AN INTELLIGENT METHOD TO ESTIMATE THE INERTIA MATRIX OF A ROBOT ARM FOR ACTIVE FORCE CONTROL USING ON-LINE NEURAL NETWORK TRAINING SCHEME

Shamsul Bahri Hussein
Hishamuddin Jamaluddin
Musa Mailah

Department of Applied Mechanics
Faculty of Mechanical Engineering
Universiti Teknologi Malaysia
81310 Skudai

ABSTRACT

This paper presents a new intelligent controller algorithm comprising an on-line multi-layer artificial neural network (ANN) training scheme to estimate the inertia matrix of the robot arm to enhance the performance of the active force control (AFC) scheme. The robot under study is a planar two-link rigid robot which is subjected to a non-linear disturbance torques acting at the robot joints. The algorithm has two stages, namely the ANN training stage and the implementation stage. During the training stage, the proposed ANN scheme trains the ANN parameters (weights and biases) for a period of time by utilising the back-propagation (BP) learning method. After a sufficient training period, the training session is switched off, and the ANN is ready to be used in the implementation stage of the intelligent AFC-ANN controller scheme. The results of the training and implementation stages are shown and discussed. It is shown that *the proposed controller scheme is very effective and robust. The simulation is accomplished using MATLAB® software.*

1.0 INTRODUCTION

Robotic control is an ever-growing research area. The main users of robotic application are in areas of automotive and manufacturing industry, remote and tele-operation manipulation and autonomous system. From all of the application areas stated above, the first one holds the largest growth rate in using robot [1]. The main tasks involving industrial robots are spray painting, welding, material handling and parts assembly operation. In the field of remote and tele-operation manipulation, robots are used to manipulate objects in hostile environments (such as those involving radioactive and corrosive working condition), underwater (submersible) operation and in space exploration. In this type of application, the robot is controlled by a human operator from a remote location. Autonomous robots are used to perform unstructured tasks that require higher level of intelligence and decision making control system. One typical application of autonomous robot is unmanned vehicle for inter-planetary exploration.

Research on intelligent control system for robotic application related to the position and force control, has been carried out for more than two decades. The popular approach for solving the intelligent control system is by implementing ANN to function as a controller itself, or as a part of a controller system, and estimating the model or parameters, architectural design and weight. However, very few papers have studied the combination of both to solve for highly non-linear dynamical system problem such as robot force control.

In robot force control, the two most cited methods are the impedance force control (IFC) as described in [2] and the hybrid position-force control (HPFC) [3]. However, there is a number of drawbacks in the two schemes. The

---

® MATLAB is a trademark of The Math Works Inc performance of the IFC is dependant on how accurate the dynamic model of the robot is achieved. In HPFC, the desired position and force have to be specified by the user and need to be updated when the robot works with different environment (different external force) or when the robot internal frictional force changes. There is yet another robot force control method known as active force control (AFC). AFC operates mainly on measured and estimated parameters, which would definitely lessen the computational burden. Some advantages of the AFC strategy are that it has a fast decoupling property and it can be applied to variable loading conditions. The performance of the AFC depends mainly on how accurate the inertia matrix of the robot arm is estimated. The inertia matrix can be estimated via crude approximation, look-up table, iterative learning technique or artificial neural network algorithm [4].

2.0 THE ROBOT MATHEMATICAL MODEL

The robot used in this work is a two-link rigid planar robot. The configuration of the robot is shown in Figure 1.

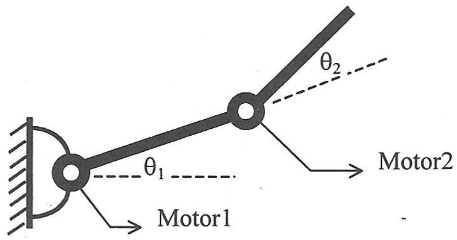

Figure 1 The two-link rigid planar robot configuration

The principles of robot dynamics can be found in many literatures [3, 4 and 5]. The general equation of robot dynamic can be derived by using the Newton-Euler or Lagrange-Euler method, and is described in the following equation:

$$T_q = H(\theta)\ddot{\theta} + h(\theta,\dot{\theta}) + G(\theta) + T_d \tag{1}$$

where

T_q is the vector of actuator torques

$H(\theta)$ is $N \times N$ dimensional manipulator and actuator inertia matrix

$h(\theta,\dot{\theta})$ is the vector of Coriolis and centrifugal torques

$G(\theta)$ is the vector of gravitational torques

T_d is the vector of external disturbances torques

The details of the mathematical model and the mechanical properties of the two-link rigid robot used in this work can be found in [6] and [7]. The robot is restricted to move in the horizontal plane, and is given a task to follow a circular trajectory path with a specified radius. The robot is also subjected to an external disturbance force (T_d) in the form of a sinusoidal function acting as a non-linear type of disturbance. The sum-squared track error (SSE) is monitored to study the performance of the proposed scheme.

3.0 THE ACTIVE FORCE CONTROL (AFC) STRATEGY

AFC is a control method first introduced by Hewit and Burdess [8]. This method is derived from the Newton's second law of motion for a rotating mass, i.e.

$$\sum T = I\alpha \tag{2}$$

where

T is the sum of all torques acting on the body

I is the mass moment of inertia of the rotating mass

α is the angular acceleration

For a robot system which has a serial configuration, the equation of motion becomes

$$T_a + Q = I(\theta)\alpha \tag{3}$$

where

T_a is the applied torque

Q is the disturbance torque

$I(\theta)$ is the mass moment of inertia of the robot arm and is a function of the

joint angle θ.

α is the angular acceleration of the robot arm

The idea of the AFC approach is that, if the value of the disturbance torques can be approximated with an acceptable accuracy, then it could be used to decouple the actual disturbance torque (Q) from the applied torque. This will make the system remains stable even under variable external forces. The estimated disturbance torque can be obtained using the following relationship:

$$Q' = I'\alpha' - T' \tag{4}$$

where the superscript (') denotes a measured or estimated quantity. The applied torque T' can be physically measured using a torque sensor and α' can be measured using an accelerometer. Meanwhile, I' can be obtained by several means such as by the simple crude estimation or by just assuming a perfect model. The more recent development has adopted the iterative learning technique and ANN to estimate the value of I' [6].

Figure 2 shows the schematic diagram of the AFC method applied to a robot arm together with the resolved motion acceleration controller (RMAC), as explained in [7]. There are two types of controllers employed, i.e. the proportional-derivative-RMAC (PD-RMAC) and the AFC. The PD-RMAC is employed to calculate the reference acceleration command vector (θ_{ddref}), which is required for the control signal (I_c).

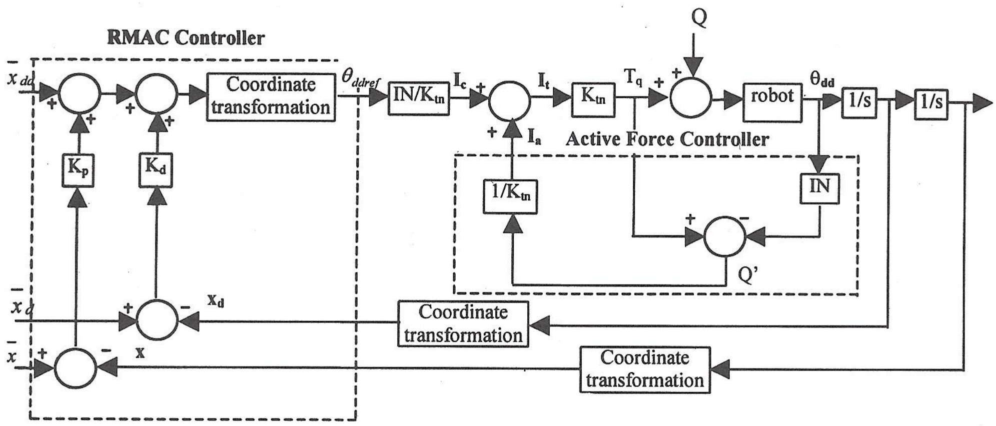

Figure 2 A block diagram of the active force controller applied to the robot force control. The calculation involves the inverse kinematics transformation of the robot arm.

This is fed into the AFC loop where the actual disturbance compensation is taking place assuming that both the acceleration and the torque vectors are suitably measured.

It is shown that the estimated disturbance torque (Q') can be computed by the following equation:

$$Q' = IN\ \theta_{dd} - T_q \tag{5}$$

and

$$T_q = K_{tn}\ I_t \tag{6}$$

where

θ_{dd} is the acceleration signal

IN is the estimated inertia matrix

T_q is the applied control torque

K_{tn} is the motor constant

I_t is the controlled current.

Previous research [7] has incorporated an intelligent technique based on ANN with off-line training, to estimate the inertia matrix (IN). It has been shown that by incorporating the ANN, the inertia matrix can be effectively estimated.

The result is very impressive compared to the other methods of estimation. However, the ANN controller used in the system is trained off-line by using the supervised learning method. The drawback of the off-line training is that the accuracy of the network's generalised solution is really dependent on the accuracy of the training data. On the other hand, if the training is done on-line, where the learning data is extracted from the actual running system, the result of the network generalisation can be improved.

4.0 INTELLIGENT ROBOT CONTROLLER SCHEME INVOLVING NEURAL NETWORK

There are numerous literatures on robotic control using ANN. Some of the works can be found in [6, 9, 10 and 11]. Some implement the ANN as a controller itself as in [10] and others incorporate the ANN as part of the controller system [9] and [11]. Some of the works are explained briefly in the following sections.

4.1 Computed–torque with neural network control (Jung [10])

This scheme implements the computed-torque control approach. Two NN controllers are employed to estimate the inertia matrix $\hat{H}(\theta)$ and the Coriolis, centrifugal, gravitational and disturbance torques $\hat{h}(\theta,\dot{\theta})$. The NN controllers are trained off-line before it is being implemented in the on-line application. During the on-line operation, the NN controller is updated to compensate for the uncertainties in the robot dynamics. Figure 3 shows the block diagram of the NN compued-torque controller scheme.

4.2 Adaptive NN control scheme (Pham and Oh [9])

Another adaptive NN control scheme is presented by Pham and Oh [9]. This scheme is applied for controlling the articulated robot with n joints that carries

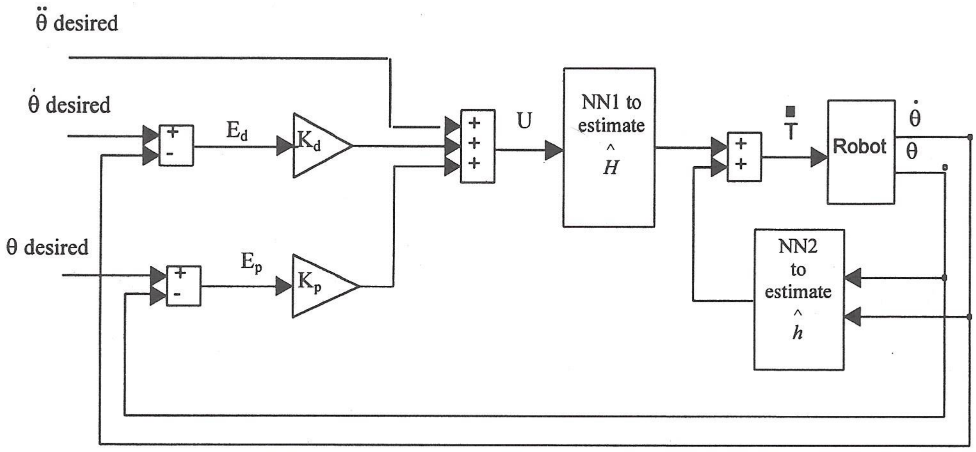

Figure 3 Two NN controllers implemented in computed-torque control strategy for on-line operation

variable load in a MIMO plant. The control system comprises of three neural networks, as depicted in Figure 4. The first neural network, $\hat{\Psi}$ (NN1), is to learn the forward dynamics of the robot, in which, the input to the NN1 is the torque from the controller (u_c) and the output is the estimated position of the robot, $\hat{\varphi}_n$. The second NN, $\hat{\Phi}$ (NN2), is to learn the inverse dynamics of the robot. The input to the NN2 is the actual robot position, φ_n, and the output is the estimated torque. The third NN $\hat{\Phi}^c$ is a copy of NN2 but is used to control the robot. The first and second NNs are trained on-line to give the system the ability to adapt to any changes. In the diagram, $\hat{\Phi}^c$ is a function of the desired position φ_d and $\hat{e}$ is the predicted total error.

The controller input to the robot is given by:

$$u_c = \hat{\Phi}_c + K_f e \tag{7}$$

where

$\hat{\Phi}_c$ is the output of $\hat{\Phi}^c$

K_f is the feedback controller gain

e is the actual position error

The value of $\hat{e}$ is obtained as follows:

$$\hat{e}(k+1) = K_1 e(k) + K_2 [\varphi_d(k+1) - \hat{\varphi}(k+1)] \quad (8)$$

where K_1 and K_2 are suitable constants.

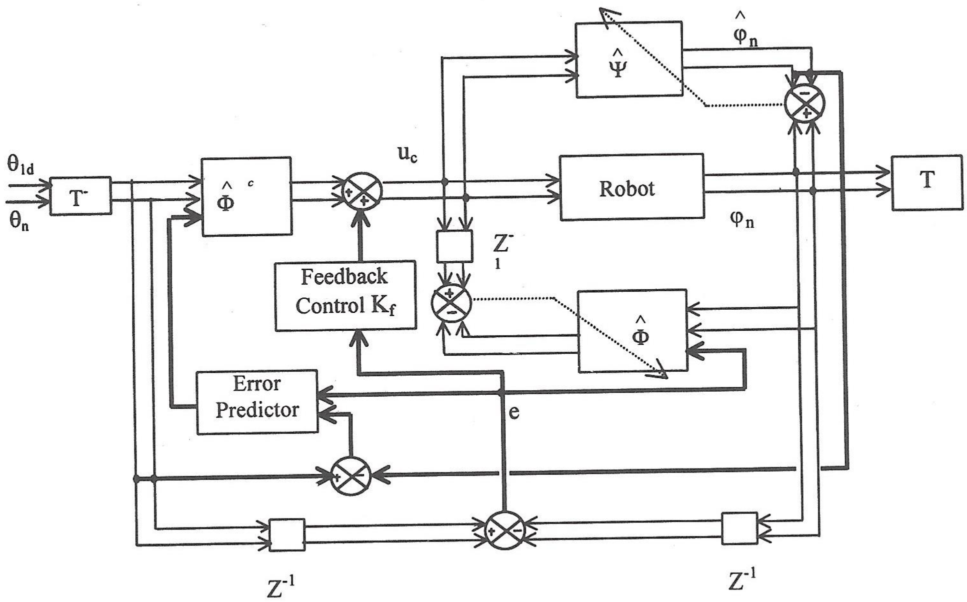

Figure 4 The adaptive NN controller scheme for robot force control

Another interesting feature of this scheme is that it employs the modified Jordan network (MJN). The MJN consists of a recursive network with an additional state layer. The purpose of the state layer is to allow the network to be trained to represent arbitrary dynamic systems.

5.0 THE PROPOSED ON-LINE INTELLIGENT AFC–ANN CONROLLER SCHEME

The proposed on-line intelligent AFC-ANN scheme involves two stages, namely the on-line training stage and the implementation stage. The schematic diagram of the ANN set-up is shown in Figure 5. The dashed line indicates the on-line training scheme, which is activated during training stage only. During this stage, the value of ANN parameters, i.e. weights and biases, are adjusted to reduce the position error of the robot arm. When training is sufficient, the on-line training scheme will be deactivated, and the feed-forward network (FFN) is ready for the implementation stage. The ANN is trained using BP algorithm. It should be noted that the error signal for the training network is not obtained directly from the FFN output, but rather from the actual positional error of the robot arm. By this way, the network can be trained using actual error to be controlled.

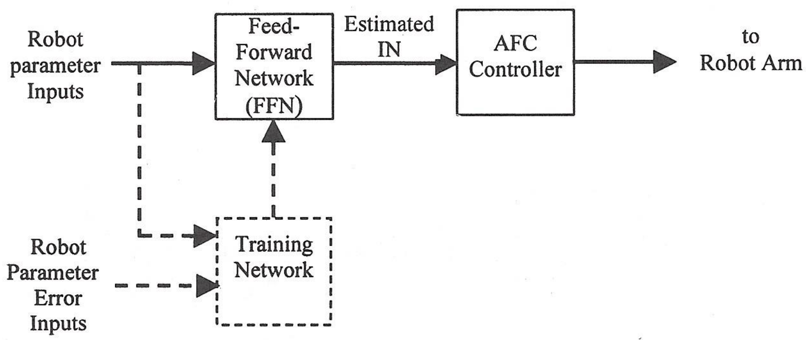

Figure 5 Schematic diagram of the neural network set-up with feed-forward network and training network

5.1 The neural network structure

The structure of the neural network implemented is the multi-layer feed-forward network with single hidden layer and five hidden nodes. The input parameters for the FFN are the angular position of joint 1 and joint 2. The transfer function for all the hidden nodes is log-sigmoid, and for the output nodes positive-linear. The network structure configuration is shown in Figure 6.

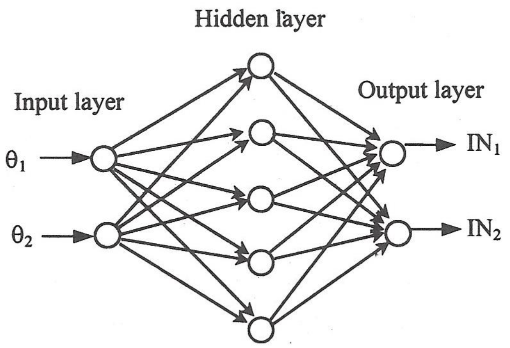

Figure 6 The structure of the multi-layer feed-forward network

Figure 7 shows the complete schematic diagram of the AFC-ANN intelligent controller scheme. An external non-linear disturbance torque (or force) (Q) is introduced to the robot arm end-effector. This is to study the effectiveness of the AFC-ANN controller scheme. The thick line shown in the figure indicates the implementation of the proposed on-line ANN sub-system to estimate the *IN* of the robot arm for the AFC loop.

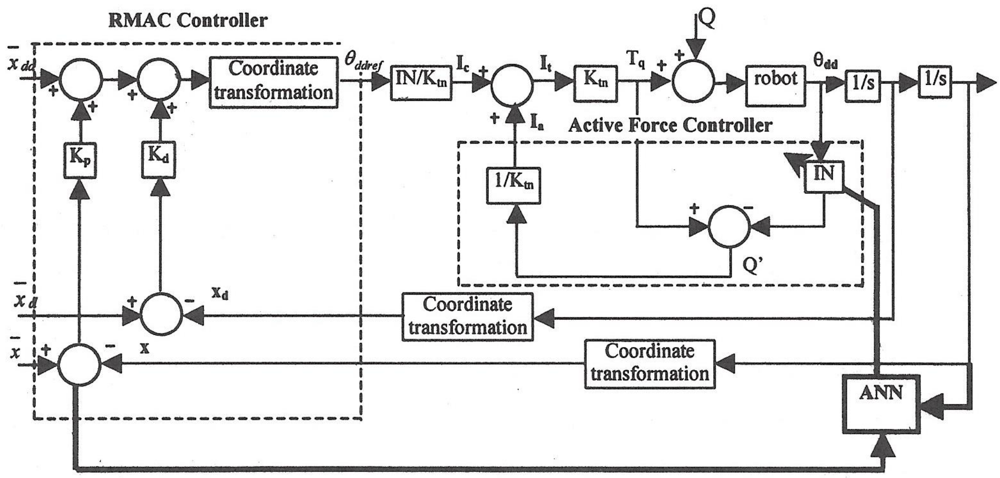

Figure 7 The schematic diagram of the on-line AFC-ANN controller for robot force control

6.0 SIMULATION AND RESULTS

The simulation is carried out using MATLAB software. In the simulation, the robot is commanded to move in a specified trajectory. In the study, the reference trajectory is a circular path with a specified diameter. A continuous non-linear disturbance with specified lower and upper limits is introduced at both the robot joints. In order to ensure that the network always estimate the inertia matrix values within the acceptable range, a lower and upper boundary limits of the network output are set. The learning rate and the momentum rate for the BP method are 0.1 and 0.05 respectively.

6.1 The result of the on-line training stage (Stage 1)

The training session is run for 20 seconds to allow sufficient time for the network to learn. Figure 8 shows result of the tracking error at this stage. As shown in the figure, during the first three seconds of the training period i.e., the first complete circle of the trajectory path, the tracking error is changing very fast. The highest tracking error is 3.3×10^{-3} m at time t = 2 seconds. This shows that the estimation of the inertia matrix by the network is not good enough.

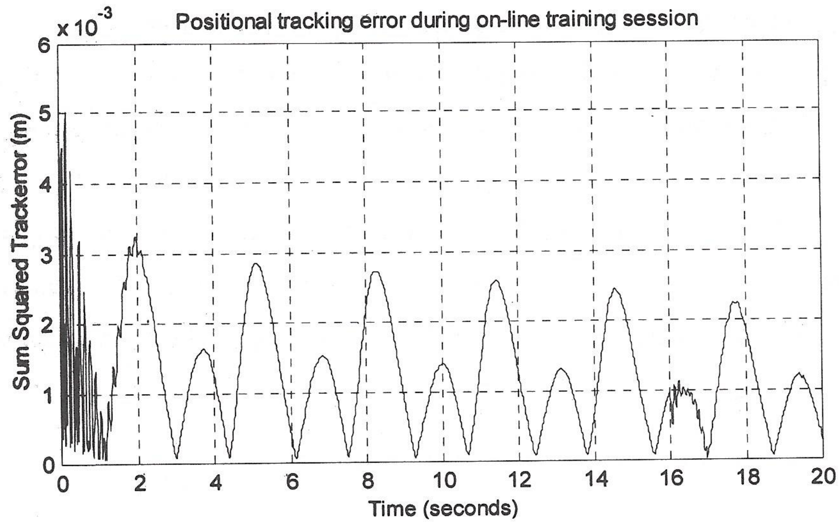

Figure 8 The tracking error result during on-line training stage

However, as time increases, the network keeps on learning until the sum-squared tracking error falls to approximately 2.3×10^{-3} m. This demonstrates that after 20 seconds of training, the estimation of inertia matrix by the network is better. The estimated inertia matrix for both robot arms during the on-line training stage is shown in Figure 9.

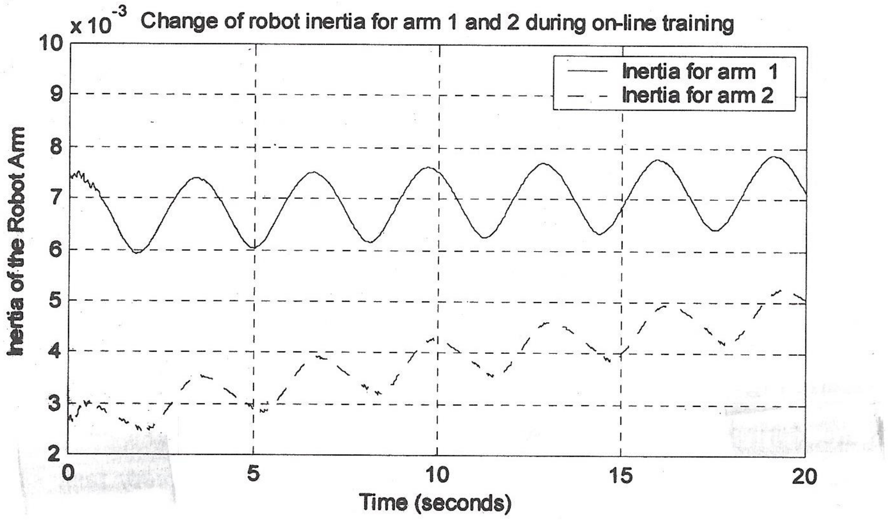

Figure 9 The estimated inertia matrix during on-line training stage

6.2 The result of the on-line implementation stage (Stage 2)

In this stage, the trained network parameters obtained from the on-line training stage is used for the feed-forward network. The simulation is run for five seconds to study the effectiveness of the method. The result of the tracking error is shown in Fig. 10.

As predicted, the result shows a significant improvement compared to the untrained network in Fig. 9. At time t = 2 seconds, the maximum tracking error is 0.6×10^{-3} m compared to 3.3×10^{-3} m considering similar time duration in the training stage.

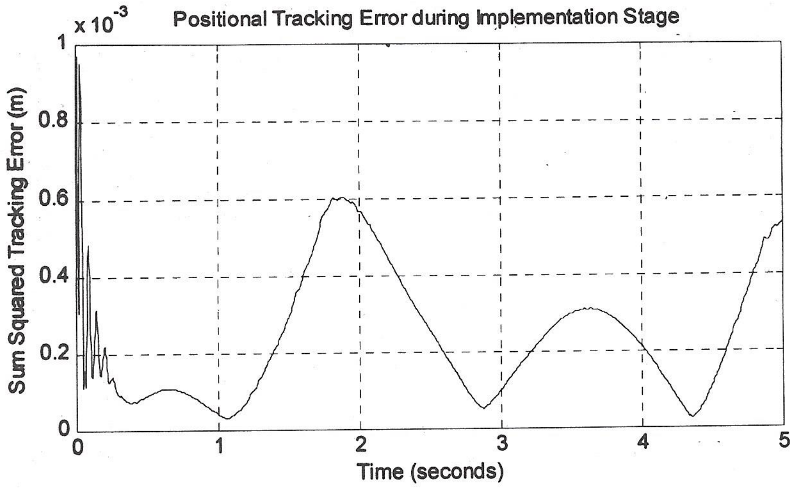

Figure 10 The tracking error result during on-line implementation stage

7.0 DISCUSSION AND CONCLUSION

From the simulation, it is observed that the proposed scheme can learn and react to the external non-linear disturbance torque acting at the robot arms, as long as the disturbance torque does not exceed the limit of the network learning capability and the motors torque themselves. The network learning capability is dependent on the number of hidden nodes and the learning and momentum rate, since more non-linear disturbance torque needs more nodes and different learning and momentum rates in order to obtain a good inertia estimation values.

This work shows the AFC-ANN with the on-line training scheme is capable in controlling the robot arm even under unknown non-linear disturbance torque. Since the training is done on-line, this scheme can be applied to any types of robot configuration.

Since the network structure in this simulation is designed by 'trial and error' method, the chosen network may not be the optimum network design for that particular problem. Thus, it is a challenge to develop an algorithm, which can search for the optimum network design. For future work, other method should be incorporated to the AFC-ANN scheme to solve this problem. One of the possibilities is to incorporate the evolutionary computation method to evolve the network until the optimum network is obtained.

In conclusion, this work shows that the proposed on-line AFC-ANN scheme is capable of becoming a robust intelligent robot controller method.

REFERENCES

1. Malcolm D. R., *Robotics: An Introduction*, PWS-Kent Publishing, 2^{nd} ed., Boston, 1988.
2. Lewis F.L., Abdallah C.T., Dawson D.M., *Control of Robot Manipulators*, Ed. John Griffin, Macmillan Pub., New York, 1993.
3. Lasky T.A. and Hsia T.C., Force Control of Robot Manipulators, *Applied Control*, Ed. S. G. Tzafestas, Marcel Dekker inc., New York, pp.639-661, 1993.
4. Slotine J.J.E and Li W., On the adaptive control of robot manipulators, *Int. J. Robotics Research*, 1987.
5. Asada H. and Slotine J.J.E., *Robot Analysis and Control*, John Wiley and Sons, 1986.
6. Mailah M., *Intelligent Active Force Control Using Neural Network and Iterative Learning Algorithms*, Ph.D Thesis, University of Dundee, 1998.
7. Mailah M., A Simulation Study on the Intelligent Active Force Control of a Robot Arm using Neural Network, *Jurnal Teknologi UTM*, No.30(D), June 1999, pp. 55-78.
8. Hewit J.R. and Burdess J.S., Fast Dynamic Decoupled Control for Robotics Using Active Force Control, *Mechanism and Machine Theory*, Vol. 16, No.5, 1981, 535-542.
9. Pham D. T. and Oh S. J., Adaptive Control of a Robot Using Neural Networks, *Robotica*, vol. 12, 1994, pp. 553-561.

10. Jung S., *Neural Network Controllers for Robot Manipulators*, Ph.D thesis, University of California Davis, 1996.

11. Rana A. and Zalzala A. M. S., A Neural Network Based Collision Detection Engine for Multi-Arm Robotic Systems, *International Journal of Intelligent Control Systems*, Vol. 2, No. 4, 1998, pp. 531-558.