# Building up a Control Algorithm Desired Compensation Adaptive with Reduced On-Line Computation for Robot Almega 16

**Vo Thu Ha**

*University of Economics – Technology for Industries, Ha Noi city, Viet Nam*

*Received: June 06, 2016; accepted: June 9, 2017*

**Abstract**

*This paper presents a build of improving algorithm adaptive control in joint space for the motion system of the Almega16 manipulator. The proposed controller eliminates the need for on-line computation of the regression matrix by replacing joints position and velocity with the desired joints position and velocity. In a adaptive control algorithm, the kinetic parameters identification system always provides state information update on the time-variant parameters. The true value is compared to the reference set-point and its evaluation result is input to the controller for adjustment. The results from Matlab - Simmechanic simulations and experiments show that the motion system of Robot Almega16 satisfies the requirement of a control system: the errors of rotating joints quickly converge to zero within a short transient time, so that closed-loop system is stable based on Lyapunov method.*

Keywords: Robot Almega 16, Desired compensation Adaptation Law, Lyapunov method.

## 1. Introduction

We has studied to develop adaptive control algorithm Li – Slotine [2, 6] for Robot Almega 16 and the new approach has shown the highlighted advantage, namely the system worked exactly without knowing the dynamical parameters of the robot dynamics Almega 16, the controller solved the problem by estimating the parameters based on law Li - Slotine, the controller solved the problem by estimating the parameters based on law Li - Slotine, by the way the number of calculation would be reduced significantly while ensuring Robot Almega 16's operational flexibility, concurrently rejecting the erroneous components of the joint angle & the last phase's positional errors, as a result, the Almega 16 motion system works stably and precisely with a small transitional period. However, adaptive controller still has weak points, it requires a large amount of the online caculation and unstable with interferences. One of the disadvantages of the adaptive[*]Li-Slotine control algorithm in [2, 6] is that the regression matrix (e.g., the matrix $\mathbf{Y}(.)$ ) used as feedforward compensation must be calculated on-line. The regression matrix must be calculated on-line since it depends on the measurements of the joint position $\mathbf{q}$ and velocity $\dot{\mathbf{q}}$ . However, one – line computation of the regression matrix can be very difficult if one desires to control a manipulator with many degrees of freedom. To eliminate the need for on – line computation of the regression matrix, we will now examine the desires compensation adaptation law. The desires compensation adaptation law eliminates the need for on – line computation of the regression matrix by replacing $\mathbf{q}$ and $\dot{\mathbf{q}}$ with the desired joint position $\mathbf{q_d}$ and $\dot{\mathbf{q}}_\mathbf{d}$ velocity. That is the desired compensation adaptation law only depends on desired trajectory information; therefore, the desired compensation adaptation law regression matrix can be calculated a priori off - line. This paper will be design and simulate the desired compensation adaptation law for the three-link arm is almega 16 robot.

**Fig. 1.** Six-link Almega 16 arm

## 2. Object Control

The Almega 16 robot is shown in Fig. 1, as follows [6]. This is a vertical welding robot with fast, rhythmic and precise movement characteristics, including six–link axes, each one link axes is

[*] Corresponding author: (+84) 913.024.989
Email:vtha@uneti.edu.vn.

equipped with a permanent magnet synchronous servo motor and closed loop control. In the article using only three-link axes as the research object, specifically the main specifications of the three joints as follows.

First joint: Rotation angle: $\pm 135^0$. Center tops from top to bottom: 28cm. Center line of axis I to the center of the cylinder: 35cm. Second joint: Rotation angle: $\pm 135^0$. The length between the center of the axis I and II is 65cm. Third joint: Angle of rotation: $90^0$ and $-45^0$. The length between the two centers of axis I and II is 47cm. The total volume of the Almega16 Robot: V = 0,12035 $m^3$. Total weight of the robot: 250kg. The mass of joints is as follows:

$m_0$ = 100 kg, $m_1$ = 67 kg, $m_2$ = 52 kg, $m_3$ = 16 kg,

$m_4$ = 10 kg, $m_5$ = 4 kg, $m_6$ = 1 kg.

The motion system Almega16 Robot is a nonlinear system that has constant model parameters and is interfering with the channel between the component motion axes. According to the literature as follows [3], the first three joints have fully integrated the dynamics of the freedom arm. The motor connected to the joint is usually a planetary gear and Small air gap. It is influenced by friction such as static friction, friction, viscous friction and so on. Therefore, the first three joints are the basic chain that ensures movement in 3D (X, Y, Z) space. The basis for the study of the next steps in robot manipulator motion systems. The problem with the controller is that: should design the quality control ensures precise orbit grip that does not depend on the parameters of the model uncertainty and the impact on channel mix between match-axis error between joint angles and the angle joints actually put a small (<0.1%).

## 3. Desired Compensation Adaptation law

### *3.1 Dynamic Model of Robot Manipulators*

The dynamic of an n-link rigid manipular, [1],[2],[3],[4],[6] can be written as

$$.\tau = \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{H}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) + \mathbf{F_d}\dot{\mathbf{q}}\ . \tag{1}$$

Where $\mathbf{q}$ is the *n x 1* joint variable vector, $\boldsymbol{\tau}$ is a *n x 1* generalized torque vector $\mathbf{M}(\mathbf{q})$ is the nxn inertia matrix, $\mathbf{H}(\mathbf{q},\dot{\mathbf{q}})$ is the *n x 1* Coriolis/centripetal vector, $\mathbf{G}(\mathbf{q})$ is the *n x1* gravity vector and $\mathbf{F_d}$ is the *n x n* positive-definite, diagonal matrix that is used to represent the dynamic coefficients of friction, and all other quantities are as defined in Chapter 3, [3].

### *3.2 Controller Design*

The controller design problem is as follows: Given the desired trajectory $\mathbf{q}_d(t)$, and with some or all the manipulator parameters being unknown, derive a control law for the actuator torques such that the manipulator output $\mathbf{q}(t)$ tracks the desired trajectory after an initial adaptation process.

In Chapter 6, [3], adaptive control of robot manipulator involves separating the known time functions from the unknown constant parameters is given by

$$\mathbf{Y}(.)\theta = \mathbf{M}(\mathbf{q})(\ddot{\mathbf{q}}_d + \dot{\mathbf{e}}) + \mathbf{H}(\mathbf{q},\dot{\mathbf{q}})(\dot{\mathbf{q}}_d + \mathbf{e}) + \mathbf{G}(\mathbf{q}) + \mathbf{F_d}\dot{\mathbf{q}} \tag{2}$$

where $\mathbf{Y}(.)$ is the *n x r* regression matrix that depends only on known time functions of the actual and desired trajectory, and $\theta$ is the *r x 1* vector of unknown constant parameters.

We define the joint tracking error to be

$$e = \mathbf{q}_d - \mathbf{q} \tag{3}$$

In the desired compensation adaptation law, this separation of parameters from time functions is given by

$$\mathbf{Y}_d(.)\theta = \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}}_d + \mathbf{H}(\mathbf{q}_d,\dot{\mathbf{q}}_d)\mathbf{q}_d + \mathbf{G}(\mathbf{q_d}) + \mathbf{F}_d\dot{\mathbf{q}}_d \tag{4}$$

where $\mathbf{Y}_d(.)$ is the *n x r* regression matrix that depends only on known functions of the desired trajectory? Note that if we substitute $\mathbf{q_d}$ and $\dot{\mathbf{q}}_\mathbf{d}$ for $\mathbf{q}$ and $\mathbf{q_d}$ respectively, into (2), the regression matrix formulation given by (2) is equivalent to that given by (4).

Utilizing the regression matrix formulation given in (4), the desired compensation adaptation law is formulated as

$$\tau = \mathbf{Y}_d(.)\hat{\theta} + \mathbf{k_v}\mathbf{r} + \mathbf{k_p}\mathbf{e} + \mathbf{k_a}\left\|\mathbf{e}\right\|^2\mathbf{r} \tag{5}$$

where $\mathbf{k_v}, \mathbf{k_p}, \mathbf{k_a}$ are scalar, constant, control gains, $\hat{\theta}$ is the *r x 1* vector of parameter estimates, and the filtered tracking error is defined as

$$\mathbf{r} = \mathbf{e} + \dot{\mathbf{e}} \tag{6}$$

The desired compensation adaptation law given by (5) is quite similar to adaptive controllers discussed in Chapter 6, [3] with the exception of the term $\mathbf{k_a}\left\|\mathbf{e}\right\|^2\mathbf{r}$ in (5). It turns out that this additional term is used to compensate for the difference between $\mathbf{Y}(.)\theta$ and $\mathbf{Y}_d(.)\theta$ given in (2) and (4), respectively. This difference between the actual regression matrix and the desired regression matrix formulations can be quantified as

$$\left\|\tilde{\mathbf{Y}}\right\| \le \zeta_1 \left\|\mathbf{e}\right\| + \zeta_2 \left\|\mathbf{e}\right\|^2 + \zeta_3 \left\|r\right\| + \zeta_4 \left\|r\right\|\left\|\mathbf{e}\right\| \qquad (7)$$

where

$$\tilde{\mathbf{Y}} = \mathbf{Y}(.)\theta - \mathbf{Y}_d\theta \qquad (8)$$

And $\zeta_1, \zeta_2, \zeta_3$ and $\zeta_4$ are positive bounding constants that depend on the desired trajectory and the physical properties of the specific robot configuration (i.e., link mass, link length, friction coefficients, etc.).

To analyze the stability of the controller given by (5), we must form the corresponding error system.

First, we rewrite (1) in terms of $\mathbf{Y}(.)\theta$ and $\mathbf{r}$ defined in (2) and (6), respectively. That is, we have

$$\mathbf{M(q)\dot{r}} = -\mathbf{H(q,\dot{q})r} + \mathbf{Y(.)}\theta - \boldsymbol{\tau} \qquad (9)$$

Adding and subtracting the term $\mathbf{Y}(\cdot)\theta$ on the right-hand side of (9).

$$\mathbf{M(q)\dot{r}} = -\mathbf{H(q,\dot{q})r} + \mathbf{Y}_d\mathbf{(.)}\theta + \tilde{\mathbf{Y}} - \boldsymbol{\tau} \qquad (10)$$

where $\tilde{\mathbf{Y}}$ is defined in (8). Substituting the control given by (5) into (10) yields the error system

$$\begin{aligned} \mathbf{M(q)\dot{r}} &= -\mathbf{H(q,\dot{q})r} + \mathbf{Y}_d\mathbf{(.)}\theta + \tilde{\mathbf{Y}} + \mathbf{Y}_d\mathbf{(.)}\hat{\theta} \\ &\quad -\mathbf{k_v r} - \mathbf{k_p e} - \mathbf{k_a}\left\|\mathbf{e}\right\|^2 \mathbf{r} \\ &\Leftrightarrow \\ \mathbf{M(q)\dot{r}} &= -\mathbf{H(q,\dot{q})r} - \mathbf{k_v r} - \mathbf{k_p e} \\ &\quad -\mathbf{k_a}\left\|\mathbf{e}\right\|^2 \mathbf{r} + \tilde{\mathbf{Y}} + \mathbf{Y}_d\mathbf{(.)}\tilde{\theta} \end{aligned} \qquad (11)$$

The parameter adaptive update law is

$$\dot{\hat{\theta}} = -\dot{\tilde{\theta}} = \Gamma \mathbf{Y}_d^T(.)\mathbf{r} \qquad (12)$$

where $\Gamma$ is an $r\ x\ r$ positive defined, diagonal, constant, adaptive gain matrix, and the parameter error is defined by

$$\tilde{\theta} = \theta - \hat{\theta} \qquad (13)$$

### *3.3 A Globally Stable Adaptive Controller*

To derive the control algorithm and adaptation law, we consider the Lyapunov function candidate

$$V(t) = \tfrac{1}{2}\mathbf{r^T M(q) r} + \tfrac{1}{2}\mathbf{k_p e^T e} + \tfrac{1}{2}\tilde{\theta}^T \Gamma^{-1}\tilde{\theta} \qquad (14)$$

Differentiating (14) with respect to time yields

$$\dot{V}(t) = \tfrac{1}{2}\mathbf{r^T \dot{M}(q) r} + \mathbf{r^T M(q)\dot{r}} + \mathbf{k_p e^T \dot{e}} + \tilde{\theta}^T \Gamma^{-1}\dot{\tilde{\theta}} \qquad (15)$$

since scalar quantities can be transposed. Substituting (11) into (15)

$$\begin{aligned} \dot{V}(t) &= \mathbf{k_p e^T \dot{e}} - \mathbf{k_v r}^T\mathbf{r} - \mathbf{k_p r}^T\mathbf{e} - \mathbf{k_a}\left\|\mathbf{e}\right\|^2 \mathbf{r}^T\mathbf{r} \\ &\quad + \mathbf{r}^T\tilde{\mathbf{Y}} + \tfrac{1}{2}\mathbf{r}^T\left(\mathbf{\dot{M}(q)} - 2\mathbf{H(q,\dot{q})}\right)\mathbf{r} \\ &\quad + \tilde{\theta}^T\Gamma^{-1}\dot{\tilde{\theta}} + \mathbf{r}^T\mathbf{Y}_d(.)\tilde{\theta} \end{aligned} \qquad (16)$$

where we have used the property of skew-symmetric to eliminate the term $\tfrac{1}{2}\mathbf{r}^T\left(\mathbf{\dot{M}(q)} - 2\mathbf{H(q,\dot{q})}\right)$. Let us define the updated law in (12), it is easy to see that second line (16) is equal to zero, therefore, by invoking the definition of $\mathbf{r}$ given in (6), (16) simplifies to

$$\dot{V}(t) = -\mathbf{k_p e^T e} - \mathbf{k_v r}^T\mathbf{r} - \mathbf{k_p r}^T\mathbf{e} - \mathbf{k_a}\left\|\mathbf{e}\right\|^2 \mathbf{r}^T\mathbf{r} + \mathbf{r}^T\tilde{\mathbf{Y}} \qquad (17)$$

From (17), we can place an upper bound on $V$ in the following manner:

$$\dot{V}(t) \le -\mathbf{k_p}\left\|\mathbf{e}\right\|^2 - \mathbf{k_v}\left\|\mathbf{r}\right\|^2 - \mathbf{k_a}\left\|\mathbf{e}\right\|^2\left\|\mathbf{r}\right\|^2 + \left\|\mathbf{r}\right\|\left\|\tilde{\mathbf{Y}}\right\| \qquad (18)$$

A new upper bound $\dot{V}$ can be obtained by substituting (7) into (18) to yield

$$\begin{aligned} \dot{V}(t) &\le -\mathbf{k_p}\left\|\mathbf{e}\right\|^2 - \mathbf{k_v}\left\|\mathbf{r}\right\|^2 - \mathbf{k_a}\left\|\mathbf{e}\right\|^2\left\|\mathbf{r}\right\|^2 + \zeta_1\left\|\mathbf{e}\right\|\left\|\mathbf{r}\right\| \\ &\quad + \zeta_2\left\|\mathbf{e}\right\|^2\left\|\mathbf{r}\right\| + \zeta_3\left\|\mathbf{r}\right\|^2 + \zeta_4\left\|\mathbf{r}\right\|^2\left\|\mathbf{e}\right\| \end{aligned} \qquad (19)$$

by rearranging the second line of (19), it can be written as

$$\begin{aligned} \dot{V}(t) &\le -\mathbf{k_p}\left\|\mathbf{e}\right\|^2 - \mathbf{k_v}\left\|\mathbf{r}\right\|^2 - \mathbf{k_a}\left\|\mathbf{e}\right\|^2\left\|\mathbf{r}\right\|^2 + \zeta_1\left\|\mathbf{e}\right\|\left\|\mathbf{r}\right\| \\ &\quad -\zeta_2\left\|\mathbf{e}\right\|^2\left[\tfrac{1}{2} - \left\|\mathbf{r}\right\|\right]^2 - \zeta_4\left\|\mathbf{r}\right\|^2\left[\tfrac{1}{2} - \left\|\mathbf{e}\right\|\right]^2 \\ &\quad + \left(\zeta_2 + \zeta_4\right)\left\|\mathbf{e}\right\|^2\left\|\mathbf{r}\right\|^2 + \left(\tfrac{\zeta_2}{4}\right)\left\|\mathbf{e}\right\|^2 + \left(\zeta_3 + \tfrac{\zeta_4}{4}\right)\left\|\mathbf{r}\right\|^2 \end{aligned} \qquad (20)$$

After collecting common terms in (20), it can be rewritten as

$$\begin{aligned} \dot{V}(t) &\le -\left(\mathbf{k_p} - \tfrac{\zeta_2}{4}\right)\left\|\mathbf{e}\right\|^2 - \left(\mathbf{k_v} - \zeta_3 - \tfrac{\zeta_4}{4}\right)\left\|\mathbf{r}\right\|^2 \\ &\quad + \zeta_1\left\|\mathbf{e}\right\|\left\|\mathbf{r}\right\| - \zeta_2\left\|\mathbf{e}\right\|^2\left[\tfrac{1}{2} - \left\|\mathbf{r}\right\|\right]^2 - \zeta_4\left\|\mathbf{r}\right\|^2\left[\tfrac{1}{2} - \left\|\mathbf{e}\right\|\right]^2 \\ &\quad - \left(\mathbf{k_a} - \zeta_2 - \zeta_4\right)\left\|\mathbf{e}\right\|^2\left\|\mathbf{r}\right\|^2 \end{aligned} \qquad (21)$$

by noting that if the control gain $\mathbf{k_a}$ is adjusted in accordance with

$$\mathbf{k_a} > \zeta_2 + \zeta_4 \qquad (22)$$

We can see that the terms on the second line of (21) will all be negative; therefore, we can obtain the new upper bound on $\dot{V}(t)$.

$$\dot{V}(t) \le \left(\mathbf{k_p} - \zeta_2/4\right)\|\mathbf{e}\|^2 - \left(\mathbf{k_v} - \zeta_3 - \zeta_4/4\right)\|\mathbf{r}\|^2 + \zeta_1\|\mathbf{e}\|\|\mathbf{r}\| \quad (23)$$

By rewriting (23) in the matrix form

$$\dot{V}(t) \le -\mathbf{x}^{\mathrm{T}}\mathbf{Q}\mathbf{x} \quad (24)$$

where

$$\mathbf{Q} = \begin{bmatrix} \mathbf{k_p} - \zeta_2/4 & -\zeta_1/2 \\ -\zeta_1/2 & \mathbf{k_v} - \zeta_3 - \zeta_4/4 \end{bmatrix} \text{ and } \mathbf{x} = \begin{bmatrix} \|\mathbf{e}\| \\ \|\mathbf{r}\| \end{bmatrix},$$

We can establish sufficient conditions on $\mathbf{k_p}$ and $\mathbf{k_v}$ such that the matrix $\mathbf{Q}$ in (24) is positive definite, we can see that if

$$\mathbf{k_p} > \zeta_1/2 + \zeta_2/4 \quad (25)$$

and

$$\mathbf{k}_v > \zeta_1/2 + \zeta_3 + \zeta_4/4 \quad (26)$$

The matrix $\mathbf{Q}$ definite in (24) will be positive definite, $\dot{V}(t)$ will be negative semidefinite.

We now detail the type of stability for the tracking error. First, since $\dot{V}(t)$ is negative semidefinite, we can state that $V$ is upper bounded. Using the fact that $V$ is upper bounded, we can state that $\mathbf{e}, \dot{\mathbf{e}}, \mathbf{r}$ and $\tilde{\theta}$ are bounded. Since $\mathbf{e}, \dot{\mathbf{e}}, \mathbf{r}$ and $\tilde{\theta}$ are bounded, we can use (11) to show that $\dot{\mathbf{r}}, \ddot{\mathbf{q}}$ and hence $\dot{V}(t)$ in (17) are bounded. Second, note that since $\mathbf{M(q)}$ is lower bounded as delineated by the positive-definite property of the inertia matrix, we can state that $\dot{V}(t)$ given in (14) is lower bounded, we can use Barbalat' lemma (see Chapter 2,[3]) to state that

$$\lim_{t\to\infty} \dot{V}(t) = 0$$

Therefore, from the argument above and (24), we know that.

$$\lim_{t\to\infty} \begin{bmatrix} \mathbf{e} \\ \mathbf{r} \end{bmatrix} = 0 \quad (27)$$

from (27), we can also determine the stability result for the velocity tracking error. Specifically, from (6), note that $\mathbf{r}$ is defined to be stable first-order differential equation in terms of the variable $\mathbf{e}$; therefore, by standard linear control arguments, we can write

$$\lim_{t\to\infty} \dot{\mathbf{e}} = 0 \quad (28)$$

This result informs us that if the controller gains are selected according to (22), (25), and (26), the tracking errors $\mathbf{e}$ and $\dot{\mathbf{e}}$ are asymptotically stable. From the analysis above, all we can say about the parameter error is that it remains bounded. The adaptive controller just derived is summarized in Table 1 and depicted in Fig. 2.

**Table 1** Desired Compensation Adaptation law

| | |
|---|---|
| **Torque Controller** | $\tau = \mathbf{Y}_d(.)\hat{\theta} + \mathbf{k_v}\mathbf{r} + \mathbf{k_p}\mathbf{e} + \mathbf{k_a}\|\mathbf{e}\|^2\mathbf{r}$<br>where<br>$\mathbf{Y}_d(.)\theta = \mathbf{M(q)}\ddot{\mathbf{q}}_d + \mathbf{H}(\mathbf{q}_d, \dot{\mathbf{q}}_d)\mathbf{q}_d$<br>$+ \mathbf{G}(\mathbf{q_d}) + \mathbf{F}_d\dot{\mathbf{q}}_d$<br>$\mathbf{r} = \mathbf{e} + \dot{\mathbf{e}}$ |
| **Update rule** | $\dot{\hat{\theta}} = -\dot{\tilde{\theta}} = \Gamma\mathbf{Y}_d^T(.)\mathbf{r}$ |
| **Stability** | Tracking error $\mathbf{e}$ and $\dot{\mathbf{e}}$ are an asymptotical state. Parameter estimate $\hat{\theta}$ is bounded |
| **Comments** | Controller gain $\mathbf{k_a}, \mathbf{k_p}$ and $\mathbf{k_v}$ must be sufficiently large. |

**Table 2** The Parameter of Desired Compensation Adaptation law

| Symbol | The parameter | The Parameter value of the joint axis |
|---|---|---|
| $\mathbf{q_d}$ | Desired joint position | $\mathbf{q_{d1}} = \mathbf{q_{d2}} = \mathbf{q_{d3}} = 1(t)$<br>$\mathbf{q_{d1}} = \mathbf{q_{d2}} = \mathbf{q_{d3}} = \sin t$ |
| $\mathbf{k_v}$ | Scalar | $\mathbf{k_v} = 250$ |
| $\mathbf{k_p}$ | Constant | $\mathbf{k_p} = 250$ |
| $\mathbf{k_a}$ | Control gain | $\mathbf{k_a} = 250$ |
| $\gamma$ | | $\gamma_1 = 25, \gamma_2 = 25, \gamma_3 = 25$ |
| $\hat{m}$ | Estimated volume | $\hat{m}_1 = \hat{m}_2 = \hat{m}_3 = 0$ |

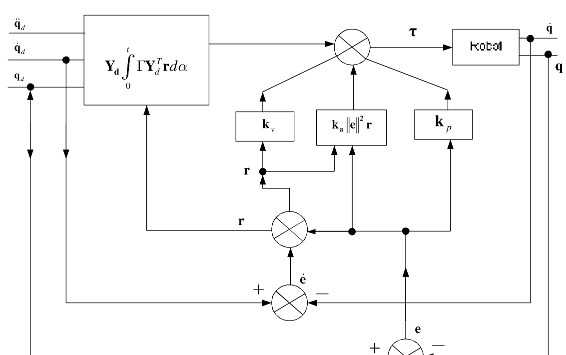


**Fig. 2.** Structure of the joint space desired compensation adaptation controller

After glancing through Table 1, we can see that as opposed to the adaptive inertia-related controller, the Desired Compensation Adaptation law has the obvious advantage of reduced on – line calculation. Specifically, the regression matrix $\mathbf{Y}_{\mathbf{d}}^{\mathbf{T}}$ depends only on the desired trajectory; therefore, the regression matrix can be off-line. We now present an example to illustrate how Table 1 can use to design an adaptive controller for the Robotic manipulator.

## 4. Desired Compensation Adaptation law for the Three –Link Arm

### *4.1 The problem*

We wish to design and simulate the Desired Compensation Adaptation law given in Table 1 for a three-link arm in Fig. 1. The dynamics for this Robot arm are given in [6]. Assuming that the friction is negligible and the link lengths are exactly known to be of length 1m each, the Desired Compensation Adaptation law can be written as:

$$\boldsymbol{\tau}_1 = \mathbf{Y}_{11}\hat{m}_1 + \mathbf{Y}_{12}\hat{m}_2 + \mathbf{Y}_{13}\hat{m}_3 + \mathbf{k}_v\mathbf{r}_1 + \mathbf{k}_p\mathbf{e}_1 + \mathbf{k}_a\|\mathbf{e}\|^2\mathbf{r}_1$$

$$\boldsymbol{\tau}_2 = \mathbf{Y}_{21}\hat{m}_1 + \mathbf{Y}_{22}\hat{m}_2 + \mathbf{Y}_{23}\hat{m}_3 + \mathbf{k}_v\mathbf{r}_2 + \mathbf{k}_p\mathbf{e}_2 + \mathbf{k}_a\|\mathbf{e}\|^2\mathbf{r}_2$$

$$\boldsymbol{\tau}_3 = \mathbf{Y}_{31}\hat{m}_1 + \mathbf{Y}_{32}\hat{m}_2 + \mathbf{Y}_{33}\hat{m}_3 + \mathbf{k}_v\mathbf{r}_2 + \mathbf{k}_p\mathbf{e}_3 + \mathbf{k}_a\|\mathbf{e}\|^2\mathbf{r}_2$$

(29)

where

$\mathbf{r}_1 = \mathbf{e}_1 + \dot{\mathbf{e}}_1, \mathbf{r}_2 = \mathbf{e}_2 + \dot{\mathbf{e}}_2, \mathbf{r}_3 = \mathbf{e}_3 + \dot{\mathbf{e}}_3$ and

$\|\mathbf{e}\|^2 = \mathbf{e}_1^2 + \mathbf{e}_2^2 + \mathbf{e}_3^2$

In the expression for the control torque, the regression matrix $\mathbf{Y}_{\mathbf{d}}(.)$ is given by

$$\mathbf{Y}_{\mathbf{d}}\left(\ddot{\mathbf{q}}_{\mathbf{d}}, \dot{\mathbf{q}}_{\mathbf{d}}, \mathbf{q}_{\mathbf{d}}\right) = \begin{bmatrix} \mathbf{Y}_{11} & \mathbf{Y}_{12} & \mathbf{Y}_{13} \\ \mathbf{Y}_{21} & \mathbf{Y}_{22} & \mathbf{Y}_{22} \\ \mathbf{Y}_{31} & \mathbf{Y}_{32} & \mathbf{Y}_{33} \end{bmatrix} \quad (30)$$

where $\mathbf{Y}_{11}, \mathbf{Y}_{12}, \mathbf{Y}_{13}, \mathbf{Y}_{21}, \mathbf{Y}_{22}, \mathbf{Y}_{23}, \mathbf{Y}_{31}, \mathbf{Y}_{32}, \mathbf{Y}_{33}$ defined in [5]. Formulating the adaptive update rule as given in Table 1, the associated parameter estimate vector is

$$\hat{\phi} = \begin{bmatrix} \hat{m}_1 \\ \hat{m}_2 \\ \hat{m}_3 \end{bmatrix}$$

With the adaptive update rules

$$\dot{\hat{m}}_1 = \gamma_1\left[\mathbf{Y}_{11}\mathbf{r}_1 + \mathbf{Y}_{21}\mathbf{r}_2 + \mathbf{Y}_{31}\mathbf{r}_3\right]; \dot{\hat{m}}_2 = \gamma_2\left[\mathbf{Y}_{12}\mathbf{r}_1 + \mathbf{Y}_{22}\mathbf{r}_2 + \mathbf{Y}_{32}\mathbf{r}_3\right]$$

$$\dot{\hat{m}}_3 = \gamma_3\left[\mathbf{Y}_{13}\mathbf{r}_1 + \mathbf{Y}_{23}\mathbf{r}_2 + \mathbf{Y}_{33}\mathbf{r}_3\right]$$

(31)

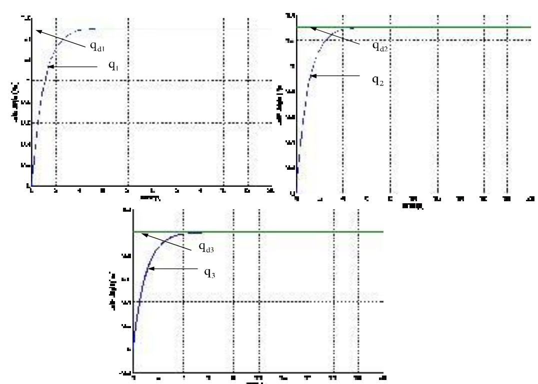


**Fig. 3.** Desired Compensation Adaptation controller with steady-state position error eliminated

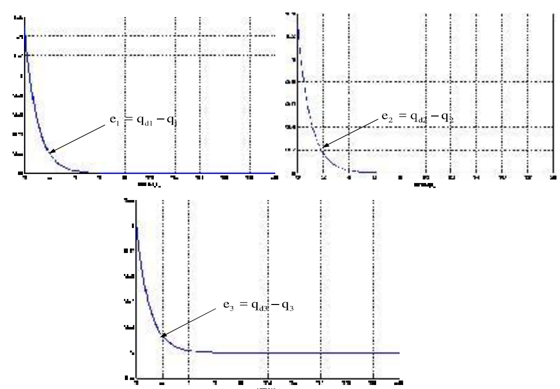


**Fig. 4.** Desired Compensation Adaptation controller with the errors between joints angles

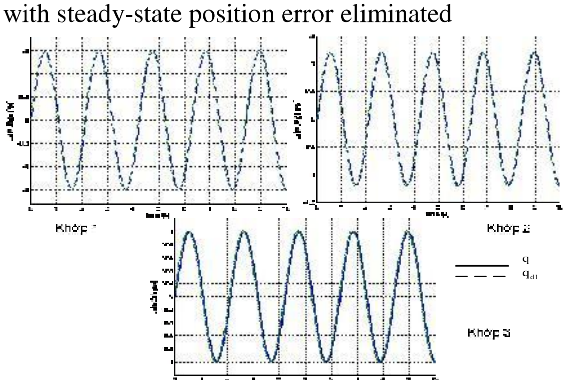


**Fig. 5.** Desired Compensation Adaptation controller with steady-state position error eliminated

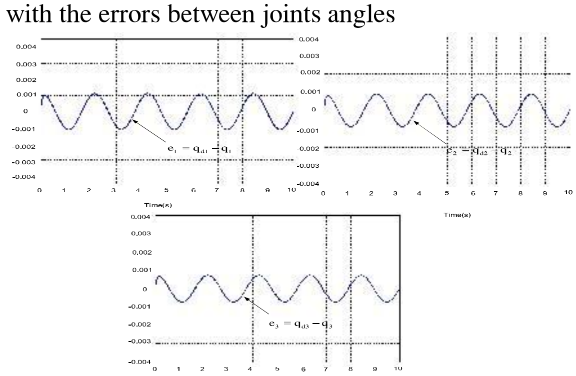


**Fig. 6.** Desired Compensation Adaptation controller with the errors between joints angles

### *4.2 Simulation*

Afer building up the algorithms and control programs, we will proceed to run the simulation program to test computer program. the Desired Compensation Adaptation law was Simulink with Table 2.

After simulation we have results position and position tracking error is depicted Fig. 3, Fig. 4, Fig. 5, Fig. 6.

#### *4.2.1 Desired joint position is 1(t)*

Comment: the robot Almega 16 motion has meet controlled requirements: Steady – state error of joint angle conveges to zero very fast with transient time is small.

#### *4.1.2 Desired joint position is sin(t)*

Comment: The desired trajectory and the real trajectory of the Almega 16 robot have a small error and the transition period of the system is very fast, the mean position error of the total three joints was very low (~ 0.002%).

As illustrated in the Fig. , the position tracking error is both asymptotically stable. Parameter estimate $\hat{j}$ is bounded. Controller gain $\mathbf{k}_a$ , $\mathbf{k}_p$ and $\mathbf{k}_v$ must be sufficiently large.

## 5. Conclusion

As research in robot control has progressed over the last couple of year, many robot controls began to focus on implementation issues. That is, implementation concerns, such as the reduction of on-line computation is causing the researcher to rethink the previous theoretical development of robot controllers so that these concerns are addressed. This paper addresses the problem of re-proofing the desired compensation adaptive control for Robot Almega 16 Robot. The desired compensation adaptive law to resolve the regression matrix $\mathbf{Y}_d^T$ depends only on the desired trajectory; therefore, the regression matrix can be off-line. Thus the volume of mathematics in the control algorithm to reduce more than controls algorithm to research. The simulation results in software Matlab – Simmechanics shows that the Robot motion has meet controlled requirements: Steady – state error of joint angle converges to zero very fast and transient time is small.

## References


[1] A. K, Robot Arm Dynamics and Control, (1974)

[2] John J. C, Introduction to robotics mechanics and control: Silma, Inc, (1995).

[3] Neil M, Frank LL, Robot Manipulator Control Theory and Practice: Marcel Dekker, (2004).

[4] Bin Y, Nonlinear Adaptive Robust Control: The invitation lecture in the Summer School for Mechatronics in Danang Vietnam (2002).

[5] Lorenzo Sciavico and Bruno Siciliano, Modeling and control of Robot Manipulator: McGraw-Hill Company, (1993).

[6] Ha Vth, Some control solutions aim to improve the quality of industrial hand motions Ph.D. thesis, (2012).