^+J)]^T y_s \tag{60}$$

where $k_s \in \mathbb{R}$ is a positive gain and $y_s(q) \in \mathbb{R}$ is a positive definite sub-task function designed as

$$y_s \triangleq \exp(-c\psi(q)) \tag{61}$$

in which $\psi(q) \in \mathbb{R}$ is picked specific to the sub-task and $J_s(q) \in \mathbb{R}^{1\times n}$ is defined as

$$J_s \triangleq \frac{\partial y_s}{\partial q}. \tag{62}$$

Differentiating (61) and then substituting (56) and (60) yields

$$\dot{y}_s = -k_s \left\|J_s(I_n - J^+J)\right\|^2 y_s + J_sJ^+(\dot{x}_d + \alpha e) - J_s r. \tag{63}$$

To investigate the stability of the subtask controller, a Lyapunov function, shown with $V_s(y_s) \in \mathbb{R}$, is introduced as

$$V_s \triangleq \frac{1}{2}y_s^2 \tag{64}$$

provided the boundedness of the closed-loop system and $\|J_s(I_n - J^+J)\| > 0$ is ensured, the time derivative of the Lyapunov function is obtained as

$$\dot{V}_s \leq -cy_s^2 + \kappa \tag{65}$$

for some positive constants κ and c. Substituting (64) into the right-hand side of (65) and then solving the differential inequality yields

$$|y_s(t)| \leq \sqrt{y_s^2(0)\exp(-2ct) + \frac{\kappa}{c}} \tag{66}$$

where UUB of y_s is ensured thus meeting the secondary control objective, in addition to the task space tracking control objective.

Appendix B
Properties of AFL Structure

The MFs used in (13) is detailed in this part. First, one dimensional MF is presented on the interval of $[u_j, u_{j+2}]$ such that $L_j : [u_j, u_{j+1}] \to \mathbb{R}$ and $R_j : [u_{j+1}, u_{j+2}] \to \mathbb{R}$ are, respectively, left and right functions given as

$$\phi_j(x) = \begin{cases} L_j(x) \text{ if } x \in [u_j, u_{j+1}] \\ R_j(x) \text{ if } x \in [u_{j+1}, u_{j+2}] \\ 0 \quad \text{otherwise} \end{cases}. \tag{67}$$

The given left and right functions provide the following features $L_j(u_j) = 0$, $L_j(u_{j+1}) = 1$, $\frac{dL_j}{dx}|_{x=u_j} = \frac{dL_j}{dx}|_{x=u_{j+1}} = 0$, $R_j(u_{j+1}) = 1$, $R_j(u_{j+2}) = 0$, $\frac{dR_j}{dx}|_{x=u_{j+1}} = \frac{dR_j}{dx}|_{x=u_{j+2}} = 0$. To satisfy the above conditions, a third-order polynomial of the form a, b, c, d scalar coefficients of $ax^3 + bx^2 + cx + d$ can be used both left and right for functions. An example function ϕ_j for $u_j = -1$, $u_{j+1} = 0$, $u_{j+2} = 1$ is shown in Fig. 7. The multidimensional MFs are defined based on the 1-D MF given in (67). In particular, $\mu_{l,m}(x_l) = \phi_{l,m}(p_{l,m} - \Delta_{l,m}x_l)$ where $p_{l,m}$ and $\Delta_{l,m}$ is the center and width of the MF via the lth input for the mth MF as

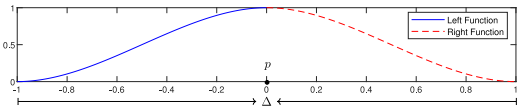


Fig. 7. 1-D MF for $\Delta = 2$, $p = 0$.

$$\phi_{l,m}(p_{l,m} - \Delta_{l,m}x_l) = \begin{cases} L_m(p_{l,m} - \Delta_{l,m}x_l) \text{ if } p_{l,m} - \Delta_{l,m}x_l \in [-1, 0] \\ R_m(p_{l,m} - \Delta_{l,m}x_l) \text{ if } p_{l,m} - \Delta_{l,m}x_l \in [0, 1] \\ 0 \quad \text{otherwise} \end{cases} \tag{68}$$

for any input given as $x = [x_1, \ldots, x_n]^T \in \mathbb{R}^n$. The output of multidimensional fuzzy network $g(x) = [g_1(x), \ldots, g_m(x)]^T \in \mathbb{R}^m$ is obtained as

$$g_k(x) = \sum_{j_n=1}^{N_n} \cdots \sum_{j_1=1}^{N_1} w_{k,(j_1,\ldots,j_n)}\mu_{j_1,\ldots,j_n}(x) \quad k = 1, \ldots, m \tag{69}$$

where N_i $i = 1, \ldots, n$ denote the number of fuzzy rules and $\mu_{j_1,\ldots,j_n}(x) = \mu_{1,j_1}(x_1)\cdots\mu_{n,j_n}(x_n)$ in which the multiplication operation is used to make inferences from fuzzy if–then rules in connection rules between each input of MFs. An estimation of the entries of $g(x)$ presented in (69) is obtained as

$$\hat{g}_k(x) = \sum_{j_n=1}^{N_n} \cdots \sum_{j_1=1}^{N_1} \hat{w}_{k,(j_1,\ldots,j_n)}\hat{\mu}_{j_1,\ldots,j_n}(x) \tag{70}$$

which can be introduced in matrix vector form as $\hat{g}(x) = \hat{w}^T\hat{\phi}(\hat{p}, \hat{\Delta}, x)$.

References

[1] Y. Nakamura, *Advanced Robotics: Redundancy and Optimization*. Boston, MA, USA: Addison-Wesley Longman Publishing Company, Inc., 1990.
[2] O. Khatib, "Dynamic control of manipulators in operational space," in *Proc. IFTOMM Congr. Theory Mach. Mechanisms*, New Delhi, India, 1983, pp. 1–10.
[3] P. Hsu, J. Hauser, and S. Sastry, "Dynamic control of redundant manipulators," *J. Robot. Syst.*, vol. 6, no. 2, pp. 133–148, 1989.
[4] F. Caccavale, C. Natale, B. Siciliano, and L. Villani, "Resolved-acceleration control of robot manipulators: A critical review with experiments," *Robotica*, vol. 16, no. 5, pp. 565–573, 1998.
[5] J. Yuan, "Closed-loop manipulator control using quaternion feedback," *IEEE J. Robot. Autom.*, vol. 4, no. 4, pp. 434–440, Aug. 1988.
[6] H. Seraji, "Configuration control of redundant manipulators: Theory and implementation," *IEEE Trans. Robot. Automat.*, vol. 5, no. 4, pp. 472–490, Aug. 1989.
[7] Z.-X. Peng and N. Adachi, "Compliant motion control of kinematically redundant manipulators," *IEEE Trans. Robot. Automat.*, vol. 9, no. 6, pp. 831–836, Dec. 1993.
[8] B. Siciliano and O. Khatib, *Handbook of Robotics*. Secaucus, NJ, USA: Springer, 2008.

[9] J. Nakanishi, R. Cory, M. Mistry, J. Peters, and S. Schaal, "Operational space control: A theoretical and empirical comparison," *Int. J. Robot. Res.*, vol. 27, no. 6, pp. 737–757, 2008.

[10] S. Lyu and C. C. Cheah, "Data-driven learning for robot control with unknown Jacobian," *Automatica*, vol. 120, pp. 109–120, 2020.

[11] C.-C. Cheah, S. Kawamura, S. Arimoto, and K. Lee, "PID control of robotic manipulator with uncertain Jacobian matrix," in *Proc. IEEE Int. Conf. Robot. Automat.*, 1999, pp. 494–499.

[12] C. C. Cheah, M. Hirano, S. Kawamura, and S. Arimoto, "Approximate Jacobian control with task-space damping for robot manipulators," *IEEE Trans. Autom. Control*, vol. 49, no. 5, pp. 752–757, May 2004.

[13] B. Xian, M. S. de Queiroz, D. M. Dawson, and I. D. Walker, "Task-space tracking control of robot manipulators via Quaternion feedback," *IEEE Trans. Robot. Automat.*, vol. 20, no. 1, pp. 160–167, Feb. 2004.

[14] E. Zergeroglu, D. M. Dawson, I. D. Walker, and P. Setlur, "Nonlinear tracking control of kinematically redundant robot manipulators," *IEEE Trans. Mechatronics*, vol. 9, no. 1, pp. 129–132, Mar. 2004.

[15] E. Tatlicioglu, M. L. McIntyre, D. M. Dawson, and L. Walker, "Adaptive non-linear tracking control of kinematically redundant robot manipulators," *Int. J. Robot. Automat.*, vol. 23, no. 2, pp. 98–105, 2008.

[16] E. Tatlicioglu, D. D. Braganza, T. C. Burg, D. M. Dawson, and I. Walker, "Adaptive control of redundant robot manipulators with sub-task objectives," *Robotica*, vol. 27, pp. 873–881, 2009.

[17] K. Cetin, E. Tatlicioglu, and E. Zergeroglu, "An extended Jacobian-based formulation for operational space control of kinematically redundant robot manipulators with multiple subtask objectives: An adaptive control approach," *J. Dyn. Syst., Meas., Control*, vol. 141, no. 5, pp. 051 011–1–11, 2019.

[18] D. Braganza, W. Dixon, D. Dawson, and B. Xian, "Tracking control for robot manipulators with kinematic and dynamic uncertainty," *Int. J. Robot. Automat.*, vol. 23, no. 2, pp. 117–126, 2008.

[19] W. E. Dixon, "Adaptive regulation of amplitude limited robot manipulators with uncertain kinematics and dynamics," *IEEE Trans. Autom. Control*, vol. 52, no. 3, pp. 488–493, Mar. 2007.

[20] M. Galicki, "Finite-time trajectory tracking control in a task space of robotic manipulators," *Automatica*, vol. 67, pp. 165–170, 2016.

[21] R. Colbaugh and K. Glass, "Robust adaptive control of redundant manipulators," *J. Intell. Robot. Syst.*, vol. 14, no. 1, pp. 68–88, 1995.

[22] U. Ozbay, H. T. Sahin, and E. Zergeroglu, "Robust tracking control of kinematically redundant robot manipulators subject to multiple self-motion criteria," *Robotica*, vol. 26, no. 6, pp. 711–728, 2008.

[23] Y. Ren, Y. Zhou, Y. Liu, Y. Gu, M. Jin, and H. Liu, "Robust adaptive multi-task tracking control of redundant manipulators with dynamic and kinematic uncertainties and unknown disturbances," *Adv. Robot.*, vol. 31, no. 9, pp. 482–495, 2017.

[24] K. Dogan, E. Tatlicioglu, E. Zergeroglu, and K. Cetin, "Learning control of robot manipulators in task space," *Asian J. Control*, vol. 20, no. 3, pp. 1003–1013, 2018.

[25] K. Cetin, E. Tatlicioglu, and E. Zergeroglu, "On operational space tracking control of robotic manipulators with uncertain dynamic and kinematic terms," *J. Dyn. Syst., Meas., Control*, vol. 141, pp. 011 001–1-7, 2019.

[26] S. Commuri and F. L. Lewis, "Adaptive-fuzzy logic control of robot manipulators," in *Proc. IEEE Int. Conf. Robot. Automat.*, Minneapolis, MN, USA, 1996, pp. 2604–2609.

[27] S. Commuri, F. L. Lewis, and K. Liu, "Approximation-based neural network and fuzzy logic control," in *Proc. IFAC World Congr.*, San Francisco, CA, USA, 1996, pp. 5274–5279.

[28] S. Commuri and F. L. Lewis, "Design and stability analysis of adaptive-fuzzy controllers for a class of nonlinear systems," in *Proc. IEEE Int. Conf. Decis. Control*, Kobe, Japan, 1996, pp. 2729–2730.

[29] E. S. Conkur and R. Buckingham, "Clarifying the definition of redundancy as used in robotics," *Robotica*, vol. 15, no. 5, pp. 583–586, 1997.

[30] D. N. Nenchev, "Redundancy resolution through local optimization: A review," *J. Robot. Syst.*, vol. 6, no. 6, pp. 769–798, 1989.

[31] B. Siciliano, "Kinematic control of redundant robot manipulators: A tutorial," *J. Intell. Robot. Syst.*, vol. 3, no. 3, pp. 201–212, 1990.

[32] D. M. Dawson, M. M. Bridges, Z. Qu, and M. Jamshidi, *Nonlinear Control of Robotic Systems for Environmental Waste and Restoration*. Englewood Cliffs, NJ, USA: Prentice Hall, 1995.

[33] F. L. Lewis, D. M. Dawson, and C. T. Abdallah, *Robot Manipulator Control: Theory and Practice*. Boca Raton, FL, USA: CRC Press, 2003.

[34] L.-X. Wang and J. M. Mendel, "Fuzzy basis functions, universal approximation, and orthogonal least-squares learning," *IEEE Trans. Neural Netw.*, vol. 3, no. 5, pp. 807–814, Sep. 1992.

[35] L.-X. Wang, "Fuzzy systems are universal approximators," in *Proc. IEEE Int. Conf. Fuzzy Syst.*, San Diego, CA, USA, 1992, pp. 1163–1170.

[36] J. L. Castro, "Fuzzy logic controllers are universal approximators," *IEEE Trans. Syst., Man, Cybern.*, vol. 25, no. 4, pp. 629–635, Apr. 1995.

[37] M. Krstic, P. V. Kokotovic, and I. Kanellakopoulos, *Nonlinear and Adaptive Control Design*. Hoboken, NJ, USA: Wiley, 1995.

[38] E. Kreyszig, *Advanced Engineering Mathematics*. Hoboken, NJ, USA: Wiley, 2009.

[39] F. L. Lewis, A. Yesildirek, and K. Liu, "Multilayer neural-net robot controller with guaranteed tracking performance," *IEEE Trans. Neural Netw.*, vol. 7, no. 2, pp. 388–399, Mar. 1996.

[40] O. Kuljaca, F. Lewis, J. Gadewadikar, and K. Horvat, "Elastic adaptive fuzzy logic controller," *J. Commun. Comput.*, vol. 7, no. 12, pp. 74–82, 2010.

[41] F. L. Lewis, S. Jagannathan, and A. Yesildirek, *Neural Network Control of Robot Manipulators and Non-Linear Systems*. Boca Raton, FL, USA: CRC Press, 2020.

[42] W. He and Y. Dong, "Adaptive fuzzy neural network control for a constrained robot using impedance learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 4, pp. 1174–1186, Apr. 2018.

[43] W. Sun, S.-F. Su, J. Xia, and V.-T. Nguyen, "Adaptive fuzzy tracking control of flexible-joint robots with full-state constraints," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 449, no. 11, pp. 2201–2209, Nov. 2019.

[44] Y. Li, Y. Yin, and D. Zhang, "Adaptive task-space synchronization control of bilateral teleoperation systems with uncertain parameters and communication delays," *IEEE Access*, vol. 6, pp. 5740–5748, 2018.

B. Melih Yilmaz received the B.Sc. and M.Sc. degrees in electrical and electronics engineering from Nigde University, Nigde, Turkey, in 2014 and 2016, respectively. He is currently working toward the doctoral degree in control systems with the Department of Electrical and Electronics Engineering, Ege University, Izmir, Turkey.

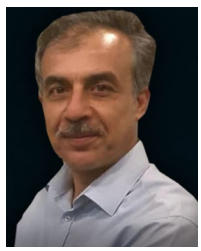
His research interests include control systems, robotic, fuzzy logic, neural networks, control of marine vessels, and learning, robust, and adaptive control of nonlinear systems.

Enver Tatlicioglu received the B.Sc. degree in electrical and electronics engineering from Dokuz Eylul University, Izmir, Turkey, in 1999, and the Ph.D. degree in electrical and computer engineering from Clemson University, Clemson, SC, USA, in 2007.

He joined the Department of Electrical and Electronics Engineering, Izmir Institute of Technology, Izmir, where he served as the Department Chair between 2017 and 2020. He is currently with the Department of Electrical and Electronics Engineering, Ege University, Izmir. His research interests include the broad areas of control and robotics.

Aydogan Savran was born in Konya, Turkey, in 1971. He received the B.S. degree in electrical and electronics engineering, from Dokuz Eylul University, Izmir, Turkey, in 1990, and the M.S. degree in electronics and communication engineering and the Ph.D. degree in aeronautical engineering from Istanbul Technical University, Istanbul, Turkey, in 1993 and 2000, respectively.

He is currently a Professor with the Department of Electrical and Electronics Engineering, Ege University, Izmir. His research interests include control systems, neural networks, fuzzy logic, and deep learning.



Musa Alci received the B.S. degree in electrical and electronics engineering and the M.Sc. degree in electrical engineering from the Technical University of Istanbul, Istanbul, Turkey, respectively. He received the Ph.D. degree in electrical and electronics engineering form Sakarya University, Sakarya, Turkey, in 1999.

He is currently a Professor with the Department of Electrical and Electronics Engineering, Ege University. His research interests include fuzzy logic, neural networks, and system identification and control.