COMPUTATIONAL RESEARCH PROGRESS IN APPLIED SCIENCE & ENGINEERING (CRPASE)

CRPASE: TRANSACTIONS OF
ELECTRICAL, ELECTRONIC AND COMPUTER ENGINEERING

Journal homepage: http://www.crpase.com

CRPASE: Transactions of Electrical, Electronic and Computer Engineering 7 (1) Article ID: 2313, 1–8, March 2021 ISSN 2423-4591

Research Article

$\mathscr{L}_1$ Adaptive Controller Design for SRS Robots in the Presence of Data Loss and Uncertainty

Hossein Ahmadian[1*], Vahid Kamrani[2], Afsaneh Davari[3], Sajjad Shahghoian Ghahfarokhy[2]

[1] *Department of Electrical Engineering, Amirkabir University of Technology (Tehran Polytechnic), Tehran, Iran*
[2] *Department of Electrical Engineering, University of Science and Technology, Tehran, Iran*
[3] *Department of Electrical Engineering, Vali-e-Asr University of Rafsanjan, Iran*

Keywords

$\mathscr{L}_1$ Adaptive Control, Lyapunov Based Control, Model Reference Adaptive Control (MRAC), Projection Performance, Shoulder Rehabilitation System (SRS).

Abstract

In this paper, the $\mathscr{L}_1$ Adaptive Control, Model Reference Adaptive Control (MRAC) and Lyapunov-based control have been used to manage the position of the arm of the Shoulder Rehabilitation Robot (SRS) with three degrees of freedom. In this way, dynamic and kinematic equations of the robot have been used, which are generally nonlinear, and the state-space model of order 6 has been assigned. In this paper, $\mathscr{L}_1$ adaptive control law includes several basic elements such as state feedback for placement of robot poles to the position of desired poles and the role of system stability, feedback compensator to eliminate steady-state error, the parameter matching for system compensation in the result of state estimation error and parameter estimation error and first-order filters with steady-state gain 1 to increase system robustness, system stability margin and as a result, the bandwidth of the system is increasing. The parameter matching algorithm playing the role of estimating the nonlinear vector of the rehabilitation robot system uses the projection performance, which is a powerful tool to estimate the system parameter with a high nonlinear degree. The obtained results show the appropriate performance, stability and robustness of the $\mathscr{L}_1$ adaptive control method compared to the MRAC and Lyapunov-based method and even other adaptive methods.

1. Introduction

Over the past few years, robotics science has grown dramatically with the use of rehabilitation. Rehabilitation means restoring the health of an injured person and making their life normal by using specific exercises and treatments. Rehabilitation is a large part of the health and therapeutic services and also helps people who had a stroke, orthopedic surgery, burn, spinal cord injuries, balance problems, etc. as much as possible to overcome their problems and gain healthy performances. A shoulder rehabilitation robot with three degrees of freedom is used for the treatment and rehabilitation of patients with neuromuscular disorders [1-3].

Many of the dynamical systems that need to be controlled have unknown parameters that are either constant or slowly changing. Adaptive control is a powerful way to control such systems. The main idea of adaptive control is that the unknown parameters of the system or its controllers are estimated based on the measured signals in a timely manner and used as the input of control calculations. Due to the fact that the adaptive control systems are designed both of linear systems and nonlinear systems (of which we mainly focus on nonlinear systems), their analysis and design can be done by Lyapunov's theory [4], [5]. The purpose of the adaptive control is to adapt and update the controller at the same time as the parameters change. In fact, adaptive control combines the control rule which is usually designed under the conditions of the system's known parameters, with the parameters derived from the online estimator, called adaptive law [6-8].

* Corresponding Author: Hossein Ahmadian
E-mail address: h.ahmadianpr@aut.ac.ir

Received: 8 December 2020; Revised: 21 February 2021; Accepted: 28 March 2021

Please cite this article as: H. Ahmadian, V. Kamrani, A. Davari, S. Shahghoian Ghahfarokhy, $\mathscr{L}_1$ Adaptive Controller Design for SRS Robots in the Presence of Data Loss and Uncertainty, Computational Research Progress in Applied Science & Engineering, CRPASE: Transactions of Electrical, Electronic and Computer Engineering 7 (2021) 1–8, Article ID: 2313.

$\mathscr{L}_1$ adaptive control is one of the control systems for compensating linear and nonlinear systems under uncertain conditions. This approach will be done based on the $\mathscr{L}_1$ norm theory, in addition to conventional adaptive control features such as MRAC which has a higher robustness and stability margin characteristics. In fact, the $\mathscr{L}_1$ adaptive control is a changed MRAC scheme, the architecture principles of which is basically based on the internal model. The $\mathscr{L}_1$ adaptive controller actually contains the filter and the largest values of the unknown parameters. Performance boundaries are the key to ensuring the performance of adaptive control. Due to the internal model architecture, the controller reaches an appropriate delay margin even in a high adaptation. In this controller, the design of the filter is important, even linear theory also can be used. The $\mathscr{L}_1$ adaptive control has been tested in a variety of applications, including flight control for missiles, aircraft and spacecraft. The key features of The $\mathscr{L}_1$ adaptive control architecture are used to ensure resistance in the face of rapid adaptations [9]-[13].

In this way, in [14] mentioned that the main issue is the designing and developing of the control system for the development of an unmanned submarine device. These devices have nonlinear dynamics and connections and tend to be presented in a variety of time characteristics. The controller must be tuned and analyzed before being implemented in the natural environment [15]. In this study, the first goal is to show the adaptive network capability of the fuzzy interface system namely ANFIS which is used to model unmanned submarine vehicles (UUVs). The second goal is to design a fuzzy controller using the ANFIS model. The input and output data from UUV have been used to model ANFIS. The weakness of this work is having a time delay in tracking the desired path. In [16] another paper studied the dynamic and motion state of a submarine vehicle under the remote control (ROV) with six degrees of freedom. They changed the efficiency of the method of Sliding mode on ROVs by using a robust adaptive fuzzy control algorithm. Fuzzy algorithm had been utilized, to estimate output disturbances for the dynamic model ROV online and robust control law and adaptive control law were used to compensate the estimated errors, however, tracking has some errors. In [17], investigated the strategy of robust adaptive control with the speed limit on ROVs. In this paper, firstly, the robust control has been considered and then, the robust adaptive control strategy has been improved with the estimation of online parameters. The weakness of this method is magnetizing of the error integral.

In [18], using $\mathscr{L}_1$ adaptive control on a submarine robot in their research. In their paper, they have achieved good results about controlling the depth and circulation around the axis y by using the above-mentioned control method, which has high robustness and stability. The weakness of this article is tracking with distortion, in fact, output signal tracks desired path with distortion and fluctuation.

In this paper, it is attempted to design and simulate a controller in order to control a shoulder rehabilitation robot with three degrees of freedom, which is called as "Shoulder Rehabilitation System (SRS)" robot, according to the features and tasks which it was asked to do. The control methods which are used include: $\mathscr{L}_1$ adaptive control method, MRAC and Lyapunov-based control. The purpose of these controllers is to enable the robot to respond appropriately against uncertainties in the system as well as modeling errors. By using the adaptive control method, the control rules can be used in a way to adapt the system to be stable and the device will be able to put in the right direction and the desired routes according to the needs. In this paper, regardless of extracted dynamic and kinematic equations, the control system equations are defined and also due to the ability of this control system, the Lyapunov-based controllers, the MRAC and the $\mathscr{L}_1$ adaptive control are defined, simulated and compared. The final results present the desired and very appropriate performance of the $\mathscr{L}_1$ adaptive control method compared to the MRAC method and Lyapunov-based control.

This paper is organized as follows: In Section 2, at first, the characteristics of the S.R.S robot are described, and then the dynamic and kinematic equations are obtained. Section 3 addresses the design of the controller and describes the design and formulation of the Lyapunov-based controls, MRAC, and the $\mathscr{L}_1$ adaptive control and also two criteria for calculating the error integral are presented. In Section 4, the simulation and analysis of the results are discussed. Finally, Section 5 is devoted to the conclusions.

2. Dynamics and Kinematics Equations of the SRS Robot

2.1. Robot Specifications

The structure of the robot is shown in Figure 1 and Table 1 presents the whole mechanical properties of the robot at a glance [19]. These properties include body segment length to meter (L), body segment weight in kilogram (W), moments of inertia with respect to the center of mass and expressed in the coordinate system on the center of mass and the center of mass expressed in the base coordinate system in kilograms of square meter (IM) and center of mass In meters relative to the reference coordinate system (CM) for each of the three parts: 1. Shoulder joint (SJ) (from point 1 in figure 1 to the shoulder joint position), 2. Arm (AZ) (from shoulder joint position to point 2 in Figure 1) and 3. Arm keeper with human member (AHZ) is calculated.

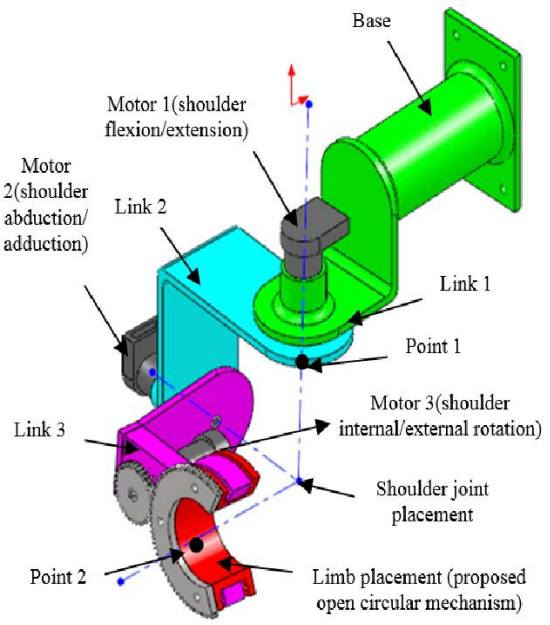

Figure 1. Degrees of freedom and different parts of the S.R.S robot [19]

Table 1. Features of the entire robot at a glance

Body		SJ	AZ	AHZ
L(*m*)		0.21	0.12	0.035
W(*kg*)		3.8	3.6	4.5
IM ($kg.m^2$)	I_{xx}	0.0934	0.0728	0.0175
	I_{yy}	0.0511	0.0148	0.0621
	I_{zz}	0.0559	0.0164	0.0719
CM(*m*)	*x*	-0.193	-0.111	-0.0067
	y	0.0096	0.00471	-0.044
	z	-0.174	-0.337	-0.447

2.2. Direct Kinematics of the Robot

The kinematic model of the robot was performed according to the dynamical model of the upper human member. Figure 2 shows the assignment of the coordinate axes in accordance with Denavit-Hartenberg (DH) law. The DH parameters for this robot were obtained in accordance with Table 2. a_i, d_i, α_i and θ_i are link length, link offset, link twist and joint angle respectively. z_0, z_1 and z_2 the rotation axes are 3 degrees of freedom of the shoulder. z_0 for shoulder bending / opening, z_1 for shoulder getting away / approaching and z_2 is for rotation of the internal / rotation of the external of the shoulder; therefore, according to Table 2, the transformation matrices are as follows:

$$\mathbf{T}_2^1 = \begin{pmatrix} c_2 & 0 & s_2 & 0 \\ s_2 & 0 & -c_2 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad \mathbf{T}_1^0 = \begin{pmatrix} c_1 & 0 & -s_1 & 0 \\ s_1 & 0 & c_1 & 0 \\ 0 & -1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{1}$$

$$\mathbf{T}_3^2 = \begin{pmatrix} c_3 & -s_3 & 0 & 0 \\ s_3 & c_3 & 0 & 0 \\ 0 & 0 & 1 & d_2 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Thus, the homogeneous transformation matrix, which correlates the final operator coordinates with the base coordinates, is obtained as follows:

$$\mathbf{T}_3^0 = \mathbf{T}_1^0\mathbf{T}_2^1\mathbf{T}_3^2 = \begin{pmatrix} c_1c_2c_3 - s_1s_3 & -c_1c_2s_3 - s_1c_3 & c_1s_2 & d_2c_1s_2 \\ s_1c_2c_3 + c_1s_3 & -s_1c_2s_3 + c_1c_3 & s_1s_2 & d_2s_1s_2 \\ -s_2c_3 & s_2s_3 & c_2 & d_1 + d_2c_2 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{2}$$

In which, according to Table 1, $d_2 = 0.12(m), d_1 = 0.21(m)$, s_i and c_i are $cos(\theta_i)$ and $sin(\theta_i)$ respectively.

Table 2. DH parameters of the robot

Joint i	a_i	d_i	α_i	θ_i
1	0	d_1	$-\frac{\pi}{2}$	θ_1
2	0	0	$\frac{\pi}{2}$	θ_2
3	0	d_2	0	θ_3

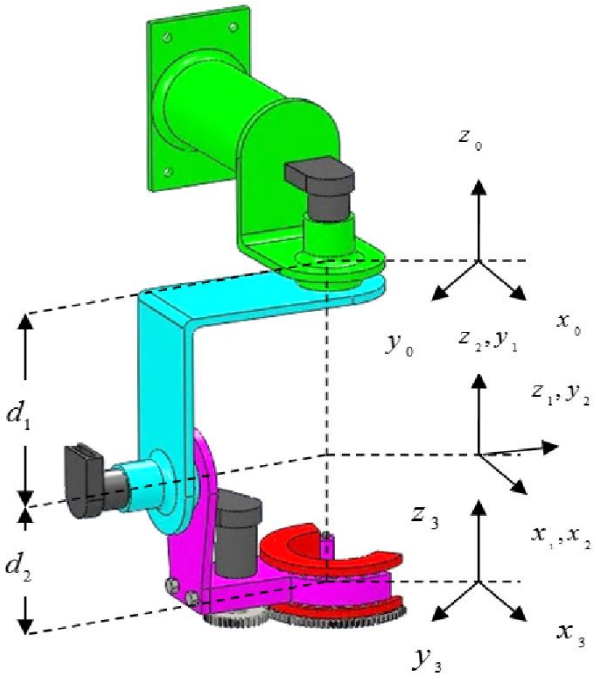

Figure 2. Link frame assignments in DH convention [19]

2.3. Robot Dynamics

The dynamics of the robot examines the robot's movement in terms of the forces and moments that make it move. In this paper, robot dynamics is derived using the generalized D'Alembert method. This method gives more efficient equations compared to Lagrange and Newton Euler methods. Besides, the computational burden is remarkably reduced which is ideal for control purposes. Generally, robot dynamics is expressed as follows [19]:

$$\boldsymbol{\tau} = \mathbf{D}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{H}(\mathbf{q},\dot{\mathbf{q}}) + \mathbf{G}(\mathbf{q}) \tag{3}$$

where $\boldsymbol{\tau}$ is a 3*1 vector expressing joint torques. $\mathbf{D}(\mathbf{q})$ is a 3*3 inertia matrix, $\mathbf{H}(\mathbf{q},\dot{\mathbf{q}})$ is a 3*1 vector of centrifugal and Coriolis terms and $\mathbf{G}(\mathbf{q})$ is a 3*1 vector of gravity term. $\mathbf{q}$, $\dot{\mathbf{q}}$ and $\ddot{\mathbf{q}}$ are 3*1 vector expressing joint position, velocity and acceleration respectively. For this robot, the dynamic equations are obtained as follows:

$$\begin{pmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{pmatrix} = \begin{pmatrix} D_{11} & D_{12} & D_{13} \\ D_{12} & D_{22} & D_{23} \\ D_{13} & D_{23} & D_{33} \end{pmatrix} \begin{pmatrix} \ddot{q}_1 \\ \ddot{q}_2 \\ \ddot{q}_3 \end{pmatrix} + \begin{pmatrix} H_1 \\ H_2 \\ H_3 \end{pmatrix} + \begin{pmatrix} G_1 \\ G_2 \\ G_3 \end{pmatrix} \tag{4}$$

where $\mathbf{D}(\mathbf{q})$, $\mathbf{H}(\mathbf{q},\dot{\mathbf{q}})$ and $\mathbf{G}(\mathbf{q})$ are provided in appendix.

3. Controller Design

Control algorithms applied on rehabilitation robots are designed considering two major aims: (1) passive rehabilitation in which the patient remains passive and the robot moves the patient's hand through a predefined trajectory and (2) active rehabilitation in which the patient initiates the movement and is partially assisted or resisted by

the robotic device [16]. In this paper the designed controllers to be used in the first case (passive rehabilitation).

3.1. Lyapunov-based Controller with Integral Action

In this section, the design and implementation of the Lyapunov-based controller with integral action are discussed. The designed controller should effectively track the desired trajectory and reject disturbance and other system uncertainties. System dynamic model can be stated as follows:

$$\mathbf{D}(\mathbf{q})\ddot{\mathbf{q}}+\mathbf{H}(\mathbf{q},\dot{\mathbf{q}})+\mathbf{G}(\mathbf{q})=\boldsymbol{\tau}+\mathbf{d} \tag{5}$$

If $\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}}=\mathbf{H}(\mathbf{q},\dot{\mathbf{q}})$ one can write the control input to the system as follows:

$$\boldsymbol{\tau}=\mathbf{D}(\mathbf{q})\ddot{\boldsymbol{\xi}}+\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\boldsymbol{\xi}}+\mathbf{G}(\mathbf{q})-\mathbf{K}_{\mathbf{D}}\boldsymbol{\sigma}-\hat{\mathbf{d}} \tag{6}$$

where, $\hat{\mathbf{d}}$ is an estimate of $\mathbf{d}$. In summary, the Lyapunov based controller with integral action can be given as follows:

$$\begin{aligned}
&\boldsymbol{\tau}=\mathbf{D}(\mathbf{q})\ddot{\boldsymbol{\xi}}+\mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\boldsymbol{\xi}}+\mathbf{G}(\mathbf{q})-\mathbf{K}_{\mathbf{D}}\boldsymbol{\sigma}-\hat{\mathbf{d}}\\
&\dot{\hat{\mathbf{d}}}=\mathbf{K}_{\mathbf{I}}\boldsymbol{\sigma},\hat{\mathbf{d}}(\mathbf{0})=\mathbf{0}\\
&\dot{\boldsymbol{\xi}}=\dot{\mathbf{q}}_{\mathbf{d}}-\boldsymbol{\Lambda}\mathbf{e}\\
&\boldsymbol{\sigma}=\dot{\mathbf{q}}-\dot{\boldsymbol{\xi}}=\dot{\mathbf{e}}+\boldsymbol{\Lambda}\mathbf{e}\\
&\mathbf{e}=\mathbf{q}-\mathbf{q}_{\mathbf{d}}
\end{aligned} \tag{7}$$

where, $\mathbf{K}_{\mathbf{D}}=k_d\mathbf{I}_{3\times3}$, $\mathbf{K}_{\mathbf{I}}=k_i\mathbf{I}_{3\times3}$ and $\boldsymbol{\Lambda}=\lambda\mathbf{I}_{3\times3}$ are three positive definite matrices.

3.2. Model Reference Adaptive Control (MRAC)

Assume that the dynamics of the linear time-invariant system (due to the matrices component being independent of time) is presented as follows [9]:

$$\begin{aligned}
&\dot{\mathbf{x}}(t)=\mathbf{A}_m\mathbf{x}(t)+\mathbf{B}\left(\mathbf{u}(t)+\boldsymbol{\theta}^T\mathbf{x}(t)\right),\ \mathbf{x}(0)=\mathbf{x}_0\\
&\mathbf{y}(t)=\mathbf{C}^T\mathbf{x}(t)
\end{aligned} \tag{8}$$

where $\mathbf{x}(t)\in\mathbb{R}^n$ is the state of the system (measured), $\mathbf{A}_m\in\mathbb{R}^{n\times n}$ is a known Hurwitz matrix that defines the desired dynamics for the closed-loop system, $\mathbf{B}$ and $\mathbf{C}\in\mathbb{R}^n$ are known constant vectors, $\boldsymbol{\theta}\in\mathbb{R}^n$ is a vector of unknown parameters, $\mathbf{u}(t)\in\mathbb{R}^m$ is the control signal vector and $\mathbf{y}(t)\in\mathbb{R}^l$ regulated output vector. The *m* is the number of inputs and *l* is the number of outputs. In general, the system dynamics is equal to:

$$\begin{aligned}
&\dot{\mathbf{x}}(t)=\mathbf{A}\mathbf{x}(t)+\mathbf{B}\mathbf{u}(t),\ \mathbf{A}=\mathbf{A}_m+\mathbf{B}\boldsymbol{\theta}^T\\
&\mathbf{y}(t)=\mathbf{C}^T\mathbf{x}(t),\ \mathbf{x}(0)=\mathbf{x}_0
\end{aligned} \tag{9}$$

For a continuous boundary signal r(t), the objective is to define an adaptive feedback signal $\mathbf{u}(t)$ such that $\mathbf{y}(t)$ tracks $\mathbf{r}(t)$ with desired specifications, while all the signals remain bounded. The MRAC architecture proceeds by considering the nominal controller:

$$\mathbf{u}_{nom}(t)=-\boldsymbol{\theta}^T\mathbf{x}(t)+\mathbf{K}_g\mathbf{r}(t) \tag{10}$$

where K_g is the feedforward gain and for eliminating the system steady-state error. The mentioned gain is as follows:

$$\mathbf{K}_g=\left(\mathbf{C}^T\mathbf{A}_m^{-1}\mathbf{B}\right)^{-1} \tag{11}$$

The direct MRAC is given by

$$\mathbf{u}(t)=-\hat{\boldsymbol{\theta}}^T(t)\mathbf{x}(t)+\mathbf{K}_g\mathbf{r}(t) \tag{12}$$

where $\hat{\boldsymbol{\theta}}(t)\in\mathbb{R}^n$ is the estimate of $\boldsymbol{\theta}$. Substituting (12) into (8) yields the closed-loop system dynamics

$$\begin{aligned}
&\dot{\mathbf{x}}(t)=\left(\mathbf{A}_{\mathbf{m}}-\mathbf{B}\tilde{\boldsymbol{\theta}}^T(t)\right)\mathbf{x}(t)+\mathbf{B}\mathbf{K}_g\mathbf{r}(t),\mathbf{x}(0)=\mathbf{x}_0\\
&\mathbf{y}(t)=\mathbf{C}^T\mathbf{x}(t)
\end{aligned} \tag{13}$$

where $\tilde{\boldsymbol{\theta}}(t)\triangleq\hat{\boldsymbol{\theta}}(t)-\boldsymbol{\theta}$ denotes the parametric estimation error. Letting $\mathbf{e}(t)\triangleq\mathbf{x}_m(t)-\mathbf{x}(t)$ be the tracking error signal, the tracking error dynamics can be written as

$$\dot{\mathbf{e}}(t)=\mathbf{A}_{\mathbf{m}}\mathbf{e}(t)+\mathbf{B}\tilde{\boldsymbol{\theta}}^T(t)\mathbf{x}(t),\mathbf{e}(0)=\mathbf{0} \tag{14}$$

The update law for the parametric estimate is given by

$$\dot{\hat{\boldsymbol{\theta}}}(t)=-\Gamma\mathbf{x}(t)\mathbf{e}^T(t)\mathbf{P}\mathbf{B},\ \hat{\boldsymbol{\theta}}(0)=\boldsymbol{\theta}_0 \tag{15}$$

where $\Gamma\in\mathbb{R}^+$ is the adaptation gain and $\mathbf{P}=\mathbf{P}^T>0$ by solving the algebraic Lyapunov equation

$$\mathbf{A}_m^T\mathbf{P}+\mathbf{P}\mathbf{A}_m=-\mathbf{Q} \tag{16}$$

For arbitrary $\mathbf{Q}=\mathbf{Q}^T>0$. The block diagram of the closed-loop system is given in Figure 3.

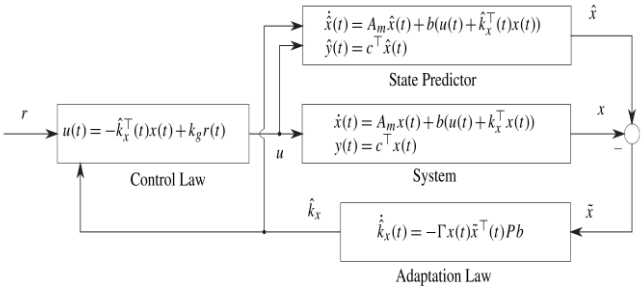

Figure 3. Closed-loop direct MRAC architecture [9]

3.3. $\mathscr{L}_1$ Adaptive Controller

In fact, the $\mathscr{L}_1$ adaptive control is a changed MRAC scheme, that its architecture principles had basically based on the internal model. The $\mathscr{L}_1$ controller actually contains the filter and the largest values of the unknown parameters. performance boundaries are the key to ensuring performance for adaptive control. Due to the internal model architecture, the controller reaches an appropriate delay margin even in high adaptation. In this controller, the design of the filter is important, even using linear theory can also be used. The $\mathscr{L}_1$ adaptive control has been tested in a variety of applications, including flight control for missiles, aircraft, and spacecraft. The key features of the $\mathscr{L}_1$ adaptive control architecture are to ensure resistance in the face of rapid adaptations [9].

Consider the class of systems

$$\begin{aligned}
&\dot{\mathbf{x}}(t)=\mathbf{A}_{\mathbf{m}}\mathbf{x}(t)+\mathbf{B}\left(\mathbf{u}(t)+\mathbf{f}(\mathbf{x}(t),t)\right),\ \mathbf{x}(0)=\mathbf{x}_0\\
&\mathbf{y}(t)=\mathbf{C}^{\mathbf{T}}\mathbf{x}(t)
\end{aligned} \tag{17}$$

where $\mathbf{A}_m\triangleq\mathbf{A}-\mathbf{B}\mathbf{k}_m^T$ is the state of the system (measured); $\mathbf{u}(t)\in\mathbb{R}^m$ is the control signal vector; $\mathbf{B}$ and $\mathbf{C}\in\mathbb{R}^n$ are known constant vectors, $\mathbf{A}$ is the known $n\times n$, $(\mathbf{A},\ \mathbf{B})$ controllable; $\mathbf{f}(\mathbf{x}(t),t)\approx\boldsymbol{\theta}^T\|\mathbf{x}(t)\|_\infty+\boldsymbol{\sigma}(t)\in\mathbb{R}^n$ is unknown nonlinear term, which $\boldsymbol{\theta}^T(t)$ and $\boldsymbol{\sigma}(t)$ as time-varying uncertainty and the time-varying disturbances, respectively.

It is assumed that $\boldsymbol{\theta}(t) \in \Theta, |\boldsymbol{\sigma}(t)| \le \Delta_0 \ \forall t \ge 0$, and $\mathbf{y}(t) \in \mathbb{R}^l$ is the regulated output vector. M is the number of inputs and l is the number of outputs. In this section, we present an adaptive control solution ($\mathcal{L}_1$ adaptive control), which ensures that the system output $\mathbf{y}$(t) follows a given piecewise-continuous bounded reference signal $\mathbf{r}$(t) with quantifiable transient and steady-state performance bounds.

Consider the control structure

$$\mathbf{u}(t) = \mathbf{u}_{\mathrm{m}}(t) + \mathbf{u}_{\mathrm{ad}}(t), \ \mathbf{u}_{\mathrm{m}}(t) = -\mathbf{k}_m{}^T \mathbf{x}(t) \tag{18}$$

where $\mathbf{k}_m \in \mathbb{R}^n$ renders $\mathbf{A}_m \triangleq \mathbf{A} - \mathbf{B}\mathbf{k}_m{}^T$ Hurwitz, while $\mathbf{u}_{\mathrm{ad}}$(t) is the adaptive component, to be defined shortly. The static feedback gain $\mathbf{k}_m$ leads to the following partially closed-loop system:

$$\begin{aligned} &\dot{\mathbf{x}}(t) = \mathbf{A}_m \mathbf{x}(t) + \mathbf{B}\left(\boldsymbol{\theta}^T \mathbf{x}(t) + \mathbf{u}_{ad}(t)\right), \ \mathbf{x}(0) = \mathbf{x}_0 \\ &\mathbf{y}(t) = \mathbf{C}^T \mathbf{x}(t) \end{aligned} \tag{19}$$

For the linearly parameterized system in (19), considered the state predictor

$$\begin{aligned} &\dot{\hat{\mathbf{x}}}(t) = \mathbf{A}_m \hat{\mathbf{x}}(\mathrm{t}) + \mathbf{B}\left(\hat{\boldsymbol{\theta}}^T(t)\mathbf{x}(t) + \mathbf{u}_{ad}(t)\right), \ \hat{\mathbf{x}}(0) = \mathbf{x}_0 \\ &\hat{\mathbf{y}}(t) = \mathbf{C}^T \hat{\mathbf{x}}(t) \end{aligned} \tag{20}$$

where $\hat{\mathbf{x}}(t) \in \mathbb{R}^n$ is the state of the predictor and $\hat{\boldsymbol{\theta}}^T(t) \in \mathbb{R}^n$ is the estimate of the parameter $\boldsymbol{\theta}$, governed by the following projection-type adaptive law:

$$\dot{\hat{\boldsymbol{\theta}}}(t) = \Gamma \, \mathrm{Pr}oj\left(\hat{\boldsymbol{\theta}}(t), -\tilde{\mathbf{x}}^T(t)\mathbf{P}\mathbf{B}\mathbf{x}(t)\right), \ \hat{\boldsymbol{\theta}}(0) = \hat{\boldsymbol{\theta}}_0 \tag{21}$$

where $\tilde{\mathbf{x}}(t) \triangleq \hat{\mathbf{x}}(t) - \mathbf{x}(t)$ is the prediction error, $\Gamma \in \mathbb{R}^+$ is the adaptation gain, and $\mathbf{P} = \mathbf{P}^T > 0$ solves the algebraic Lyapunov equation

$$\mathbf{A}_m{}^T \mathbf{P} + \mathbf{P}\mathbf{A}_m = -\mathbf{Q} \tag{22}$$

For arbitrary $\mathbf{Q} = \mathbf{Q}^T > 0$. The projection defined as [6]:

$$\mathbf{f}(\boldsymbol{\theta}) = \frac{(\varepsilon_\theta + 1)\boldsymbol{\theta}^T\boldsymbol{\theta} - \theta_{\max}^2}{\varepsilon_\theta \theta_{\max}^2}$$

$$proj(\boldsymbol{\theta}, \mathbf{y}) \triangleq \begin{cases} \mathbf{y} & if \ \mathbf{f}(\boldsymbol{\theta}) < 0 \\ \mathbf{y} & if \ \mathbf{f}(\boldsymbol{\theta}) \ge 0 \ and \ \nabla \mathbf{f}^T \mathbf{y} \le 0 \\ \mathbf{y} - \dfrac{\nabla \mathbf{f}}{\|\nabla \mathbf{f}\|} \left\langle \dfrac{\nabla \mathbf{f}}{\|\nabla \mathbf{f}\|}, \mathbf{y} \right\rangle \mathbf{f}(\boldsymbol{\theta}) & \\ if \ \mathbf{f}(\boldsymbol{\theta}) \ge 0 \ and \ \nabla \mathbf{f}^T \mathbf{y} > 0 & \end{cases} \tag{23}$$

The Laplace transform of the adaptive control signal is defined as

$$\mathbf{u}_{ad}(s) = -\mathbf{C}(s)(\hat{\boldsymbol{\eta}}(s) - \mathbf{k}_{\mathbf{g}}\mathbf{r}(s)) \tag{24}$$

where $\mathbf{r}(s)$ and $\hat{\boldsymbol{\eta}}(s)$ are the Laplace transforms of $\mathbf{r}(t)$ and $\hat{\boldsymbol{\eta}}(t) \triangleq \hat{\boldsymbol{\theta}}^T(t)\mathbf{x}(t)$, respectively. $\mathbf{K}_g = \left(\mathbf{C}^T \mathbf{A}_m{}^{-1} \mathbf{B}\right)^{-1}$, and $\mathbf{C}(s)$ is a BIBO-stable and strictly proper transfer function matrix with DC gain $\mathbf{C}(0) = \mathbf{I}$, and its state-space realization assumes zero initialization. The $\mathcal{L}_1$ adaptive control architecture with its main elements is represented in Figure 4.

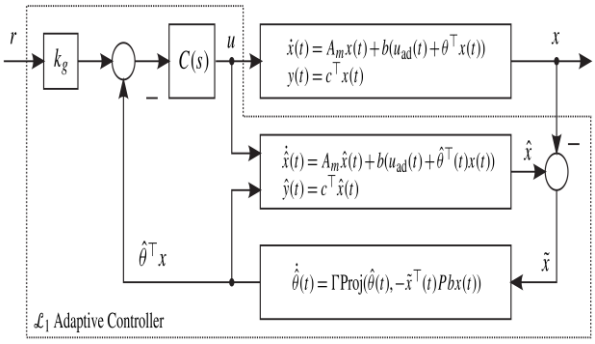

Figure 4. Closed-loop L_1 adaptive system [9]

3.4. Error Calculation Methods

In order to better compare the methods, the Lyapunov-based method, the MRAC and the $\mathcal{L}_1$ adaptive control method, the surface is below the error curve (error integral) is considered. This value is calculated by the two criteria defined below.

a. Error integral criterion with trapezoidal rule (TR)

In numerical analysis, the trapezoidal rule is a way of calculating the surface under the curve (the definite integral). The trapezoidal rule uses linear approximation and can to converted the function diagram with linear approximation into a trapezoidal series and then, by calculating the sum of their areas, the integral of the function is obtained. The relation to this law is defined as follows [20]:

$$\int_a^b f(x)dx \approx (b-a)[\frac{f(a)+f(b)}{2}] \tag{25}$$

b. criterion Integral Absolute Magnitude Error (IAE)

This criterion is defined as [20]**:**

$$IAE = \int_0^{pst} |e(t)|dt \tag{26}$$

4. Simulation Results

In this section, simulation, analysis and comparison of the three control methods discussed in the previous sections, the Lyapunov-based method, the MRAC method, and the $\mathcal{L}_1$ adaptive control method are checked. The state-space model of the SRS robot in terms of the equations obtained in the preceding sections is as follow:

$$\begin{aligned} &\mathbf{A}_m = \begin{pmatrix} -\mathbf{I}_{3*3} & \mathbf{I}_{3*3} \\ 0 & -\mathbf{I}_{3*3} \end{pmatrix}, B = \begin{pmatrix} \mathbf{0} \\ \mathbf{D}^{-1} \end{pmatrix}, \\ &\mathbf{u}_{ad}(t) = \mathbf{T}_{3*1}, \boldsymbol{\theta} = \begin{pmatrix} \mathbf{0} \\ -\mathbf{D} - \mathbf{H} \end{pmatrix}^T \end{aligned} \tag{27}$$

where I is a 3*3 identity matrix; T is model torques; D and C are as defined in the Appendix. The desired trajectory is stated as $\boldsymbol{\theta}_d = \begin{bmatrix} 1 - \cos(t) & 2\sin(t) & 0.5\sin(t) \end{bmatrix}^T$ which is free of singular point. In Lyapunov-based method assumed that $k_d = 40, k_i = 20$ and $\lambda = 1$. It is also assumed that the perturbation is entered into the system as Figure 5.

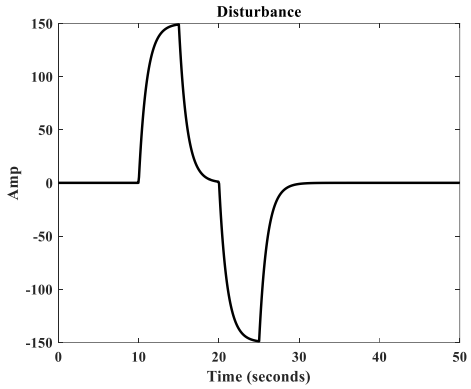

Figure 5. The disturbance profile applied as the system input

The following uncertainties are also considered in the system model:

$$\begin{cases} L_1 = 0.21m & if\ t \le 20\sec \\ L_1 = 0.5m & if\ t > 20\sec \end{cases}, \begin{cases} m_2 = 3.6kgr & if\ t \le 30\sec \\ m_2 = 4kgr & if\ t > 30\sec \end{cases} \quad (28)$$

Finally, a white noise with a mean zero and a variance one as noise is input to the torque. It is also assumed that data loss has occurred since50 seconds. The following simulation results are brought without loss data for all three methods and with data loss for $\mathscr{L}_1$ adaptive control and the MRAC.

a. Results without data loss

Simulation results for the Lyapunov-based method in Figure 6 has been shown. As you can see, in this method, tracking of the desired path is distorted due to disturbance, noise and uncertainty, and according to Table 3, there is a large error integral in comparison with the MRAC method and $\mathscr{L}_1$ adaptive method. Figure 7 shows the simulation results for the MRAC method. Figure 7 shows that in this case, despite disturbance and noise and uncertainties, it performs better than the Lyapunov-based method, and has a lower error according to Table 3, but suffers from a small distortion than the disturbance. The simulation results in Figure 8 is related to the $\mathscr{L}_1$ adaptive control, where can be seen, the output tracked the desired path and the state predictor as well, and even in the presence of disturbances and noise, and uncertainty, there has been no disturbance in tracking, and is very appropriate compared to the two methods Lyapunov-based and MRAC and has much less error according to Table 3.

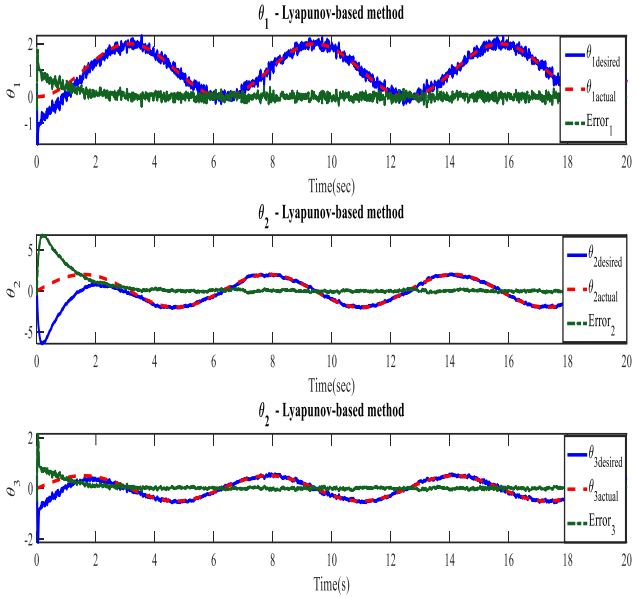

Figure 6. Desired path tracking with Lyapunov-based controller

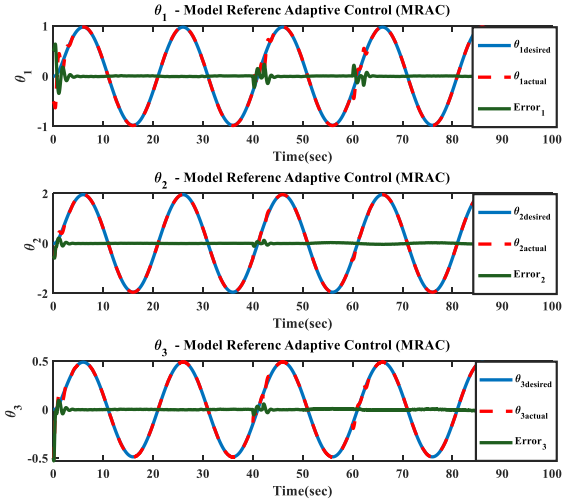

Figure 7. Desired path tracking with MRAC without data loss

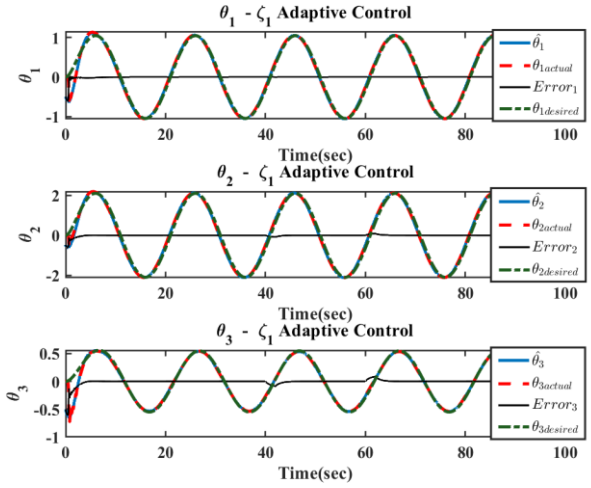

Figure 8. Desired path tracking and state predictor with $\mathscr{L}_1$ Adaptive Control without data loss

Table 3. Comparison of error integral values between different methods

		$\mathscr{L}_1$ *Adaptive*	MRAC	Lyapunov-based
TR	θ_1	0.19	0.22	0.99
	θ_2	0.33	0.94	8
	θ_3	0.17	0.49	1
	θ_{total}	0.69	1.65	9.99
IAE	θ_1	0.64	3.0695	4.79
	θ_2	0.92	1.8964	18.53
	θ_3	0.67	0.993	2.87
	θ_{total}	2.23	5.9589	26.19

b. Results with data loss

In this case, after the 50-second time that the data loss occurs, used a prior data and determines the optimal coefficients to predict a step ahead and continue the tracking process [14]. For this purpose, the following relation is used:

$$\hat{\mathbf{\theta}}(t)=\frac{t}{\alpha}\mathbf{u}(t-1)+(c-a)\mathbf{u}(t-2)-(c\times a)\hat{\mathbf{\theta}}(t-1)-(a\times f)\hat{\mathbf{\theta}}(t-2) \quad (29)$$

where $\alpha=0.36, a=0.16, c=0.05$ and $f=0.1$.

The results of the simulation for the MRAC and $\mathscr{L}_1$ adaptive control are presented in Figurs 9, 10. As can be seen, even with loss data, the $\mathscr{L}_1$ adaptive controller has a decent performance. While the MRAC has a lot of distortion and error.

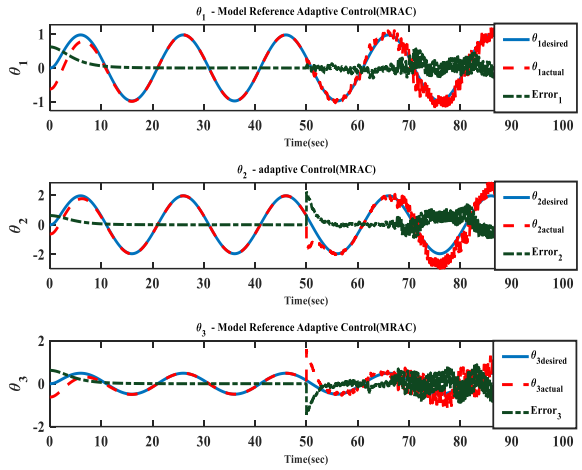

Figure 9. Desired path tracking with MRAC with data loss

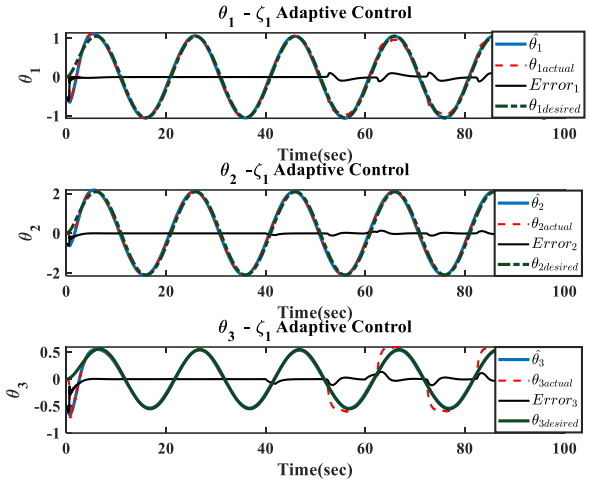

Figure 10. Desired path tracking and state predictor with $\mathscr{L}_1$ Adaptive Control with data loss

5. Conclusion

In this paper, according to the state space model of the S.R.S robot, the Lyapunov-based, MRAC and $\mathscr{L}_1$ Adaptive Controllers in the presence of loss data, time delay and uncertainty were designed. The results of design show the desired and very appropriate performance of the L_1 Adaptive Control method compared to the MRAC method and Lyapunov-based control and even other adaptive methods, which provides objectives such as stability, tracking reference inputs with the highest accuracy (minimum steady state error), and especially the margin of stability compared to the other conventional adaptive control methods. In fact, with this kind of control, it is possible to increase system bandwidth and thus increase the degree of freedom for the user and increase the flexibility of the system.

Appendices: Appendix A

In Eq. (4):

$$\begin{pmatrix}\tau_1\\ \tau_2\\ \tau_3\end{pmatrix}=\begin{pmatrix}D_{11} & D_{12} & D_{13}\\ D_{12} & D_{22} & D_{23}\\ D_{13} & D_{23} & D_{33}\end{pmatrix}\begin{pmatrix}\ddot{q}_1\\ \ddot{q}_2\\ \ddot{q}_3\end{pmatrix}+\begin{pmatrix}H_1\\ H_2\\ H_3\end{pmatrix}+\begin{pmatrix}G_1\\ G_2\\ G_3\end{pmatrix} \quad (30)$$

Let us define $m_i, i=1,2,3$ link masses and inertia tensor matrix as:

$$\mathbf{I_i}=\begin{pmatrix}I_{ixx} & 0 & 0\\ 0 & I_{iyy} & 0\\ 0 & 0 & I_{izz}\end{pmatrix}, i=1,2,3 \quad (31)$$

Using Table 2 transformation matrices will be derived as follows:

$$\mathbf{T}_2^1=\begin{pmatrix}c_2 & 0 & s_2 & 0\\ s_2 & 0 & -c_2 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 0 & 1\end{pmatrix},\ \mathbf{T}_1^0=\begin{pmatrix}c_1 & 0 & -s_1 & 0\\ s_1 & 0 & c_1 & 0\\ 0 & -1 & 0 & d_1\\ 0 & 0 & 0 & 1\end{pmatrix} \quad (32)$$

$$\mathbf{T}_3^2=\begin{pmatrix}c_3 & -s_3 & 0 & 0\\ s_3 & c_3 & 0 & 0\\ 0 & 0 & 1 & d_2\\ 0 & 0 & 0 & 1\end{pmatrix}$$

Where, c_i and s_i are $\cos(\theta_i)$ and $\sin(\theta_i)$ respectively.

Let $\mathbf{P_i}$ be the first three elements of the last column of $\mathbf{T}_i^0$

and $\mathbf{r}_i = \begin{pmatrix} r_{ix} & r_{iy} & r_{iz} \end{pmatrix}^T, i = 1,2,3$ is link centre of masses vector. Defining $\bar{\mathbf{c}}_\mathbf{i} = \mathbf{r_i} - \mathbf{P_{i-1}}$, $i = 1,2,3$ one can obtain:

$$
\begin{aligned}
&D_{11} = I_{1yy} + m_1(r_{1x}^2 + r_{1y}^2) + m_2(r_{2x}^2 + r_{2y}^2) + \\
&m_3(r_{3x}^2 + r_{3y}^2) + (I_{2zz} + I_{3zz})c_2^2 + (I_{2xx} + I_{3xx}c_3^2 + I_{3yy}s_3^2)s_2^2 \\
&D_{12} = (I_{3yy} - I_{3xx})s_2s_3c_3 + m_2(r_{2x}c_{2z}s_1 - r_{2y}c_{2z}c_1) \\
&+m_3(r_{3x}c_{3z}s_1 - r_{3y}c_{3z}c_1) \\
&D_{13} = I_{3zz}c_2 + m_3((r_{3x}c_{3x} + r_{3y}c_{3y})c_2 - \\
&(r_{3x}c_{3z}c_1 + r_{3y}c_{3z}s_1)s_2) \\
&D_{22} = I_{2yy} + I_{3xx} + (I_{3yy} - I_{3xx})c_3^2 + m_2(c_{2y}^2 + c_{2z}^2 + \\
&c_1^2(c_{2x}^2 - c_{2y}^2) + 2c_{2x}c_{2y}c_1s_1) + \\
&m_3(c_{3y}^2 + c_{3z}^2 + 2c_{3x}c_{3y}c_1s_1 + c_1^2(c_{3x}^2 - c_{3y}^2)) \\
&D_{22} = I_{2yy} + I_{3xx} + (I_{3yy} - I_{3xx})c_3^2 + \\
&m_2(c_{2y}^2 + c_{2z}^2 + c_1^2(c_{2x}^2 - c_{2y}^2) + 2c_{2x}c_{2y}c_1s_1) \\
&+m_3(c_{3y}^2 + c_{3z}^2 + 2c_{3x}c_{3y}c_1s_1 + c_1^2(c_{3x}^2 - c_{3y}^2)) \\
&D_{33} = I_{3zz} + m_3(c_{3x}^2(1 - c_1^2s_2^2) - 2c_{3x}c_{3y}c_1s_1s_2^2 \\
&-2c_{3x}c_{3z}c_1c_2s_2 + c_{3y}^2c_1^2s_2^2 + c_{3y}^2c_2^2 \\
&-2c_{3y}c_{3z}s_1s_2c_2 + c_{3z}^2s_2^2)
\end{aligned}
\tag{33}
$$

$$
\begin{aligned}
&H_1 = \dot{\theta}_1^2(m_1(c_{1x}r_{1y} - c_{1y}r_{1x}) + m_2(r_{2y}c_{2x} - r_{2x}c_{2y}) \\
&+m_3(c_{3x}r_{3y} - c_{3y}r_{3x})) + \dot{\theta}_1\dot{\theta}_2(2(I_{2xx} - I_{2zz} + I_{3yy} \\
&-I_{3zz})c_2s_2 + 2m_2(r_{2x}c_{2z}c_1 + r_{2y}c_{2z}s_1) + 2m_3(r_{3x}c_{3z}c_1 \\
&+r_{3y}c_{3z}s_1)) + \dot{\theta}_1\dot{\theta}_3(2(I_{3yy} - I_{3xx})c_3s_3s_2^2 + m_3(c_{3x}r_{3y} \\
&-r_{3x}c_{3y} + r_{3y}c_{3x}c_2 - c_{3y}r_{3x}c_2 - c_{3z}r_{3y}c_1s_2 + \\
&c_{3z}r_{3x}s_1s_2)) + \dot{\theta}_2^2(m_2(r_{2y}c_{2x}c_1^2 - r_{2x}c_{2y}s_1^2 + \\
&c_1s_1(r_{2y}c_{2y} - r_{2x}c_{2x})) + m_3(r_{3y}c_{3x}c_1^2 - r_{3x}c_{3y}s_1^2 + \\
&c_1s_1(r_{3y}c_{3y} - r_{3x}c_{3x})) + (I_{3yy} - I_{3xx})c_2c_3s_3) + \dot{\theta}_2\dot{\theta}_3((I_{3xx} \\
&-I_{3yy} - I_{3zz})s_2 + 2(I_{3yy} - I_{3xx})s_2c_3^2 + m_3((c_{3y}r_{3x} + \\
&c_{3x}r_{3y})c_1s_1s_2 - c_{3x}r_{3x}s_1^2s_2 - c_{3y}r_{3y}c_1^2s_2)) + \dot{\theta}_3^2 \\
&(m_3c_2(r_{3y}c_{3x} - r_{3x}c_{3y}))
\end{aligned}
$$

$$
\begin{aligned}
&H_2 = \dot{\theta}_1^2((I_{2zz} - I_{2xx} + I_{3yy} + I_{3zz} - I_{3xx}c_3^2 + \\
&I_{3yy}c_3^2)c_2s_2 - m_2c_{2z}(c_{2y}s_1 + c_{2x}c_1) - m_3c_{3z} \\
&(c_{3x}c_1 + c_{3y}s_1)) + \dot{\theta}_1\dot{\theta}_3((I_{3xx} - I_{3yy} + I_{3zz} - \\
&2I_{3xx}c_3^2 + 2I_{3yy}c_3^2)s_2 + 2m_3(c_{3z}^2s_2 - \\
&c_{3x}c_{3z}c_1c_2 - c_{3y}c_{3z}c_2s_1)) + \dot{\theta}_3^2(m_3((c_{3z}^2 - \\
&c_{3y}^2 - c_{3x}^2c_1^2 + c_{3y}^2c_1^2)c_2s_2 + c_{3y}c_{3z}s_1 + \\
&c_{3x}c_{3z}c_1 - 2c_{3x}c_{3z}c_1c_2^2 - 2c_{3y}c_{3z}c_2^2s_1 - \\
&c_{3x}c_{3y}c_1c_2s_1s_2)) + \dot{\theta}_2\dot{\theta}_3(2(I_{3xx} - I_{3yy})c_3s_3 + \\
&2m_3(c_{3x}c_{3y}c_2 + c_{3y}c_{3z}c_1s_2 - c_{3x}c_{3z}s_1s_2 - \\
&2c_{3x}c_{3y}c_1^2c_2 + c_{3x}^2c_1c_2s_1 - c_{3y}^2c_1c_2s_1)) \\
&H_3 = \dot{\theta}_1^2((I_{3xx} - I_{3yy})c_3s_2^2s_3 + m_3(c_{3y}c_{3z}c_1s_2 - \\
&c_{3x}c_{3z}s_1s_2)) + \dot{\theta}_1\dot{\theta}_2((I_{3xx} - I_{3yy} - I_{3zz} - 2I_{3xx}s_3^2 \\
&+2I_{3yy}s_3^2)s_2 + 2m_3c_{3z}(c_{3y}c_2s_1 + c_{3x}c_1c_2 - c_{3z}s_2)) \\
&+\dot{\theta}_2^2((I_{3yy} - I_{3xx})c_3s_3 + m_3((c_{3y}^2 - c_{3x}^2)c_1s_1c_2 + \\
&c_{3x}c_{3y}c_1^2c_2 + c_{3x}c_{3z}s_1s_2 - c_{3y}c_{3z}c_1s_2 - c_{3x}c_{3y}c_2)) \\
&G_1 = 0 \\
&G_2 = -g_0(m_2(c_{2x}c_1 + c_{2y}s_1) + m_3(c_{3x}c_1 + c_{3y}s_1)) \\
&G_3 = g_0m_3s_2(-c_{3x}s_1 + c_{3y}c_1)
\end{aligned}
$$

References

[1] P. Ioannou, B. Fidan, Advances in design and control, Adaptive control tutorial 11 (2006) 389.

[2] T. Nef, R. Riener, ARMin-design of a novel arm rehabilitation robot, 9th International Conference on Rehabilitation Robotics, 2005. ICORR 2005., IEEE (2005) 57–60.

[3] A. Nedjatia and B. Vizvárib, Robot Path Planning by Traveling Salesman Problem with Circle Neighborhood: modeling, algorithm, and applications, (2020) arXiv preprint arXiv:2003.06712.

[4] H. Ahmadian, H. A. Talebi, I. Sharifi, $\mathscr {L} _ {1} $ Adaptive Controller Design for Single-Link Flexible Joint Manipulator with Fuzzy-PID Filter, In 2020 28th Iranian Conference on Electrical Engineering (ICEE). IEEE (2020) p–1-6.

[5] I. Gregory, C. Cao, E. Xargay, N. Hovakimyan, X. Zou, L1 adaptive control design for NASA AirSTAR flight test vehicle, AIAA guidance, navigation, and control conference, (2009) 5738.

[6] J. Yang, H. Su, Z. Li, D. Ao, R. Song, Adaptive control with a fuzzy tuner for cable-based rehabilitation robot, International Journal of Control, Automation Systems 14 (2016) 865–875.

[7] A. Ebrahimi and J. Addeh, Classification of ECG Arrhythmias Using Adaptive Neuro-Fuzzy Inference System and Cuckoo Optimization Algorithm, Computational Research Progress in Applied Science and Engineering (2015) 134–140.

[8] H. Behbahani, H. Ziari, A. Amini, V. N. M. Gilani, and R. Salehfard, Investigation of un-signalized roundabouts delay with adaptive-network-based fuzzy inference system and fuzzy logic, Computational Research Progress in Applied Science & Engineering 2 (2016) 1–7.

[9] N. Hovakimyan, C. Cao, $\mathscr{L}$1 adaptive control theory: Guaranteed robustness with fast adaptation, SIAM2010.
[10] H. Lee, S. Snyder, N. Hovakimyan, $\mathcal {L} _1 $ Adaptive Output Feedback Augmentation for Missile Systems, IEEE Transactions on Aerospace Electronic Systems 54 (2017) 680–692.
[11] H. Beikzadeh, G. Liu, Trajectory tracking of quadrotor flying manipulators using L1 adaptive control, Journal of the Franklin Institute 355 (2018) 6239–6261.
[12] F. Peter, M. Leitão, F. Holzapfel, Adaptive augmentation of a new baseline control architecture for tail-controlled missiles using a nonlinear reference model, AIAA Guidance, Navigation, and Control Conference (2012) 5037.
[13] H. Jafarnejadsani, D. Sun, H. Lee, N. Hovakimyan, Optimized L1 Adaptive Controller for Trajectory Tracking of an Indoor Quadrotor, Journal of Guidance, Control, Dynamics 40 (2017) 1415–1427.
[14] S. Salman, S.A. Anavatti, T. Asokan, Adaptive fuzzy control of unmanned underwater vehicles, Indian Journal of Geo-Marine Sciences 40 (2011) 168–175.
[15] P. Trslić, E. Omerdic, G. Dooly, D. Toal, Neuro-Fuzzy Dynamic Position Prediction for Autonomous Work-Class ROV Docking, Sensors 20 (2020) 693.
[16] A. Marzbanrad, M. Eghtesad, R. Kamali, A robust adaptive fuzzy sliding mode controller for trajectory tracking of ROVs, 2011 50th IEEE Conference on Decision and Control and European Control Conference, IEEE, 2011, pp. 2863-2870.
[17] M. Torabi, M. Sharifi, G. Vossoughi, Robust adaptive sliding mode admittance control of exoskeleton rehabilitation robots, Scientia Iranica 25 (2018) 2628–2642.
[18] D. Maalouf, A. Chemori, V. Creuze, L1 adaptive depth and pitch control of an underwater vehicle with real-time experiments, Ocean Engineering 98 (2015) 66–77.
[19] M. Babaiasl, A. Ghanbari, S. Noorani, Anthropomorphic mechanical design and Lyapunov-based control of a new shoulder rehabilitation system, Engineering Solid Mechanics 2 (2014) 151–162.
[20] R. Maiti, K.D. Sharma, G. Sarkar, Linear Consequence-Based Fuzzy Parallel Distributed Compensation Type $ L_ {1} $ Adaptive Controller for Two Link Robot Manipulator, IEEE Transactions on Circuits Systems I: Regular Papers 66 (2019) 3978–3990.