Robotica (2005) volume 23, pp. 781–784. © 2005 Cambridge University Press
doi:10.1017/S0263574705001992 Printed in the United Kingdom

Control of a robot manipulator and pendubot system using artificial neural networks

Joseph Constantin*, Chaïban Nasr and Denis Hamad†

(Received in Final Form: May 9, 2005)

SUMMARY
The paper introduces artificial neural networks for the conventional control of robotic systems for better tracking performance. Different advanced dynamic control techniques are explained and a new second order recursive algorithm has been developed to tune the weights of the neural network. The problem of real-time control of a Pendubot system in difficult situations has been addressed. Examples, such as positioning and balancing structures, are presented and performances are compared to a conventional PD controller.

KEYWORDS: Pendubot system; Artificial neural network; Tracking performance.

I. INTRODUCTION

In manufacturing systems robots are used to carry out different tasks. In these tasks, the end effectors are required to move from one position to another, or to follow some desired trajectories as close as possible at a fast operational speed.[1] Adaptive controllers with linear parameters have been used for trajectory tracking in robot control.[2] However, instability may occur due to some bounded uncertain disturbances or plant non-linearity.[3] Therefore the robotic community has been interested to implement an ANN controller in a direct closed-loop in order to improve the control performance without modifying the architecture of the system.

This work introduces a comparative study between a second order recursive learning algorithm and the back-propagation method for different ANN controllers implemented in a direct closed-loop for better tracking performance in robot control. We address also the problem of real-time control of the Pendubot system using ANN. Performances are compared to a conventional PD controller.

II. SUBSTANCE OF TOPIC

The dynamic equations of a 2D robot arm (Figure 1) are given by:[4]

$$M(\vec{q})\ddot{\vec{q}} + \vec{h}(\vec{q}, \dot{\vec{q}}) + \vec{g}(\vec{q}) = \vec{u} \qquad (1)$$

* Department of EE, Faculty of Engineering I, Lebanese University EL ARZ Street, Tripoli-North (Lebanon).
† Laboratoire d'Analyse des Systèmes du Littoral. Université du Littoral Côte d'Opale – 50 rue Ferdinand Buisson, Cedex 699 62228 Calais (France).
Corresponding author. J. Constantin, P.O. Box 70-327, Antelias (Lebanon); E-mail: cjoseph@inco.com.lb

The pendubot model is a special case of these equations because the second joint is unactuated (Figure 2).[5,6]

Three control methods have been introduced to control the manipulator. The first method[7] involves training a feed-forward neural controller in conjunction with a fixed controller (Figure 3). The second method[8] employs an ANN that anticipates the torque control (Figure 4). The third method[8] uses an ANN that anticipates the desired input of the closed loop system (Figure 5). To improve the back-propagation method,[9] we develop a second recursive order

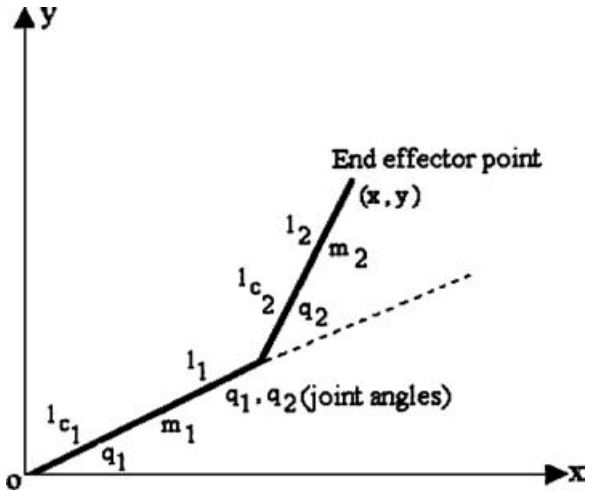


Fig. 1. Dynamic planar robot arm.

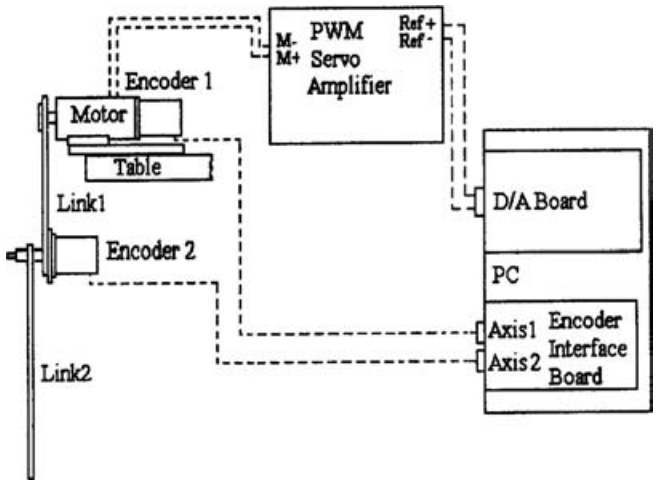


Fig. 2. The Pendubot's interface with the controller.

https://doi.org/10.1017/S0263574705001992 Published online by Cambridge University Press

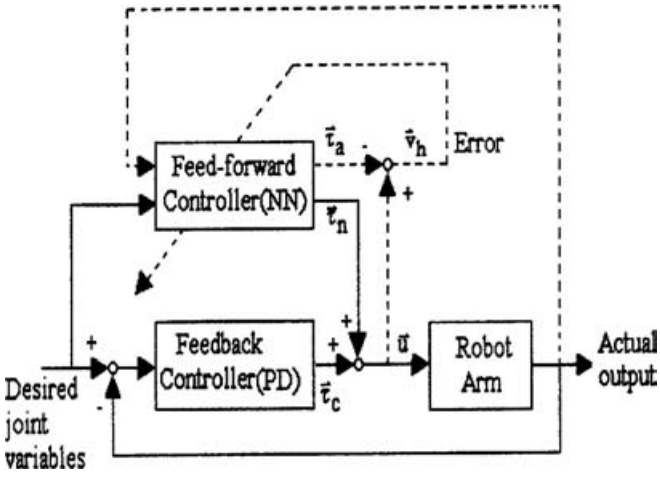


Fig. 3. Feed-forward controller trained by minimizing the torque error.

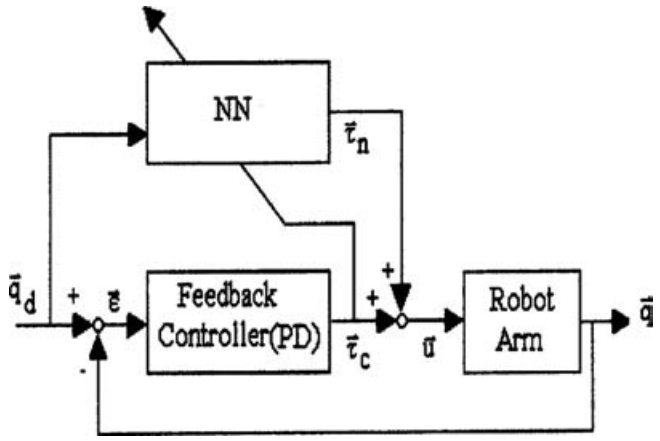


Fig. 4. Feedback error learning control.

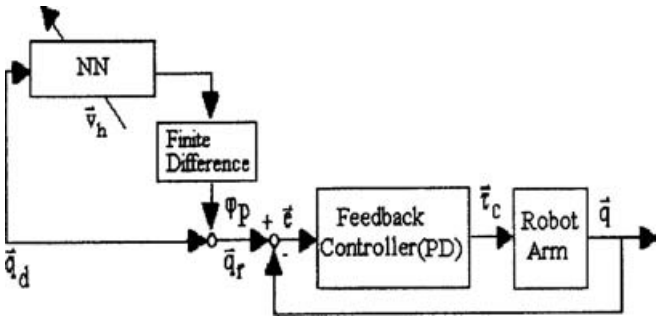


Fig. 5. Neural network that anticipates the desired input.

algorithm which tunes the weights based on the equation:

$$\vec{W}_k = \vec{W}_{k-1} + \eta(\lambda_k I + R_k)^{-1}\psi_k(\vec{W}_{k-1}) \times k(\vec{W}_{k-1}) + \alpha\Delta\vec{W}_{k-1} \quad (2)$$

where $R_k(\vec{W}_{k-1}) = \frac{k-1}{k}R_{k-1}(\vec{W}_{k-1}) + \frac{1}{k}\psi_k(\vec{W}_{k-1})\psi_k^T(\vec{W}_{k-1})$, $\psi_k(\vec{W}_{k-1}) = \frac{\partial \vec{y}_k^T}{\partial \vec{W}_{k-1}}$, $\vec{\varepsilon}_k(\vec{W}_{k-1})$ is the error, $\vec{y}_k$ is the network output and λ_k is introduced to make $(\lambda_k I + R_k)$ positive-definite.

Two neural control techniques have been used to control the pendubot (Figures 6 & 7). The first technique[10] uses the

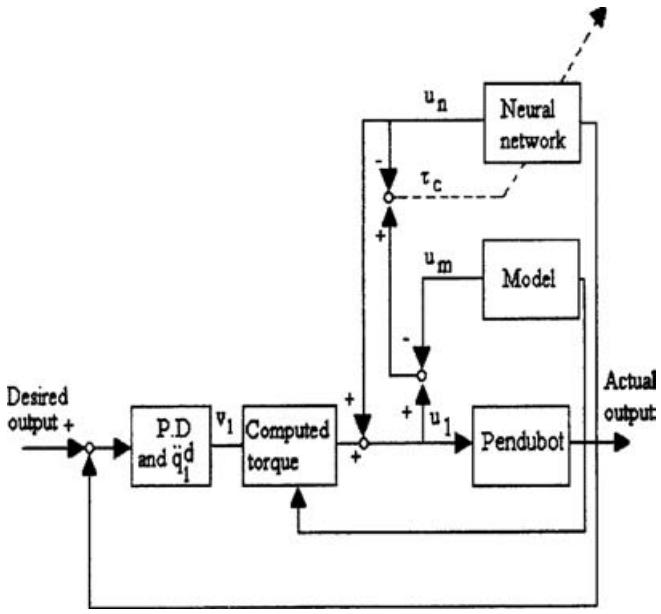


Fig. 6. Controller using the computed torque method with neural network.

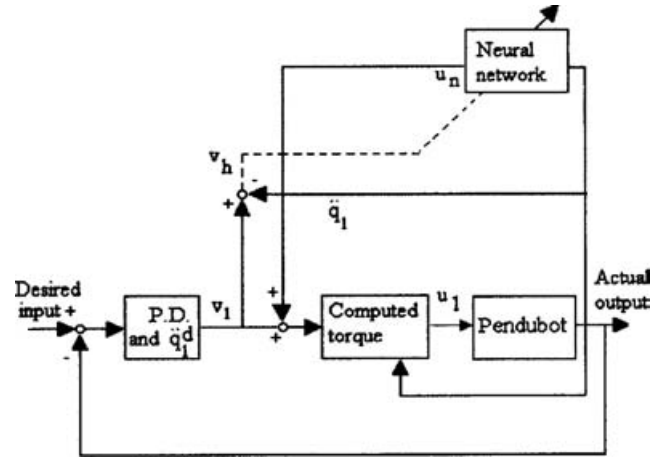


Fig. 7. Neural network compensating at controller level.

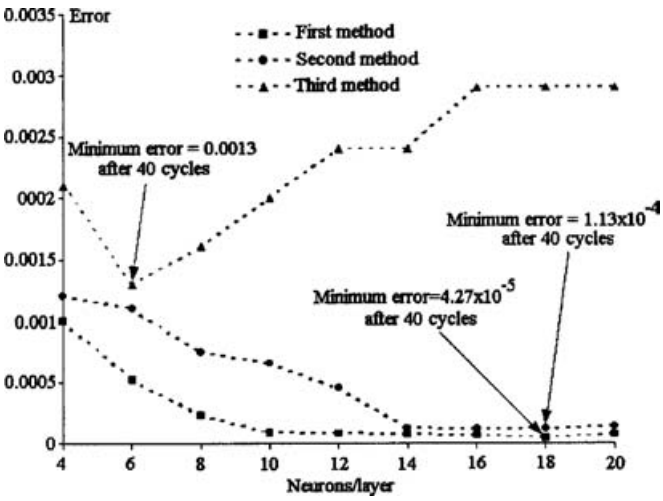


Fig. 8. Optimized architecture of the neural network for first order algorithm.

https://doi.org/10.1017/S0263574705001992 Published online by Cambridge University Press

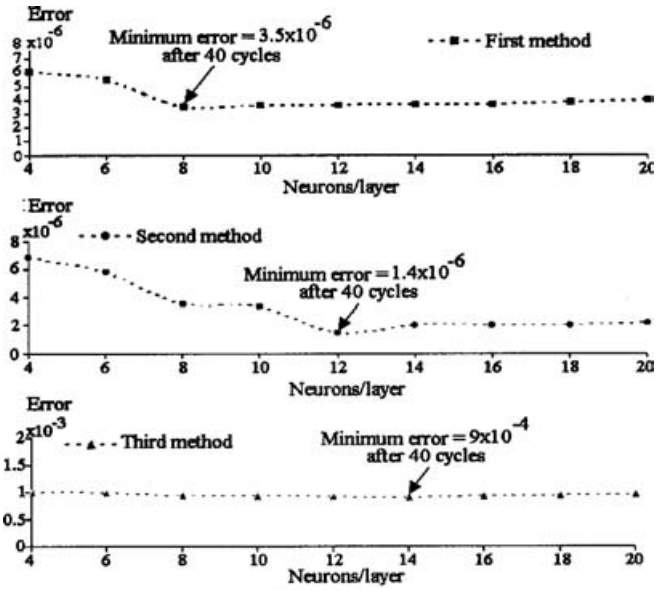


Fig. 9. Optimized architecture of the neural network for second order recursive algorithm.

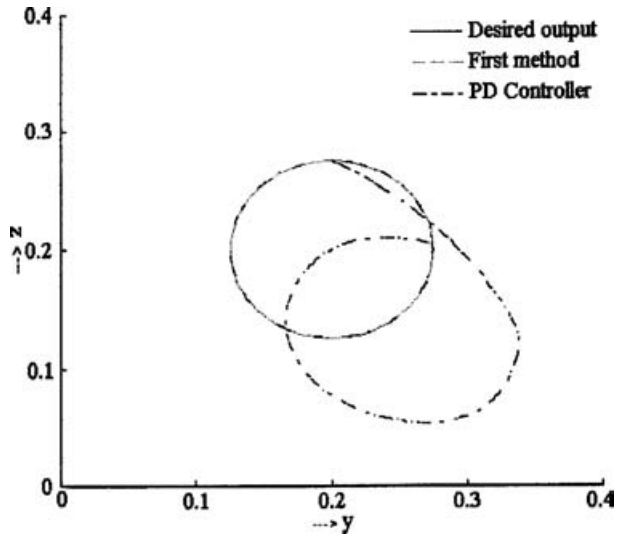


Fig. 10. End-effector path in the yz Cartesian space for second order recursive algorithm.

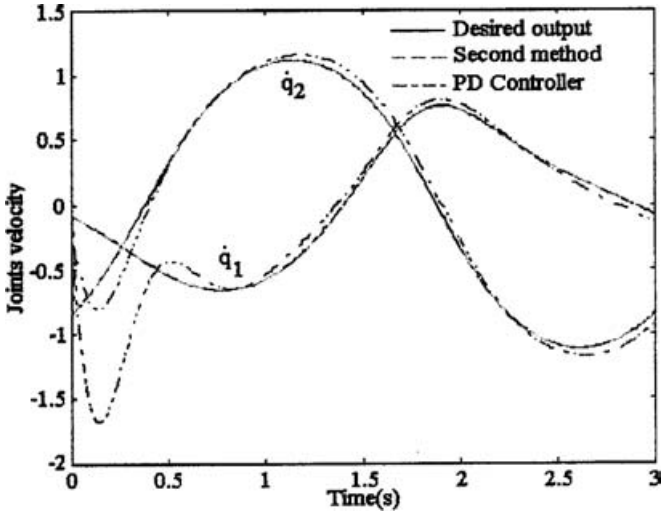


Fig. 11. Desired and actual joints velocity trajectories for second order recursive algorithm.

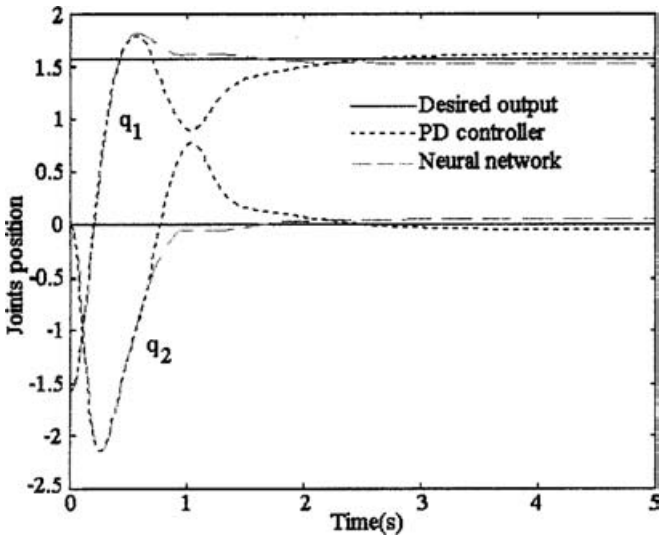


Fig. 12. Desired and actual joint position responses using friction control.

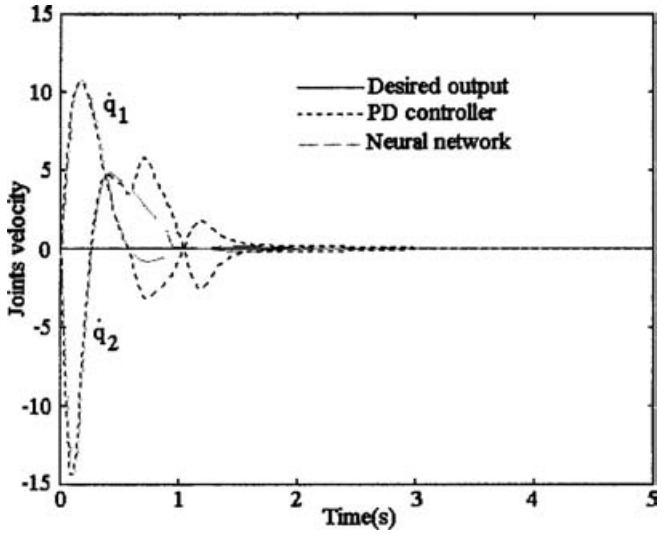


Fig. 13. Desired and actual joint velocity responses using friction control.

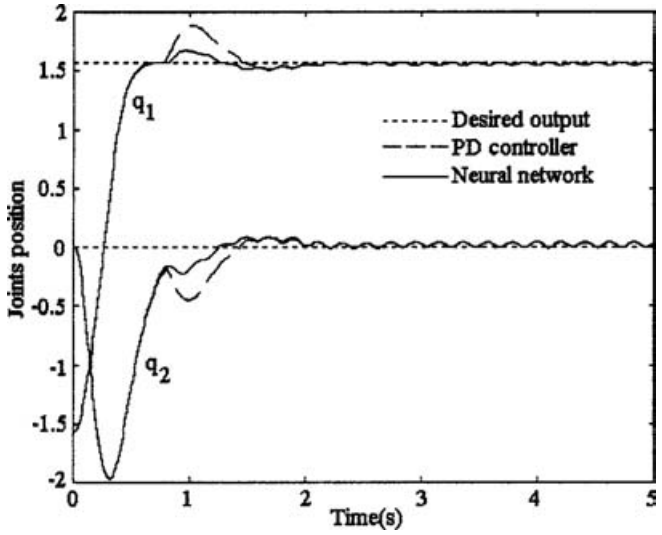


Fig. 14. Desired and actual joint position responses.

https://doi.org/10.1017/S0263574705001992 Published online by Cambridge University Press

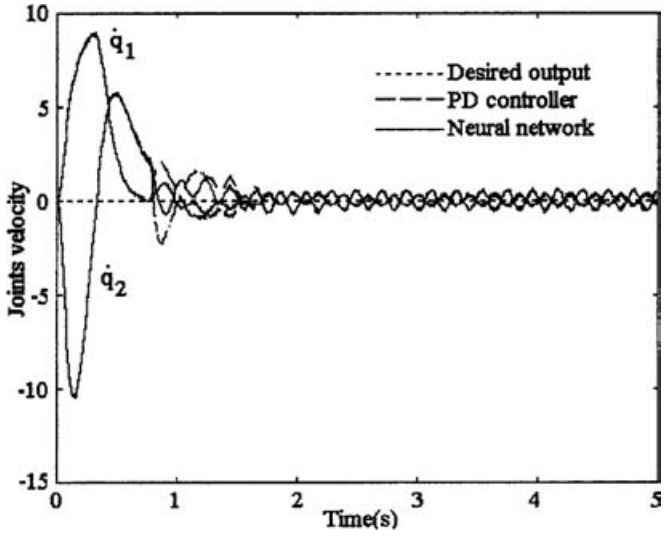


Fig. 15. Desired and actual joint velocity responses.

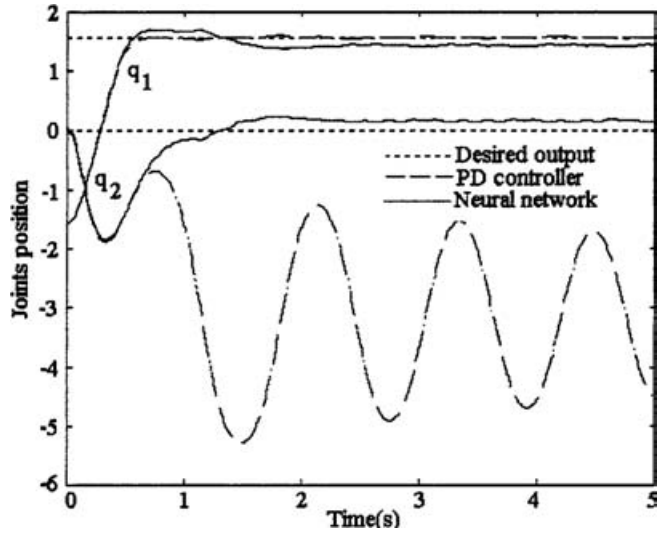


Fig. 16. Desired and actual joint position responses for control with Payload.

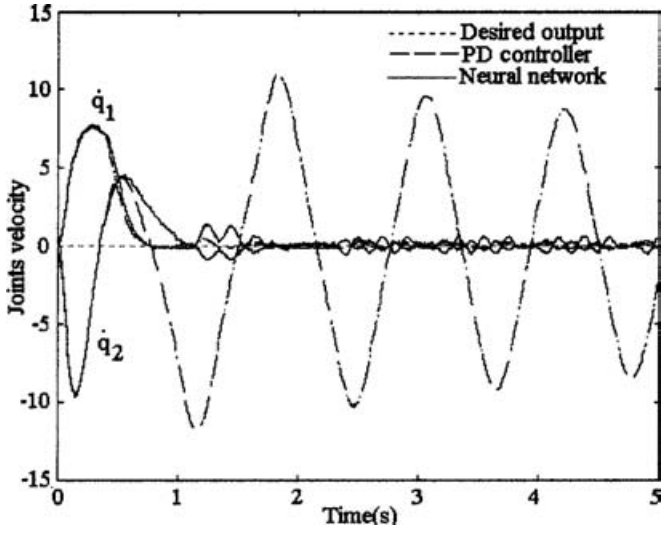


Fig. 17. Desired and actual joint velocity responses for control with payload.

ANN together with the model to compensate uncertainties. The second technique[11] calculates the torque vector as the sum of the ANN output and the PD control signal. A comparative study between the first order and second order recursive algorithms on circular trajectory for three control methods shows that the second order recursive algorithm is better in convergence for all optimal architectures (Figures 8, 9, 10 & 11). For the Pendubot, we show during simulations (Figures 12 & 13) and experimental study (Figures 14, 15, 16 & 17) that ANN techniques minimize the tracking error and stabilize the system in case of uncertainties.

III. CONCLUSION

A second order recursive learning algorithm has been developed and the robustness of this algorithm has been proved in comparison with the first order learning algorithm for three control methods. During the experimental phase, we show the efficiency of ANN to stabilize the pendubot in case of uncertainties. We conclude that ANN is a very important technique which must be generalized in robotics and automation in order to assure stability and better tracking performance.

References

1. A. S. Poznyak, W. Yu, W. N. Sanchez and J. P. Perez, "Nolinear adaptive trajectory tracking using dynamic neural networks," *IEEE Trans. Neural Network* **10**, 1403–1411 (1999).
2. L. L. Whitcomb, A. A. Rizzi and D. E. Koditschek, "Comparative experiments with a new adaptive controller for robot arm," *IEEE Trans. Robotics and Automation* **9**(1), 59–69 (1993).
3. J. S. Reed and P. A. Ioannou, "Instability analysis and robust adaptive control of robotic manipulators," *IEEE Trans. Robotics and Automation* **5**(3), 381–386 (1989).
4. S. S. Ge, T. H. Lee and C. J. Harris, *Adaptive Neural Networks Control of Robotic* Manipulators, *World Scientific Series in Robot and Intelligent Systems* (World Scientific Publishing, 1998) **Vol. 19**.
5. M. W. Spong and D. J. Block, "The pendubot: A mechatronic system for control research and education," *34th IEEE conf. on Decision and Control*, New Orleans (1995) pp. 555–556.
6. M. W. Spong, "The control of underactuated mechanical systems," *Plenary Lecture at the first International Conference on Mechatronics*, Mexico city (1994) pp. 26–29.
7. R. A. Jacobs and M. I. Jordan, "Learning piecewise control strategies in a modular neural network architecture," *IEEE Trans. on Systems, Man and Cybernetics* **23**, 337–345 (1993).
8. S. Jung and T. C. Hsia, "New neural network control technique for non-model based robot manipulator control," *IEEE Trans. on Systems, Man and Cybernetics* **1**, 2928–2933 (1995).
9. S. Haykin, *Neural Networks a Comprehensive Foundation*, (Prentice Hall, 2nd edition, 1998).
10. A. Ishiguro, T. Furuhashi, S. Okuma and Y. Uchikawa, "A neural network compensator for uncertainties for robotics manipulators," *IEEE Trans. on Industrial Electronics* **39**(6), 565–570 (1992).
11. S. Jung and T. C. Hsia, "On reference trajectory modification approach for Cartesian space neural network control of robot manipulators," *Proc. of IEEE International Conference on Robotics and Automation*, Nagoya (1995) pp. 575–580.

https://doi.org/10.1017/S0263574705001992 Published online by Cambridge University Press