An Improved Dynamic Feedback Control Technique For Robot Manipulators In Wireless Communication Networks

Udeh Ikemefuna James, Offia Innocent S., Ihedioha Ahmed C,
Department of Electrical and Electronics Engineering, Enugu State University of Science and Technology. Enugu Nigeria
Email: jamesudeh145@yahoo.com

***Abstract*—In this paper a new feedback control scheme, is developed by incorporating the conventional PID control approaches into the learning control system of Artificial Neural Network (ANN) Controller. The proposed system possesses both learning and robustnenss properties, and thereby is able to handle robotic systems as well as certain classes of non-linear and uncertain dynamic systems. Its robust learning control property illustrates the capability of working in either iterative or repetitive control mode with respect to the different control objectives. Theoretical analyses and substantial discussions have been presented to disclose the inherent relationships between the Manipulators non-linearity and uncertainties with respect to, position control, path tracking, error detection, objective trajectory categories, reset conditions, use of derivative signals and learning control modes.**

Keywords—Artificial Neural Network (ANN), Controllers, Error Detection, Derivative Signals Manipulators.

1. INTRODUCTION

The concept of automated machines dates to antiquity with myths of mechanical begins brought to life. Automata, or humanlike machines, also appeared in the clockwork figures of medieval churches, and 18th-century watchmakers were famous for their clever mechanical creatures.

Feedback (self-correcting) control mechanisms were used in some of the earliest robots and are still in use today. An example of feedback control is a watering trough that uses a float to sense the water level. When the water falls past a certain level, the float drops, opens a valve, and releases more into the trough. As the water rises, so does the float. When the float reaches a certain height, the valve is closed and the water is shut off. (Jin 2003)

The first true feedback controller was the Watt governor, invented in 1788 by the Scottish Engineer James Watt. This device featured two metal balls connected to the drive shaft of a steam engine and also coupled to a valve that regulated the flow of steam. As the engine speed increased, the balls swung out due to centrifugal force, closing the valve. The flow of steam to the engine was decreased, thus regulating the speed.

Feedback control, the development of specialized tools, and the division of work into smaller tasks that could be performed by either workers or machines were essential ingredients in the automation of factories in the 18th century. As technology improved, specialized machines were developed for tasks such as placing caps on bottles or pouring liquid rubber into molds. These machines, however, had none of the versatility of the human arm; they could not reach for objects and place them in a desired location. (Lance 2009).

2. CONTROLLERS WITH ARTIFICIAL NEURAL NETWORKS (ANN)

There are lots of articles about the use of ANN in the control of manipulators, (Tetsuro, 2007) discussed the possibility of using ANN as a controller for robotic manipulators and compared the ANN with adaptive control method. Although both ANN and adaptive controllers show good performance, these proved that a neural network with two linear layer is equal to an adaptive controller, while by use of a three layer neural network with nonlinear function for the second layer output and a linear function for the third layer output the ANN shows better performance in systems with high nonlinearities. (Onder, 2001) there is a good comparison of different ANN structures as a controller for a three DOF robot manipulator. Feed forward Neural Networks (FNN), Radial Basis Function Neural Networks (RBFNN), Runge-Kutta Neural Networks (RKNN) and Adaptive Neuro Fuzzy Inference systems (ANFIS) are compared to each other and different learning methods are evaluated. There are different approaches for implementation of controllers using ANN. The simplest structure is introduced in (Bin, 1998). In this approach first the network is trained with samples from model of robot and after the convergence of robot output is guaranteed, the model is replaced with real system and the robot trains in real-time in order to adapt to the real system. By using such approach one don't need to calculate the inverse dynamics of the system and implementation is easy however by changing the desired trajectory we have to train the robot again and repeat the learning procedure. Figure 2.1 shows the structure of this robot. By using reinforcement learning methods it is possible to enhance the learning time considerably however there is no guarantee for the convergence of output in a specific limited time.

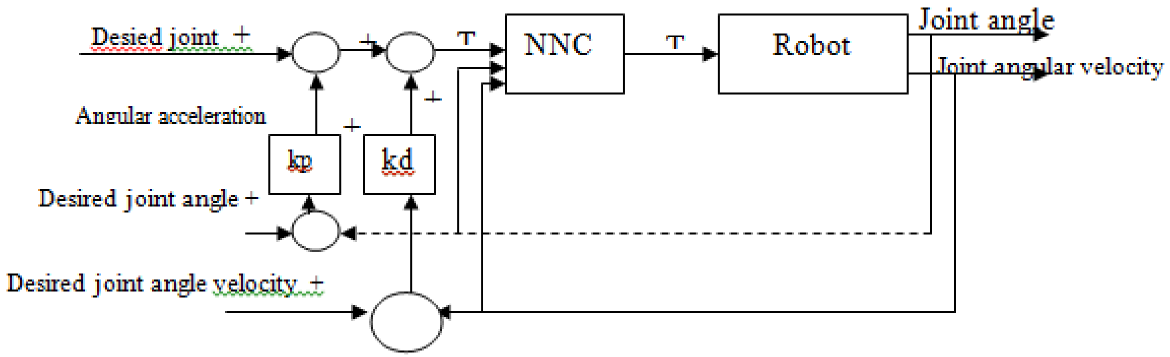

Figure 2.1 the ANN controller proposed by Bin Jin

Akio Ishighuro trained ANN such that the network output compensates for the error between the real system and the model. Figure 2.2 shows the structure of this model (Okay 2001).

3. INVERSE DYNAMIC FEEDBACK CONTROL ANALYSIS

The manipulator is considered as a nonlinear MIMO system described by

It can be shown that this is possible because:

- The model is linear in the control input u;
- The matrix M(q) is invertible for any configuration of the manipulator.

Let us choose the control input u (based on the state feedback):

$(M(q)q + C(q.q)q + Dq + g(q) = u$

Or, in short: $M(q)q + n(q.q) = u$ - - - - - 3.1

The goal is now to define a control input u such that the overall system can be regarded as a linear MIMO system.

This result (global linearization) can be achieved by using a nonlinear state feedback.

$u = M(q)y + n(q, q)$ - - - - - - 3.2

it follows that

$Mq + n = My + n$ and thus (since M is invertible) $\rightarrow q = y$

Where y is the new input of the system

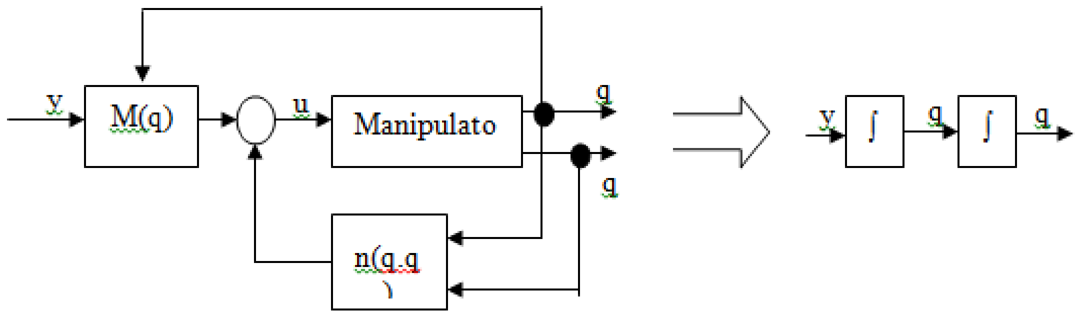

Fig. 3.1 Inverse dynamic control scheme

This is called an inverse dynamics control scheme because the inverse dynamics of the manipulator must be calculated and compensated.

As long as y affects only qi = qi) the overall system is linear and decoupled with respect to y

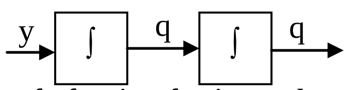

Fig 3.2 Transfer function of an inverse dynamic controller

Now, it is necessary to define a control law y that stabilizes the system.

By choosing.

$y = - Kpq - K_Dq + r$

from q = y it follows

$q + K_Dq + Kpq = r$ - - - - 3.3

that is asymptotically stable if the matrices K_P, K_D are positive-definite.

If matrices K_P, K_D are diagonal matrices defined by

$K_P = \text{diag}\{w^2ni\}$ $K_D = \text{diag}\{2\delta iwni)$

Where K_P is a square (n x n) positive-defined matrix.

The dynamics of the ι-th component is characterized by the natural frequency ω_{ni} and by the damping coefficient.

A predefined trajectory q_d can be tracked by defining

Journal of Multidisciplinary Engineering Science and Technology (JMEST)
ISSN: 3159-0040
Vol. 2 Issue 7, July - 2015

$r = q_d + K_D q_d + K_P q_d$

Then, the dynamics of the tracking error is:

$$q + K_D q + Kpq = 0 \quad - \quad - \quad - \quad 3.4$$

.

4. OBSERVATIONS/SIMULATIONS RESULTS

The ANN is design in MATLAB Simulink environment and tested with simulator from Matlab Robotics Toolbox. Two robots used to testify the performance of the controller. First robot is a two DOF robot and second robot is a puma560. The parameters for this 6DOF robot are extracted from real robot PUMA560.

The motion of robot between two points in joint space is illustrated in the figure, 4.1 Robot path is generated by the 'jtraj' Function from the robotic toolbox Tang, et al (1996) and is a polynomial to the order of 5. The values for the PID controller are in the graphs the trajectory path of the robot as well as error of the controller (the error between desired path and robot path) is shown. The other graphs show a comparison between the output of PID part of the controller and the output of ANN. It is shown that as the training continues the PID output reduces and the ANN generates the main signal for the robot control. As it

The error is not null if and only if q (o) ≠ 0, q(0) ≠ 0 and converges to zero with a dynamics defined by K_P, K

is obvious in the system the network solves the inverse dynamics of the system very fast and a significant improvement is observed after the first cycle of training. The number of neurons in the network may affect the performance of the system and as the number of neurons increases the error of the system decreases however, the network needs more training data. One idea is that at the first cycles of training the network is constructed with a few neurons and as the previous networks to achieve better performance (Slotine 2007).

The manipulator performance is poor before the first robot cycle. Here, ANN controller is yet to recognize the system's dynamics and to begin its learning/training process. This is because the training data are not yet sufficient.`

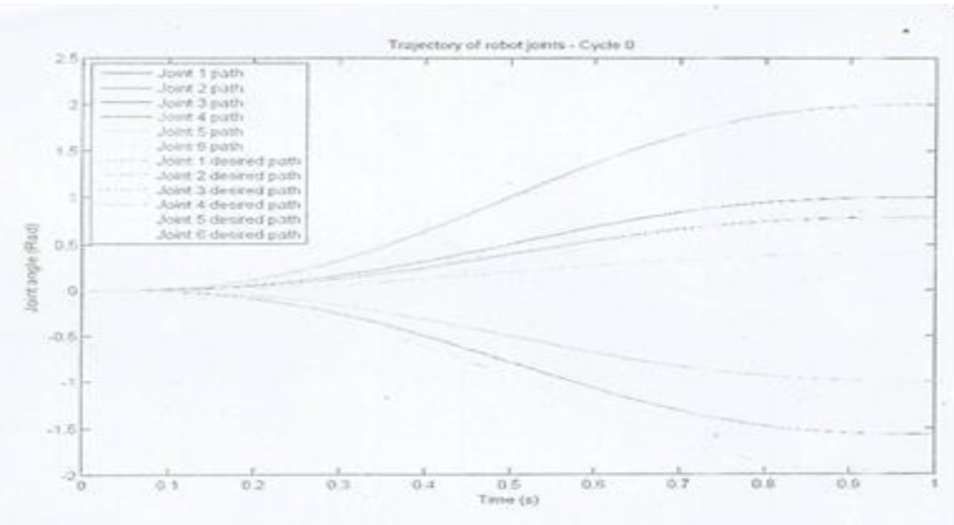

Fig 4.1

At the 10th robot cycle the PID output indicates noise convergence for joint 6 This is good result because the actual positioning and placing object in Cartesian space is determined by this joint

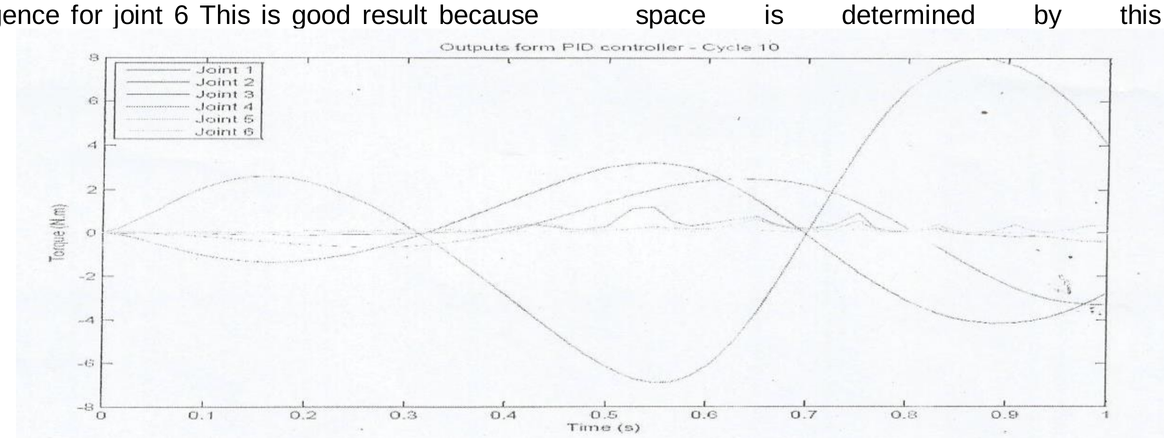

Fig 4.2

The graph below is the output of ANN controller after 10 successful cycles. It is shown that as the training continues the ANN output normalizes and generates a linear signal used for the robot control.

.

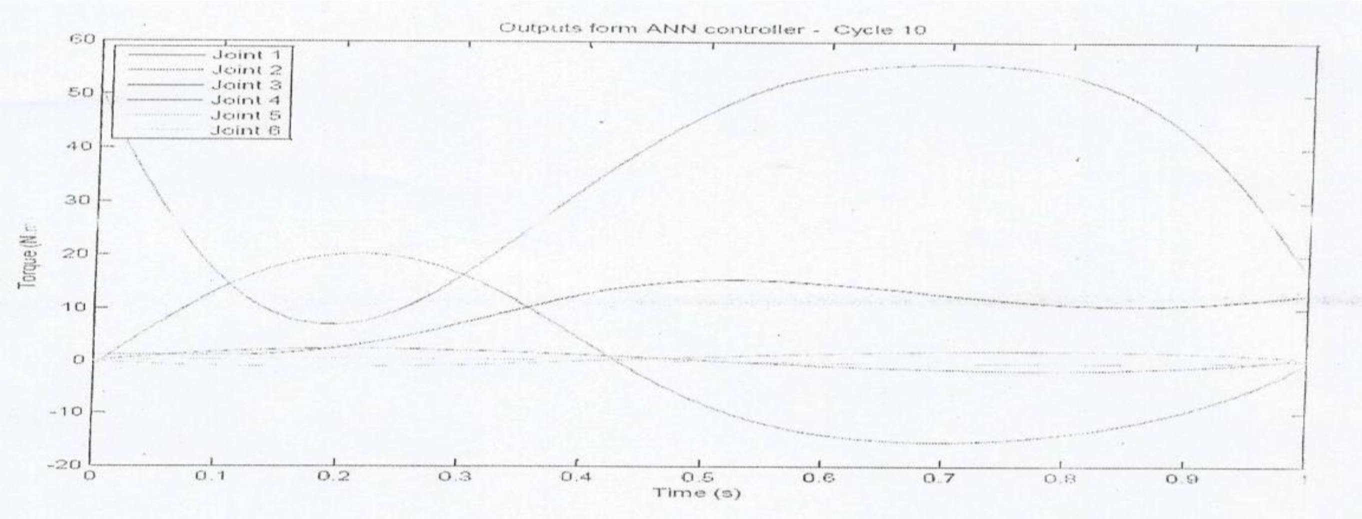

Fig 4.3

Disturbance signal applied to actuators control voltage to check the performance of the controller .

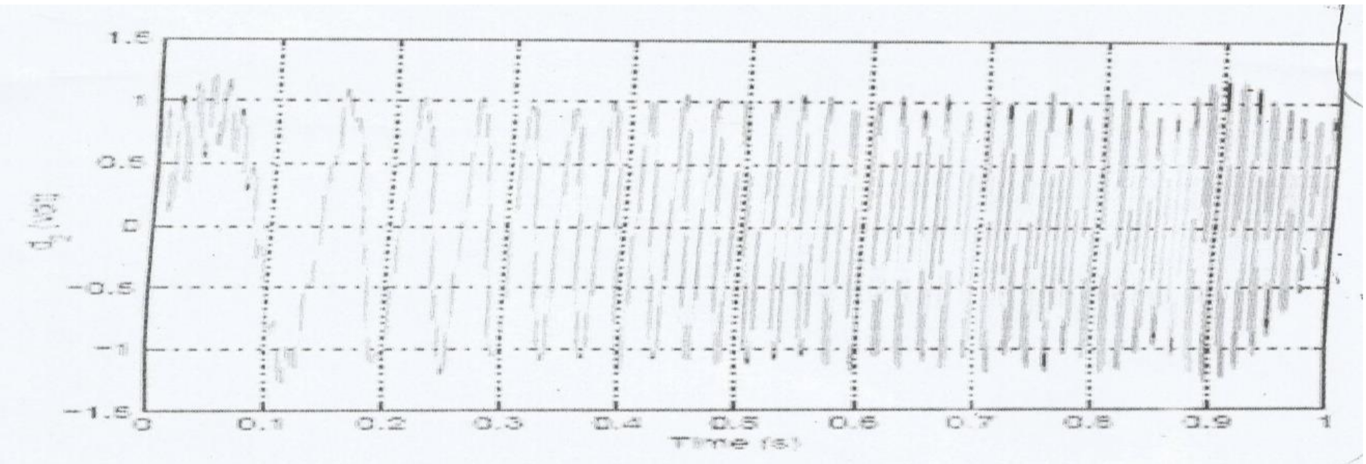

Fig 4.4

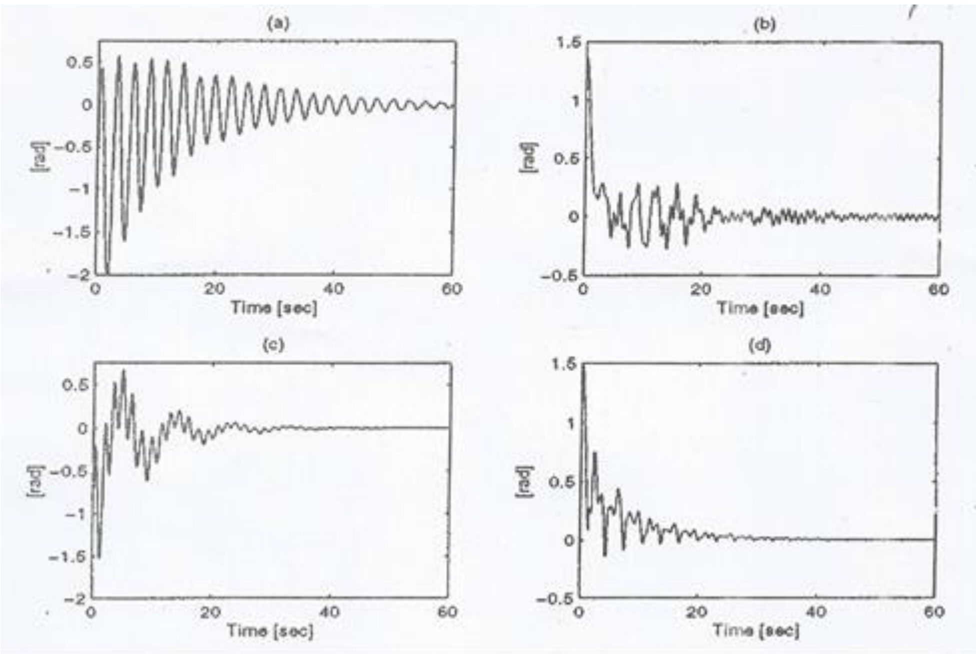

Fig. 4.5 position tracking error comparison for ANN vs. ANN/PID controllers using:

(a) link 1, (b) link 2 => ANN controller (c) link 1, and (d) link 2 => ANN/PID (the proposed) controller.

Comparative Analysis

In order to demonstrate the effect performance this controller over recently developed controllers, one now compare the simulation results of the proposed ANN/PID controller with the regular ANN controller. It is not surprising that asymptotic tracking is obtained even though the controller's gains do not satisfy all the aforementioned conditions. In fact, PID gain as low as 0.01 is chosen and the error is observed to still converge at zero after some time, showing the robustness of this control technique.

5. CONCLUSION

In this paper an efficient method for control of a manipulator with arbitrary degrees of freedom was achieved. The ANN/PID controller is used to identify and solve both the dynamics and the kinematics feedback of a manipulator. Position, orientation control of manipulator end-effector can be accurately be attained in real time. Although this controller did not performing well during the first cycle of a new path, with help of its learning algorithm, the associated errors where brought to zero during subsequent cycles. The controller design is independent from parameters of the system, but the controller learns the system's parameters during its operation. The other advantage is that since the two controllers are in parallel it is easier to design with lower price and faster performance.

REFERENCES

(1) Bin Jin (1993). "Artificial Neural Networks Based Controller for robotic Manipulatpr". Pp. 149-155. Shanghai University Press.

(2) Hang C.C. Shuzhi S., (1999). "Adaptive Neural Network Feedback Control of Robot Manipulators in Work Space", pp. 57 – 102. Shanghai.

(3) Frank A.V., Lewis L., Suresh Jagannathan., (2010). "A Neural Network Control of Robot Manipulators and Nonlinear Systems" p 200-237, Oxford.

(4) Mark Spong and Vidyasagar M.C., (2009). "Robotics Dynamics and Control", pp. 76-108., Minneapolish Inc.LA.

(5) Onder Efe and Okyay Kaynak. (2001), "Comparative Study of Neural Network Structures in Identification of Non-Linear Systems". Vol. 2. p. 101-120., Bogazici University Press, 80815 Isanbul.

(6) Kenneth Levenberg (1944). "A Method for the Solution of Certain Non-Linear Problems in Least Sequare". Vol. 7: pp. 164-168., Grove Press. USA.

(7) Gill, P.E., Murray, W. and Wright, M.H. (1981), "Practical Optimization", Vol. 3 pp. 1029-1037, Academic Press, London.

(8) Liang, C.G and Lance, G.M. (2009), "A Differentiable Null Space Method for Constrained Dynamic Analysis", p 405-411.

(9) Loria A. and Nijmeijer H. "Bounded Output Feedback Tracking Control of Fully-Actuated Euler-Lagrange Systems", Systems and Control Letters.

(10) Slotine J.J.E and W. Li, (2007), "Applied Nonline Control" pp. 23-37. Prentice Hall, NJ: Englewood Cliff.

(11) Watanabe K, Tang J., Nakamura, M., Koga S., and Fukuda T., (March 1996) "A Fuzzy-Gaussian Neural Network and its Application to Mobile Robot Control", vol. 4, No 2, pp. 193. IEEE Trans. Control Syst. Technology.