*Jurnal Mekanikal*
*Jun 2002, Bil. 13, 50 - 63*

# INTELLIGENT ACTIVE FORCE CONTROL OF A ROBOTIC ARM USING GENETIC ALGORITHM

Musa Mailah[1]
Wong Min Yee
Hishamuddin Jamaluddin

Faculty of Mechanical Engineering
Universiti Teknologi Malaysia
81310 Skudai, Johor
MALAYSIA
Email: musa@fkm.utm.my

## ABSTRACT

*The main requirement of an active force control (AFC) applied to a dynamical system is the estimation of the inertia matrix, **IN** to compensate for the disturbances and uncertainties in the system. In this paper, genetic algorithm (GA) is used to estimate suitable value of **IN** of a robotic manipulator necessary for the implementation of the AFC strategy through a simulation study. A set of constant torques at the joints is deliberately introduced as the disturbance mechanism to test the effectiveness of the proposed scheme. The results show that the GA used in the study being a stochastic and global optimizer successfully computes appropriate **IN** value to effect the control action. The proposed scheme exhibits a high degree of robustness and accuracy as the track error is bounded within an acceptable range of value even under the influence of the introduced disturbance.*

## 1.0 INTRODUCTION

Robot force control is concerned with the physical interaction of the robot's end effector with the external environment in the forms of applied forces/torques, changes in the mass payloads and constrained elements. A number of control methods has been proposed to achieve stable and robust performance ranging from the classical proportional-derivative (PD) control [1] to the more recent intelligent control technique. A system is said to be robust when the system performs with acceptable degree of accuracy, stability and reliability in the presence of disturbances, parametric uncertainties and varied operating conditions. The PD control is simple, efficient and provides stable performance when the operational speed is low and there are very little or no disturbances. The performance however is severely affected with the increase in speed and presence of disturbances. Adaptive control technique have been proposed [2, 3, 4] and to a certain degree succeeded in overcoming this problem – providing better performance and robustness in a wider range of system operating parameters but at the expense of involving complex mathematical manipulation. The

[1] To whom all correspondence should be forwarded.

implementation of the adaptive control method in real time poses a problem due to the complexity of the models involved and more often than not, most of the works are done through simulation. There is an emerging class of adaptive control methods are increasingly being used in robotic systems [5, 6, 7]. Active force control (AFC) has been demonstrated to be superior compared to the conventional methods [8, 9] in dealing with compensating a variety of disturbances. A distinct advantage about this method is the practical realization of the system in which the method bases its concept on using mainly the estimated or measured values of certain parameters to effect its compensating action. AFC has a fast decoupling property and it can be applied to variable loading conditions.

Since late 1980's, researchers have tried to implement the artificial intelligent methods, i.e., artificial neural network (ANN), evolutionary computation (EC), and fuzzy logic in robot control to either function as a robot controller itself, or as part of the controller system. More recently, some researchers have incorporated genetic algorithm (GA) to control the robot. Some of them incorporate the GA with other classical controller such as PID controller, and some incorporate the GA with the ANN controller. In this paper, a GA-based AFC method is used to control a rigid two-link horizontal planar robotic arm. GA is used to estimate the inertia matrix of a robot arm, which is required in the AFC feed forward loop. The effectiveness of this scheme to compensate external disturbances is studied from the track error plotted. We called the scheme AFCAGA – an acronym for *Active Force Control And Genetic Algorithm.*

This paper is organized as follows. Section 2 presents a description of the problem statement. Sections 3 and 4 cover the fundamentals of both the AFC and GA. The dynamic model or the general equation of motion of a robot manipulator is described in Section 5. The integration of GA and AFC is applied to the manipulator and subsequently, the simulation results are studied and discussed in Section 6. Finally, the conclusions are given in Section 7.

## 2.0 PROBLEM STATEMENT

AFC is a force control strategy originated in [8, 10] and is primarily designed to ensure that a system remains stable and robust even in the presence of known or unknown disturbances. In AFC, the system mainly uses the estimated or measured values of a number of identified parameters to effect its compensation action. In this way, we can reduce the mathematical complexity of the robotic system, which is known to be highly coupled and non-linear. However, the main drawback of AFC is the acquisition of the estimated inertia matrix that is required by the AFC feed-forward loop. Previous methods rely heavily on either perfect modeling of the inertia matrix, crude approximation or the reference of a look-up table, which obviously require prior knowledge of the estimated inertia matrix. Although the methods are quite effective to implement, they lack in systematic approach and flexibility to compute the inertia matrix. Thus, a search for better ways to generate efficiently suitable estimated inertia matrix is sought. If a suitable method of estimating the inertia matrix can be found, then the practical value of implementing AFC scheme is considerably enhanced. Obviously, intelligent methods are viable options and should be exploited to achieve the objective as already described in [11, 12]. While there are some other adaptive techniques used to solve this difficulty, we propose yet another strategy, which is simple, effective and globally optimum, to be incorporated into the AFC method to control the robot arm. The learning approach applied is through the use of genetic algorithm. In this method, the inertia matrix (IN) of the arm in the AFC controller is estimated automatically via GA as the arm is commanded to execute a prescribed task accurately even in the presence of disturbances.

## 3.0 ACTIVE FORCE CONTROL (AFC)

The full mathematical analysis of the AFC scheme can be found in [8, 13]. It has been shown that disturbances can be effectively eliminated via the compensating action of the AFC strategy. Figure 1 shows a schematic of AFC scheme applied to control a robot arm.

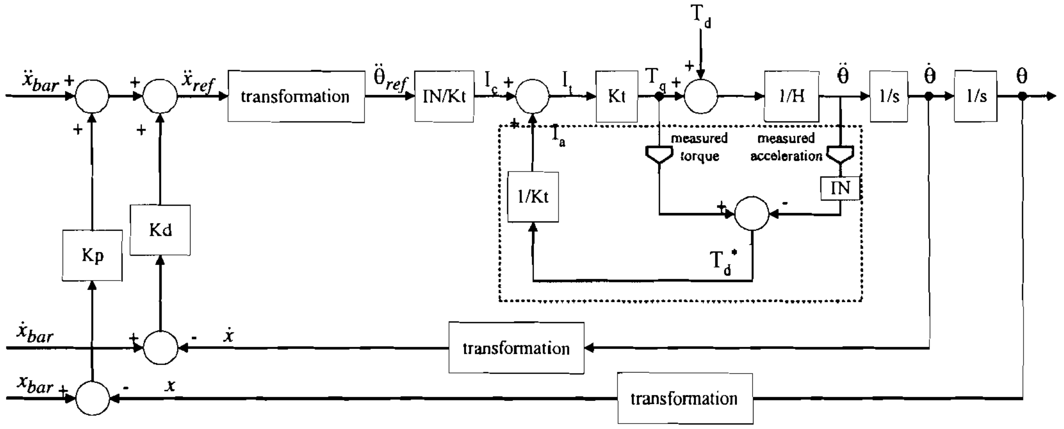


Figure 1 The AFC scheme applied to a robot arm

The notation used in Figure 1 is as follows:

| | |
|---|---|
| $\theta$ | vector of positions in joint space |
| $K_p$, $K_d$ | PD controller gains |
| $K_t$ | motor torque constant |
| $I_a$ | current command vector |
| $I_a$ | compensated current vector |
| $I_t$ | armature current for the torque motor |
| **IN** | estimated inertia matrix |
| $T_d$* | estimated disturbance torque |
| $T_q$ | applied torque (measured) |

$x$, $x_{bar}$ vectors of actual and desired positions respectively in *Cartesian* space

$\ddot{\theta}_{ref}$ , $\ddot{x}_{ref}$ reference acceleration vectors in joint and *Cartesian* spaces

In AFC, it is essential that we obtain the physical measurements of the acceleration ($\ddot{\theta}$) of the arm and the actuated torque ($T_q$) using accelerometer and torque sensor respectively as can be seen in Figure 1. Next, the estimated inertia matrix of the arm (**IN**) has to be appropriately identified by suitable means. In this way, we could estimate the disturbances based on the measured or estimated values of the variables and could be expressed as follows:

$$T_d{}^* = T_q - \mathbf{IN}\,\ddot{\theta} \tag{1}$$

Equation 1 can be further simplified as

$$T_d{}^* = K_t\, I_t - \mathbf{IN}\,\ddot{\theta} \tag{2}$$

where

$$T_q = K_t\, I_t \tag{3}$$

In this case, instead of measuring the torque directly, we measure the torque current $I_t$ and then multiply this value with the torque constant $K_t$ which of course gives the value of the required actuated torque. While the measurement part is obvious, the inertia matrix can be obtained using a number of methods such as crude approximation, reference of a look-up table or intelligent method. Note that the arm is assumed to operate horizontally; hence we consider only the diagonal elements of the estimated inertia matrix **IN** and that for convenience we denotes these as $IN_{11}=IN_1$ and $IN_{22}=IN_2$. The off-diagonal terms $IN_{12}$ and $IN_{21}$ are disregarded, i.e., $IN_{12}=IN_{21}=0$, since it has been shown that this coupling term can be safely ignored by AFC strategy [8].

In addition to the above, we include a PD controller employing resolved motion acceleration control (RMAC) as described in [13] which can improve the overall performance of the control scheme. RMAC is governed by the following equation:

$$\ddot{x}_{ref} = \ddot{x}_{bar} + K_d\,(\dot{x}_{bar} - \dot{x}) + K_p\,(x_{bar} - x) \tag{4}$$

In AFC, it is shown that a robotic system subjecting to disturbances remains stable and robust through the compensating action of the control strategy. In other word, the system remains stable in the presence of "noises". The main computational burden in AFC is the multiplication of the estimated inertia matrix (**IN**) with the angular acceleration of the arm before being fed into the AFC feed-forward loop. Apart from that, the output of the system, e.g., *Cartesian* position needs to be computed from the joint space via forward kinematics and also the controller prior to the AFC loop is determined. Knowing that the performance of the AFC depends mainly on how appropriate the inertia matrix of the robot arm is estimated, thus in this paper, the estimation of inertia matrix through the use of genetic algorithm is attempted. A brief but adequate description of the theoretical background of GA and its application to robot control will be given in the following section.

## 4.0 GENETIC ALGORITHM (GA)

Genetic algorithm search method is rooted in the mechanism of evolution and natural genetics. The interest in heuristic search algorithms with underpinnings in natural and physical processes began as early in 1970s, when Holland first proposed the concept of genetic algorithm [14]. Genetic algorithms generate a sequence of populations using selection and search mechanisms involving the process of crossover and mutation.

### 4.1 Structure and Mechanism of Genetic Algorithm

Genetic algorithm operates on a population of potential solutions applying the principle of survival of the fittest to produce better and better approximations to a solution. At each generation, a new set of approximations is created by the process of selecting individuals according to their level of fitness in the problem domain and breeding them together using operators borrowed from natural genetics. This process leads to the evolution of populations of individuals that are better suited to their environment than the individuals that they were created from, just as in natural adaptation. As illustrated in Figure 2, at the beginning of the computation, a number of individuals (population) is randomly initialized. The objective function is then evaluated for these individuals. The first/initial generation is produced. If the optimization criteria are not met, the creation of a new generation starts. Individuals are selected according to their fitness for the production of offspring. Parents are recombined to produce offsprings. All offsprings will be mutated with a certain probability. The fitness of the offspring is then computed. The offsprings are inserted into the population replacing the parents, producing a new generation. This cycle is performed until the optimization criteria are reached.

### 4.2 Genetic-Based Active Force Control

As mentioned earlier, GA is applied to estimate the inertia matrix of a robot arm in the control loop. Based on the information of the track error (denoted by *e*), GA is applied to estimate IN and fed it again into the AFC loop as can be observed in Figure 3. The cycle will be continued until a set of appropriate IN value is obtained.

Figure 4 shows the proposed AFCAGA control scheme and how the GA component is embedded into the control strategy as the IN estimator. The box (dashed-line) represents the essence of the AFC mechanism.

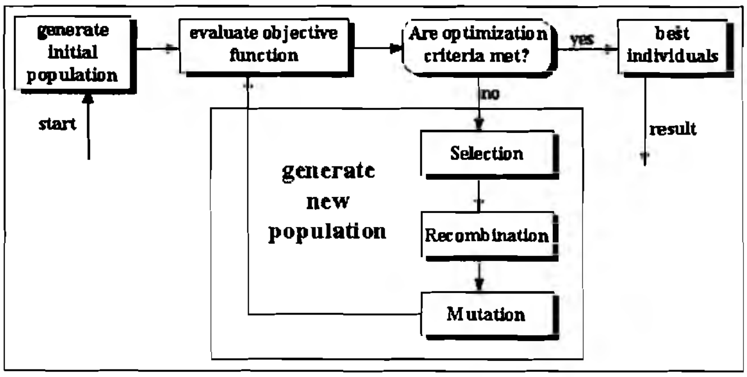


Figure 2 Structure of a single population evolutionary algorithm

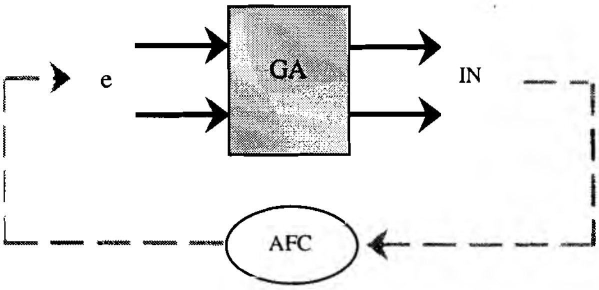


Figure 3 GA for optimizing the inertia matrix, IN

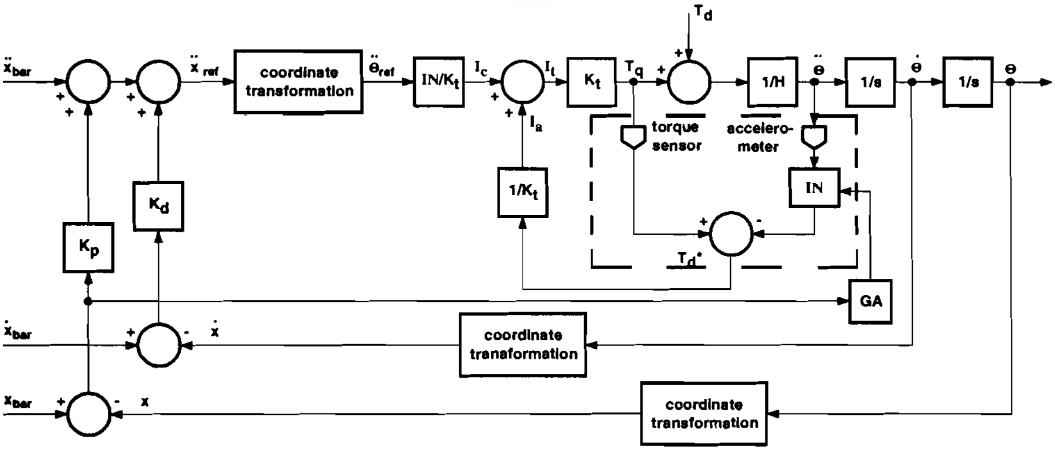


Figure 4 AFCAGA control scheme

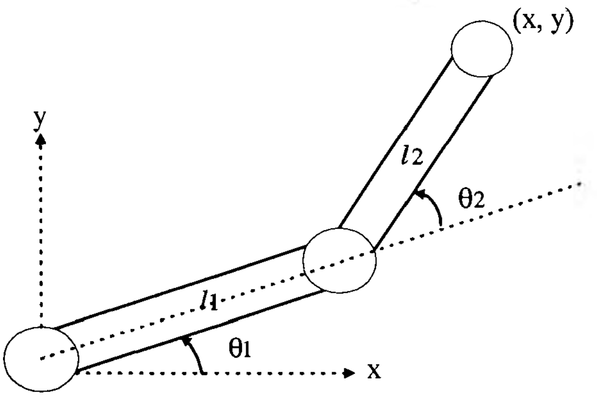
## 5.0 MATHEMATICAL MODEL OF THE ROBOT ARM

The dynamic model or the general equation of motion of a robot manipulator [15] can be described as follows:

$$T_q = H(\theta)\ddot{\theta} + h(\theta, \dot{\theta}) + G(\theta) + T_d \qquad (5)$$

where

$T_q$ vector of actuated torque
**H** $N{\times}N$ dimensional manipulator and actuator inertia matrix
$h$ vector of the *Coriolis* and centrifugal torques
$G$: vector of gravitational torque
$T_d$ vector of the disturbance torque



*Note:*
$l_1$ length of link-1
$l_2$ length of link-2
$\theta_1$ rotation of link-1
$\theta_2$ rotation of link-2
(x,y) end-point position of arm in *Cartesian* space

Figure 5 A representation of a rigid two-link planar arm

For the horizontal two-link rigid planar manipulator shown in Figure 5, its dynamic model is given by,

$$T_{q1} = H_{11}\ddot{\theta}_1 + H_{12}\ddot{\theta}_2 - h\dot{\theta}_2^{\,2} - 2h\dot{\theta}_1\dot{\theta}_2 \tag{6}$$

$$T_{q2} = H_{22}\ddot{\theta}_2 + H_{21}\ddot{\theta}_1 - h\dot{\theta}_1^{\,2} \tag{7}$$

where

$$H_{11} = m_2 l_{c1}^{\,2} + I_1 + m_2(lc_1^{\,2} + lc_2^{\,2} + 2l_1 l_{c2}\cos\theta) + I_2$$

$$H_{12} = H_{21} = m_2 l_1 l_{c2}\cos\theta + m_2 lc_2^{\,2} + I_2$$

$$H_{22} = m_2 lc_2^{\,2} + I_2$$

$$h = m_2 l_1 l_{c2}\sin\theta_2$$

where

$I$ mass moment of inertia of the link
$m$ mass of the link
$l$ length of the link
$l_c$ length of link from the joint to the center of gravity of the link

The gravitational term of the general equation of motion of the arm has been omitted since the arm is assumed to move only in a horizontal plane. As can be seen from Equations (6) and (7), the system is highly coupled, as the motion of second link will affect the dynamic behavior of first link, and vice versa. The coupling property adds to a certain extent a degree of difficulty in controlling the robot arm effectively. Thus, the dynamic model is reduced to

$$T_q = H(\theta)\ddot{\theta} + h(\theta, \dot{\theta}) + T_d \qquad (8)$$

## 6.0 SIMULATION

Simulation of the control scheme is performed using MATLAB and SIMULINK together with *GEATbx* (Genetic Evolutionary Algorithm Toolbox).

### 6.1 Simulation Parameters

The parameters used in the simulation are given as follows:

Robot parameter:

| | | |
|---|---|---|
| Link length: | $l_1 = 0.25$m | $l_2 = 0.2236$m |
| Link masses: | $m_1 = 0.3$kg | $m_2 = 0.25$kg |
| Motor masses: | $mot_{11} = 1.3$kg | $mot_{21} = 0.8$kg |
| Payload mass: | $mot_{22} = 0.1$kg | |

Controller parameters:

| | | |
|---|---|---|
| AFC controller gains: | $K_p = 750 / s^2$ | $K_d = 500 / s^2$ |
| Motor torque constants: | $K_{tn} = 0.263$ N/A | |
| AFC constants: | $K_c = 1.0$ | |

GA parameter:

| | | |
|---|---|---|
| Range of **IN** (kgm$^2$): | $0 \leq IN_1 \leq 0.15$ | $0 \leq IN_2 \leq 0.01$ |
| Number of generation: | 50 | |
| Objective function: | Fitness function, $f = \dfrac{1}{1+E}$<br>where $E = \sum_{t=0}^{T} \lvert e(t) \rvert$ | |
| Crossover probability: | 0.25 | |
| Mutation rate: | 0.035 | |
| Genes per parameter: | 10 | |

Simulation is performed with a set of constant torques, $T_d$ introduced at the joints incrementally vary from 10Nm to 50Nm. A number of results were obtained and evaluated based on the different values of $T_d$. In GA, fitness function has been chosen to evaluate the objective function using the equation $f = \dfrac{1}{1+E}$ where $E$ is sum of the track error from rest until a full circular trajectory is complete.

## 6.2 Prescribed Trajectory

A circular trajectory is generated considering the following time (t) dependent functions for the *Cartesian* coordinate:

$$x_{bar1} = 0.25 + 0.1\cos(\tfrac{V_{cut}}{0.1}t) \tag{9}$$

$$x_{bar2} = 0.2 + 0.1\sin(\tfrac{V_{cut}}{0.1}t) \tag{10}$$

where the introduced endpoint tangential velocity, $V_{cut}$ is assumed to be 0.5m/s.

Figure 6 illustrates the desired trajectory of the arm.

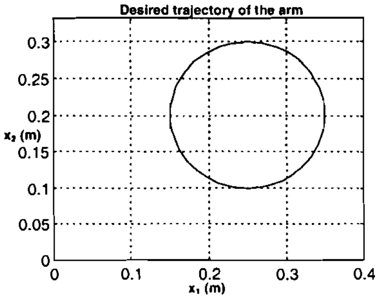


Figure 6 The desired trajectory of the arm

## 6.3 Results and Discussion

Table 1 shows the performance of GA with various magnitudes of constant torque, $T_d$ at the joints. The performance of GA is measured by the fitness and the track error of the best individual found in the whole simulation run. The same result is also illustrated in Figure 7. In each generation, the best individual is traced by its highest fitness value or the lowest track error since $f = \dfrac{1}{1+E}$, where $f$ is the fitness value while $E$ the sum of all track error sampled along a complete trajectory cycle. It should be noted that $f < 1$ for $E > 0$. It can be seen that when the introduced $T_d$ increases, the performance of GA will generally decrease. This finding is as expected because the increasing value of $T_d$ will have proportional effect to the level of the difficulty in the GA mechanism to estimate suitable **IN** value for AFC to accomplish the compensation process. However, the GA technique is able to adapt satisfactorily to this occurrence since the performance of GA is observed to be minimally affected. The statistical results show that, for every 1 Nm increase in $T_d$, the sum of track error, $E$ of the best individual found by GA at the end of simulation will only increase by 7.2 x $10^{-4}$ m. In other words, the average sensitivity of $E$ to $T_d$ caused by variation in the GA parameter is 7.2 x $10^{-4}$ m.

Table 1 Summary of the results obtained

| $T_d$ (Nm) | 10 | 20 | 30 | 40 | 50 |
|---|---|---|---|---|---|
| $E$ (m) | 0.0649 | 0.0717 | 0.0705 | 0.0738 | 0.0937 |
| Best fitness | 0.9391 | 0.9331 | 0.9342 | 0.9313 | 0.9143 |

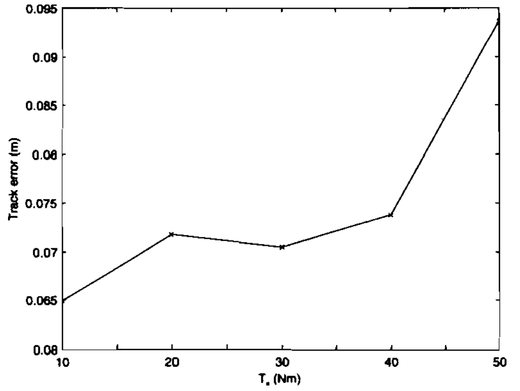


Figure 7 Track error versus constant disturbance torque at the joints, $T_d$

Figure 8 shows optimum inertia matrix, **IN**, versus $T_d$. In general, $IN_1$ is greater than $IN_2$ as expected since the inertia at joint one is always greater than joint two. The results show that the approximate range of $IN_1$ is 0.045 - 0.06 kgm$^2$ while $IN_2$ 0.005 - 0.01 kgm$^2$. It can also be observed that the GA has shown its ability to adapt the **IN** value to different working environment in terms of $T_d$.

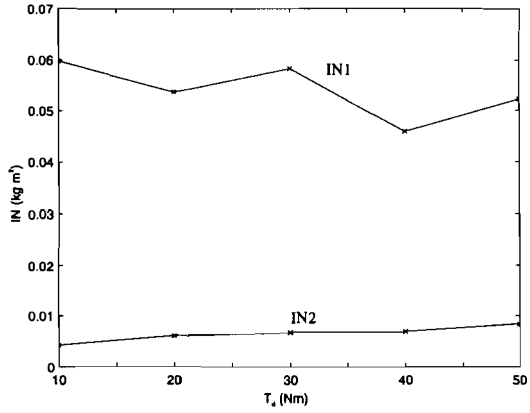
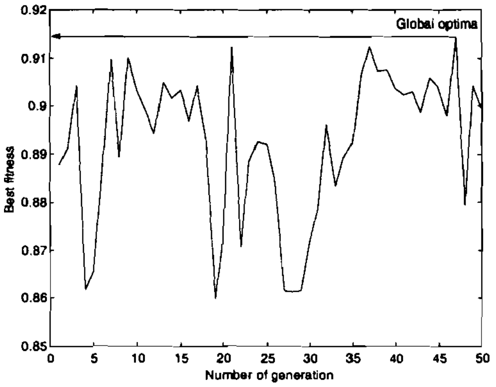


Figure 8 Optimum **IN** versus $T_d$

Figure 9 depicts the trace of the best fitness found along the generation in an arbitrary simulation. Here, a sample of $T_d$ = 50 Nm is shown. As can be seen in this figure, the GA has found a best individual of fitness 0.9143 at generation 47. Even though, as a stochastic and global optimizer, GA keeps searching for other possible peaks, which may be higher than the currently found peak.

Figures 10-15 show the track errors of AFCAGA scheme for different $T_d$ at the joints. The general trend of most of the error curves is converging with time signifying that the estimated inertia matrix of the arm is appropriately identified and adapted to the conditions imposed. Note that the maximum error occurs at the beginning of the operation. This is mainly due to the inherent static friction of the robotic system. When there is no disturbance ($T_d$ = 0 Nm), the maximum



(*Global optima: best found solution throughout the simulation)

Figure 9 Best fitness versus number of generation

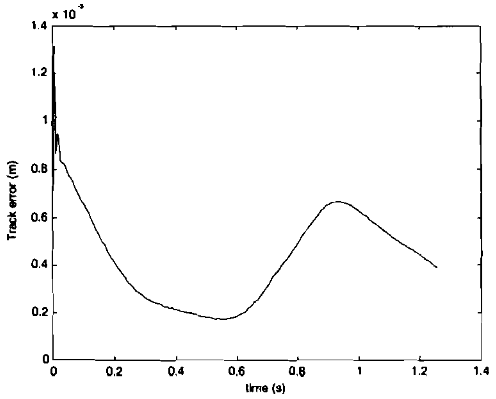


Figure 10 Track error along the trajectory for $T_d = 0$ Nm

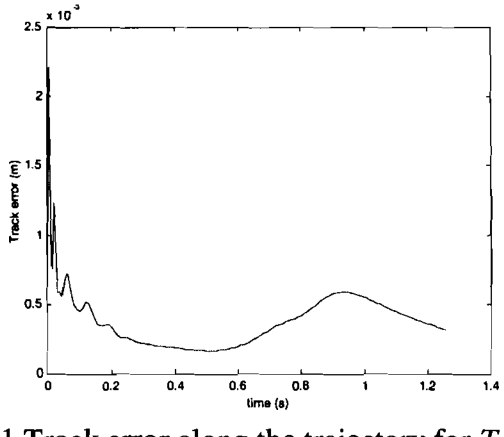


Figure 11 Track error along the trajectory for $T_d = 10$ Nm

error along trajectory is about 1.3 mm. When $T_d$ = 50 Nm, the maximum error is 2 mm. This shows a slightly decreasing performance of AFCAGA with the increase of $T_d$. However, as a whole, the performance of system with the presence of disturbances is still considered robust. In all cases, the proposed scheme is able to 'absorb' all the disturbances effectively without degrading the system's performance. Thus, AFCAGA scheme exhibits a high degree of robustness and accuracy as the track error is successfully bounded within an acceptable range (0 - 3mm). The track error within this range implies that the end effector follows or tracks the trajectory very well and almost resembling the desired trajectory.

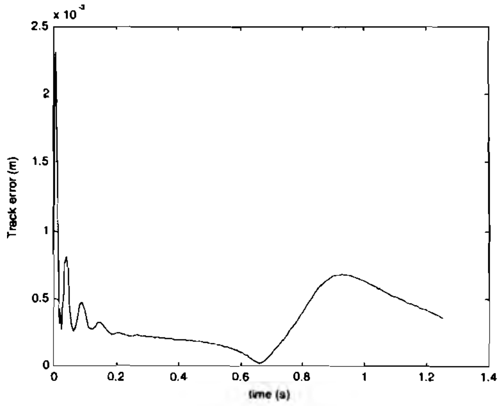


Figure 12 Track error along the trajectory for $T_d = 20$ Nm

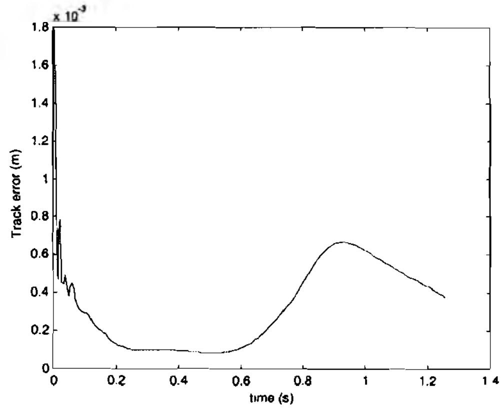


Figure 13 Track error along the trajectory for $T_d = 30$ Nm

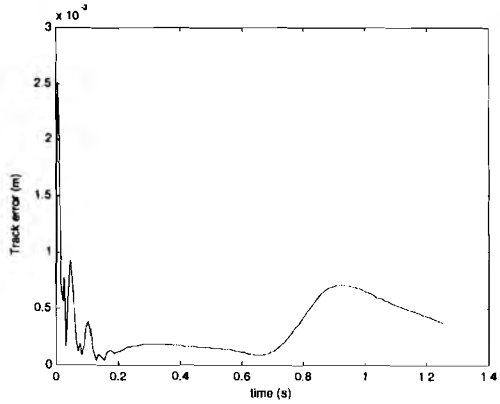


Figure 14 Track error along the trajectory for $T_d = 40$ Nm

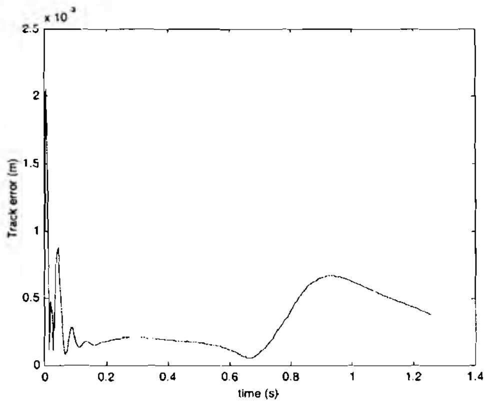


Figure 15 Track error along the trajectory for $T_d = 50$ Nm

## 7.0 CONCLUSIONS

The genetic algorithm embedded in the AFC scheme used in the study has been shown to be very effective in generating the required estimated inertia matrix automatically, which when implemented to the main control scheme with or without disturbances, produce favourable results. Thus, the integration of the GA in the AFC strategy is shown to be feasible and implementable. The trajectory track error obtained is reasonably small showing the excellent capability of AFCAGA scheme to accommodate the disturbances. For future development, this work can be extended by taking into account other form of complicated trajectories within the robot workspace. Also, other forms of disturbances and test trajectories can be considered to further investigate the robustness of the system.

## ACKNOWLEDGEMENTS

The research work was supported by a research grant (IRPA project no. 09-02-06-0172). The authors would like to express their gratitude to the Ministry of Science, Technology and Environment (MOSTE), Malaysia and Universiti Teknologi Malaysia for their support.

## REFERENCES

1. Lewis, F.L., Abdallah, C.T. and Dawson, D.M., 1993, *Control of Robot Manipulators*, MacMillan Publishing Company, New York.
2. Dubowski, S. and Desforges, D. T., 1979, "The Application of Model Referenced Adaptive Control to Robotic Manipulators", *Transactions of ASME, Journal of Dynamic Systems, Measurement and Control*, vol. 101, pp. 193-200.

3. Slotine, J-J. and Li, W., 1988, "Adaptive Manipulator Control: A case study", *IEEE Transactions on Automatic Control*, vol. 38, no. 11, pp. 905-1003.
4. Leahy, M. B., Jognson, M. A. and Rogers, S. K., 1990, "Neural Network Payload Estimation for Adaptive Robot Control", *IEEE Transactions on Neural Network*, vol. 2, no. 1, pp. 93-100.
5. Astrom, K. J. and McAvoy, T. J., 1995, "Intelligent Control: An Overview and Evaluation", in White, D. A., and Sofge, D. A. (editors), *Handbook of Intelligent Control: Neural, Fuzzy and Adaptive Approaches*, Van Nostrand Reinhold, New York, pp. 3-34.
6. Cheng , W. and Wen, J., 1994, "A Two-Time-Scale Neural Network Controller for the Tracking Control of Rigid Manipulators", *IEEE Transactions of Systems, Man and Cybernetics*, vol. 24, no. 7, pp. 991-1000.
7. Shibata, M., Murakami, T. and Ohnishi, K., 1996, "A Unified Aapproach To Position and Force Control by Fuzzy Logic", *IEEE Transactions on Industrial Electronics*, vol. 43, no. 1.
8. Hewit, J.R. and Burdess, J.S., 1981, "Fast Dynamic Decoupled Control for Robotics Using Active Force Control", *Transaction on Mechanism and Machine Theory*, Vol. 16, No.5, pp. 535-542.
9. Mailah, M., Hewit, J.R. and Meeran, S., 1996, "Active Force Control Applied to Aa Rigid Robot Arm", *Jurnal Mekanikal*, Faculty of Mechanical Engineering, Universiti Teknologi Malaysia, pp. 52-68.
10. Taylor, P.M., 1990, *Robotic Control*, MacMillan Education Ltd.
11. Mailah, M., 1998, *Intelligent Active Force Control Of A Rigid Robot Arm Using Neural Network And Iterative Learning Algorithms*, Ph.D. Thesis, University of Dundee, Dundee.
12. Mailah, M., 1999, "Trajectory Track Control of A Rigid Robotic Manipulator Using Iterative Learning Technique and Active Force Control", *Proceedings of the World Engineering Congress on Robotics and Automation*, Kuala Lumpur, pp. 107-114.
13. Hewit, J.R., 1988, *Advances in Teleoperations*, Lecture Notes on Control Aspects, CISM.
14. Holland, J.H., 1971, "Processing and Processors for Schemata" in Jack, E.L., Ed., *Associative Information Processing*, American Elsevier, pp. 127-146.
15. Slotine, J-J. and Asada, H., 1986, *Robot Analysis and Control*, John Wiley and Sons.