Abstract—This paper develops a novel regressor-free robust controller for rigid robots whose dynamics can be described using the Euler-Lagrange equations of motion. The function approximation technique (FAT) is used to represent the robot’s inertia matrix, Coriolis matrix, and gravity vector as finite linear combinations of orthonormal basis functions. The proposed controller establishes a robust FAT control framework that uses a fixed control structure. The control objectives are to track reference trajectories in worst-case scenarios where the robot dynamics are too costly to develop or otherwise unavailable. Detailed stability analysis via Lyapunov functions, the passivity property, and continuous switching laws show uniform ultimate boundedness of the closed loop dynamics. Simulation results of a three-degree-of-freedom (DOF) robot when the robot parameters are perturbed from their nominal values show good robustness of the proposed controller when compared to some well-established control methods. We also demonstrate success in real-time experimental implementation of the proposed controller, which validates practicality for real-world robotic applications.

Index Terms—Robust Control, Function Approximation Technique, Passivity, Lyapunov Stability.

I. INTRODUCTION

ROBUST controllers are designed to give a desired performance for a range of system uncertainties. Early research on robust control used feedback linearization of the nonlinear robot dynamics along a predefined trajectory [1], [2]. An exact linearization approach that used the positive definite property of a robot’s mass matrix was developed in [3].

One well-known robust control method is variable structure control. The advantage of variable structure control is that the error is driven to a switching surface, after which the system is in a sliding mode in which it will be robust to large disturbances [4]. Other robust controllers using variable structure control were developed in [5], [6]. However, variable structure control methods use discontinuous switching laws that induce chattering, which might excite high-frequency unmodeled dynamics.

The regressor-based approach has also been used to develop robust and adaptive controllers for robots. In regressor-based control, the dynamic equation of a robot is linearly parameterized as a product of the regressor matrix and a parameter vector. The regressor matrix contains the nonlinear components of the robot dynamics while the parameter vector contains robot parameters such as the link masses and moments of inertia. Most regressor-based robust control utilize fixed control structures and switching laws, while most regressor-based adaptive controllers utilize update laws to estimate the unknown parameter vector [7]. A regressor-based sliding mode controller was developed based on the variable structure control scheme [8]. A regressor-based controller for robots was developed to give accurate parameter estimation using the interval excitation condition (IE), which is much weaker than the persistent excitation (PE) condition [9]. Various forms of regressor-based controllers have been developed for robot control [10]–[13]. In recent decades, researchers have combined adaptive control and robust control to take advantage of the benefits of both methods [14], [15].

Regressor-free control methods, which forgo the use of linear parameterization into the regressor matrix and parameter vector, have been developed to mitigate the drawbacks of regressor-based robot control. Many model-free robot controllers use neural networks and fuzzy systems [16]. Neuro-fuzzy controllers use the universal approximation property and the linear parameterization property. The unknown parameters of neural network controllers and fuzzy controllers can be tuned online. Lyapunov-based techniques have been used to develop neural network controllers and fuzzy controllers that have the ability to tune their unknown parameters online with adaptation laws [17]. Neural network control has also been developed for bimanual robots to ensure stability [18]. Radial-basis-functions are well-known universal function approximators that have been used by several neuro-fuzzy controllers for the control of robots [19]–[22].

Another model-free controller for robots is the function approximation technique (FAT)-based control and several adaptive FAT controllers have been developed [23]–[27]. FAT-based control does not need the calculation of the regressor matrix. The FAT controller enables the control of robots in the presence of parametric uncertainties by employing basis functions to account for time-varying uncertainties in the robot dynamics [23]. In contrast to neuro-fuzzy control, FAT-based control does not use any hidden layers, or input-output data for regression, but rather exploits the structural properties of the dynamic equation of a robot. In the FAT controller framework, the robot’s unknown inertia matrix, Coriolis ma-
controller parameters. Over a given range of uncertainties without the need to retune the dynamic equation of a robot is too costly to develop or is controller. This controller is most attractive in scenarios where freedom (DOF) robot verify the effectiveness of the proposed computer simulations and experimental tests on a three degree-of-freedom robot. The controller is proven to provide closed functions and taking advantage of the passivity property of the controller achieves robustness by employing a fixed control scheme was based on the premise that there are large uncertainties in the robot dynamic equation. When implemented on real-world robots, adaptive control exhibits problems during transient response. Most robust adaptive FAT controllers use an update law and a robustifying term to improve robustness. The modification technique is a popular method used to robustify adaptive FAT controllers. The modification approach does not use disturbance bounds. The modification prevents the estimate of the robot parameters from growing without bounds in the presence of system uncertainties. One of the drawbacks of the modification is that when the tracking errors are small, the adaptive parameters tend to converge to zero; that is, they unlearn the gain values that made the tracking errors small. Additionally, the tracking error does not converge to zero even when disturbances are removed from the system.

The main contribution of this paper is that, for the first time, we develop a robust FAT (RFAT) controller with no update laws. The controller uses the FAT to represent the robot’s inertia matrix, Coriolis matrix, and gravity vector as finite linear combinations of orthonormal basis functions. The controller achieves robustness by employing a fixed control structure and continuous switching laws. Using Lyapunov functions and taking advantage of the passivity property of robot dynamics, the controller is proven to provide closed loop dynamics that are uniformly ultimately bounded. Computer simulations and experimental tests on a three degree-of-freedom (DOF) robot verify the effectiveness of the proposed controller. This controller is most attractive in scenarios where the dynamic equation of a robot is too costly to develop or is otherwise unavailable, and guaranteed performance is required over a given range of uncertainties without the need to retune controller parameters.

This paper is organized as follows. Section II discusses the dynamic system and reviews the adaptive FAT controller. Section III develops our new robust function approximation technique (RFAT) controller. Section IV validates RFAT controller performance via computer simulations and compares it with the adaptive FAT, adaptive passivity, and robust passivity controllers. Section V validates RFAT controller performance via experimental tests and compares it with the robust passivity controller. Section VI discusses conclusions and future research.

II. Dynamic System Description and Review of Existing Techniques

This section gives a brief description of the dynamic system, along with a review of the adaptive function approximation technique (AFAT) controller. This provides a basis for the development of the RFAT controller in Section III.

A. Dynamic System

The motion of a rigid robot can be described by the dynamic equation

\[ D(q) \ddot{q} + C(q, \dot{q}) \dot{q} + g(q) = \tau \]  

where \( D(q) \in \mathbb{R}^{n \times n} \) is the inertia matrix, \( C(q, \dot{q}) \in \mathbb{R}^{n \times n} \) is the matrix of Coriolis and centripetal forces, \( g(q) \in \mathbb{R}^n \) is the gravity vector, \( \tau \in \mathbb{R}^n \) denotes the torque input, \( q \in \mathbb{R}^n \) is the vector of generalized joint displacements, and \( n \) is the number of degrees of freedom (DOFs). Note that Eqn. (1) does not consider the effects of external forces. We note the following properties of Eqn. (1), which will be used in subsequent sections.

Property 1: The inertia matrix \( D(q) \) is a positive definite matrix with eigenvalues that satisfy \( 0 < \lambda_1(q) \leq \lambda_2(q) \leq \cdots \leq \lambda_n(q) \) such that \( \lambda_1(q)I_n \leq D(q) \leq \lambda_n(q)I_n \), where \( I_n \) is the \( n \times n \) identity matrix.

Property 2: The matrix \( D(q) - 2C(q, \dot{q}) \) is skew-symmetric; that is, \( x^T [D(q) - 2C(q, \dot{q})] x = 0 \) for any vector \( x \in \mathbb{R}^n \).

Property 3: The left-hand side of Eqn. (1) can be linearly parameterized in the form

\[ Y_r(q, \dot{q}, \ddot{q}) = \tau \]  

where \( Y_r(q, \dot{q}, \ddot{q}) \in \mathbb{R}^{n \times l} \) is called the regressor matrix, \( \theta \in \mathbb{R}^l \) is a vector of parameters, and \( l \) is the number of parameters, which is not unique, but a minimal parameterization can always be found.

B. Review of Adaptive Function Approximation Technique Controller

The Stone-Weierstrass theorem [33], [35], [36] shows that orthonormal basis functions provide a universal approximator for any nonlinear dynamic system with arbitrary accuracy. If a real-valued periodic or aperiodic function satisfies the Dirichlet conditions, then it is equal to the sum of its Fourier series within a time interval [37]. Dirichlet conditions are:

1) The function is absolutely integrable over the time period.
2) The function has a finite number of extrema over the time period.
3) The function has a finite number of discontinuities over the time period.
An aperiodic function can be assumed to be periodic if it is restricted to a limited time interval. Since the inertia matrix, Coriolis matrix, and gravity vector are functions of $q$, which is continuous, we can see that they satisfy the Dirichlet conditions [38]. This allows us to use Fourier series as our orthonormal basis functions. The AFAT controller framework [39] represents the inertia matrix, Coriolis matrix, and gravity vector as

$$
D(q(t)) = W_D^T Z_D(t) + \epsilon_D(t)
$$

$$
C(q(t), \dot{q}(t)) = W_C^T Z_C(t) + \epsilon_C(t)
$$

$$
g_q(t) = W_g^T Z_g(t) + \epsilon_g(t)
$$

where $W_D \in \mathbb{R}^{n_\beta_D \times n}$, $W_C \in \mathbb{R}^{n_\beta_C \times n}$, and $W_g \in \mathbb{R}^{n_\beta_g \times n}$ are constant weight matrices. $Z_D(t) \in \mathbb{R}^{n_\beta_D \times n}$, $Z_C(t) \in \mathbb{R}^{n_\beta_C \times n}$, and $Z_g(t) \in \mathbb{R}^{n_\beta_g}$ are matrices of basis functions. $\epsilon_D(t) \in \mathbb{R}^{n \times n}$, $\epsilon_C(t) \in \mathbb{R}^{n \times n}$, and $\epsilon_g(t) \in \mathbb{R}^n$ are time-varying approximation errors. $\beta_D$, $\beta_C$, and $\beta_g$ denote the number of basis functions used for the inertia matrix, Coriolis matrix, and gravity vector respectively. The estimates are expressed as

$$
\hat{D}(t) = \hat{W}_D^T Z_D(t)
$$

$$
\hat{C}(t) = \hat{W}_C^T Z_C(t)
$$

$$
\hat{g}_q(t) = \hat{W}_g^T Z_g(t)
$$

**Remark 1:** The dependence of $D$, $C$, and $g$ on the robot joint trajectory $q$ and its derivative $\dot{q}$ will not be indicated in the notation in the sequel for the sake of convenience. The dependence of the matrix of basis functions $Z(t)$, the approximation errors $\epsilon(t)$, and the estimate of the weight matrices $\hat{W}(t)$ on $t$ will also not be indicated in the notation in the sequel for the sake of convenience.

Define

$$
v = \dot{q}_d - \Lambda \ddot{q}
$$

$$
a = \dot{v} = \ddot{q}_d - \Lambda \dddot{q}
$$

$$
r = \dot{r} - v = \dddot{q} + \Lambda \dddddot{q}
$$

where $q_d \in \mathbb{R}^n$ is the reference trajectory, $\Lambda \in \mathbb{R}^{n \times n}$ is a tunable diagonal matrix with positive diagonal entries, and $\dddot{q} = q - q_d$. The adaptive FAT control law is

$$
\tau = \hat{W}_D^T Z_D a + \hat{W}_C^T Z_C v + \hat{W}_g^T Z_g - K r
$$

where $K \in \mathbb{R}^{n \times n}$ is a tunable diagonal matrix with positive diagonal entries. The update laws are given as

$$
\dot{\hat{W}}_D = -Q_D^1 (Z_D a r^T + \sigma_D \hat{W}_D)
$$

$$
\dot{\hat{W}}_C = -Q_C^1 (Z_C v r^T + \sigma_C \hat{W}_C)
$$

$$
\dot{\hat{W}}_g = -Q_g^1 (Z_g v r^T + \sigma_g \hat{W}_g)
$$

where $Q_D \in \mathbb{R}^{n_\beta_D \times n_\beta_D}$, $Q_C \in \mathbb{R}^{n_\beta_C \times n_\beta_C}$, and $Q_g \in \mathbb{R}^{n_\beta_g \times n_\beta_g}$ are a tunable diagonal matrix with positive diagonal entries, and $\sigma_D$, $\sigma_C$, and $\sigma_g$ are positive numbers.

**Theorem 1:** Using the control law of Eqn. (8) and the update laws of Eqns. (9), (10), and (11) results in uniform ultimate boundedness of the closed loop system.

**Proof:** See [39].

### III. Main Results

This section develops the robust function approximation technique (RFAT) controller. Eqn. (1) is rewritten using Eqns. (5), (6), and (7) to give the open loop dynamics

$$
\dot{D} + Cr + Da + Cv + g = \tau
$$

We use Eqn. (3) to rewrite Eqn. (12) as

$$
\dot{D} + Cr + W_C^T Z_D a + W_C^T Z_C v
$$

$$
+ W_g^T Z_g + \epsilon_D a + \epsilon_C v + \epsilon_g = \tau
$$

We suppose that the weight matrices $W_D$, $W_C$, and $W_g$ as well as the approximation errors $\epsilon_D$, $\epsilon_C$, and $\epsilon_g$ are uncertain. This means that there exists $W_{a_D} \in \mathbb{R}^{n_\beta_D \times n}$, $W_{a_C} \in \mathbb{R}^{n_\beta_C \times n}$, and $W_{a_g} \in \mathbb{R}^{n_\beta_g}$, such that

$$
\| \hat{W}_D \| = \| W_{a_D} - W_D \| \leq \psi_D
$$

$$
\| \hat{W}_C \| = \| W_{a_C} - W_C \| \leq \psi_C
$$

$$
\| \hat{W}_g \| = \| W_{a_g} - W_g \| \leq \psi_g
$$

Let

$$
x(t) = \begin{bmatrix} q(t) \\ \dot{q}(t) \end{bmatrix}
$$

$$
x_d(t) = \begin{bmatrix} q_d(t) \\ \dot{q}_d(t) \end{bmatrix}
$$

Since $D(q(t))$, $C(q(t), \dot{q}(t))$, and $g(q(t))$ are continuous on $q(t)$ and $\dot{q}(t)$ (or $x(t)$), Eqn. (3) implies that $\epsilon_D$, $\epsilon_C$, and $\epsilon_g$ are also continuous on $x(t)$. Hence, given $\| x(t) \| < \Delta$ where $\Delta$ is a positive number, we have

$$
\sup_{t \geq 0} \| \epsilon_D(t) \| < \psi_D
$$

$$
\sup_{t \geq 0} \| \epsilon_C(t) \| < \psi_C
$$

$$
\sup_{t \geq 0} \| \epsilon_g(t) \| < \psi_g
$$

We define the control law as

$$
\tau = \hat{W}_D^T Z_D a + \hat{W}_C^T Z_C v + \hat{W}_g^T Z_g - \omega_D a - \omega_C v - K r
$$

where

$$
\hat{W}_D = W_{a_D} + \delta W_D
$$

$$
\hat{W}_C = W_{a_C} + \delta W_C
$$

$$
\hat{W}_g = W_{a_g} + \delta W_g
$$

The additional control terms $\delta W_D$, $\delta W_C$, and $\delta W_g$ will be defined later in this section. We note that the control law of Eqn (22) is defined in terms of the fixed nominal weight matrices $W_{a_D}$, $W_{a_C}$, and $W_{a_g}$. In contrast to the AFAT control strategy, these weight matrices are not updated or changed in time. This gives the advantage of avoiding drifts in the estimate of the weight matrices, which is one of the drawbacks of the AFAT controller that was addressed using $\sigma$ modification [25], [33]. We also note that according to the Stone-Weierstrass theorem, there exist different weight matrices that yield good approximation for different reference trajectories. However, the use of a nominal weight matrix can yield desirable performance if the uncertainty bounds are not violated. This implies that good performance can
be maintained if the difference between the nominal weight matrix and the true weight matrix lies within the uncertainty bound (see Eqns. (14), (15), and (16)).

Substituting Eqn. (22) into Eqn. (13),

\[ D\dot{r} + Cr + W_D^T Z_D a + W_T^T Z_C v + W_g^T Z_g + \epsilon_D a + \epsilon_C v + \epsilon_g \]
\[ = \dot{W}_D^T Z_D a + \dot{W}_T^T Z_C v + \dot{W}_g^T Z_g - \omega_D a - \omega_C v - Kr \]

(26)

Eqn. (26) is rearranged to give

\[ D\dot{r} + Cr + Kr = \begin{bmatrix} \dot{W}_D^T - W_D^T \end{bmatrix} Z_D a + \begin{bmatrix} \dot{W}_T^T - W_T^T \end{bmatrix} Z_C v \]
\[ + \begin{bmatrix} \dot{W}_g^T - W_g^T \end{bmatrix} Z_g - \omega_D a - \omega_C v - \epsilon_g - \epsilon_C v - \epsilon_g \]

(27)

Using Eqns. (23), (24), and (25), Eqn. (27) is rewritten to give the closed loop dynamics

\[ D\dot{r} + Cr + Kr = \begin{bmatrix} \dot{W}_D^T + \delta W_D^T \end{bmatrix} Z_D a + \begin{bmatrix} \dot{W}_T^T + \delta W_T^T \end{bmatrix} Z_g \]
\[ + \begin{bmatrix} \dot{W}_g^T + \delta W_g^T \end{bmatrix} Z_C v - \omega_D a - \omega_C v - \epsilon_g - \epsilon_C v - \epsilon_g \]

(28)

In hindsight, we select continuous switching laws for the additional control terms \( \delta W_D^T \), \( \delta W_T^T \), and \( \delta W_g^T \) as

\[ \delta W_D^T = \begin{cases} -\psi_D \frac{ra^T Z_D^T}{\|ra^T Z_D^T\|} & \text{if } \|ra^T Z_D^T\| > \mu_D \\ -\psi_D \frac{ra^T Z_D^T}{\mu_D} & \text{if } \|ra^T Z_D^T\| \leq \mu_D \end{cases} \]

(29)

\[ \delta W_T^C = \begin{cases} -\psi_C \frac{rv^T Z_C^T}{\|rv^T Z_C^T\|} & \text{if } \|rv^T Z_C^T\| > \mu_C \\ -\psi_C \frac{rv^T Z_C^T}{\mu_C} & \text{if } \|rv^T Z_C^T\| \leq \mu_C \end{cases} \]

(30)

\[ \delta W_g^T = \begin{cases} -\psi_g \frac{r\zeta^T}{\|r\zeta^T\|} & \text{if } \|r\zeta^T\| > \mu_g \\ -\psi_g \frac{r\zeta^T}{\mu_g} & \text{if } \|r\zeta^T\| \leq \mu_g \end{cases} \]

(31)

where \( \mu_D, \mu_C, \mu_g \in R^+ \) are dead zone parameters, which alleviate control signal chattering.

Remark 2: Note that in contrast to traditional sliding mode control and the exact regressor used in [7], we represent time-varying uncertainties as finite combinations of basis functions. Similar to sliding mode control, the amount of parameter perturbation and disturbance the controller can handle are related to the uncertainty bounds. Higher bounds might yield better robustness, but can also induce unwanted chattering. Additionally, higher dead zone values minimize control signal chattering but degrade tracking accuracy. In practice, the dead zone values can be manually set to high values and then reduced until the desired tracking performance is achieved with acceptable chattering.

We now present the following lemma that will aid in the stability analysis of the RFAT controller.

Lemma 1: If the switching laws of Eqns. (29), (30), and (31) are used, then the following holds:

\[ r^T \left[ \dot{W}_D^T + \delta W_D^T \right] Z_D a \leq \begin{cases} 0 & \text{if } \|ra^T Z_D^T\| > \mu_D \\ \frac{\psi_D \mu_D}{2} & \text{if } \|ra^T Z_D^T\| \leq \mu_D \end{cases} \]

(32)

\[ r^T \left[ \dot{W}_T^C + \delta W_T^C \right] Z_C v \leq \begin{cases} 0 & \text{if } \|rv^T Z_C^T\| > \mu_C \\ \frac{\psi_C \mu_C}{2} & \text{if } \|rv^T Z_C^T\| \leq \mu_C \end{cases} \]

(33)

\[ r^T \left[ \dot{W}_g^T + \delta W_g^T \right] Z_g \leq \begin{cases} 0 & \text{if } \|r\zeta^T\| > \mu_g \\ \frac{\psi_g \mu_g}{2} & \text{if } \|r\zeta^T\| \leq \mu_g \end{cases} \]

Proof: See Appendix A.

For the stability analysis, the following candidate Lyapunov function is selected:

\[ V = \frac{1}{2} r^T Dr + \dot{q}^T \Lambda K \dot{q} = e^T A e \]

(34)

where

\[ e = x - x_d = \begin{bmatrix} \dot{q} \\ \ddot{q} \end{bmatrix} \]

(35)

and

\[ A = \begin{bmatrix} \frac{1}{2} \Lambda A + \Lambda K & \frac{1}{2} \Lambda D \\ \frac{1}{2} \Lambda D & \frac{1}{2} D \end{bmatrix} \]

(36)

For \( \|x\| < \Delta \), we have

\[ \tilde{X}_A(A, K) > \lambda_{\max}(A) \geq \lambda_{\min}(A) > \lambda_{\Delta}(A, K) > 0 \]

where \( \tilde{X}_A(\ldots) \) and \( \lambda_{\Delta}(\ldots) \) are positive functions of \( \Lambda \) and \( K \). Taking the time derivative of Eqn. (32),

\[ \dot{V} = r^T \dot{D} r + \frac{1}{2} r^T \dot{D} r + 2q^T \Lambda K \dot{q} \]

(37)

Evaluating Eqn. (36) along the closed loop trajectory of Eqn. (28)

\[ \dot{V} = r^T \left[ \dot{W}_D^T + \delta W_D^T \right] Z_D a + r^T \left[ \dot{W}_T^C + \delta W_T^C \right] Z_C v \]
\[ + r^T \left[ \dot{W}_g^T + \delta W_g^T \right] Z_g + \frac{1}{2} r^T \left[ \dot{D} - 2C \right] r \]
\[ - r^T \omega_D a - r^T \epsilon_D a - r^T \omega_C v - r^T \epsilon_C v - r^T \epsilon_g + 2q^T \Lambda K \dot{q} - r^T K r \]

(38)

Using Property 2, Eqn. (37) reduces to

\[ \dot{V} = r^T \left[ \dot{W}_D^T + \delta W_D^T \right] Z_D a + r^T \left[ \dot{W}_T^C + \delta W_T^C \right] Z_C v \]
\[ + r^T \left[ \dot{W}_g^T + \delta W_g^T \right] Z_g - r^T \omega_D a - r^T \epsilon_D a \]
\[ - r^T \omega_C v - r^T \epsilon_C v - r^T \epsilon_g + 2q^T \Lambda K \dot{q} - r^T K r \]

(39)
Defining
\[ \Phi = \begin{bmatrix} \Lambda K & 0 \\ 0 & K \end{bmatrix} \] (40)
where \( \Phi \) is a positive definite matrix, Eqn. (39) becomes
\[
\dot{V} = r^T \left[ W_D^T + \delta W_D^T \right] Z D \alpha + r^T \left[ W_C^T + \delta W_C^T \right] Z C v \\
+ r^T \left[ W_g^T + \delta W_g^T \right] Z g - r^T \omega g - r^T \omega_D a - r^T \epsilon_D a \\
- r^T \omega_C v - r^T \epsilon_C v - e^T \Phi e
\] (41)

Using Lemma 1, Eqn. (41) reduces to
\[
\dot{V} \leq \psi_D \mu_D + \frac{\psi_C \mu_C}{4} + \frac{\psi_g \mu_g}{4} - r^T \epsilon g - r^T \omega_D a \\
- r^T \epsilon_C v - r^T \omega_C v - e^T \Phi e
\] (42)

Using the relationships
\[
\lambda_{\min}(\Phi) ||e||^2 \leq e^T \Phi e \leq \lambda_{\max}(\Phi) ||e||^2
\] (43)
\[
-r^T \omega_D a \leq -\omega_D ||r^T|| ||a||
\] (44)
\[
-r^T \omega_C v \leq -\omega_C ||r^T|| ||v||
\] (45)
\[
-r^T \epsilon_D a \leq ||r^T|| ||e_D|| ||a|| \leq \omega_D ||r^T|| ||a||
\] (46)
\[
-r^T \epsilon_C v \leq ||r^T|| ||e_C|| ||v|| \leq \omega_C ||r^T|| ||v||
\] (47)
\[
-r^T \epsilon_g = e^T \left[ -\Lambda \epsilon_g \right] \leq ||e|| \eta
\] (48)

where
\[
\eta = ||A I|| \omega_g
\] (49)

and \( \lambda_{\min}(\Phi) \) and \( \lambda_{\max}(\Phi) \) denote the the minimum and maximum eigenvalue respectively of the positive definite matrix \( \Phi \), we write Eqn. (42) as
\[
\dot{V} \leq E \leq ||e||^2 \lambda_{\min}(\Phi) + ||e|| \eta
\] (50)

where
\[
E = \frac{\psi_D \mu_D + \psi_C \mu_C + \psi_g \mu_g}{4}
\]

Using the Cauchy inequality for \( ||e|| \eta \) in Eqn. (50), we have
\[
\dot{V} \leq -\lambda_{\min}(\Phi) ||e||^2 + \frac{\lambda_{\min}(\Phi)}{2} ||e||^2 + \frac{1}{2\lambda_{\min}(\Phi)} \eta^2 + E
\]
\[
= -\lambda_{\min}(\Phi) ||e||^2 + \frac{1}{2\lambda_{\min}(\Phi)} \eta^2 + E
\] (51)

Eqn. (51) implies that \( \dot{V} < 0 \) if
\[
||e||^2 < \left[ \frac{\eta^2}{\lambda_{\min}(\Phi)} + 2E \right] \frac{1}{2} \lambda_{\min}(\Phi) : = \delta
\] (52)

Let \( S(\delta) \) denote the smallest level set of \( V \) containing \( B(\delta) \), the ball of radius \( \delta \). Let \( B(r_1) \) denote the smallest ball containing \( S(\delta) \). All solutions of the closed loop system are uniformly ultimately bounded (UUB) with respect to \( B(r_1) \). All trajectories will reach the boundary of \( S(\delta) \) and will eventually enter the ball \( B(r_1) \) since \( \dot{V} \) is negative definite outside the set \( S(\delta) \).

We have
\[
\Lambda ||e||^2 \leq \lambda_{\min}(A) ||e||^2 \leq V \leq \lambda_{\max}(A) ||e||^2 \leq \bar{\Lambda} ||e||^2.
\] (53)

From (51) and (53),
\[
\dot{V} \leq -\lambda_{\min}(\Phi) V + \frac{1}{2\lambda_{\min}(\Phi)} \eta^2 + E
\] (54)

Let
\[
\alpha = \frac{\lambda_{\min}(\Phi)}{2\Lambda}
\] (55)

According to the comparison lemma in [40], from (54) we have
\[
V \leq \exp(-\alpha t) V(0) + \frac{1}{\alpha} \left( \frac{1}{2\lambda_{\min}(\Phi)} \eta^2 + E \right)
\] (56)

From (53) and (56),
\[
||e||^2 \leq \exp(-\alpha t) V(0) + \frac{1}{\alpha} \left( \frac{1}{2\lambda_{\min}(\Phi)} \eta^2 + E \right)
\]
\[
= \exp(-\alpha t) \frac{\bar{\Lambda}}{\Delta} ||e(0)||^2 + \delta^2.
\] (57)

**Theorem 2:** If there exist \( \Delta > \Delta_0 \geq \Delta_d > 0 \) such that for \( \|x(0)\| < \Delta_0 \) and \( \|x_d\| < \Delta_d \), and \( \Lambda \) and \( K \) are chosen such that
\[
\sqrt{\frac{\bar{\Lambda}}{\Delta}} (\Delta_0 + \Delta_d)^2 + \delta^2 + \Delta_d < \Delta
\] (58)

then using the switching laws of Eqns. (29), (30), and (31), the RFAT closed loop system dynamics of Eqn. (28) is uniformly ultimately bounded.

**Proof:** First, we show that \( x \) remains in the set \( \|x\| < \Delta \) for \( t \geq 0 \). We will prove it by contradiction. Assume that \( x \) exceeds the set \( \|x\| < \Delta \). Since \( x \) is continuous with respect to \( t \), there exists \( t_1 > 0 \) such that \( \|x(t_1)\| > \Delta \) for \( 0 \leq t < t_1 \) and \( \|x(t_1)\| = \Delta \). Then, (21) becomes
\[
\sup_{t \geq 0} ||\epsilon_g(t)|| < \omega_g.
\] (59)

Similarly, since \( A \) is a function of \( D(q) \), (35) becomes
\[
\bar{\Lambda}_A(A, K) \geq \lambda_{\max}(A) \geq \lambda_{\min}(A) \geq \Delta_A(A, K) > 0
\] (60)

for \( t_1 \geq t \geq 0 \). Thus, (48) and (53) still hold for \( t_1 \geq t \geq 0 \). As a result, (57) holds for \( t_1 \geq t \geq 0 \). From (57), we have
\[
\|x\| = \|e + x_d\| \leq ||e|| + ||x_d|| < \sqrt{\frac{\bar{\Lambda}}{\Delta}} ||e(0)||^2 + \delta^2 + \Delta_d
\]
\[
< \sqrt{\frac{\bar{\Lambda}}{\Delta}} (\Delta_0 + \Delta_d)^2 + \delta^2 + \Delta_d
\]
\[
< \Delta
\] (61)

for \( t_1 \geq t \geq 0 \). This implies \( \|x(t_1)\| < \Delta \), which contradicts our assumption above. Therefore, \( \|x\| < \Delta \) for \( t \geq 0 \).

Finally, the boundedness of the trajectory error follows the inequality in (57).
Remark 3: The nominal weight matrices \( W_{\alpha D}, W_{\alpha C}, \) and \( W_{\alpha g}, \) and the uncertainty bounds \( \psi_D, \psi_C, \psi_g, \omega_D, \) and \( \omega_C \) are regarded as tuning parameters. In practice, the uncertainty bounds and optimal nominal values can not be known exactly. However, more accurate values for the tuning parameters can be found by employing some knowledge of the physical system. In cases where there is no knowledge of the physical system, they can be tuned via trial and error. The approximations can be found by employing some knowledge of the physical system. However, more accurate values for the tuning parameters bounds and optimal nominal values can not be known exactly.

IV. Simulation Results

In this section, the RFAT controller is verified via computer simulations on a rigid robot. We compare the performance of the RFAT controller against the AFAT controller, adaptive passivity (AP) controller, and the robust passivity (RP) controller.

A. Robot Model

The robot model used as a testbed for controller implementation is a six-DOF PUMA500 robot. Although it has six DOFs, only three DOFs are implemented in the simulation: \( q_1, q_2, \) and \( q_3 \) as shown in [45]. The robot link parameters that characterize the PUMA500 robot at the Cleveland State University Controls, Robotics, and Mechatronics Laboratory are \( b = 0.410 \text{ m}, \) \( d_1 = 0.666 \text{ m}, \) \( d_2 = 0.243 \text{ m}, \) \( d_3 = 0.093 \text{ m}, \) and \( d_2 = 0.432 \text{ m}. \) The dynamic equation of the robot can be obtained via the Euler-Lagrange method [46]. The actual robot dynamics derived with the Euler-Lagrange method is used to simulate the plant. In the AP and RP controller implementation, a known regressor matrix and an uncertain parameter vector is used. In the AFAT and RFAT controller implementation, finite linear combinations of orthonormal basis functions are used in place of the regressor matrix / parameter vector. The regressor matrix and parameter vector of the PUMA500 robot are available at [47].

B. Overview of Robust Passivity Control and Simulation Parameters

The robust passivity (RP) controller is a regressor-based control method [7]. It uses a known regressor matrix and an uncertain parameter vector. The uncertain parameter vector is adjusted via a continuous switching law. Linearly parameterizing Eqn. (12) gives

\[
D\dot{\theta} + C\dot{\theta} + Y(q, \dot{q}, v, a)\theta = \tau
\]

where \( Y \in \mathbb{R}^{3 \times 1} \) is the known regressor matrix and \( \theta \in \mathbb{R}^l \) is the parameter vector. Note that \( \dot{Y} \) differs from \( Y_r \) in Eqn. (2) because \( Y \) is independent of joint acceleration while \( Y_r \) is dependent on joint acceleration [29]. The independence of \( Y \) from joint acceleration holds true for all robots that can be modeled using the Euler-Lagrange equation of motion. More details can be found in [7]. The control law is given as

\[
\tau = Y\dot{\theta} - K\theta
\]

where \( \dot{\theta} = \theta_0 + \delta\theta, \theta_0 \in \mathbb{R}^l \) is the nominal parameter vector, and \( l = 10 \) for the PUMA500 robot described in Section IV-A. With the parametric uncertainty bounded by \( \rho \in \mathbb{R}^l_+ \) such that \( \|\theta - \hat{\theta}\| \leq \rho, \) the additional control term \( \delta\theta \) is defined by the continuous switching law

\[
\delta\theta = \begin{cases} -\rho\frac{Y^T\tau}{\|Y^T\tau\|} & \text{if } \|Y^T\tau\| > \Upsilon \\ -\rho\frac{Y^T\tau}{\Upsilon} & \text{if } \|Y^T\tau\| \leq \Upsilon \end{cases}
\]

where \( \Upsilon \in \mathbb{R}_+ \) is a deadzone parameter. The RP controller parameters were tuned manually to give good controller performance. The deadzone was selected as \( \Upsilon = 0.1, \) the uncertainty bound was selected as \( \rho = 10, \) and the controller gains were selected as \( K = \text{diag} [10 \ 20 \ 10] \) and \( \Lambda = \text{diag} [2 \ 10 \ 2] \) (see Eqn. (7)). The nominal parameter vector was selected as

\[
\theta_0 = [2.9625 \quad 1.5893 \quad 0.0421 \quad 0.3369 \quad -0.1171 \quad 7.6163 \quad 1.1870 \quad -0.7620 \quad 7.6547 \quad 39.6898]^T
\]

C. Overview of Adaptive Passivity Control and Simulation Parameters

The adaptive passivity (AP) controller is a regressor-based control method [7]. It also uses a known regressor matrix and an uncertain parameter vector. The uncertain parameter vector is adjusted via an update law. The control law is the same as Eqn. (63). The update law is given as

\[
\dot{\theta} = -\Gamma^{-1}Y^T\tau
\]

where \( \Gamma \in \mathbb{R}^{l \times l} \) is tunable diagonal matrix with positive diagonal entries, and \( l = 10. \) The AP controller parameters were tuned manually to give good controller performance. The AP controller gains were selected as \( K = \text{diag} [10 \ 20 \ 10] \) and \( \Lambda = \text{diag} [2 \ 10 \ 2] \). The update law gain was selected as \( \Gamma = 60 \times I_{10}. \) The initial parameter vector was chosen as \( \hat{\theta}(0) = 0 \in \mathbb{R}^{10}. \)

D. RFAT Controller Simulation Parameters

The RFAT controller was manually tuned to give good controller performance. The controller gains were selected as \( K = \text{diag} [10 \ 20 \ 10] \) and \( \Lambda = \text{diag} [2 \ 10 \ 2] \). The uncertainty bounds were selected as \( \psi_D = \psi_C = \psi_g = 5, \) and \( \omega_D = \omega_C = 0.2. \) The 20-term Fourier series is selected as the basis function such that

\[
z_{ij} = \begin{bmatrix} 1 & \cos(w_1t) & \cos(w_2t) & \cos(w_3t) & \sin(w_3t) \\ \cos(w_4t) & \sin(w_4t) & \cdots & \cos(w_{20}t) & \sin(w_{20}t) \end{bmatrix}^T \in \mathbb{R}^{20}
\]
for $i, j \in [1, 3]$ where $w_k = \frac{2k\pi}{T}$ for $k \in [2, 20]$. The value for $T$ was chosen as $T = \pi$. Based on our experience, we recommend tuning the nominal weight matrix of the gravity vector, as well as the nominal weight matrices that correspond to the diagonal of the inertia and Coriolis matrices respectively, while setting all other elements of the nominal matrices to zero. This process enhances the simplicity in selection of the nominal weight matrices. However, we note that the use of different nominal weight matrices while keeping the uncertainty bounds fixed can affect the performance of the RFAT controller. The difference in controller performance will mainly be observed in the transient response.

The nominal weight matrices and basis function matrices were arbitrarily selected. The nominal weight matrices were

\[
W_{D_{ij}} = W_{C_{ij}} = [0.01 \ 0 \ \cdots \ 0]^T \in \mathbb{R}^{20} \ \forall i = j \\
W_{D_{ij}} = W_{C_{ij}} = [0 \ \ 0 \ \cdots \ 0]^T \in \mathbb{R}^{20} \ \forall i \neq j \\
W_{og_1} = W_{og_3} = [0.01 \ 0 \ \cdots \ 0]^T \in \mathbb{R}^{20} \\
W_{og_2} = [-40 \ 0 \ \cdots \ 0]^T \in \mathbb{R}^{20}
\]

for $(i, j) \in [1, 3]$. The dead zone values for the switching laws of the RFAT controller were selected as $\mu_D = \mu_C = \mu_g = 0.5$, and $\omega_D = \omega_C = 0.2$.

### E. AFAT Controller Simulation Parameters

The AFAT controller was manually tuned to give good controller performance. The controller gains were selected as $K = \text{diag} \ [10 \ 20 \ 10]$ and $\Lambda = \text{diag} \ [2 \ 10 \ 2]$. The basis functions used were the same as the ones used in Section IV-D. The update law gains were selected as $Q_{D}^{-1} = Q_{C}^{-1} = 5 \times I_{180}$, and $Q_{g}^{-1} = 150 \times I_{60}$. The initial weight matrices were selected as $\hat{W}_{D_{ij}}(0) = \hat{W}_{C_{ij}}(0) = \hat{W}_{g_{i}}(0) = 0 \in \mathbb{R}^{20}$ for $(i, j) \in [1, 3]$.

### F. Simulation

We note that the gains $K$ and $\Lambda$ are common to the AP, RP, AFAT, and RFAT controllers. The gain values were kept the same to get a good basis for comparing the controller performances (see Sections IV-B, IV-C, IV-D, and IV-E).

**Simulation 1**

Here, we use the periodic reference trajectories $q_{1d} = \sin(2t)$, $q_{2d} = 0.25 \sin(2t)$, and $q_{3d} = 0.5 \sin(2t) - \frac{\pi}{2}$. We simulate the AP, RP, AFAT, and RFAT controllers on the nominal robot model when zero initial tracking errors are used.

Figures 1 and 2 show the controller tracking performance and control signals respectively on the nominal robot model when zero initial tracking errors were used. We see good tracking by the AP, RP, AFAT, and RFAT controllers with reasonable control signals. Although their transient responses differ, the RFAT controller had the least tracking error, with a root-mean-square error (RMSE) error value of 0.014 rad, 0.003 rad, and 0.004 rad for $q_1$, $q_2$, and $q_3$ respectively. We note that in steady-state, the tracking errors of the RP, AFAT, and RFAT controllers never really converge to zero, but stay bounded within a small region around 0 rad. This is because of the uniform ultimate boundedness of the RP, AFAT, and RFAT controllers (see [7], and Theorems 1 and 2 respectively). Over time, the steady-state tracking errors for the AP controller should converge to a zero value because of the AP controller’s global convergence property.

We show the RFAT controller approximation error for the robot’s inertia matrix, Coriolis matrix, and gravity vector in Fig. 3. We note that the approximation errors of the weight matrices are not given because the actual weight matrices are unknown. However, regression methods could be used to evaluate the weight matrices that correspond to the robot dynamics. We see that the estimates, though periodic due to the periodic nature of the reference trajectories, do not converge to their true values but remain bounded. We hypothesize that the simultaneous approximation of the inertia matrix, Coriolis matrix, and gravity vector affects the convergence of the estimates since the RFAT controller design prioritizes reference...
Simulation 2

Here, we test the robustness of the AP, RP, AFAT, and RFAT controllers to a time-varying load on the third link of the robot. We use the periodic reference trajectories \(q_{1d} = \sin(2t)\), \(q_{2d} = 0.25\sin(2t)\), and \(q_{3d} = 0.5\sin(2t) - \frac{\pi}{2}\). We use zero initial tracking errors to simulate the nominal robot model, while keeping the all controller parameters unchanged. We add an unknown time-varying load to the third link of the robot after the AP, RP, AFAT, and RFAT controllers have all reached steady-state. We vary the mass of the third link using the equation

\[
m_3 = m_{3o} + 5(1 - \cos(5t)) \quad \forall t > 5
\]

where \(m_{3o}\) is the nominal mass of the third link. Figure 5 shows the tracking performance when the time-varying load is added to the third link of the robot. We see that when the time-varying load is added to the robot’s third link after 5 s, the RFAT controller maintains good reference trajectory tracking while AP, RP, and AFAT controllers do not give good performance. We note that the effect of the time-varying load is more profound on the robot’s second link, which is evident by the larger tracking errors of \(q_2\) when compared to \(q_1\) and \(q_3\). Despite the fact that the RP controller is a robust controller, we see that the RP controller gave unsatisfactory performance, especially in \(q_2\). This is because the addition of the time-varying load violated the uncertainty bounds of the RP controller. Increasing the uncertainty bounds of the RP controller might yield better tracking but it induces unwanted control signal chattering. The AP and AFAT controllers, despite being adaptive controllers that do not use uncertainty bounds, do not give satisfactory tracking for all robot joints. This is because the update laws of the AP and AFAT controllers can not keep up with the time-varying load. This is in line with one of the major drawbacks of adaptive control, which is decreased performance when the robot’s parameters change rapidly. We note that the AP and AFAT controllers will show improved performance when a slower time-varying load is used. The good tracking of the RFAT controller in Fig. 5 shows good robustness of the RFAT controller when compared to the AP, AFAT, and RP controllers.

Simulation 3

Here, we compare the robustness performance of the RFAT controller against the RP controller by performing random parameter perturbations, while keeping the controller parameters unchanged. We do not compare them against the AP and AFAT controllers because the AP and AFAT controllers do not have fixed control structures like the RFAT and RP controllers. We evaluate the performance of the RFAT and RP controllers over 100 Monte Carlo simulations, where each simulation includes a random perturbation of the robot parameters from their nominal values in the range \([-30\%, +30\%]\). We evaluate controller robustness performance by computing the tracking RMSE and root-mean-square (RMS) control signal values of each joint.

Figure 6 shows the RFAT and RP controller tracking performance over 100 Monte Carlo simulations. We see that the
RFAT controller performs better than the RP controller by giving lower tracking RMSE for all three joints of the robot. The RFAT controller tracking RMSE (mean ± one standard deviation) are 0.0118 ± 0.0027 rad, 0.0032 ± 0.0009 rad, and 0.0028 ± 0.0004 rad for \(q_1\), \(q_2\), and \(q_3\) respectively. The RP controller tracking RMSE (mean ± one standard deviation) are 0.0253 ± 0.0121 rad, 0.0285 ± 0.0353 rad, and 0.0928 ± 0.1138 rad for \(q_1\), \(q_2\), and \(q_3\) respectively. The tracking RMSE of the RFAT controller in Fig. 6 demonstrates good robustness of the RFAT controller. Figure 7 shows the RFAT and RP controller RMS control signal values over 100 Monte Carlo simulations. The RMS control signal values for both controllers are reasonable and do not exceed 30 Nm, 100 Nm, and 10 Nm for \(q_1\), \(q_2\), and \(q_3\) respectively.

V. EXPERIMENTAL RESULTS

The robot used is the six degree-of-freedom PUMA500 robot, but only the first three degrees of freedom are used for the experimental verification. The three joints of the PUMA500 robot are controlled by brushless DC motors. The motors are coupled with incremental encoders that capture position and velocity data from the robot joints. A servo amplifier is used as a power source to deliver the voltage commanded by the controller to the motors. For a control scheme implementation, input constants are used to capture the overall amplifier gains and motor gear ratios for each joint. The input constants convert torque to voltage and are 0.0543 Nm/V, 0.0806 Nm/V, and 0.1078 Nm/V for \(q_1\), \(q_2\), and \(q_3\) respectively. Real-time instrumentation and control is handled by a dSPACE DS-1202 system and associated software with a sampling frequency of 1 kHz. The experimental setup is shown in Fig. 8. Here, we validate the RFAT controller via experimental tests on a real-world robotic system by comparing it with the RP and AFAT controllers.

A. RFAT Controller Parameters

The controller gains were manually tuned online to give as little control signal chattering as possible while maintaining good reference trajectory tracking and reasonable control signal magnitudes. The dead zone values for the switching laws of the RFAT controller were \(\mu_D = \mu_C = \mu_S = 0.5\) (see Eqns. (29), (30), and (31)). The controller gains were selected as \(K = \text{diag}[15 \ 20 \ 15]\) and \(D = \text{diag}[2 \ 10 \ 2]\).

For the RFAT controller implementation, we use the 10-term Fourier series as the matrix of basis function (See Section IV-D). We note that the number of basis function is \(\beta_D = \beta_C = \beta_S = 10\). We use the same nominal weight matrices \(W_{\alpha(i)}\) as the ones used in the simulation (See Section IV-D).

We note that although a larger number of basis functions might yield better performance, computational issues such as...
larger matrices and the need for larger memory arise during real-time implementation. This is why a trade-off between accuracy and computational efficiency is favorable during real-time implementation.

The uncertainty bounds were manually tuned online to reduce chattering and were selected as $\psi_D = 2$, $\psi_C = \psi_g = 3$, and $\omega_D = \omega_C = 0.2$.

### B. RP Controller Parameters

We use the same regressor matrix and nominal parameter vector (see Section IV-B) for the controller implementation. The controller gains were manually tuned online to give as little control signal chattering as possible while maintaining good reference trajectory tracking and reasonable control signal magnitudes. The controller gains were selected as $K$ and $\Lambda = \text{diag}[2 \ 10 \ 2]$. The deadzone was selected as $\Upsilon = 0.4$ and the uncertainty bound was selected as $\rho = 6$. We note that dead zone values below 0.4 induce unwanted chattering in our experiment.

### C. AFAT Controller Parameters

We use the same basis function and initial weight matrices (see Section IV-E) for the controller implementation. The controller gains were manually tuned online to give good performance. The controller gains were selected as $K = \Lambda = \text{diag}[2 \ 10 \ 2]$. The update law gains were selected as $Q_D^{-1} = Q_C^{-1} = 0.1 \times I_{180}$, and $Q_g^{-1} = 1 \times I_{60}$.

### D. Experiment

We note that the gains $K$ and $\Lambda$ are common to the RP and RFAT controllers. The gain values were kept the same to get a good basis for comparing the controller performances.

Here, we use the periodic reference trajectories $q_{1d} = 0.5 \sin(2t)$, $q_{2d} = 0.25 \sin(2t)$, and $q_{3d} = 0.5 \sin(2t) - \frac{\pi}{4}$ to evaluate the performance of the RFAT, RP, and AFAT controllers. We use zero initial tracking errors for this experiment.

Figure 9 shows the trajectory tracking performance for joints $q_1$, $q_2$, and $q_3$ respectively. The RFAT controller gave better tracking performance than the RP and AFAT controllers. The tracking RMSE values for the RFAT controller were 0.0243 rad, 0.0063 rad, and 0.0192 rad for $q_1$, $q_2$, and $q_3$, respectively. The tracking RMSE values for the RP controller were 0.0256 rad, 0.0186 rad, and 0.0383 rad for $q_1$, $q_2$, and $q_3$, respectively. The tracking RMSE values for the AFAT controller were 0.1557 rad, 0.0499 rad, and 0.0973 rad for $q_1$, $q_2$, and $q_3$, respectively. We note that although the AFAT controller gave the worst tracking performance, the tracking errors eventually reduce as time progresses beyond the 6 s time shown in the figure.

Figure 10 shows the control signals of the RFAT, RP, and AFAT controllers. We see that the control signals are reasonable with little chattering. We note that the RFAT control signals remained within the amplifier saturation limits of ±5 V for $q_1$, ±10 V for $q_2$, and ±5 V for $q_3$. The RP control remained within the amplifier saturation limits of ±5 V for $q_1$, ±10 V for $q_2$, but exceeded the amplifier saturation limit of ±5 V for $q_3$ during the transient response. However, this did not cause any instabilities in the robotic system. We note that increasing the uncertainty bound $\Upsilon$ of the RP controller induces high frequency dynamics of the robot, leading to unwanted control signal chattering. We also note that the oscillatory nature of the AFAT control signal during the transient response, if large enough, can destabilize the system. This was why higher update gain values could not be used to improve the transient performance of the AFAT controller.
Here, we evaluate the performance of the RFAT and RP controllers by using a reference trajectory that has a time-varying phase and a regulation phase. The reference trajectories were selected as

\[
q_1 = \begin{cases} 
0.5 \sin(2t) & \text{if } 0 \leq t < 2 \\
0.5 \sin(4) & \text{if } t \geq 2
\end{cases}
\]

\[
q_2 = \begin{cases} 
0.25 \sin(2t) & \text{if } 0 \leq t < 2 \\
0.25 \sin(4) & \text{if } t \geq 2
\end{cases}
\]

\[
q_3 = \begin{cases} 
0.5 \sin(2t) - \frac{\pi}{2} & \text{if } 0 \leq t < 2 \\
0.5 \sin(4) - \frac{\pi}{2} & \text{if } t \geq 2
\end{cases}
\]

Figure 11 shows the reference trajectory tracking performance for joints \(q_1\), \(q_2\), and \(q_3\) respectively when the RFAT and RP controllers were implemented. We see that the RFAT controller also gave better tracking performance when compared to the RP controller. The tracking RMSE for the RFAT controller were 0.0143 rad, 0.0043 rad, and 0.0249 rad for \(q_1\), \(q_2\), and \(q_3\) respectively. The tracking RMSE for the RP controller were 0.0153 rad, 0.0262 rad, and 0.0669 rad for \(q_1\), \(q_2\), and \(q_3\) respectively. We see that the RFAT and RP controllers gave reasonable control signals in Fig. 12.

Experiment 3

Here, we use the same reference trajectories as Experiment 1. We evaluate the RFAT and RP controller performance by using nonzero initial tracking errors. We use the initial conditions \(-0.2\), and \(-1.2\) for \(q_1\), \(q_2\), and \(q_3\) respectively. Although that both controllers give good tracking despite the use of poor initial conditions for all joints. The RFAT controller gave the least tracking error of 0.0511, 0.0190, and 0.0615 for \(q_1\), \(q_2\), and \(q_3\) respectively. The good reference trajectory tracking performance of the RFAT controller, which does not need the computation of a regressor matrix and parameter vector, shows the practical applicability of the RFAT controller to robots, especially in scenarios where the dynamic equation of a robot is unavailable or is too costly to develop. A summary of the experimental results is shown in Table I.

VI. DISCUSSION AND CONCLUSION

In this paper, we developed a novel controller called the robust function approximation technique (RFAT) controller. The RFAT controller, which uses a fixed control law, was designed to give desirable performance over a given range or uncertainties if the uncertainty bounds are not violated. The uniform ultimate boundedness of the RFAT controller was shown via detailed stability analysis. This is the first time a robust controller that eliminates the need for update laws has been developed using the FAT framework.

The performance of the RFAT was verified on a three-DOF PUMA500 robot via computer simulations. In the simulation, the RFAT controller was evaluated and compared with the...
TABLE I

<table>
<thead>
<tr>
<th>Experiment</th>
<th>q1 RMSE (rad)</th>
<th>q2 RMSE (rad)</th>
<th>q3 RMSE (rad)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Experiment 1</td>
<td>RP controller</td>
<td>0.0256</td>
<td>0.0186</td>
</tr>
<tr>
<td></td>
<td>AFAT controller</td>
<td>0.1557</td>
<td>0.0499</td>
</tr>
<tr>
<td></td>
<td>RFAT controller</td>
<td>0.0243</td>
<td>0.0063</td>
</tr>
<tr>
<td>Experiment 2</td>
<td>RP controller</td>
<td>0.0153</td>
<td>0.0262</td>
</tr>
<tr>
<td></td>
<td>RFAT controller</td>
<td>0.0143</td>
<td>0.0043</td>
</tr>
<tr>
<td>Experiment</td>
<td>RP controller</td>
<td>0.0570</td>
<td>0.0287</td>
</tr>
<tr>
<td></td>
<td>RFAT controller</td>
<td>0.0511</td>
<td>0.0190</td>
</tr>
</tbody>
</table>

AFAT, RP, and AP controllers. We showed that the RFAT controller gave superior performance over the AFAT, RP, and AP controllers when a fast-varying payload was added to the robot. The RFAT controller also showed superior performance over the RP controller when the robot parameters were randomly perturbed from their nominal values.

The performance of the RFAT was also verified on a three-DOF PUMA500 robot via experimental tests in the Control, Robotics and Mechatronics Lab at Cleveland State University. The experimental results showed that the RFAT controller gave better tracking when compared to the RP controller. The experimental tests also showed the ease of tuning the RFAT controller for real-world robotic applications. We note that the implementation of the RFAT controller (in simulation and in real-time) becomes easier when the nominal weight matrix of the gravity vector, as well as the nominal weight matrices that correspond to the diagonal of the inertia and Coriolis matrices respectively are tuned, while setting all other nominal matrices to zero (see Section IV-D).

The RFAT controller is proposed as an alternative to the AFAT controller. The use of switching laws make it easier to improve the transient performance of the RFAT controller. On the other hand, improving the transient response of the AFAT controller might be harder to achieve due to its use of update laws. Higher update law gains might improve the transient response but can also destabilize the system because they induce unwanted oscillations. When compared to the AFAT controller, the RFAT controller will be the better choice for robot control when the uncertainties are not too large and bounded, and where guaranteed performance in the presence of unmodeled dynamics and external disturbances are required. Future work could include developing a systematic approach of optimally selecting the nominal weight matrices and the number of basis functions to improve robustness of the controller.

In this paper, we utilized three switching laws to account for uncertainties in the inertia matrix, Coriolis matrix, and gravity vector respectively. The use of three switching laws can be advantageous in scenarios where the uncertainties or disturbances have a more profound effect on a certain part of the robot dynamics, say the Coriolis matrix for instance. This allows for a conservative approach to not overcompensate for the effects of disturbances to attain good performance. Further research could be done on improving the convergence of the RFAT controller’s estimates, despite the lack of information about the dynamic equation of the robot. However, several
adaptive FAT controllers in the literature achieve simplicity and reduce computational cost by lumping the unknown dynamics and approximating them using a single update law [26], [30], [48]. Our future work will involve developing a more compact form of the RFAT controller by reducing the number of switching laws to improve computational efficiency, while maintaining good robustness. The results in this paper are the first steps towards developing purely robust FAT-based controllers. Finally, we note that the source code used to generate the RFAT controller simulation results in this paper are available at [47].

**Appendix A**

**Proof of Lemma 1**

The proof for Lemma 1 is divided into three sections. First, we evaluate $r^T [\hat{W}_D^T + \delta W_D^T] Z_D a$ using the switching law of Eqn. (29). Second, we evaluate $r^T [\hat{W}_C^T + \delta W_C^T] Z_C v$ using the switching law of Eqn. (30). Third, we evaluate $r^T [\bar{W}_g^T + \delta W_g^T] Z_g$ using the switching law of Eqn. (31).

First, we evaluate $r^T [\hat{W}_D^T + \delta W_D^T] Z_D a$ by using Eqn. (29). If $||r a^T Z_D^T|| > \mu_D$, we have

$$r^T [\hat{W}_D^T + \delta W_D^T] Z_D a = r^T W_D^T Z_D a - \psi_D \frac{r^T r a^T Z_D^T Z_D a}{||r a^T Z_D^T||}$$

(66)

Since the expression $r^T r a^T Z_D^T Z_D a$ is scalar, we can define the relations

$$r^T r a^T Z_D^T Z_D a = ||r||^2 ||a^T Z_D^T||^2$$

(67)

$$||r a^T Z_D^T|| \leq ||r|| ||a^T Z_D^T||$$

(68)

Rewriting Eqn. (66) using Eqns. (67) and (68) gives

$$r^T [\hat{W}_D^T + \delta W_D^T] Z_D a \leq r^T W_D^T Z_D a - \psi_D ||r|| ||a^T Z_D^T||$$

(69)

Using the fact that

$$r^T W_D^T Z_D a \leq ||r|| ||W_D|| ||a^T Z_D^T|| \leq \psi_D ||r|| ||a^T Z_D^T||$$

Eqn. (69) reduces to

$$r^T [\hat{W}_D^T + \delta W_D^T] Z_D a \leq 0 \quad \text{if} \quad ||r a^T Z_D^T|| > \mu_D$$

(70)

Alternatively, using Eqn. (29), if $||r a^T Z_D^T|| \leq \mu_D$, we have

$$r^T [\hat{W}_D^T + \delta W_D^T] Z_D a = r^T [\hat{W}_D^T - \frac{\psi_D}{\mu_D} r a^T Z_D^T] Z_D a$$

(71)

Using the relationships in Eqns. (67) and (68), Eqn. (71) becomes

$$r^T [\hat{W}_D^T + \delta W_D^T] Z_D a = r^T W_D^T Z_D a - \frac{\psi_D}{\mu_D} ||r||^2 ||a^T Z_D^T||^2$$

$$\leq \frac{\psi_D}{\mu_D} ||r||^2 ||a^T Z_D^T||^2$$

(72)

Defining $\Omega = ||r|| ||a^T Z_D^T||$, the R.H.S of Eqn. (72) has a critical point at $\Omega = \frac{\psi_D}{2 \mu_D}$ and a local maximum such that

$$r^T [\hat{W}_D^T + \delta W_D^T] Z_D a \leq \frac{\psi_D \mu_D}{4} \quad \text{if} \quad ||r a^T Z_D^T|| \leq \mu_D$$

(73)

Therefore, from Eqns. (70) and (73),

$$r^T [\hat{W}_D^T + \delta W_D^T] Z_D a \leq \begin{cases} 0 & \text{if} \ ||r a^T Z_D^T|| > \mu_D \\ \frac{\psi_D \mu_D}{4} & \text{if} \ ||r a^T Z_D^T|| \leq \mu_D \end{cases}$$

Finally, we note that the proof of $r^T [\hat{W}_C^T + \delta W_C^T] Z_C v$ and $r^T [\bar{W}_g^T + \delta W_g^T] Z_g$ follows from the proof of $r^T [\hat{W}_D^T + \delta W_D^T] Z_D a$.

**References**


Donald Ebeigbe received the B.Eng. degree in Electrical and Electronic Engineering from Federal University of Technology Akure, Nigeria, in 2011. He is currently a PhD candidate in the Department of Electrical Engineering and Computer Science, Cleveland State University. His research interests include control theory, robotics, and optimization.

Thang Nguyen received the B.Eng. and M.S. degrees from Hanoi University of Science and Technology in 2002 and 2004 respectively, and the Ph.D. degree from Rutgers University in 2010, all in Electrical Engineering. He is a Postdoctoral Scholar in the Department of Mechanical Engineering, Northern Arizona University. He held teaching and research positions at various universities in the USA, Australia, the UK, and Vietnam. His research interests include robotics, optimization, cyberphysical systems, and control theory.

Hanz Richter is a Professor of Mechanical Engineering at Cleveland State University (CSU). He earned the Ph.D. and an M.Sc. degrees from Oklahoma State University, and a B.Sc. from the Pontificia Universidad Catolica del Peru, Lima, Peru, all in mechanical engineering. He conducts research in the broad areas of control, robotics and mechatronics with applications to biomedical robotics, aerospace propulsion, high-precision motion control and smart sensors. His research funding has been supported by the US National Science Foundation, NASA, the State of Ohio, the Parker-Hannifin Corporation and the Cleveland Clinic Foundation.

Dan Simon received his BS degree from Arizona State University in 1983, his MS degree from the University of Washington in 1987, and his Ph.D. degree from Syracuse University in 1991, all in electrical engineering. He has had 14 years of industrial experience in several engineering fields, including aerospace, automotive, biomedical, process control, and software development. He joined Cleveland State University in 1999, where he is currently a professor and the Associate Vice President for Research. His teaching and research interests include evolutionary algorithms, computer intelligence, and control theory. He has over 200 peer-reviewed publications and is the author of the books Optimal State Estimation (John Wiley & Sons, 2006), Evolutionary Optimization for Engineering (John Wiley & Sons, 2012), and Evolving Systems with Biogeography-based Optimization (John Wiley & Sons, 2017). His publications, teaching materials, and research-related software can be downloaded from his website at https://academic.csuohio.edu/simonid/.