KNOWLEDGE-BASED TRAJECTORY ERROR PATTERN METHOD APPLIED TO AN ACTIVE FORCE CONTROL SCHEME

Endra Pitowarno[1], Musa Mailah, Hishamuddin Jamaluddin

Department of Applied Mechanics, Faculty of Mechanical Engineering
Universiti Teknologi Malaysia
81310 Skudai, Johor Darul Takzim, Tel: 07-5534756, Fax: 07-5566159
E-mail: pgep@graduan.fkm.utm.my, musa@fkm.utm.my, hishamj@fkm.utm.my

Abstract: The active force control (AFC) method is known as a robust control scheme that dramatically enhances the performance of a robot arm particularly in compensating the disturbance effects. The main task of the AFC method is to estimate the inertia matrix in the feedback loop to provide the correct (motor) torque required to cancel out these disturbances. Several intelligent control schemes have already been introduced to enhance the estimation methods of acquiring the inertia matrix such as those using neural network, iterative learning and fuzzy logic. In this paper, we propose an alternative scheme called Knowledge-Based Trajectory Error Pattern Method (KBTEPM) to suppress the trajectory track error of the AFC scheme. The knowledge is developed from the trajectory track error characteristic based on the previous experimental results of the crude approximation method. It produces a unique, new and desirable error pattern when a trajectory command is forced. An experimental study was performed using simulation work on the AFC scheme with KBTEPM applied to a two-planar manipulator in which a set of rule-based algorithm is derived. A number of previous AFC schemes are also reviewed as benchmark. The simulation results show that the AFC-KBTEPM scheme successfully reduces the trajectory track error significantly even in the presence of the introduced disturbances.

Key Words: Active force control, estimated inertia matrix, robot arm, trajectory error pattern, knowledge-based.

1. INTRODUCTION

Robot control is a challenging research area in the field of robotic in which a number of problems need to be addressed or solved in order to acquire good coordinated motion and force controlled performance of the robotic system while it is performing specific task. One of the major problems encountered is the inherent non-linear dynamic characteristics of the robotic system that often needs rigorous and complex mathematical model analysis. Implementing the model directly into the system

controller is often complicated. The most popular and proven method that is widely employed is the classical control method such as the Proportional-Integral-Derivative (PID) controller. Although this method is simple, practical and robust (to a certain degree), it possesses severe limitation when it comes to high-speed operation or in the presence of significant 'noises' in the system [1] as the control system losses its effectiveness in the process. These two undesirable conditions normally trigger further non-linearity effects and uncertainties into the system. Thus, it is an obvious indication that pure classical control alone is not enough to address and overcome the problem. Eventually, many researchers have tried other methods which usually include the classical method alongside their proposed strategies as means to realize an overall robust, efficient and effective controller[2-6]. One of these methods is based on the incorporation of intelligent control mechanism. Some researchers in this area claim to have achieved a degree of enhancement (value-added) in terms of performance when applied to robotic system using features such as neural network [7], fuzzy logics [8], and iterative learning algorithm [9]. Most of the proposed intelligent control methods focus on the computation or estimation of the robot inverse dynamics via suitable intelligent algorithm or structure while in some other application, it is used to estimate parameter/s of interest.

In this paper, we propose another intelligent method to indirectly control a robot arm using a knowledge-based method. In a number of cases for non-linear system (including robotic), the knowledge-based element is found to be useful in enhancing the decision support system [10]. Using the knowledge component of a particular system, a semantic network analysis was successfully implemented [11], [12]. Once the knowledge component is clearly defined, the probability of the occurrence of event should also be determined. One of

---

[1] Corresponding author

the popular method to approximate the probability of an event is *Bayes Rule* [13].

The technique is applied to an active force control of a robotic manipulator. We call the system *A*ctive *F*orce *C*ontrol with *K*nowledge-*B*ased *T*rajectory-*E*rror-*P*attern *M*ethod (AFC-KBTEPM). The first part of the paper highlights the fundamental concepts of AFC and KBTEPM followed by a brief review of other AFC methods including AFC based on crude approximation (AFCCA)[14], the AFCANN (AFC And Neural Network) [15], and AFCAFL (AFC And Fuzzy Logic) [16]. Simulation study of the AFC schemes was then performed. Consequently, the analysis and discussion of the obtained results obtained are given followed by conclusion and further works that could be carried out.

Figure 1 shows a schematic diagram of an AFC scheme. AFC show that the system subjecting to a number of disturbances remains stable and robust via the compensating action of the control strategy.

The detailed mathematical analysis of the AFC scheme is obtained[4].

From the Newton's second law of motion for a rotating mass, the sum of all torques (*T*) acting on the body is the product of the mass moment of inertia (*I*) and the angular acceleration (*α*) of the body in the direction of the applied torque,

$$\sum T = I\,\alpha$$

For a robot system with serial configuration,

$$T + Q = I(\theta)\,\alpha \tag{2}$$

where,

2. ACTIVE FORCE CONTROL

Application of AFC to robot arm is first proposed by Hewit and Burdess [4]. The aim of this control method is to ensure that the system is stable and robust even in the presence of known or unknown disturbances. The main advantage of the method is that it provides a compensating action using mainly the estimated or measured values of certain parameters. This method gives benefits on reducing the mathematical complexity of the robot system that is known to be highly coupled and non-linear[4].

T is the applied torque

Q is the disturbance torques

$I(\theta)$ is mass moment of inertia or the robot arm and θ is the robot joint angle

α is the angular acceleration of the robot arm

We can obtain a measurement of *Q′* of *Q* as

$$Q' = I'\,\alpha' - T' \tag{3}$$

where the superscript ′ denotes a measured or computed (or estimated) quantity. *T′* can be measured by using a current sensor (with conversion) while *α′* with accelerometer. *I′* may be obtained by assuming a perfect model or simply crude approximation.
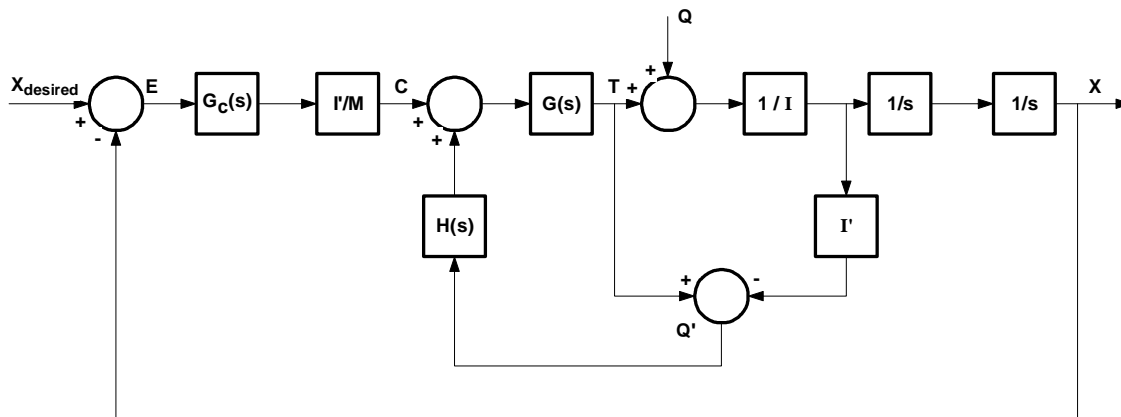

Fig. 1: A schematic diagram of an AFC scheme

For the above open loop system, the estimated variable *Q′* is passed through a function *G*(*s*) before subtraction

from a command vector *C* at a summing junction. If there is a suitable choice of *G*(*s*), the output *X* can be

made invariant with respect to the disturbances Q. In this case the selection of proper estimated inertia matrix I' becomes very important.

Thus, the main computational burden in AFC is the multiplication of the estimated inertia matrix with the angular acceleration of the arm before being fed into the AFC feed forward loop. Apart from that, the output X needs to be computed from the joint angle θ via forward kinematics and also the controller $G_c(s)$ be determined.

3. MATHEMATICAL MODEL OF THE ROBOT ARM

The general equation of motion (or dynamic model) of a robot arm can be described as follows [14]:

$$T_q = \mathbf{H}(\theta)\,\theta_{dd} + h(\theta, \theta_d) + G(\theta) + Q \tag{4}$$

where,

T_q is the vector of actuator torques

$\mathbf{H}(\theta)$ is the $N \times N$ dimensional manipulator and actuator inertia matrix

$h(\theta, \theta_d)$ is the vector of the *Coriolis* and centrifugal torques

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

Q is the vector of the external disturbance torques

For a horizontal two-link rigid planar manipulator as shown in Fig. 2, its dynamic model is given by,

$$T_{q1} = H_{11}\theta_{dd1} + H_{12}\theta_{dd2} - h\theta_{d2}{}^2 - 2h\theta_{d1}\theta_{d2} \tag{5}$$

$$T_{q2} = H_{22}\theta_{dd2} + H_{21}\theta_{dd1} - h\theta_{d1}{}^2 \tag{6}$$

where,

$$H11 = m2lc12 + I1 + m2(lc12 + lc22 + 2\ l1lc2\cos\Box2) + I2 \tag{7}$$

$$H12 = H21 = m2l1lc2\cos\Box2 + m2lc22 + I2 \tag{8}$$

$$H22 = m2lc22 + I2 \tag{9}$$

$$h = m2l1lc2\sin\Box2 \tag{10}$$

m is the vector of link masses

l is the vector of link lengths

l_c is the vector of link lengths from the joint to the center of gravity of the link
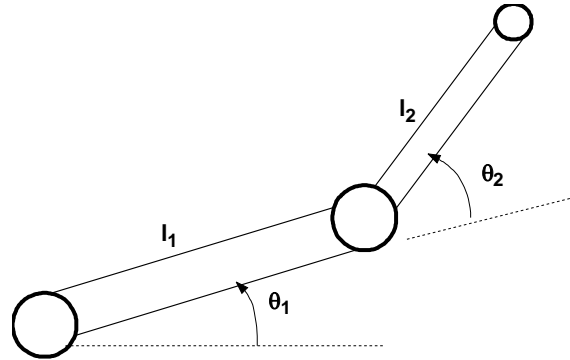

Fig. 2: A representation of a two-link robot arm

In this study, the gravitational term in Equation (6) has been omitted since the arm is assumed to move only in a horizontal plane. Fig. 3 shows the complete schematic diagram of the robot arm including the AFC for the robot arm shown in Fig. 2.
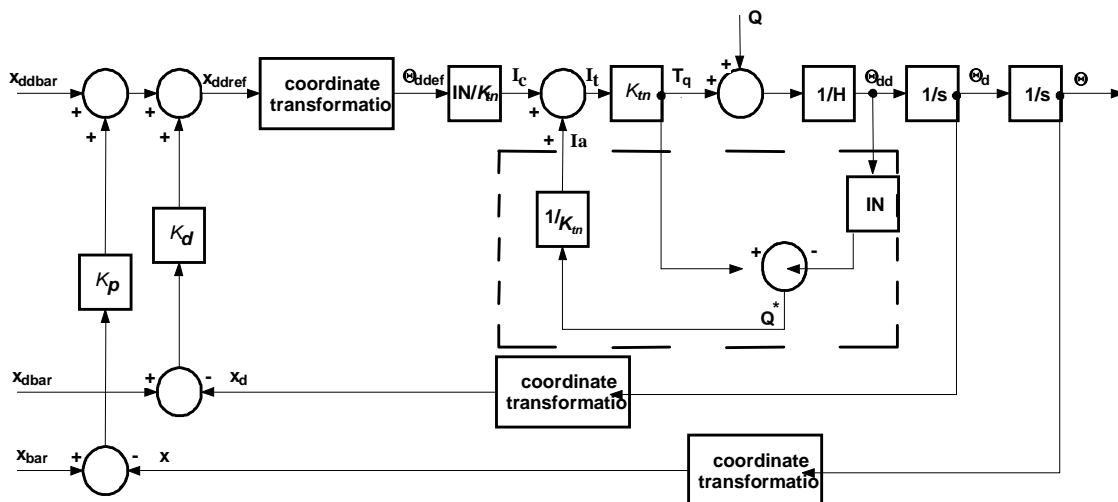

Fig. 3: Two-link robot arm with AFC

The notation used in Fig. 3 is as follows:

K_{tn} is the motor torque constant

K_p and K_d are the controller gains

x is a vector of positions in *Cartesian* space

θ is a vector of positions in joint space

I_c is the current command vector

I_a is the compensated current vector

I_t is the armature current for the torque motor

IN is the estimated inertia matrix

$Q*$ is the estimate of all the disturbance torques

T_q is the applied control torque

The subscript *d* expresses the velocity vector while *dd* is for the acceleration vector. The left hand side of Fig. 3 represents the *x* vector (shown by subscript extension - *bar*) of the desired trajectories and the extension *ref* represents the reference vector. The controller $G_c(s)$ chosen here is a resolved-motion-acceleration-control (RMAC) using a proportional-derivative (PD) element. The RMAC produces the acceleration command vector signal θ_{ddref} which when multiplied by a decoupling transfer function will give the required command vector to the main AFC loop.

From Fig. 3 we have,

$$G(s)H(s) = 1 \tag{11}$$

where, $G(s) = K_{tn}$ and $H(s) = 1/K_{tn}$

The equation for the estimate disturbance term is,

$$Q* = \mathbf{IN}\ \theta_{dd} - T_q \tag{12}$$

where, $T_q = K_{tn} I_t$. From this expression, torque can be calculated by multiply the I_t to the motor constant K_{tn}, where I_t can be measured by using current sensor.

4. THE NEW PROPOSED SCHEME

The real environment of the robot system is often and most likely unpredictable. The system's input error function may easily change whenever disturbance is introduced while performing a specific task. Ideally, this input function should be considered as one of the main parameter to be manipulated so that an accurate and robust control can be achieved by the proposed control strategy. The AFC schemes have been excellent in giving a high level of robust performance by minimizing this error function. Here, we introduce a simplification and refinement of the input function using knowledge-based method applied to a two-link planar manipulator operated in a horizontal plane. This method is accomplished in view of getting better performance by means of suitably managing the error function. The idea of AFC-KBTEPM scheme can be seen in Fig. 4. A complete schematic diagram of the proposed scheme is shown in Fig. 5. The block containing the term KBTEPM is developed and modified from the fixed crude's value of the estimated inertia matrix obtained in the previous study [14]. By giving an input function from the actual task, xbar, the estimated inertia matrix value IN depends on the 'characteristic' of the actual task as well as the angular acceleration of the arm, □dd. The application of the new method is best demonstrated first by reviewing the result of the previous AFCCA method applied to the robot arm as described in the next section. All the results were based on the simulation works performed using the MATLAB and SIMULINK software packages
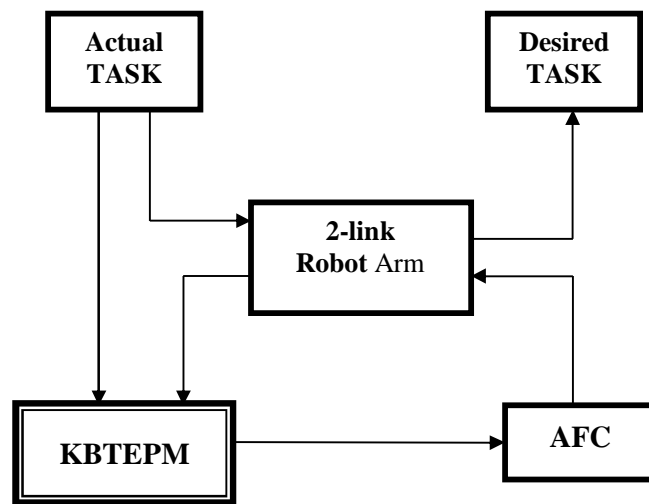

Fig. 4: The idea of AFC-KBTEPM scheme

4.1 KNOWLEDGE INVESTIGATION

The knowledge that will be used as criterion-based rules in the proposed scheme is explored from the previous experimental results of AFCCA method [14]. For the purpose of showing the suggested method using knowledge-based, the same simulation of the AFCCA scheme is reviewed. Figure 6 shows the SIMULINK® model diagram of the scheme.

A preliminary simulation of the AFCCA scheme should be first performed to explore the trajectory track error characteristic. For a fair and one-to-one comparison, all the simulation and other important common parameters for all the robot control schemes should be kept exactly the same. The mentioned parameters are as follows:
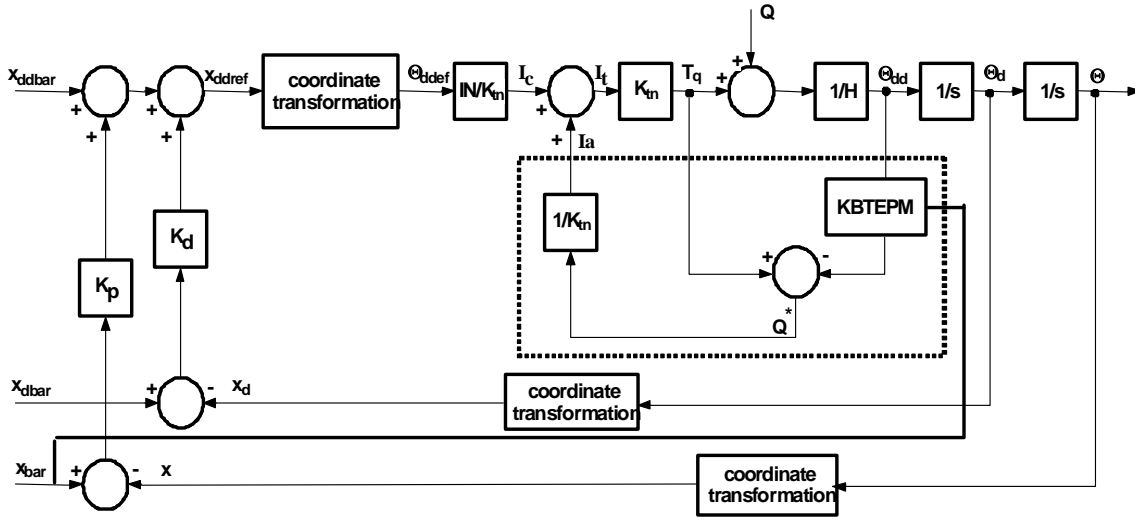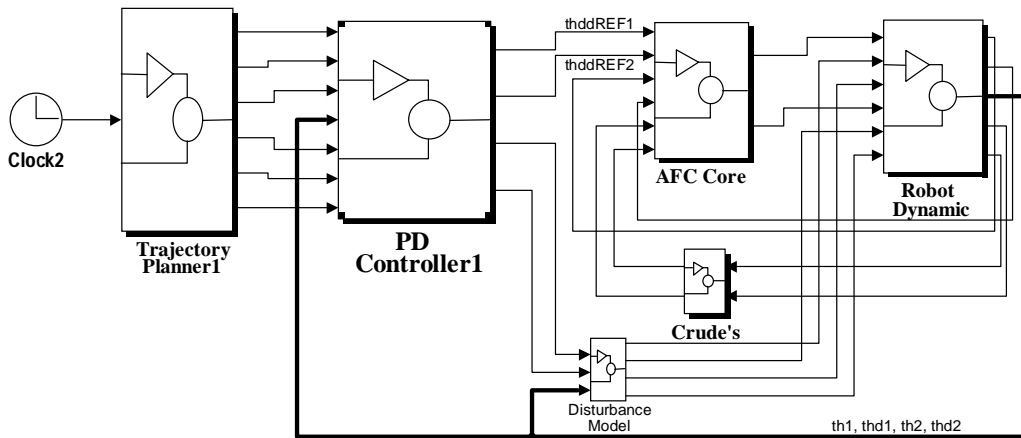

Fig. 5: A schematic of the AFC-KBTEPM

Fig. 6: The SIMULINK® model of the AFCCA scheme

Robot parameters:

Link length: l1 = 0.25 m,   l2 = 0.22 m

Link mass: m1 = 0.3 kg,   m2 = 0.25 kg

Motor mass: mot11 = 1.3 kg   ,mot21 = 0.8 kg

Payload mass: mot22 = 0.1 kg

Controller parameters of RMAC:

Controller gain: Kp = 750 /s,   Kd = 500 /s2

Motor torque constant: Ktn = 0.263 Nm/A

Main simulation parameters:

Integration algorithm: Gear

Simulation time start, tstart: 0.00s

Simulation time stop, tstop: varies

Minimum step size: 0.01s

Maximum step size: 0.10s

Sampling time: 0.01s

Trajectory: Circular form with centre (0.25, 0.2) in *Cartesian* space and a radius of 0.1m.

Other parameters:

Endpoint tangential velocity: V_{cut} = 0.2 m/s

Disturbances: a spring attached at the end of the second link with stiffness k = [0, 200, 400] N/m. k = 0 N/m denotes no disturbances. Figure 7a to 7c show the graphical results of the initial knowledge investigation experiments in AFCCA scheme considering three disturbance conditions; no disturbance, k = 200 N/m and k = 400 N/m. The Track Error (TE) represents the trajectory track error of the robotic arm. Figures 7b and 7c both show a unique form of the curve consisting of the 'hill and valley' pattern that is repeated for each cycle of the circular-form trajectory command apart from the first cycle.
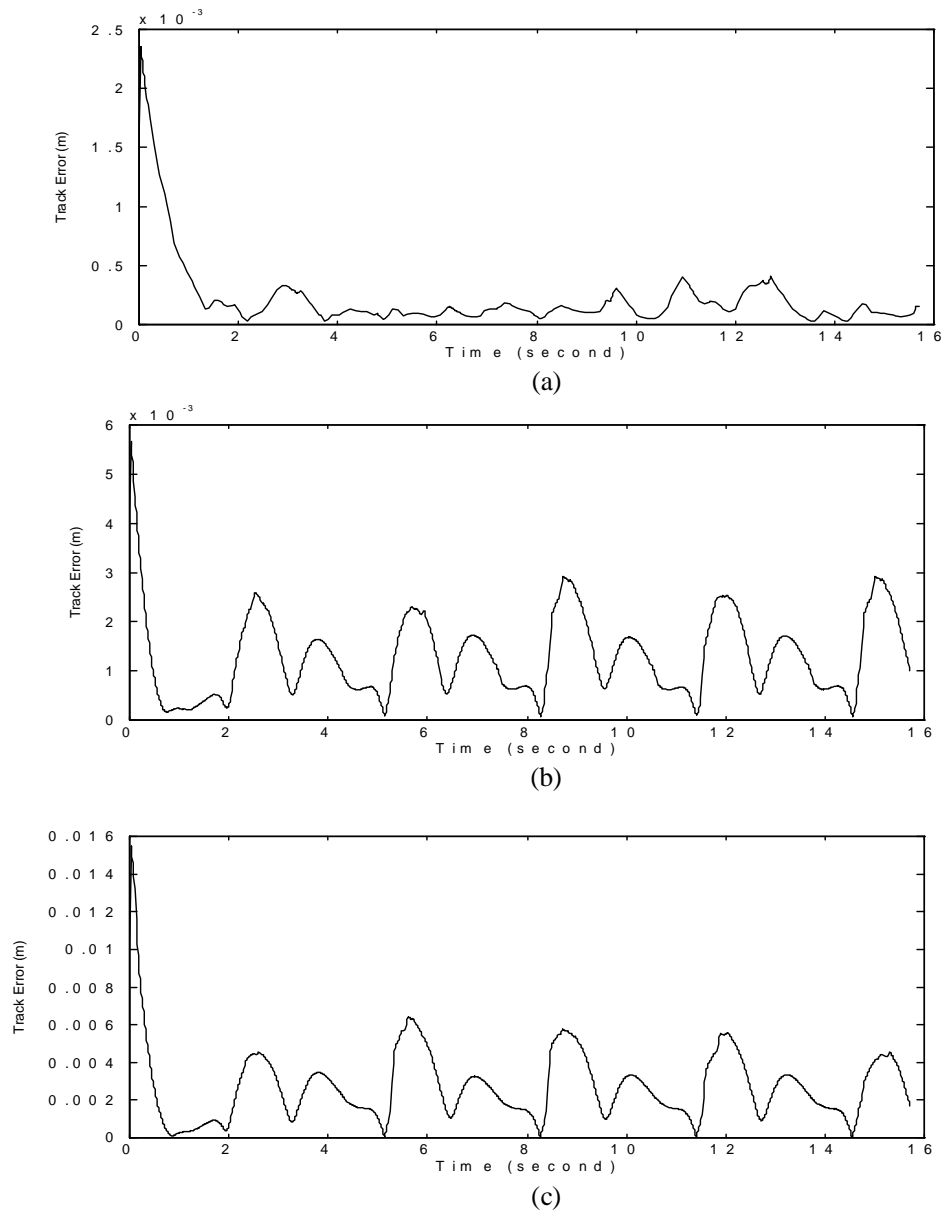

(a)

(b)

(c)

Fig. 7: Track error of AFCCA scheme: (a) k = 0 N/m, (b) k = 200 N/m, and (c) k = 400 N/m.

Figure 8 shows that each loop (cycle) of the circular-form trajectory which is generated by the control scheme and at the same time producing a trajectory track error with two 'hills' and two 'valleys' respectively at 0.5π or 1.5π for the hills and at π or 2π for the valleys. The error is maximum when it is in the form of a hill and becomes minimum when it is a valley. This phenomenon gives us an idea to suppress the error by estimating the probability of the hill and the valley patterns (evidences).By using *Bayes* Rule [13], the posterior probability, an expression is derived as follows:

$$p(H_i \mid e_j) = \frac{p(e_j \mid H_i).p(H_i)}{\sum_i p(e_j \mid H_i).p(H_i)} \qquad (13)$$

where,

H_i is a hypothesis (the occurrence of hill or valley)

e_j is an item of evidence (hill and valley)

The posterior probability of each hill or valley is determined by weighting the prior probability occurrence of the hill or valley. By evaluating the trajectory track error of the circular-form trajectory command, we can conclude that the probability of the 'hill error' and or 'the valley error' is equal to 1 (in the study, a number of trials have been experimented involving loops tried for more than 10 cycles; which give consistent result). This fact (knowledge) of the characteristic can be used to construct an inference mechanism consisting of a set of suitable rules or reasons to be applied in the proposed scheme as expressed in Fig. 9.

For the rules, the optimum parameter for a 'crude value' is set to be equal to 0.95 [14]. The adding/ subtracting values are denoted by (++) or (- -) signs and the proposed expression can be obtained as follows:

$$KVkbs = \left| \frac{Yo}{\sqrt{Xo^2 + Yo^2}} \right| Kgain \qquad (14)$$

where,

KVkbs: add/subtract value of the crude value

Kgain: calibrating gain parameter

Xo, *Yo*: relative *Cartesian* coordinate axis measured from centre of the circular-form trajectory.

The value of *Kgain* would be investigated in this study. In fact, the 'bad' setting of the *KVkbs* will cause an incidental instability for the multiple loops. This will be explained later in the paper.

The MAIN KNOWLEDGE Rule:

IF   <The   Trajectory   Form>   IS
                     <Circular Form>

THEN   DO   <Rule1>

ELSE   <Find others Knowledge Based
                     Data's Library>

END

Rule1:

IF   Absolute (Yo/Xo) INCREASE

THEN   [Crude's   VALUE]new = [Crude's
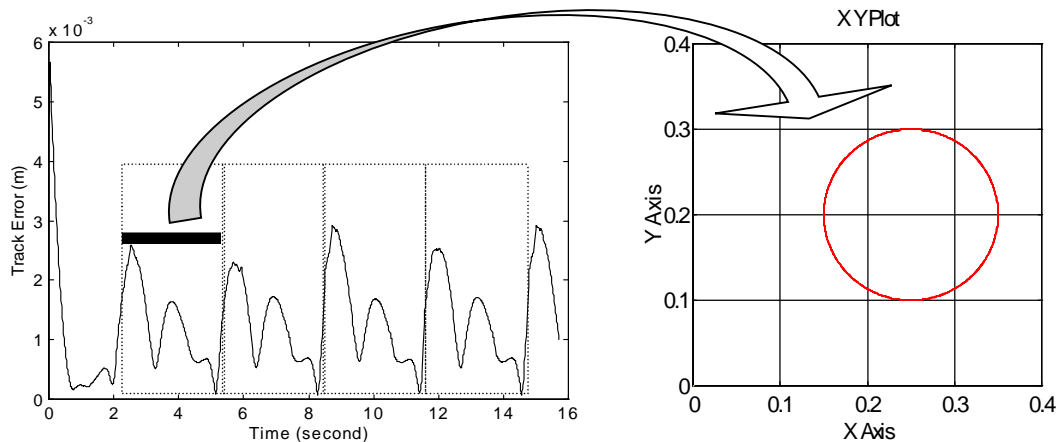VALUE]old ++

ELSE   [Crude's   VALUE]new = [Crude's
VALUE]old - -

END

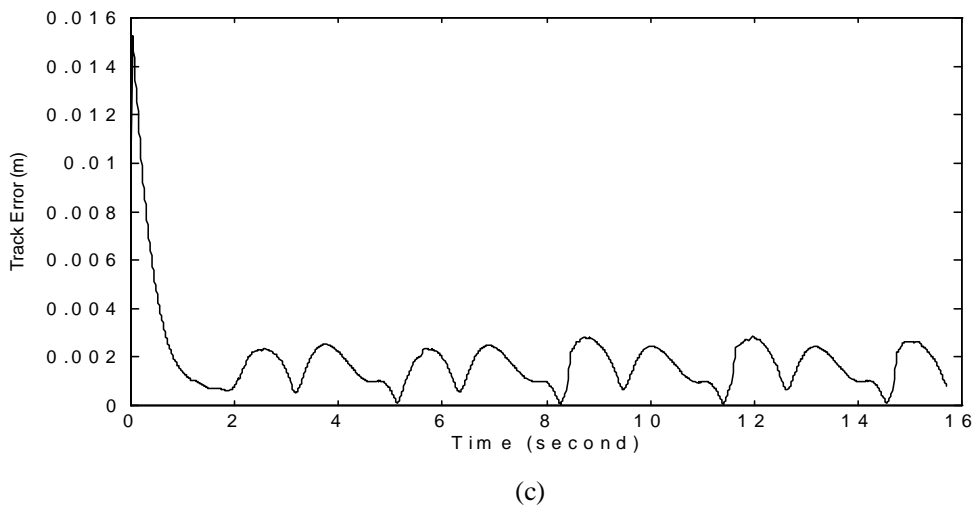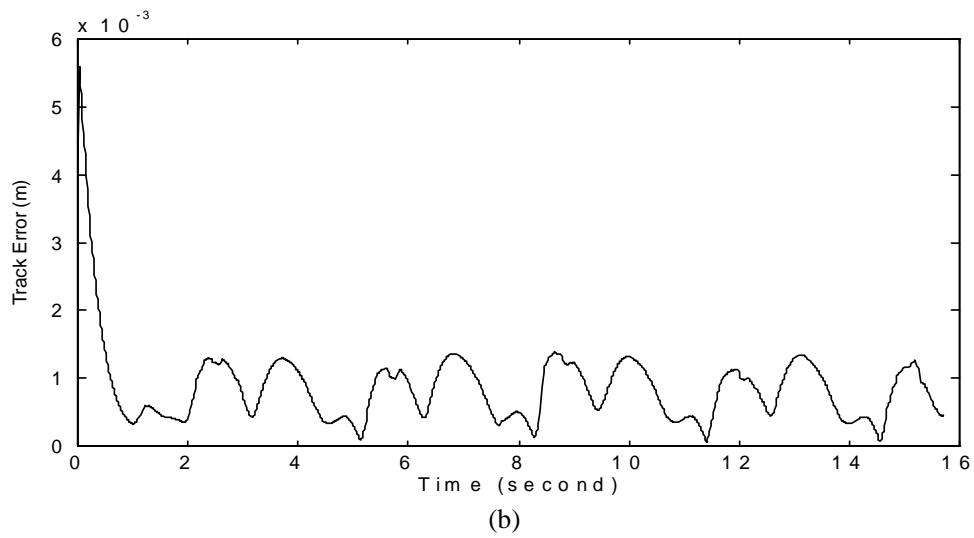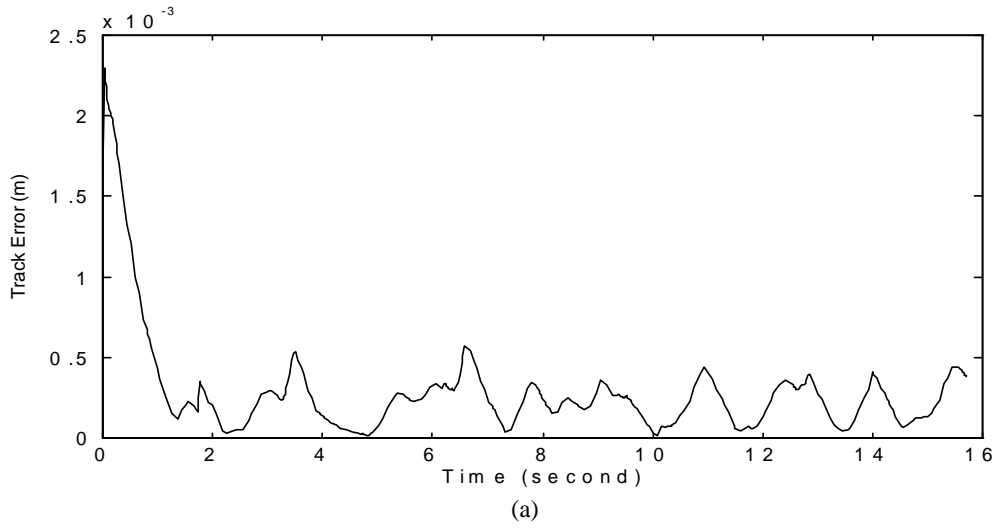**Fig. 8:** The error generated over one cycle of the circular-form trajectory

5. SIMULATION

5.1 SIMULATION OF THE PROPOSED SCHEME

Figure 10 shows a SIMULINK model of the proposed AFC-KBTEPM sch. Fig. 11 shows the simulation results when *Kgain* = 0.01. According to the experimental results when *Kgain* value ranging from 0.005 to 0.050 and with a spring force disturbance, an optimum *Kgain* obtained is 0.02. The trajectory track error at $k = 0$ N/m, $k = 200$ N/m and $k = 400$ N/m are shown respectively in Fig. 12.

5.2 SIMULATION OF THE AFCANN SCHEME

To ensure a fairly comparison the simulation of the AFCANN[14] with the same parameters is presented.

Figure 13 shows a SIMULINK® model of the AFCANN scheme. Besides the main common simulation parameters, the following parameters were used in the AFCANN scheme only:

The result of the training process is shown in Fig. 14. Figure 15 shows the AFCANN simulation results for the same conditions

Network type: Feed-forward Neural Network

eme.

Architecture: 2-5-2

Training algorithm: Error-Back Propagation

Learning rate: $lr = 2.5$

Momentum constant: $mc = 1.75$

Error goal: $eg = 0.05$

5.3 SIMULATION OF THE AFCANN SCHEME

To ensure a fairly comparison the simulation of the AFCANN[14] with the same parameters is presented.

Figure 13 shows a SIMULINK® model of the AFCANN scheme. Besides the main common simulation parameters, the following parameters were used in the AFCANN scheme only:

The result of the training process is shown in Fig. 14. Figure 15 shows the AFCANN simulation results for the same conditions

Network type: Feed-forward Neural Network

Architecture: 2-5-2

Training algorithm: Error-Back Propagation

Learning rate: $lr = 2.5$

Momentum constant: $mc = 1.75$

Error goal: $eg = 0.05$

Fig. 10: A SIMULINK model of the AFC-KBTEPM scheme

(a)

(b)

(c)

Fig. 11: Track Errors of AFC-KBTEPM for various loading conditions at *Kgain* = 0.01: (a) *k* = 0 N/m, (b) *k* = 200 N/m, and (c) *k* = 400 N/m
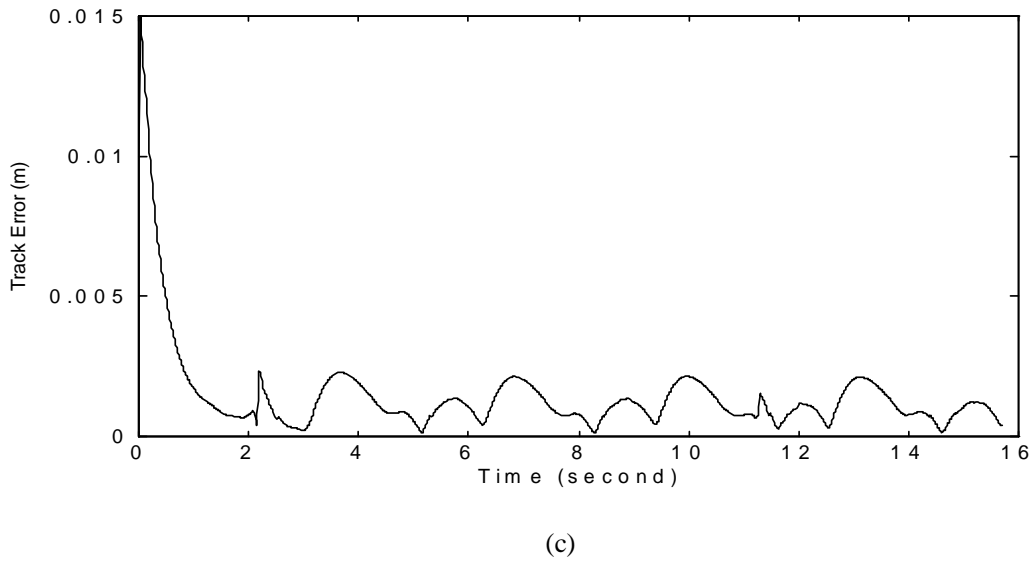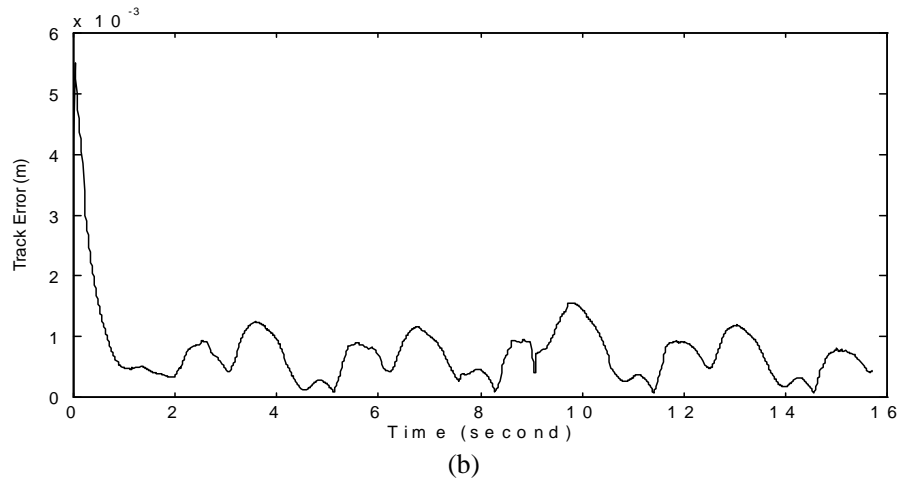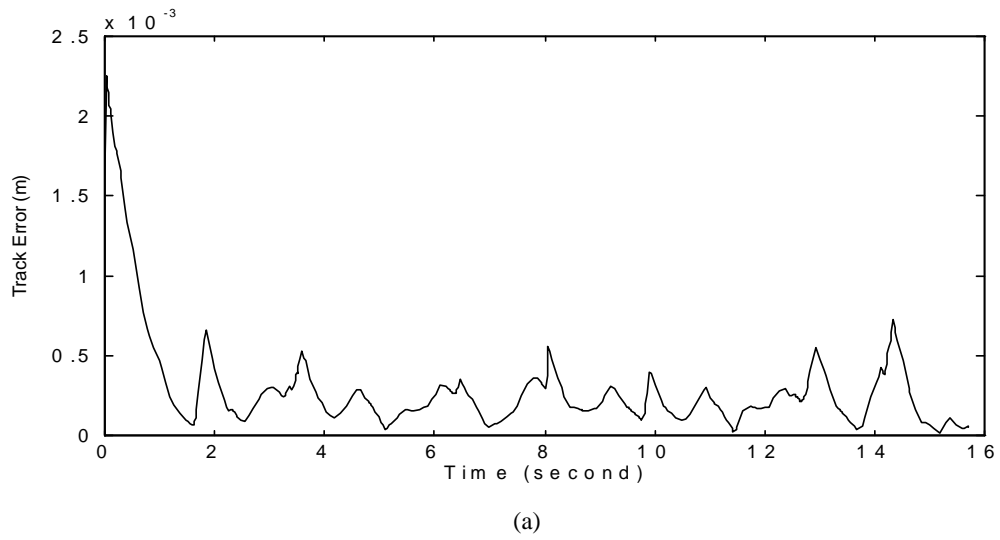
(a)

(b)

(c)

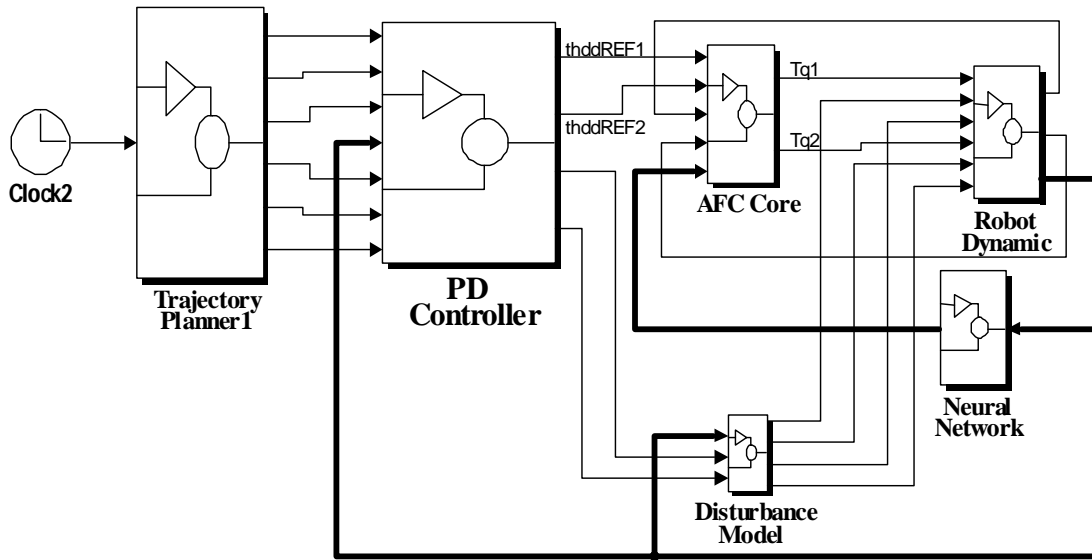**Fig. 12:** Track errors of AFC-KBTEPM at *Kgain* = 0.02: (a) *k* = 0 N/m, (b) *k* = 200 N/m, and (c) *k* = 400 N/m.

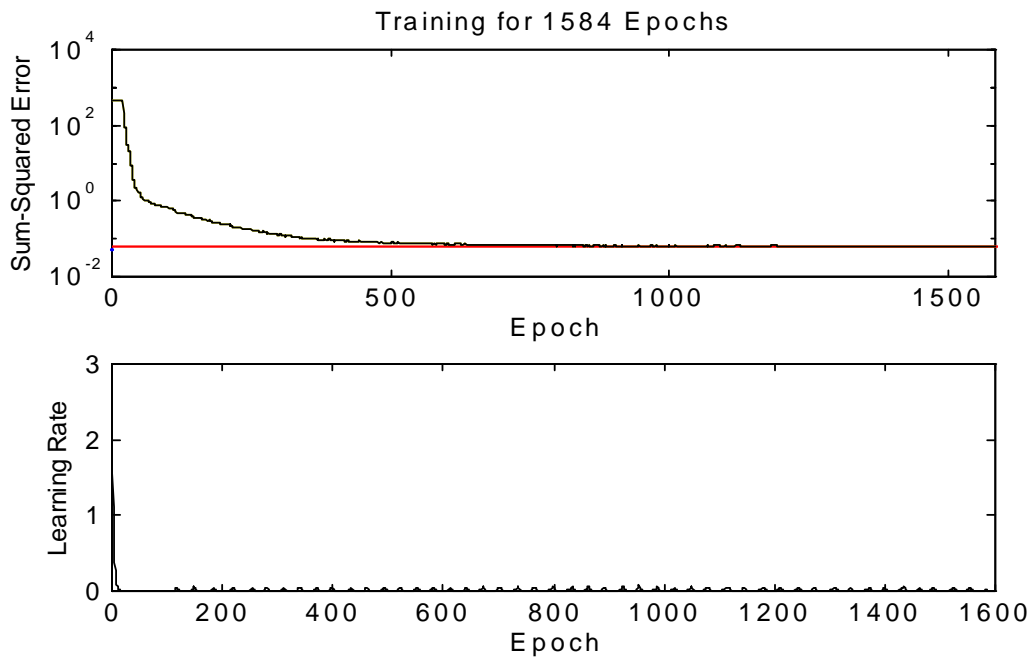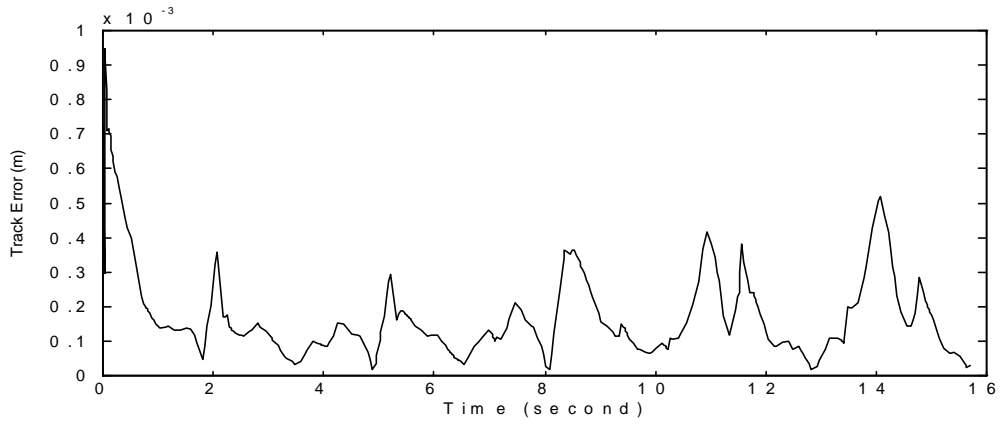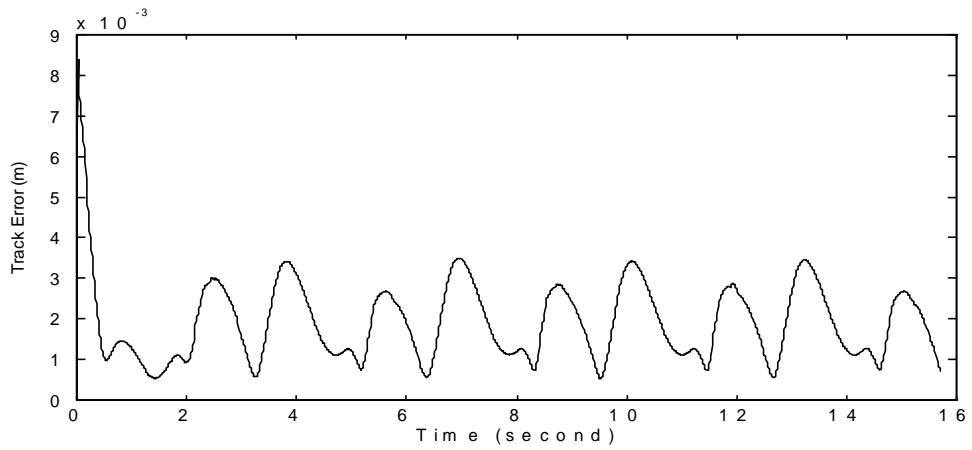**Fig. 13:** The SIMULINK® model of AFCANN

Fig. 14: Training results .

(a)

(b)
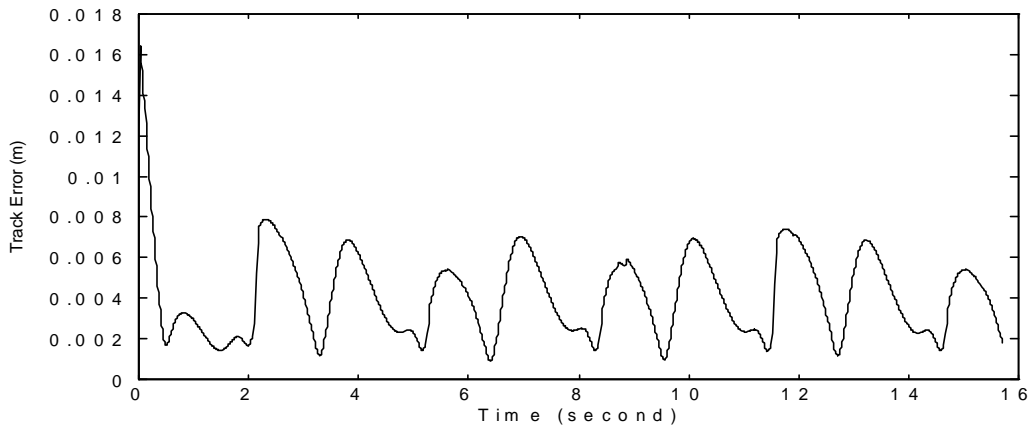
(c)

Fig. 15: Track errors for the selected AFCANN schemes: (a) $k = 0$ N/m, (b) $k = 200$ N/m, and (c) $k = 400$ N/m

6. RESULTS AND DISCUSSION

By evaluating the results shown in Figs. 7, 10, 11, 14 and 15, a number of points can be highlighted for the purpose of comparison. We divided this analysis into three groups according to various types of the applied disturbances, i.e. for no disturbances, disturbance with k = 200 N/m and disturbance with k = 400 N/m. The latter disturbance is deliberately considered to verify the effectiveness of the proposed scheme.

6.1 CASE 1: NO DISTURBANCES (K = 0 N/M)

Figure 16 shows the superimposed track errors of the AFCCA, AFCANN and AFC-KBTEPM schemes when there are no applied disturbances.

It is clear that the AFCANN scheme has the least track error while the AFC-KBTEPM scheme shows more irregularity in shapes compared to others. However, it is useful to note that the peaks of the error pattern are less than 1 mm (after 2 s). The AFCANN scheme seems to produce the least error with the initial error registered at approximately 1 mm compared to an amplitude of more than 2 mm for the other two schemes. Although AFC-KBTEPM has a series of higher peak errors throughout the simulation period, its starting error is slightly lower than those produced by AFCCA.
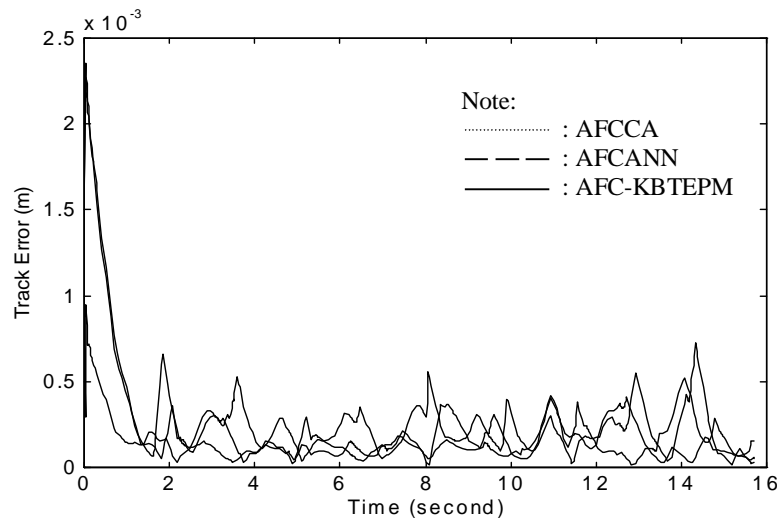
6.2 CASE 2: DISTURBANCE WITH K = 200 N/M

Figure 17 shows a comparison of the track errors for various AFC schemes with the spring stiffness, k = 200 N/m.

The trajectory track errors for all the schemes increase significantly when a spring disturbance with k = 200 N/m is applied to the end-effector of the robot arm. The most significant error pattern with high peaks is observed in the AFCANN scheme as shown in Fig. 16. The series of peak reaches more than 3 mm in amplitude. This is an indication of the system's declining performance in terms of its robustness. This is followed by the AFCCA scheme. On the other hand, AFC-KBTEPM produces a very stable performance with the error pattern below the 2 mm margin.

6.3 CASE 3: DISTURBANCE WITH K = 400 N/M

Figure 18 shows the track error patterns of the AFC schemes under study; this time considering a disturbance with k = 400 N/m.

Again, it can be seen that the AFCANN scheme exhibits the worst performance; generating the largest error pattern compared to the other two schemes. Evidently, the AFC-KBTEPM scheme is the best performer with the peak errors reaching only the 2 mm mark while the performance of the AFCCA scheme lies in between the two schemes.
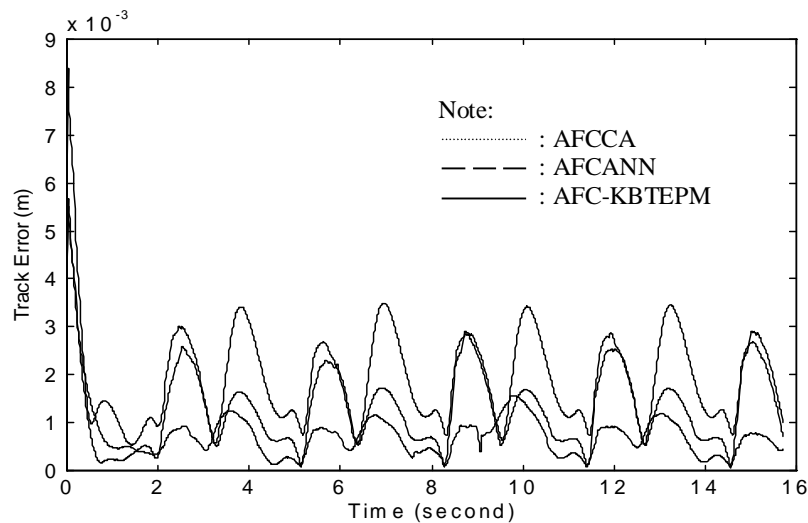

Fig. 16: The track errors of various schemes at no disturbances

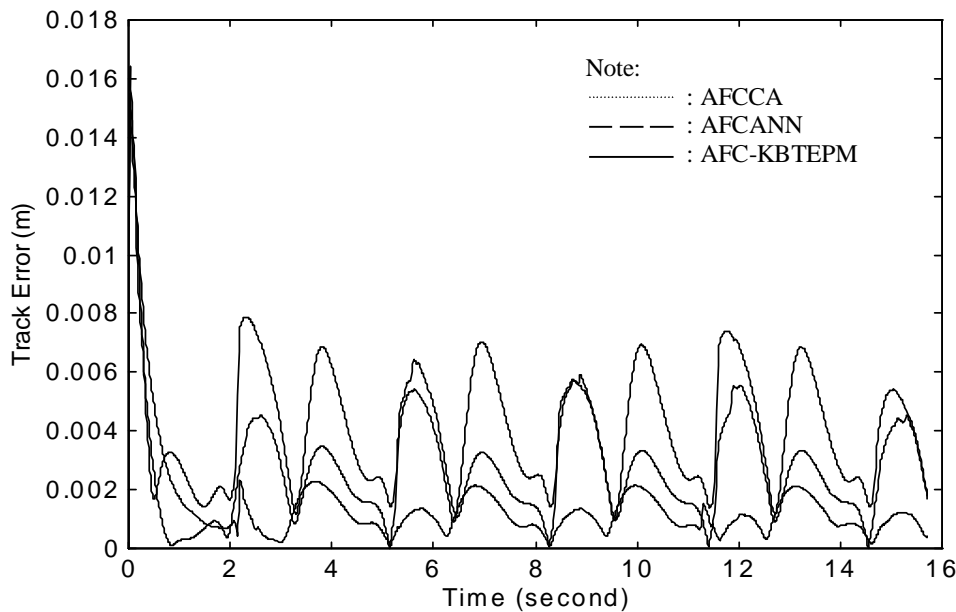**Fig. 17:** Comparison of the track errors at $k = 200$ N/m

Fig. 18: Comparison of the track errors at $k=400$N/m

7. CONCLUSIONS

The advantage of using of the feature of trajectory-form of the robot as knowledge for reasoning investigation that will be applied as basis of rule-based algorithm has been proven in the application of the AFC within the simulation study. Although the AFCCA itself is basically suitable enough implemented at any condition however it still needs to be more precise in certain job. This investigation has shown that the AFC-KBTEPM is able to enhance the performance of the AFC

significantly. The trajectory track error of the AFC-KBTEPM scheme has marginally increased in the range of 1 to 2 mm only when the disturbance is applied. This means the proposed control scheme is proven to be very robust in handling the effect of the disturbances. However, the trajectory-form (task) features extraction and the proper adjustment of the parameter *Kgain* of the proposed scheme seem to provide fresh challenges for the researchers to fully explore the potentials of the method. This will be the main subject of further investigation.

ACKNOWLEDGEMENTS

REFERENCES

[1] H. Asada and J. J. E. Slotine, Robot Analysis and Control, John Wiley and Sons Inc., 1986.

[2] J.Y.S. Luh, M.W. Walker and R.P.C. Paul, Resolved-Acceleration Control of Mechanical Manipulators, IEEE Trans. on Automatic Control, Vol. AC-25, No. 3, pp. 468-474, June 1980.

[3] M.H. Raibert and J.J. Craig, Hybrid Position/Force Control of Manipulators, Trans. of the ASME, Journal of Dynamic Systems, Measurement and Control, No. 102, pp. 126-133, June 1981.

[4] J. R. Hewit and J.S. Burdess, Fast Dynamic Decoupled Control for Robotics Using Active Force Control, Mechanism and Machine Theory, Vol. 16, No. 5, pp. 535-542, 1981.

[5] S. Arimoto, S. Kawamura and F. Miyazaki, Bettering Operation of Robots by Learning, Robotic Systems, pp 123-140, 1984.

[6] F. Lange and G. Hirzinger, Learning Force Control with Position Controlled Robots, Proc. of IEEE International Conference on Robotics and Automation, Minneapolis, April 1996.

[7] M. Kawato., Y. Uno, R. Isobe and R. Suzuki, Hiearchical Neural Network Model For Voluntary Movement With Application to Robotics, IEEE Control Systems Magazine, Vol. 8 , pp 8-15, 1988.

[8] K. Ohnishi, M. Shibata and T. Murakami, A Unified Approach to Position and Force Control by Fuzzy Logic, IEEE Trans. on Industrial Electronics, Vol. 43, No. 1, pp 81-7, February 1996.

[9] P. Bondi, G. Casalino and Gambardella, On the Iterative Learning Control Theory for Robotic Manipulators, IEEE Journal of Robotics and Automation, Vol. 4, No. 1, pp 14-22, Feb.1988.

[10] K. Hori, An Ontology of Strategic Knowledge: Key Concepts and Applications, Knowledge-Based Systems, Vol.13, Elsevier, pp.369-374, 2000.

[11] J. A. Barnden, Semantic Networks, The Handbook of Brain Theory and Neural Networks, The MIT Press, pp. 854-857, 1995.

[12] J. G. Schmolze, W. Snyder, Detecting Redundancy Among Production Rules Using Term Rewrite Semantics, Knowledge-Based Systems Vol.12 , Elsevier, pp.3-11, 1999.

[13] S. Growe, T. Schroeder, C.–E. Liedtke, Use of Bayesian Networks as Judgement Calculus in a Knowledge-Based Image Interpretation System, IAPRS Journal, Vol. XXXIII, pp.101-110, 2000.

[14] Musa Mailah, Intelligent Active Force Control of a Rigid Robot Arm Using Neural Network and Iterative Learning Algorithms, Ph.D Thesis, University of Dundee, 1998.

[15] Musa Mailah, 'A Simulation Study on the Intelligent Active Force Control of A Robot Arm Using Neural Network', Jurnal Teknologi (D), UTM, pp 55-78, June 1999.

[16] Musa Mailah and Nurul Izzah Ab Rahim, 'Intelligent Active Force Control of A Robot Arm Using Fuzzy Logic', IEEE International Conference on Intelligent Systems and Technologies TENCON 2000, Vol. II, pp 291-297, Kuala Lumpur, September 2000.