ABSTRACT

In many industrial robots the joint velocity is estimated from position measurements rather than from velocity sensors such as tachometer, which may provide signals with noise. Also in practice, robotic actuators have physical constraints that limit the amplitude of the available torques. Possible problems that could result from the implementation of controllers based on the unlimited available torque assumption include degraded link position tracking and thermal or mechanical failure. Therefore, when designing control strategies for such mechanisms, it is important to take into account those actuator limits.

Most of the previous work on the design of such output feedback controllers with bounded torque inputs has been targeted at the set-point control. From a review of the literature, it seems that the more general tracking control problem has received less attention [1, 2]. In this project, an adaptive controller with proper parameter estimation update law is designed so that the error between the time-varying desired position and actual position of the robot system goes asymptotically to zero. To achieve this, the adaptive controller uses only joint position measurement, estimates the velocity and produces a saturated torque input. With this controller, the local exponential stability for closed loop system is guaranteed.

## Problem Statement

We consider a mechanical system with n degree-of-freedom whose generalized coordinates are q1, q2,…qn. The Lagrange equations describing the motion of the system are

$$\frac{d}{dx}\left(\frac{dL}{d\dot{qr}}\right) - \frac{dL}{dqr} = \tau r \,, r = 1,2, \dots \dots , n. \quad (1)$$

Where$L = T - P$ , $T$ and $P$ are the kinetic energy and potential energy respectively, and $\tau r$ is the generalized force. The kinetic energy can be expressed as:

$$T = \frac{1}{2}\sum_{i,j}^{n} aij\dot{q}\imath\dot{q}\jmath \quad (2)$$

Where $aij$ is a function of the generalized coordinates. In the following, we denote q = [q1 q2 … qn]$^T$, then langrage equation (1 ) can be manipulated to derive:

$$D(q)\ddot{q} + F(\dot{q}, q)\dot{q} + G(q) = \tau \quad (3)$$

When the $n \times n$ matrix D(q) is positive definite and symmetric, and is related to the inertial properties of system, the vector function $F\,(\dot{q}, q)\dot{q}$is in general a nonlinear function of its arguments and $\tau = [\tau 1, \tau 2 \dots \tau n]^T$.

In the following development we only consider the system with the following two simplifying properties:

Property 1: A suitable definition of $F(q, \dot{q})$ makes the matrix $\left(\dot{D} - 2F\right)$ skew-symmetric. In particular, this is true if the elements of $F(q, \dot{q})$ are defined as:

$$Fij = \frac{1}{2}[\dot{q}^T \frac{\partial Dij}{\partial q} + \sum_{k=1}^{n} (\frac{\partial Dik}{\partial qj} - \frac{\partial Djk}{\partial qi})q\dot{k}]$$

Property 2: There exists a m-vector $\alpha$ with components depending on mechanical parameters (masses, moments of inertia, etc.), such that

$$D(q)\ddot{q} + F(\dot{q}, q)\dot{q} + G(q) = \mathrm{Y}(\mathrm{q}, \dot{\mathrm{q}}, \ddot{q})\alpha$$

Where Y is $n \times m$ matrix of known functions of $(\mathrm{q}, \dot{\mathrm{q}}, \ddot{q})$ and $\alpha$is the m-vector of inertia parameters.

The control objective can be specified as: given desired $\mathrm{qd}, \dot{\mathrm{q}}\mathrm{d}$ and $\ddot{q}d$, which are assumed to be bounded, determine a control law for $\tau$such that q asymptotically converge to qd, satisfying the following conditions :

- All the parameters in the manipulator system (3) are unknown
- Use only joint position measurements
- Bounded torque inputs are used

It is necessary to use a two-link manipulator with the parameters given in previously to verify the control algorithm.

# Introduction

## Dynamics of motion

The dynamics of robot manipulators has been given increasing attention in recent years whereas the kinematic equations express the motion of the robot without consideration of the forces and torques producing the motion, the dynamic equations describes the relationship between force and motion. In the design of robots, the equations of motion are important to be considered. The Euler-Lagrange equations are used to describe the mechanical system subject to holonomic constraints. The Euler-Lagrange approach has a simple derivation of these equations from Newton's Second Law for a one-degree-of-freedom system. In order to determine the Euler-Lagrange equations in a specific situation, one has to form the Lagrangian of the system, which is the difference between the kinetic energy and the potential energy [3].

The Euler-Lagrange equations have several very important properties that can be exploited to design and analyze feedback control algorithms. Among these are explicit bounds on the inertia matrix, linearity in the inertia parameters, and the so-called skew symmetry and passivity properties.

In general, for any system of the type considered, using the Euler-Lagrange equations leads to a system of n-coupled, second order nonlinear ordinary differential equations of the form:

$$\frac{d}{dx}\left(\frac{dL}{d\dot{q}r}\right)-\frac{dL}{dqr}= \tau r \text{ }, r = 1,2, \dots \dots , n.$$

By the number of so-called generalized coordinates that are required to describe the evolution of the system, the order $n$ of the system can be determined. We shall see that the $n$ Denavit-Hartenberg joint variables serve as a set of generalized coordinates for an n-link rigid robot.

**Kinetic Energy for an n-Link Robot**

By considering a manipulator consisting of n links, the linear and angular velocities of any point on any link can be expressed in terms of the Jacobian matrix and the derivative of the joint variables. Since in our case the joint variables are indeed the generalized coordinates, it follows In general form, the kinetic energy of the manipulator can be written in the form of:

$$K = \frac{1}{2}\dot{q}^T M(q)\dot{q}$$

Where M(q) is a dependent symmetric positive definite matrix, the matrix M is called the inertia matrix.

**Potential Energy for an n-Link Robot**

The only source of potential energy is gravity where the potential energy of the i-th link can be computed by assuming that the mass of the entire object is concentrated at its center of mass and is given by:

$$Pi = g^T r_{ci} mi$$

Where g is vector giving the direction of gravity in the inertial frame and the vector $r_{ci}$ gives the coordinates of the center of mass of link i. The total potential energy of the n-link robot is therefore:

$$P = \sum_{i=1}^{n} Pi = \sum_{i=1}^{n} g^T r_{ci} m_i$$

**Equations of Motion**

The kinetic energy is a quadratic function of the vector $\dot{q}$ of the form:

$$K = \frac{1}{2}\sum_{i,j}^{n} dij(q)\dot{q}_\imath\dot{q}_\jmath := \frac{1}{2} q^T \dot{M}(q)\dot{q}$$

Where the $n \times n$ inertia matrix M(q) is symmetric and positive definite for each q∈ $R^n$ and second the potential energy P=P(q) is dependent of $\dot{q}$. So, the Euler-Lagrange equations for such a system can be derived as follows:

$$L = K - P = \frac{1}{2}\sum_{i,j}^{n} dij(q)\dot{q}_\imath\dot{q}_\jmath - P(q)$$

$$\frac{\partial L}{\partial \dot{q_k}} = \sum_j dkj\, \dot{qj}$$

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{q_k}} = \sum_j dkj\, \ddot{q}j + \sum_j \frac{d}{dt} dkj\, \dot{q}j$$

$$=$$

$$\frac{\partial L}{\partial qk} = \frac{1}{2}\sum_{i,j} \frac{\partial dij}{\partial qk}\dot{q}\imath\dot{q}j - \frac{\partial P}{\partial qk}$$

Thus the Euler-Lagrange equation can be written:

$$\sum_j dkj\, \ddot{q}j + \left\{\sum_{i,j} \frac{\partial dkj}{\partial qi} - \frac{1}{2}\frac{\partial dij}{\partial qk}\right\}\dot{q}_\imath\dot{q}j - \frac{\partial P}{\partial qk} = \tau k$$

By interchanging the order of summation and taking advantage of symmetry, it can be shown:

$$\left\{\sum_{i,j} \frac{\partial dkj}{\partial qi} - \frac{1}{2}\frac{\partial dij}{\partial qk}\right\} \dot{q_\imath}\dot{q}\jmath = \sum_{i,j}\frac{1}{2}\left\{\sum_{i,j} \frac{\partial dkj}{\partial qi} + \frac{\partial dki}{\partial qj} - \frac{\partial dij}{\partial qk}\right\} \dot{q_\imath}\dot{q}\jmath$$

So, the Christoffel symbols of the first kind:

$$\mathrm{C_{i,j,k}} := \frac{1}{2}\left\{\frac{\partial \mathrm{dkj}}{\partial \mathrm{qi}} + \frac{\partial \mathrm{dki}}{\partial \mathrm{qj}} - \frac{\partial \mathrm{dij}}{\partial \mathrm{qk}}\right\}$$

Therefore, the matrix form of Euler-Lagrange equations will be:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) = \tau$$

Where the k, j-th element of the matrix $C(q,\dot{q})$ is defined as

$$C_{i,j,k} = \sum_{i=1}^{n} C_{ijk}(q)\, \dot{q}_i$$

$$= \sum_{i=1}^{n} \frac{1}{2}\left\{\frac{\partial dkj}{\partial qj} + \frac{\partial dki}{\partial qj} - \frac{\partial dij}{\partial qk}\right\} q\dot{}_i$$

Therefore, the equations of motion for a two link manipulator can be written as

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) = \tau$$

$M(q)$ : 2x2 symmetric positive definite inertia matrixes

$C(q,\dot{q})\dot{q}$ : 2x1 centripetal and coriolis torques vector

$\tau$ : 2x1 applied torque input vector

$g(q)$ : 2x1 gravitational torque vector

$$\begin{bmatrix} maa & mau \\ mua & muu \end{bmatrix}\begin{bmatrix} \ddot{q1} \\ \ddot{q2} \end{bmatrix} + \begin{bmatrix} h\dot{q2} & h(\dot{q1} + \dot{q2}) \\ -h\dot{q1} & 0 \end{bmatrix}\begin{bmatrix} \dot{q1} \\ \dot{q2} \end{bmatrix} + \begin{bmatrix} Ga \\ Gu \end{bmatrix} = \begin{bmatrix} u1 \\ u2 \end{bmatrix}$$

With

$maa = m1.lc1^2 + m2.(l1^2 + lc2^2 + 2.l1.lc2.\cos(q2)) + I1 + I2$

$mau = mua = m2.(lc2^2 + l1.lc2.\cos(q2)) + I2$

$muu = m2.lc2^2 + l1.lc2.\cos(q2)) + I2$

$h = -m2.l1.lc2.\sin(q2)$

$Ga = (m1.lc1 + m2.l1).(g.\cos(q1)) + m2.lc2.g.\cos(q1 + q2)$

$Gu = m2.lc2.\cos(q1 + q2)$

m1=10, m2=5, l1=l2=1, lc1=lc2=0.5, I1=10/12, I2=5/12.

## Regressor Matrix

For implementing adaptive control, the parameters (link lengths, mass, moment of inertia etc) need to be estimated [3]. So, the dynamic model equation can be re-arranged in following form to give

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) = Y(q,\dot{q},\ddot{q}).\theta$$

$Y(q,\dot{q},\ddot{q})$: n x r Regressor matrix

θ: parameter vector containing the robot and payload parameters.

Let the parameters be:

$\theta_1 = m_2.l_{c2}^2 + I_2$; $\theta_2 = m_2.l_{c2}.l_1$; $\theta_3 = m_1.l_{c1}^2 + m_2.l_1^2 + I_1$; $\theta_4 = m_1.l_{c1} + m_2.l_1$;

$\theta_5 = m_2.l_{c2}$

So, the expressions change to

$maa = \theta_1 + \theta_3 + 2.\theta_2.\cos(q_2)$

$mau = mua = \theta_1 + \theta_2.\cos(q_2)$

$muu = \theta_1$

$h = -\theta_2 .\sin(q_2)$

$Ga = \theta_4.g.\cos(q_2) + \theta_5.g.\cos(q_1 + q_2)$

$Gu = \theta_5.\cos(q_1 + q_2)$

The Regressor matrix is a 2 x 5 matrix with following elements:

$Y(q,\dot{q},\ddot{q}) = [a_{11}\ a_{12}\ a_{13} a_{14} a_{15}$

$a_{21}\ a_{22}\ a_{23}\ a_{24}\ a_{25}]$

$a_{11} = a_{21} = \ddot{q_1} + \ddot{q_2}$ ; $a_{13} = \ddot{q_1}$ ; $a_{14} = g.\cos(q_1)$; $a_{15} = g.\cos(q_1 + q_2)$;

$a_{25} = \cos(q_1 + q_2)$; $a_{23} = a_{24} = 0$; $a_{12} = (2.\ddot{q_1} + \ddot{q_2}).\cos(q_2) - (\dot{q_1}^2 + 2.\dot{q_1}.\dot{q_2}).\sin(q_2)$

$a_{22} = \ddot{q_1}.\cos(q_2) + (\dot{q_1}^2).\sin(q_2)$

## Robot model Properties

For all values of $q, \dot{q}$,x,y,z; the inertia and coriolis matrix satisfy following properties [4,5]:

$$\lambda_{Max}\{M(q)\}\|x\|^2 \geq x^T M(q)x \geq \lambda_{min}\{M(q)\}\|x\|^2,$$

$$\dot{M}(q) = C(q, \dot{q}) + C(q, \dot{q})^T,$$

$$C(x, y)z = C(x, z)y,$$

$$C(x, y + z) = C(x, y) + C(x, z),$$

$$\|C(q, \dot{q})\| \leq k_{C1}\|\dot{q}\|,$$

$$x^T \left[\frac{1}{2}\dot{M}(q) - C(q, \dot{q})\right] x = 0.$$

$\|x\|^2$ = square root $(x^T. x)$ is the norm of vector x; $\lambda_{min}\{A(x)\}$ and $\lambda_{Max}\{A(x)\}$ denote the minimum and maximum Eigen values of a symmetric positive definite matrix A(x).

Residual dynamics satisfy:

$$h(\tilde{q}, \dot{\tilde{q}}) = [M(q_d) - M(q)]\ddot{q}_d$$
$$+[C(q_d, \dot{q}_d) - C(q, \dot{q})]\dot{q}_d + [g(q_d) - g(q)],$$

$$\|h(\tilde{q}, \dot{\tilde{q}})\| \leq c_1\|\dot{\tilde{q}}\| + \frac{\delta\alpha}{\tanh(\alpha\sigma)}\|\tanh(\sigma\tilde{q})\|.$$

$\sigma$ is a strictly positive constant,

$$\begin{aligned} c_1 &= k_{C1}\|\dot{q}_d\|_M, \\ \delta &= k_g + k_M\|\dot{q}_d\|_M + k_{C2}\|\dot{q}_d\|_M^2, \\ \alpha &= 2\frac{k_1 + k_2\|\ddot{q}_d\|_M + k_{C1}\|\dot{q}_d\|_M^2}{\delta}. \end{aligned}$$

## Proposed Controller

The proposed control law [5] is given by

$$\tau = Y(q_d, \dot{q_d}, \ddot{q_d})\hat{\theta} + k_v\, tanh(\tilde{\vartheta}) + k_p tanh(\sigma\tilde{q})\, q_1 \qquad (22)$$

Where $\tilde{q} = q_d -$ q denotes the tracking error;

$k_v = \text{diag}\{k_{v1} \dots . k_{vn}\}$ and $k_p = \text{diag}\{k_{p1} \dots . k_{pn}\}$ positive definite matrices

$\sigma$ Positive constant and estimated velocity $\tilde{\vartheta}$ is obtained from the following nonlinear filter:

**Velocity estimation**

$\dot{x} = -Atanh(\tilde{\vartheta})$

$\tilde{\vartheta} = x + B\tilde{q}$

Implies that: $\tilde{\vartheta} = B\tilde{q} - \int Atanh(\tilde{\vartheta})$

Where $\mathrm{A} = \mathrm{diag}\{\mathrm{a1}, \ldots, \mathrm{an}\}, \mathrm{B} = \mathrm{diag}\{\mathrm{b1}, \ldots, \mathrm{bn}\}$ are positive definite matrices, and $\hat{\theta} \in IRr$ is the estimated parameter vector obtained from an update law [6].

**Parameter Update**

The parameter estimation update is as follows:

$$\hat{\theta} = \Gamma\left[Y^T(q_d, \dot{q_d}, \ddot{q_d})\tilde{q} - \int_0^t \{\dot{Y}^T(q_d, \dot{q_d}, \ddot{q_d})\tilde{q} + \xi Y^T(q_d, \dot{q_d}, \ddot{q_d})\,[tanh(\tilde{\vartheta}) - \tanh(\sigma\tilde{q})]\}\, dt\right]$$

$\Gamma$ is positive definite matrix; $\xi$ is strictly positive constant between $(\xi_{min}, \xi_{max})$.

**Bounded Torque**

$$|\tau_i| < \tau_i^{Max}$$

$\tau_i^{Max}$ refers to the maximum capability of torque provided by the robot actuator for the joint $i$. For robots with revolute joints, the vector of gravitational torque is bounded i.e. $k_g \geq \sup ||g(q)||$ and $0 < \tau_i^{Max} \leq k_g$

**Closed Loop system**

The closed loop system is obtained by replacing the value for $\tau$ in the robot dynamic model equation with the one in the control law above.

The parameter estimation error: $\tilde{\theta} = \theta - \hat{\theta}$ ;$\ddot{\tilde{\theta}} = -\ddot{\hat{\theta}}$.

Therefore, closed loop system equation is as follows

$$\frac{d}{dt}\begin{bmatrix} \tilde{q} \\ \dot{\tilde{q}} \\ \\ \tilde{\vartheta} \\ \tilde{\theta} \\ \\ \end{bmatrix} = \begin{bmatrix} \dot{\tilde{q}} \\ M(q)^{-1}[-C(q,\dot{q})\dot{\tilde{q}} - F_v\dot{\tilde{q}} - K_v\tanh(\tilde{\vartheta}) \\ -K_p\tanh(\sigma\tilde{q}) - h(\tilde{q},\dot{\tilde{q}}) + Y(q_d,\dot{q}_d,\ddot{q}_d)\tilde{\theta}] \\ -A\tanh(\tilde{\vartheta}) + B\dot{\tilde{q}} \\ \Gamma\left[-Y^T(q_d,\dot{q}_d,\ddot{q}_d)\dot{\tilde{q}}\right. \\ \left.+\xi Y^T(q_d,\dot{q}_d,\ddot{q}_d)[\tanh(\tilde{\vartheta}) - \tanh(\sigma\tilde{q})]\right] \end{bmatrix}$$

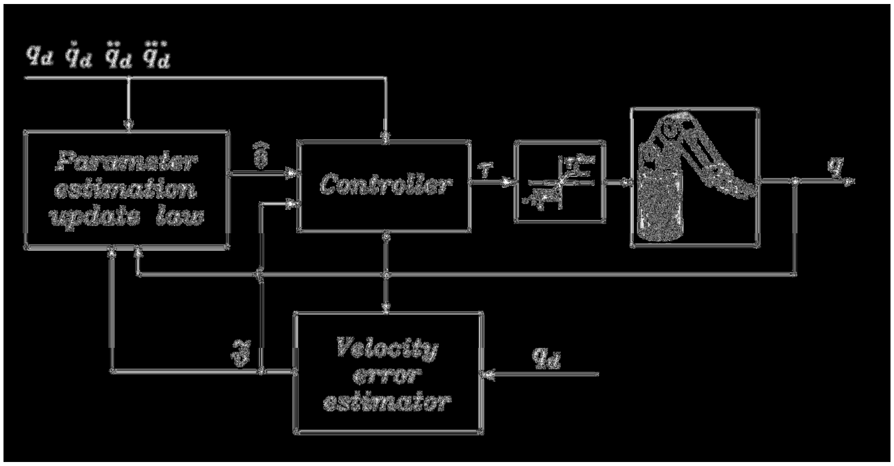
**Lyapunov function**

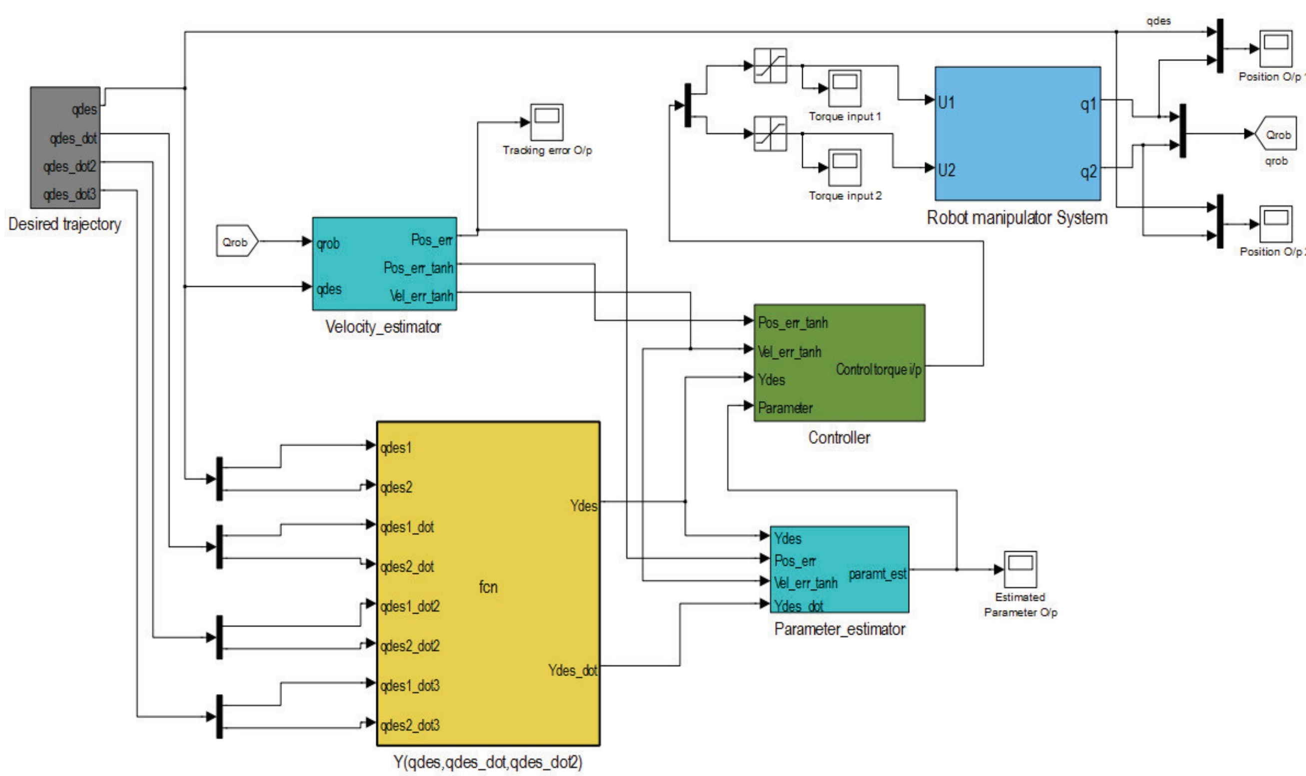
The closed loop system is globally stable in the lyapunov sense for some domain belonging to the three dimensional space. The assumptions considered to prove stability are:

$\lambda_{Min}\{k_p\} > \gamma_1$ and $\xi_{min} < \xi < \xi_{max}$. The values are defined in the appendix.

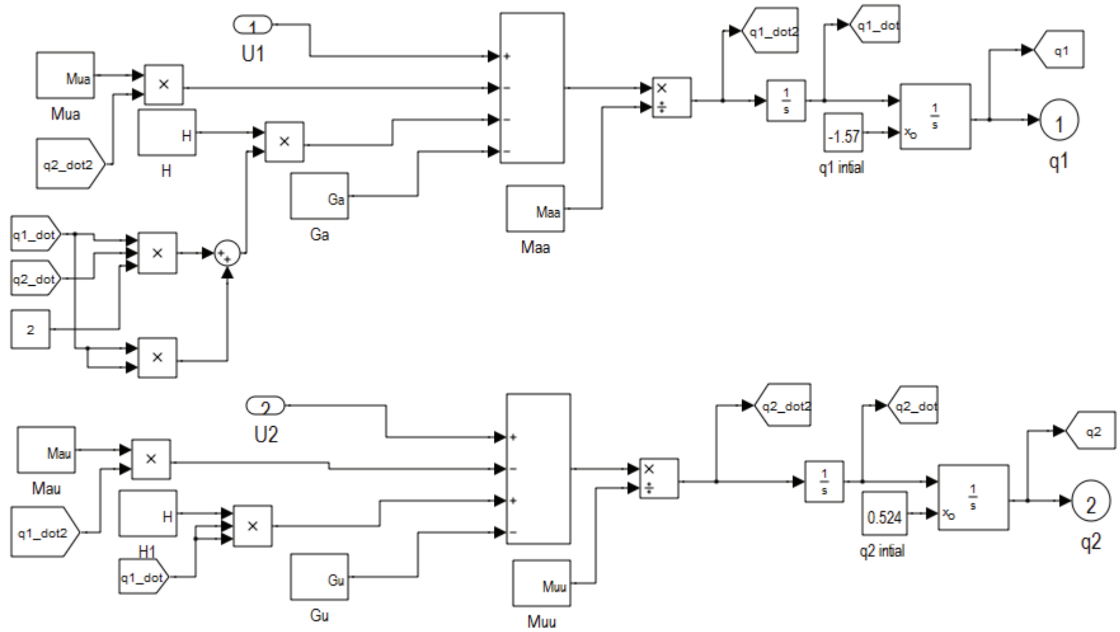
## SIMULINK MODEL

The block diagram for the adaptive OFT controller is as follows

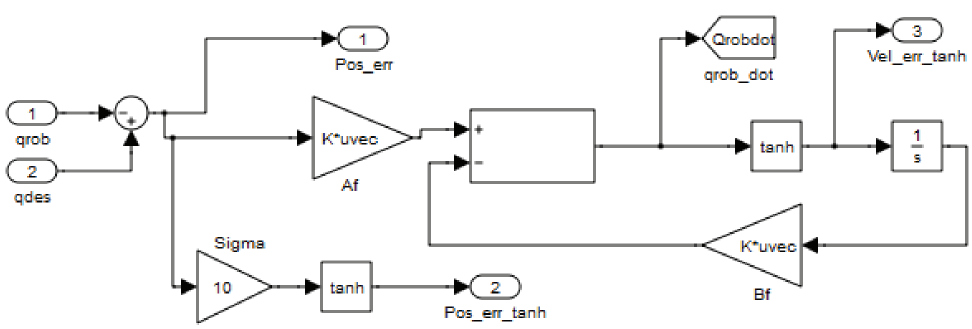


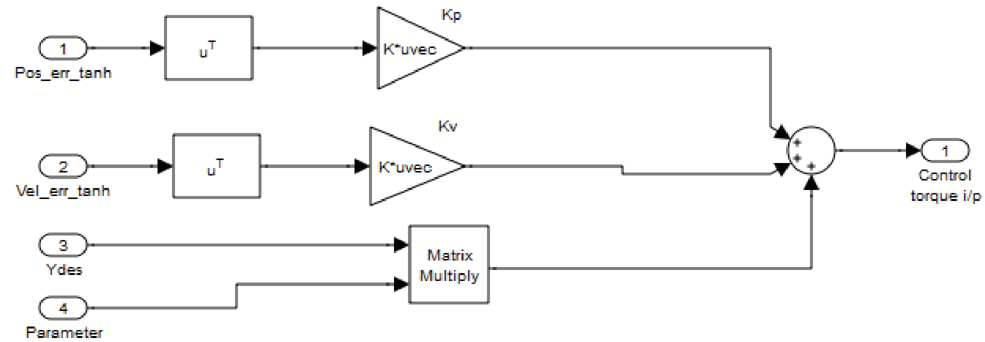
### Robot Manipulator block



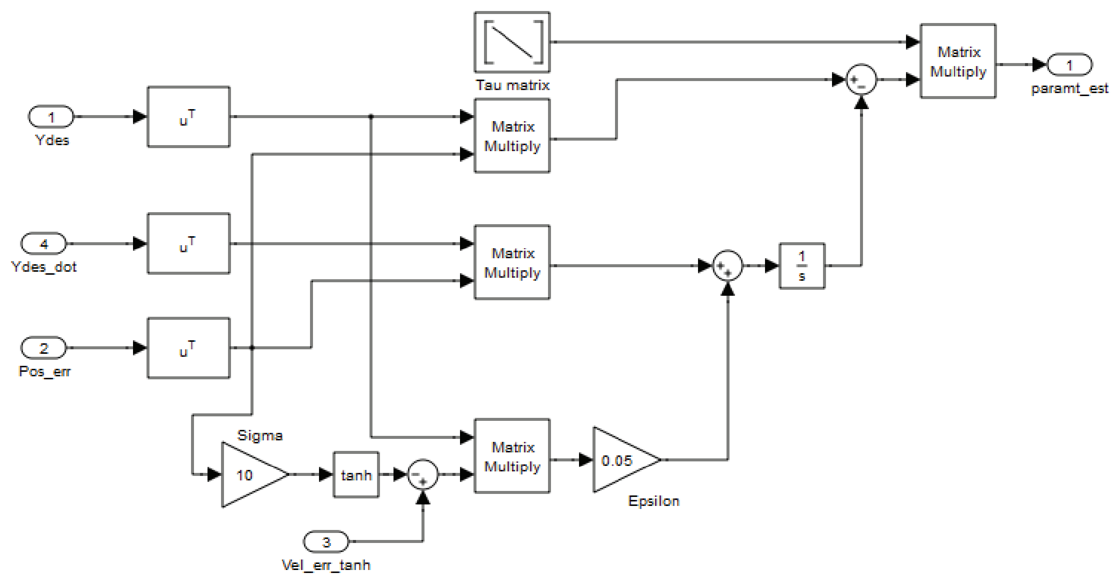

### Velocity Estimator Block



## Controller Block




## Parameter estimator Block

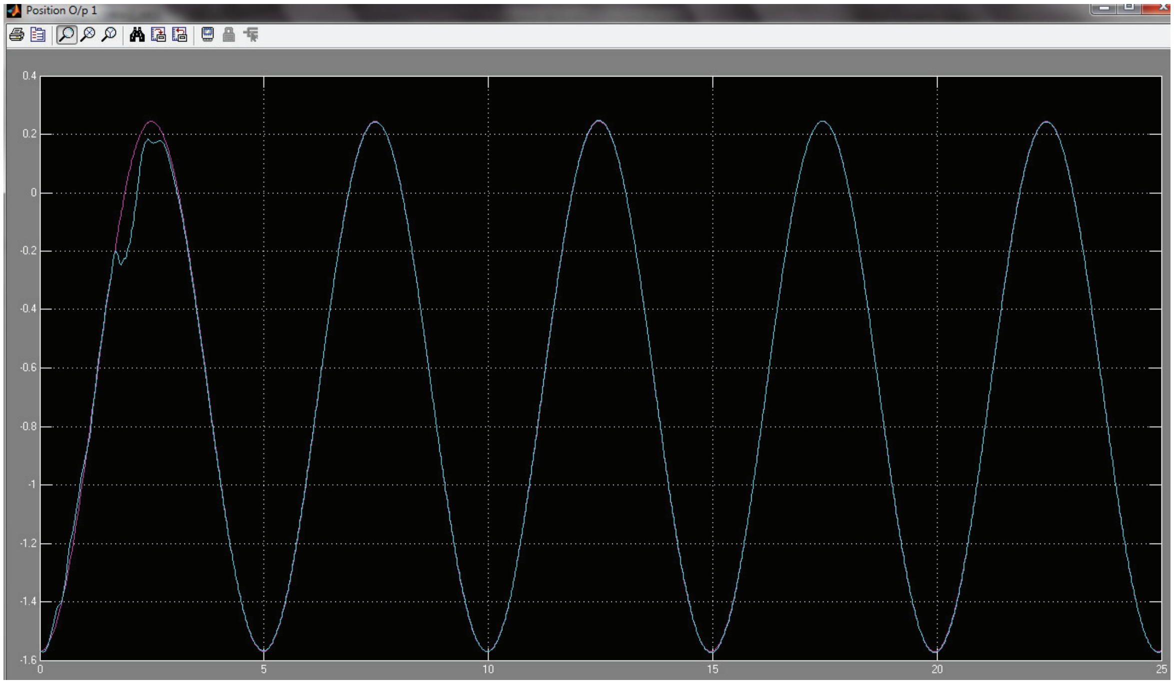


## SIMULATION RESULTS

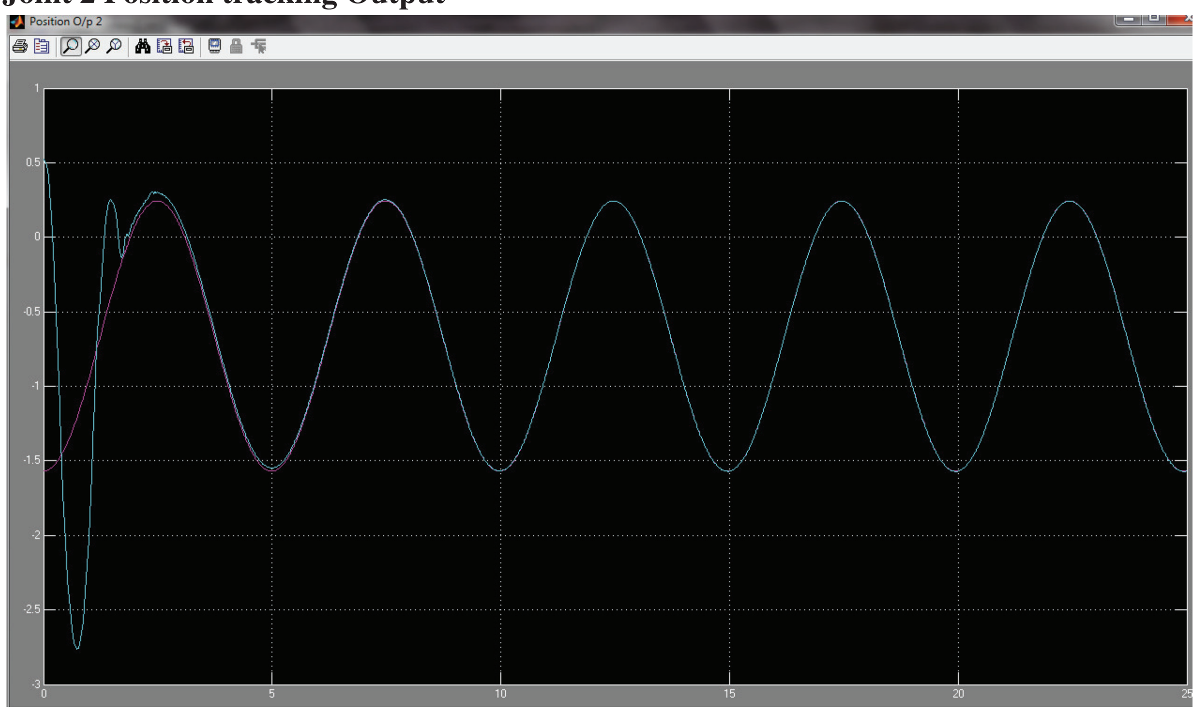
Following values are used for the tuning and limit parameters:

σ =10; $k_v$ = diagonal {250,250}; $k_p$ = diagonal {10, 10}; ξ =0.05; A = diagonal {10, 10}; B = diagonal {10, 10}; Γ = diagonal {1.6,2.5,8,10,2.5}; $\tau_1^{Max} = 150$; $\tau_2^{Max} = 35$;

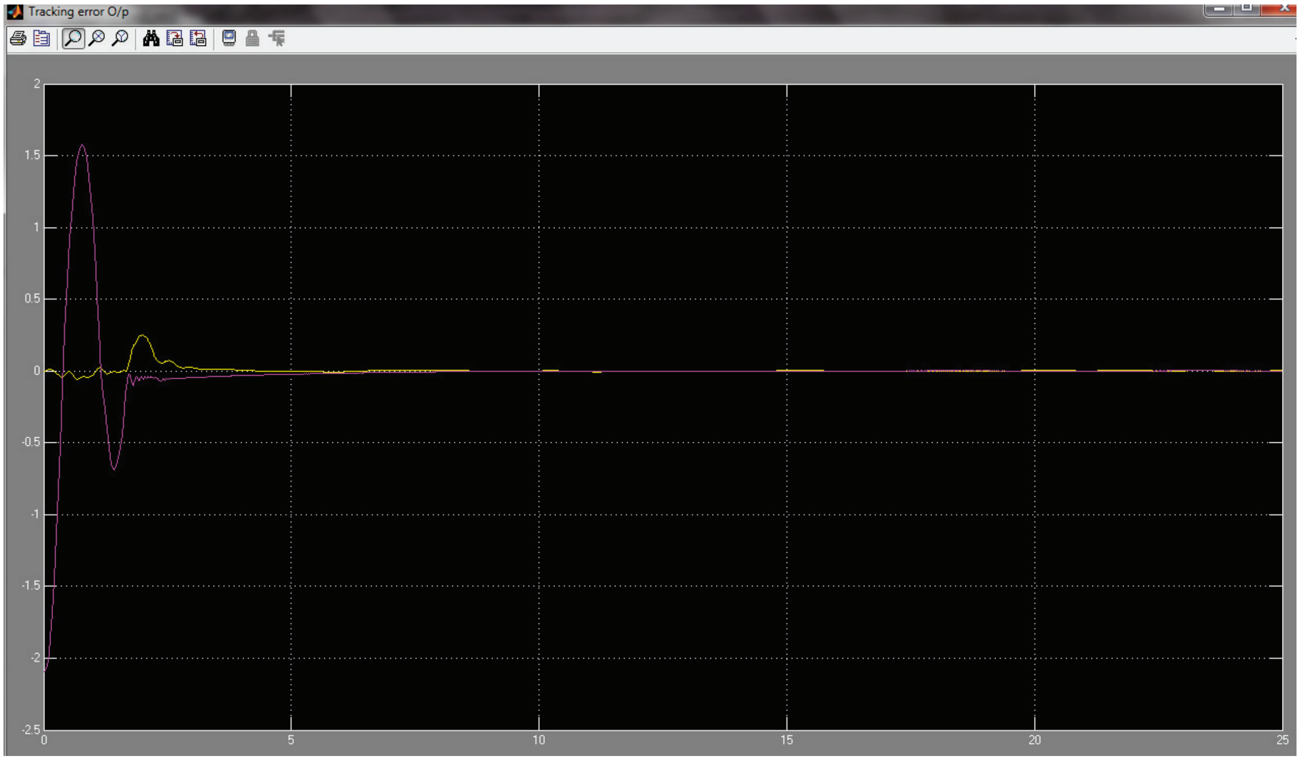
### Joint 1 Position tracking Output



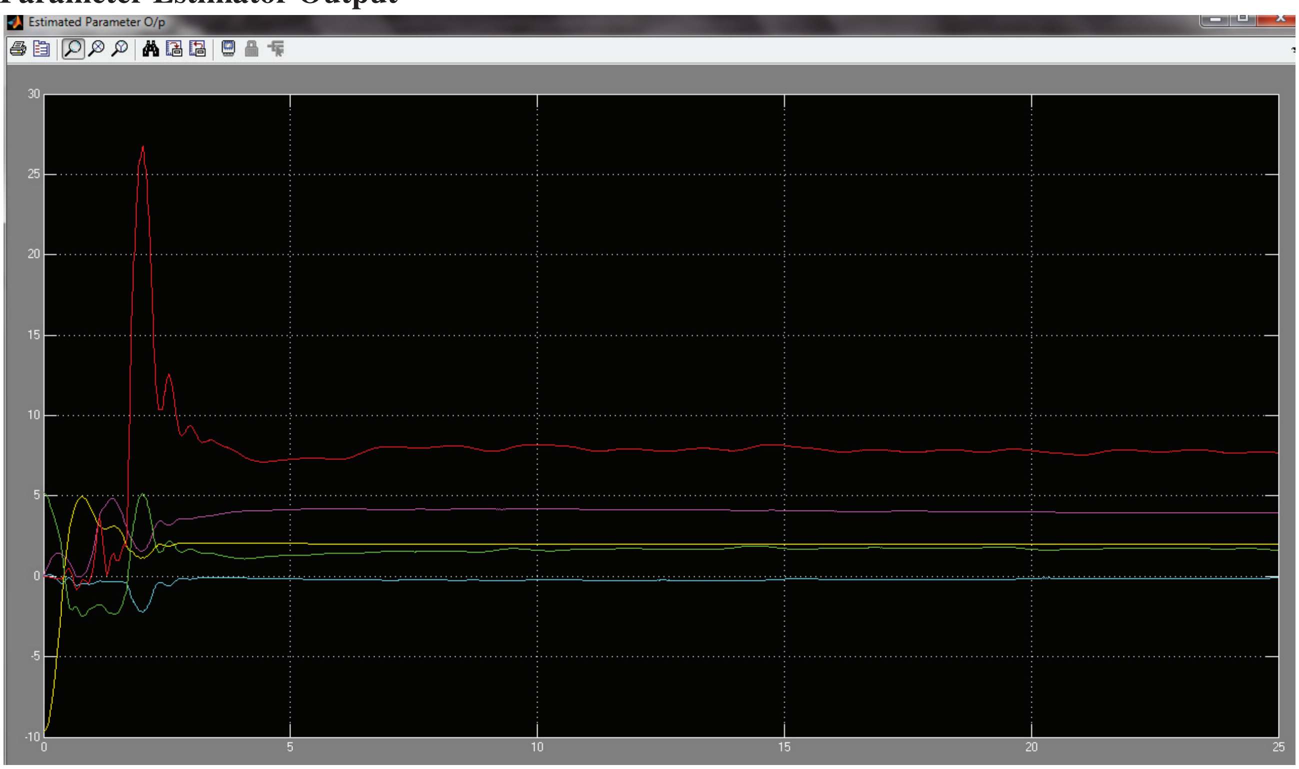

### Joint 2 Position tracking Output



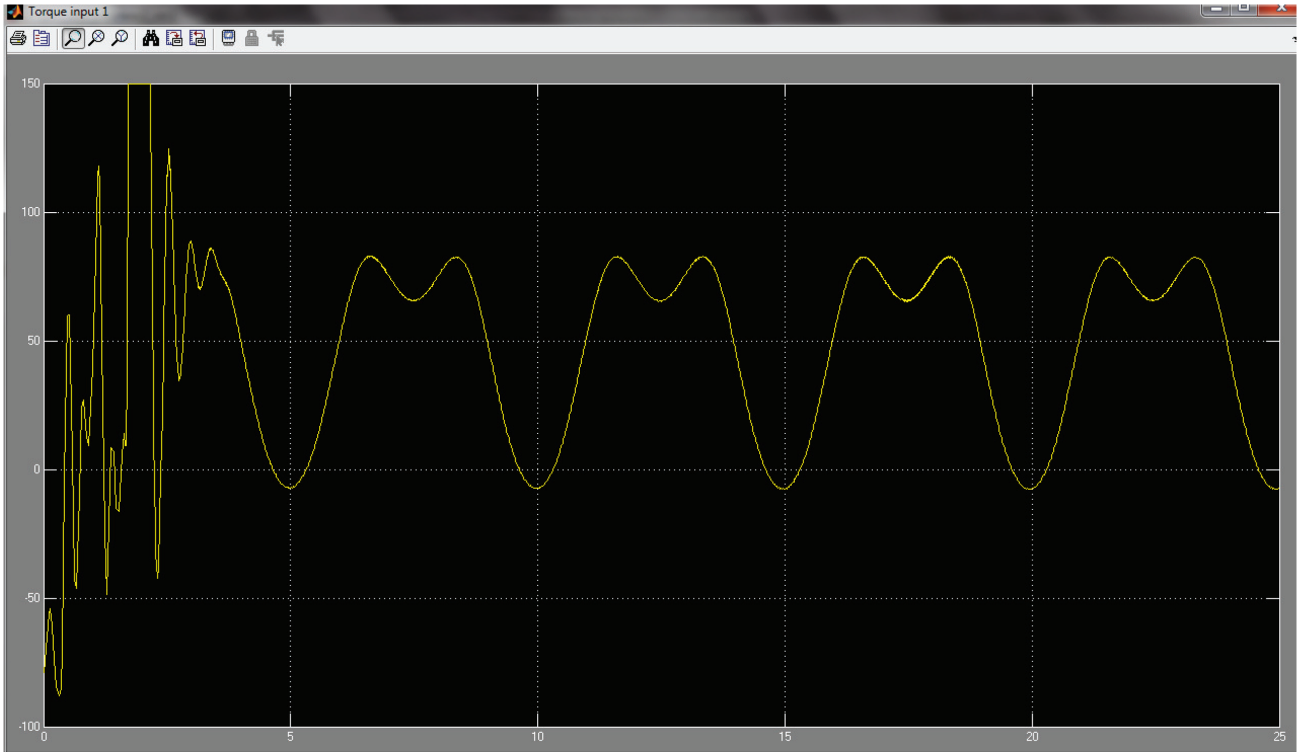
## Tracking error Output



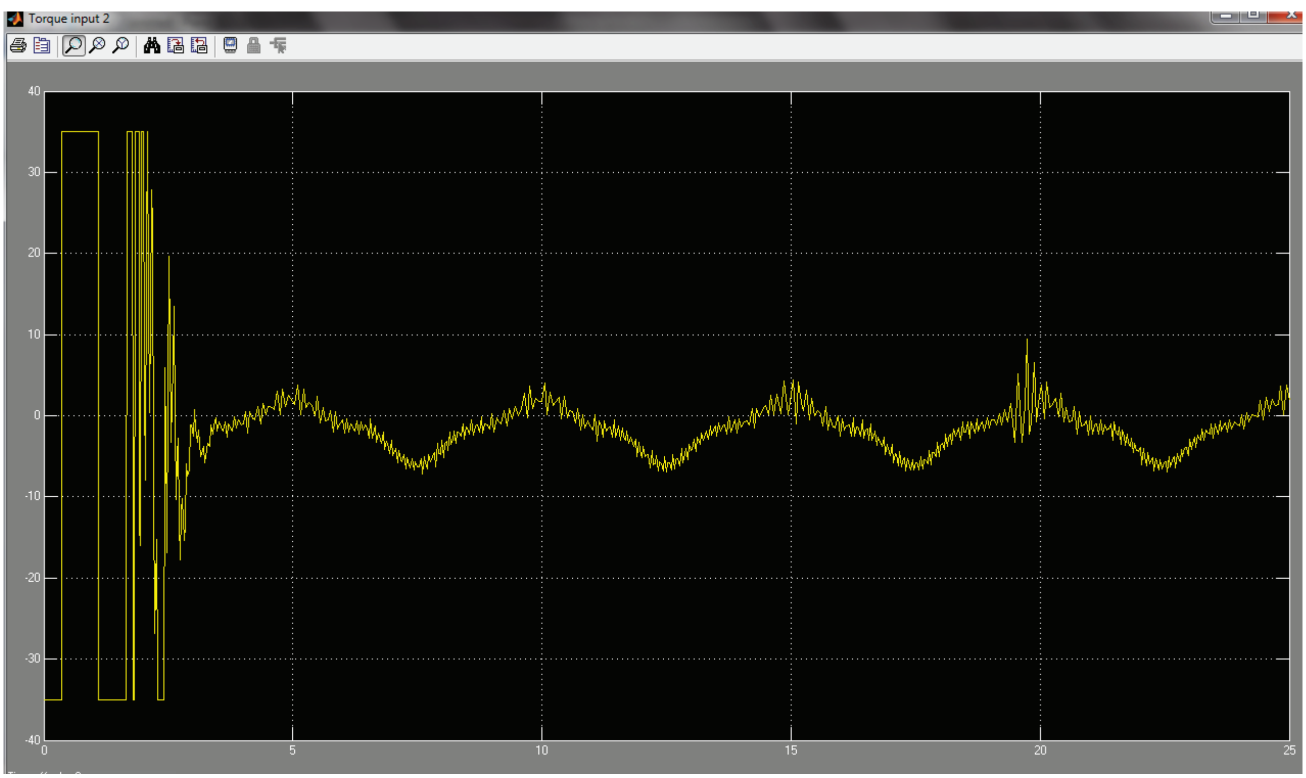

## Parameter Estimator Output



## Control Torque 1 Output




## Control Torque 2 Output



## CONCLUSION

The tracking control for the two link robot manipulator subject to parameter estimation, joint position feedback and constrained torque is studies. The model is successfully implemented in Simulink and the results are provided. The tracking error for the joint position measurements is within 0.001 radians or 0.03 degrees. The settling time is less than 5 seconds for the steady state.

## REFERENCES


[1] M.A. Arteaga, “Robot control and parameter estimation with only joint position measurements”, Automatica, Vol. 39, pp. 67–73, 2003.

[2] F. Zhang, D.M. Dawson, M.S. de Queiroz and W. Dixon, “Global adaptive output feedback tracking control of robot manipulators”, IEEE Transactions on Automatic Control, Vol. 45, No.6 pp. 1203– 1208, 2000.

[3] M.W. Spong and M. Vidyasagar, Robot dynamics and control, John Wiley and Sons, New York, 1989.

[4] L. Sciavicco and B. Siciliano, *Modeling and control of robot manipulators*, Springer, London, 2000.

[5] Javier Moreno-Valenzuela, Víctor Santibáñez, Ricardo Campa, A Class of OFT Controllers for Torque-Saturated Robot Manipulators: Lyapunov Stabilityand Experimental Evaluation, Journal of Intelligent Robot Systems (2008) 51:65–88.

[6] Javier Moreno and Salvador Gonzalez , An Adaptive Output Feedback Tracking Controller for Manipulators Subject to Constrained Torques, Proceedings of the 45th IEEE Conference on Decision & Control, December 13-15, 2006.

# APPENDIX

Matlab Function File for Regressor matrix

```
function [Ydes,Ydes_dot]  = 
fcn(qdes1,qdes2,qdes1_dot,qdes2_dot,qdes1_dot2,qdes2_dot2,qdes1_dot3,qdes2_do
t3)

a11 = qdes1_dot2 + qdes2_dot2;
a12 = (cos(qdes2)*(2*qdes1_dot2+qdes2_dot2)) - 
(sin(qdes2)*(2*qdes1_dot*qdes2_dot));
a13 = qdes1_dot2;
a14 = 9.8 * cos(qdes1);
a15 = 9.8 * cos(qdes1 + qdes2);
a21 = qdes1_dot2 + qdes2_dot2;
a22 = (qdes1_dot2*cos(qdes2))+(qdes1_dot*qdes1_dot* sin(qdes2));
a23 = 0;
a24 = 0;
a25 = cos(qdes1 + qdes2);
Ydes = [a11,a12,a13,a14,a15;a21,a22,a23,a24,a25];

b11 = qdes1_dot3 + qdes2_dot3;
b12 = (cos(qdes2)*((2*qdes1_dot3)+qdes2_dot3-(qdes1_dot*qdes1_dot)-
(2*qdes1_dot*qdes2_dot)))-
(sin(qdes2)*((2*qdes1_dot2)+qdes2_dot2+(2*qdes1_dot)+(2*qdes1_dot*qdes2_dot2)
+(2*qdes1_dot2*qdes2_dot)));
b13 = qdes1_dot3;
b14 = -9.8*sin(qdes1);
b15 = -2*9.8* sin(qdes1 + qdes2);
b21 = qdes1_dot3 + qdes2_dot3;
b22 = (sin(qdes2)*(2*qdes1_dot-qdes1_dot2)) + 
(cos(qdes2)*(qdes1_dot3+(qdes1_dot*qdes1_dot)));
b23 = 0;
b24 = 0;
b25 = -2*sin(qdes1 + qdes2);
Ydes_dot = [b11,b12,b13,b14,b15;b21,b22,b23,b24,b25];
```

To find the value of ξ

The constants $\xi_{min}$ and $\xi_{Max}$ are given by

$$\begin{aligned}\xi_{min} &= \max\{\xi_1, \xi_2\}, \\ \xi_{Max} &= \min\{\xi_3, \xi_4, \xi_5, \xi_6\},\end{aligned}$$

where

$$\xi_1 = \frac{2\gamma_1}{\sqrt{[\lambda_{min}\{K_p\} - \gamma_1][\lambda_{min}\{M(q)\}\lambda_{min}\{B\}\text{sech}^2(d) - 4\gamma_6]}},$$

$$\xi_2 = \frac{4|\lambda_{min}\{F_v\} - c_1|}{\lambda_{min}\{M(q)\}\lambda_{min}\{B\}\text{sech}^2(d) - 4\gamma_2^2},$$

$$\xi_3 = \frac{[\lambda_{min}\{K_p\} - \gamma_1]\lambda_{min}\{K_v B^{-1} A\}}{[\lambda_{min}\{K_p\} - \gamma_1]\lambda_{max}\{K_v\} + \gamma_4^2},$$

$$\xi_4 = \frac{\lambda_{min}\{M(q)\}\lambda_{min}\{B\}\text{sech}^2(d)\lambda_{min}\{K_v B^{-1} A\}}{\lambda_{min}\{M(q)\}\lambda_{min}\{B\}\text{sech}^2(d)\lambda_{max}\{K_v\} + \gamma_5^2},$$

$$\xi_5 = \frac{\sqrt{\sigma^{-1}\lambda_{min}\{K_p\}\lambda_{min}\{M(q)\}/2}}{\lambda_{Max}\{M(q)\}},$$

$$\xi_6 = \frac{\sqrt{\lambda_{min}\{K_v B^{-1}\}\lambda_{min}\{M(q)\}/2}}{\lambda_{Max}\{M(q)\}},$$

$$\begin{aligned}\gamma_4 &= \lambda_{max}\{K_v\} + \lambda_{max}\{K_p\} + \gamma_1, \\ \gamma_5 &= \lambda_{max}\{M(q)\}\lambda_{max}\{A\} + \gamma_2, \\ \gamma_6 &= k_{C1}\sqrt{n} + \gamma_3.\end{aligned}$$