Neural Network NARMA Control of a Gyroscopic Inverted Pendulum

F. Chetouane and S. Darenfed

Abstract— **The objective herein is to demonstrate the feasibility of a real-time digital control of an inverted pendulum for modeling and control, with emphasis on nonlinear auto regressive moving average based neural network (NARMA). The plant of interest is a novel Gyroscopic Inverted Pendulum (GIP) device that is nonlinear and open-loop unstable. The GIP balances a pendulum on its free knife-edge base using a flywheel driven by DC motor fixated on the top. In this application, an indirect data-based technique is taken, where a model of the plant is identified on the basis of input-output data and then used in the model-based design of a NARMA controller. The plant under digital PID control with I-adaptation provides initial stability at the beginning of a single layer NARMA neural network training. NARMA models of increasing complexity are used successively to generate input-output data for the training of multilayered NARMA models. In using a NARMA neural network the control laws are nonlinear and online adaptation of the model is possible to capture un-modeled or time-varying dynamics. Such an environment provides for experimentation, data collection, system identification and real time control strategy implementation.**

Index Terms— **Closed loop identification, feedback linearization, gyroscopic inverted pendulum, neural network NARMA control, real-time.**

I. Introduction

Most practical systems exhibit nonlinear behaviour. The three-term PID control system provides satisfactory performance, when operated about the point where local linearity holds. The resulting system performance, under a standard PID controller with fixed gains, is reduced when the controller operates over a wide region about the point of tuning. One approach to alleviate this problem is to use a nonlinear controller to improve consistency in terms of performance over a range about the point of tuning. The inverted pendulum (IP) remains an interesting plant to control engineers in terms of nonlinear behaviour coupled with its physical simplicity along with complete instability. It is widely accepted as an adequate model of a human standing still. Moreover, various types of IPs are common in academia for the study of controller design, e.g. LQG, PID, Fuzzy Logic (FL), Genetic Algorithms (GA) and Artificial Neural Networks (ANN) etc, alone or in combination. Most of the pendulums developed so far, have restoring force(s) applied somehow at the fulcrum. Various linearization techniques can be used to account for nonlinearities, such as linear compensators based on Jacobian linearization. Similarly, approximate linearization was used effectively to design a controller for an inverted pendulum [1]. Some authors have considered an alternative control action consisting of an oscillatory vertical force applied to the pendulum pivot [2]. The stabilizing effect of a fast vertical oscillation applied to the pendulum base is known from the early work of Stephenson [3]. Another control alternative is based on the application of a rotational torque to the pendulum base, as proposed in [4]. Recently hybrid LQG-Neural Controller has been studied in [5]. IPs with higher degrees of freedom are the plant of choice for control of MIMO systems [6], [7], [8], [9]. Humans manage to balance the pendulums intuitively, by applying actuation at the fulcrum, and their complicated counterparts. But applying actuation at the top of the pendulum is a novel idea, as this is somehow most biped creatures walk and balance in everyday life. Often a person spreads their arms and rotates them rapidly to balance themselves and keep from falling. There is always a process of learning various techniques based on previous set goals to balance the pendulum in vertical position. In the presented case, the GIP's fulcrum is kept in a groove so that it is only free to move on either side. Also the restoring torque is applied through a DC motor-flywheel fixated at the top. It is a freestanding pendulum where it is swung around the fulcrum to achieve stability (see Figure 1). The GIP has much less actuating power making it *a weak system*. The torque depends on the gyroscopic movement of the flywheel, where the DC motor has to be run in a min-max voltage limit (±10 V). This configuration makes the GIP an interesting and novel plant for the design of new class of controllers.

In modeling a DC motor connected to a load via a shaft, the general approach is to neglect the nonlinear effects and build a linear transfer function representation for the input–output relationship of the DC motor and the load it drives. This assumption is satisfactorily accurate as far as conventional control problems are concerned. However, when the DC motor driven flywheel operates at various speeds and rotates in two directions, the assumption that the nonlinear effects on

Manuscript received April 9, 2008. This work was supported by The National Sciences and Engineering Research Council of Canada under Grant 249484-02.

Prof. F. Chetouane is with the Electrical and Industrial Engineering Department at Université de Moncton, NB, E1A3E9 CANADA (phone: 506-863-2074; fax: 506-858-4082; e-mail: chetouf@ umoncton.ca).

Prof. S. Darenfed is with the Mechanical Engineering Department at Université de Moncton, NB, E1A3E9 CANADA (phone: 506-863-4027; fax: 506-858-4082; e-mail: darenfs@ umoncton.ca).

(Advance online publication: 20 August 2008)

the system are negligible resulted in poor control performance for the GIP. Indeed, efforts to use a transfer function based approach to design a classical PID controller resulted in poor stability. The remainder of this manuscript is organized as follow: in the next section the nonlinear dynamics of GIP is given. In section III, approximate linearization via feedback is considered in the context of a NARMA based methodology. A closed loop identification scheme is presented whereby NARMA controllers of increasing complexity are synthesized in section IV. Experimental results are given in section V, followed by concluding remarks.

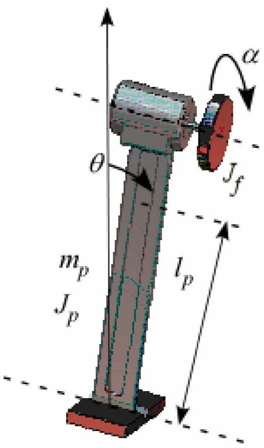

Fig. 1. The GIP is a free standing pendulum. The fulcrum is a single punctual contact at the base. A V-shaped groove allows one degree of freedom. No bearings are used.

II. GIP Dynamical Model

A. Nonlinear Dynamical Model

The GIP variables are defined as follows: angular position from vertical (θ), motor's current (i), motor's input voltage (u), flywheel angular position (α), motor-flywheel system generated torque (T_f), and the gravitational torque acting on GIP's center of gravity (T_g). The gravitational acceleration is (g). The GIP model is given by the following equations:

$$L\frac{di}{dt} + R\,i = u - K\frac{d\alpha}{dt} \quad (1)$$

$$J_f\frac{d^2\alpha}{dt^2} + b\frac{d\alpha}{dt} = K\,i \quad (2)$$

$$T_f = K\,i \quad (3)$$

$$T_g = m_p\,g\,l_p\,\sin(\theta) \quad (4)$$

$$T_f - T_g = J_p\frac{d^2\theta}{dt^2} \quad (5)$$

Where $L, R, K,$ and b are respectively, the inductance, the electric resistance, the torque constant and the friction coefficient of the DC motor. Pendulum parameter's are m_p, J_p and l_p corresponding respectively to the mass, the moment of inertia around fulcrum, and the effective length between fulcrum and centre of gravity. The moment of inertia of the flywheel is J_f.

Equations (1) to (3) describe the motor-flywheel system. Equation (4) describes the non-linear gravitational torque that tends to destabilize the pendulum (gravitational pull). Equation (5) describes the net torque that governs the GIP precession. The above equations lead to the two main equations governing the dynamic of the GIP:

$$\left(\frac{LJ_f}{K}\right)\frac{d^3\alpha}{dt^3} + \left(\frac{Lb+RJ_f}{K}\right)\frac{d^2\alpha}{dt^2} + \left(\frac{Rb+K^2}{K}\right)\frac{d\alpha}{dt} = u \quad (6)$$

$$J_p\frac{d^2\theta}{dt^2} + m_p l_p g\sin(\theta) = J_f\frac{d^2\alpha}{dt^2} \quad (7)$$

From equation (6) one can notice that u appears after three derivation of α. The later variable has the same order dynamic as the output θ (equation 7). Therefore, the relative degree of the system is $d = 3$. Information about relative degree is useful to determine the structure of the plant neural network model. The total number of state variables from (6) and (7) is five. This number represents the order of the GIP system.

B. Approximate Linearization via Feedback: an indirect data-based approach

The idea of using feedback to enhance system linearity has motivated the search for feedback compensators able to make the dynamic behaviour of the closed-loop system as close as possible to that of a linear reference model. Model matching problems can be tackled by either *model* or *data-based* techniques, according to whether the available description of the plant is a mathematical model or just a set of I/O (real or simulated) data. Data-based techniques may be classified as *direct* or *indirect*. Indirect data-based technique is a two-step methodology, where a model of the plant is identified on the basis of input-output data and then used in the model-based design of a suitable compensator. Direct data-based design techniques attempt to provide a suitable tuning of the compensator parameters, without explicitly identifying a model of the plant. An indirect data-based approach is used in this work.

The NARMA model is an exact description of the input-output behaviour of a finite dimensional nonlinear discrete time plant in a neighbourhood of an equilibrium point. It often leads to mathematically intractable nonlinear control equations and is therefore approximated by ARMA models for tractability. Though adequate for most applications [10], [11], [12], the ARMA model is only accurate for non-affine plants with small input magnitudes. To relax this restriction, NARMA-L2 was recently introduced as approximations of NARMA models with inputs larger than permitted with ARMA models [13]. They are nonlinear with respect to past outputs but linear with respect to the current input and therefore suitable for control design.

III. NARMA-L2 CONTROL

The control technique described in this section is refereed to by feedback linearization control or NARMA-L2 control. It is referred to as feedback linearization when the plant model has a particular form (companion form) and as NARMA-L2 control when the plant model can be approximated by a companion form. The first step in using NARMA-L2 control is to identify the system to be controlled. One standard model that has been used to represent general discrete-time nonlinear systems is the nonlinear autoregressive moving average (NARMA) model:

$$y(k+d) = \mathrm{F}[y(k), y(k-1),..., y(k-n+1), u(k), u(k-1),..., u(k-m+1)] \quad (8)$$

where $u(k)$ is the system input, $y(k)$ is the system output and d is the relative degree. The positive integers m and n are respectively the number of measured values of inputs and outputs. Multilayer neural networks can be used to identify the function F[.]. Denoting the network mapping by N[.] the identified model has the form:

$$\hat{y}(k+d) = \mathrm{N}[y(k), y(k-1),..., y(k-n+1), u(k), u(k-1),..., u(k-m+1)] \quad (9)$$

where $\hat{y}(k+d)$ is the estimate of $y(k+d)$. The system output is usually constrained to follow a reference trajectory $y_r(k+d)$. For stable operation of the GIP the reference trajectory is $y_r(k+d) = \theta_r(k+d) = 0, \forall k \geq 0$. Given the reference trajectory and equation (8), the control input $u(k)$ necessary to maintain the output on its reference trajectory is:

$$u(k) = \mathrm{G}[y(k), y(k-1),..., y(k-n+1), y_r(k+d), u(k-1),..., u(k-m+1)] \quad (10)$$

The adjustments of the parameters of the neural network approximating G cannot be achieved during plant real time control using static back propagation. The dynamic of back propagation is slow and computationally demanding. One solution, proposed in [13], [14], is to use a Taylor expansion of F[.] around the input. The model is given by:

$$y(k+d) = f[y(k),..., y(k-n+1), u(k-1),..., u(k-m+1)] + g[y(k),..., y(k-n+1), u(k-1),..., u(k-m+1)] \cdot u(k) \quad (11)$$

Form (11) allows solving for the control input that brings the system output to follow the reference trajectory. The resulting theoretical controller is:

$$u(k) = \frac{y_r(k+d) - f[y(k)..., u(k-1).....]}{g[y(k)..., u(k-1).....]} \quad (12)$$

The controller form in (12) can not be realizable because

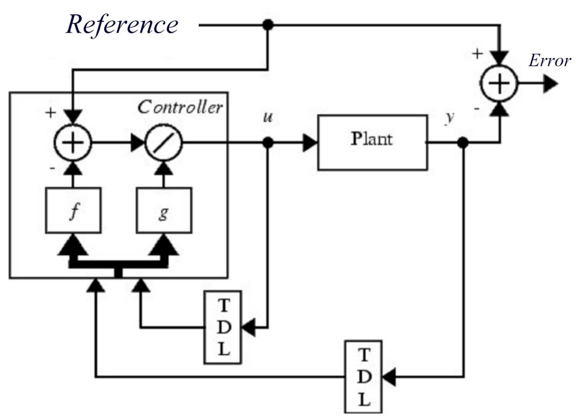

Fig. 2. The plant model is determined by two neural networks approximating functions *f* and *g* respectively. Input and output values used for approximation are continuously recorded by tapped delay lines (TDL)

input $u(k)$ computation requires the output signal $y(k)$ occurring at the same time. A more practical form is given by (13). This controller is realizable for $d \geq 2$. The controller structure is shown in Figure (2).

$$u(k+1) = \frac{y_r(k+d) - f[y(k)..., u(k).....]}{g[y(k)..., u(k).....]} \quad (13)$$

IV. APPLICATION TO IPNC CONTROL

In this section, the NARMA-L2 control technique is applied to the control of a GIP plant called the Inverted Pendulum New Class IPNC [15]. The IPNC is available with its analog PID controller (see Figure 3). The PID controller ensures stabilization using set-point adaptation technique. This consists in integrating the error value (difference between set-point and the actual angular position of pendulum), and in using it to dynamically alter the set-point given by the user. Figure 4 shows the control model in the form of a Simulink™.

The first step is an identification of the plant model. The neural network is trained to represent the forward dynamics of the system. Since the GIP is open loop unstable a Proportional-Derivative with set point adaptation was implemented based on the analog PID at hand. Therefore, the

Simulink model contains two controllers: PID controller and the designed NARMA-L2 controller.

Fig. 3. IPNC plant is a GIP with a an analog PID controller

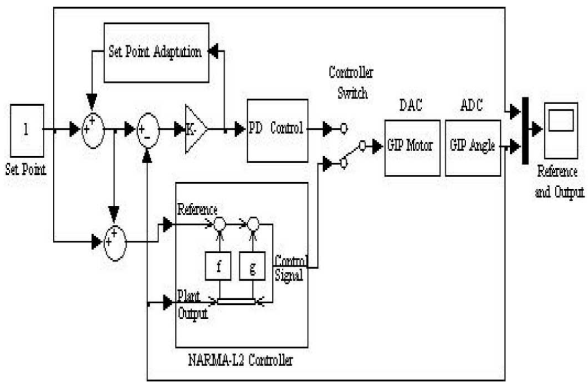

Fig. 4. Simulink identification and control model: the switch allows shifting to PID controller with integral adaptation for data collection and training. Once training is complete the NARMA-L2 controller can be selected at any time during GIP real time control. The ADC and DAC are from Wincon Quanser real time control library.

A switch block allows shifting from one controller to another at any time during GIP operation. Here the method of closed-loop system identification is used since plant input–output data are available in the form of PID control signal (GIP input voltage) and angular position (GIP output voltage with 120 volts/radian sensor transfer gain) respectively. It has been shown that for systems where both the output signal and plant input are measurable, and information on the linear regulator that relates plant input with the plant output is available; the estimates that result from the direct method and the indirect method are identical [16].

V. Experimental Results

The first set of training data was obtained under PID control of the GIP and is shown in Figure 5. Training data collection was achieved at a sampling rate of 5 kHz using 16 bit A/D channel for the angular displacement of the pendulum and 12 bit D/A channel for the control of the flywheel DC motor. A controller NARMA(5,2,1), consisting of 5 delayed output $y(k), y(k-1),..., y(k-4)$ (θ angular position), 2 delayed input $u(k), u(k-1)$ (PID control signal) and a one (1) layer was trained in batch on data obtained under PID (P=6, D=0.1 and I=0.71). A recursive Levenberg–Marquardt minimization method is used [17]. It is an intermediate method between the steepest descent and Gauss–Newton, and it has good convergence properties. The set of data is divided into a training set, a testing set and a validation set in order to avoid over fitting. A mean square error of 1.71E-05 was achieved after 50 epochs. The identified NARMA(5,2,1) is then used for real-time control of the GIP and the results are shown in Figure 6.

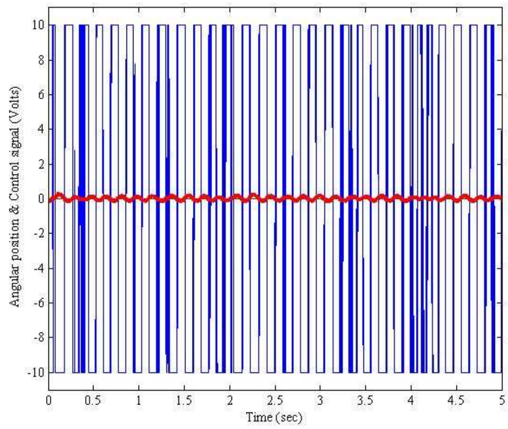

Fig. 5. PID control signal (+/- 10Volts) and sensor signal of the angular displacement around the vertical.

To improve the performance of the learning controller a strategy of incremental model complexity is adopted. Here complexity is defined in terms of the number of hidden layers L . Closed loop training data was generated using NARMA(n , m , L) and then used for the identification of a

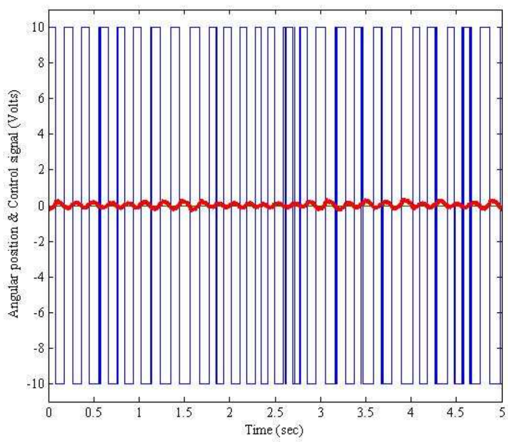

Fig. 6. NARMA(5,2,1) control signal (+/- 10Volts), and sensor signal of the angular displacement around the vertical.

NARMA(n , m , $L+1$), consisting of n delayed output, m delayed input and $L+1$ layers. Figure 7 illustrates the improvement in the performance of NARMA(5,2,3) over NARMA(5,2,1). In Figure 8, the IPNC-GIP plant is under PID control for the first 5 seconds then switched to NARMA(5,2,3) thereafter resulting in a clear improvement of the steady state error. When shifting controller at time 5 seconds, the NARMA controller shows a better voltage delivery to the DC motor compared to the PID controller and no oscillations where observed at stable position.

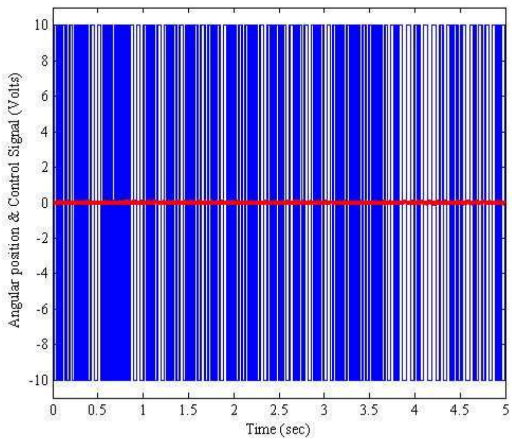

Fig. 7. NARMA(5,2,3) control signal (+/- 10Volts), and sensor signal of the angular displacement around the vertical.

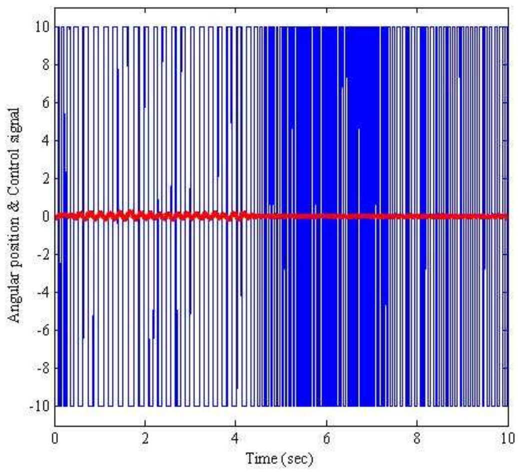

Fig. 8. PID control for 5 first seconds followed by NARMA(5,2,3) control.

The mean square error (MSE) method is most commonly used for model validation purposes and is shown in Table I. MSE was computed using validation data set, not used during the training and testing phase. It shows that the increasing complexity method consisting in training actual network using previous network generated data allows better performance when increasing gradually layer number L .

TABLE I
MEAN SQUARE ERROR

L	1	2	3	4	5	6
$MSE \times 10^5$	1.710	0.210	0.085	0.050	0.030	0.021

By increasing the number of layers from 1 to 6 the error was reduced by 80 times.

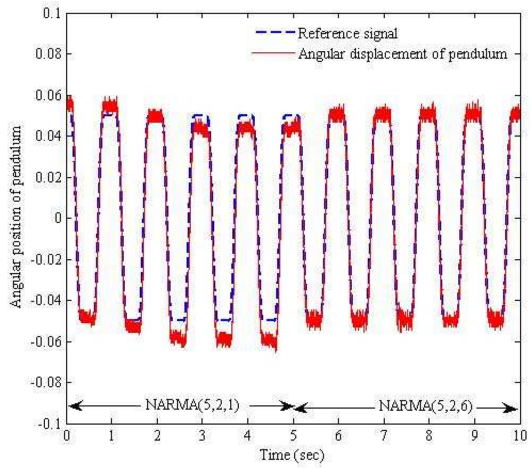

Fig. 9. Performance and generalization abilities for NARMA with increasing complexity.

The improvement of performance by reducing MSE allows better network generalization for time varying reference trajectory as shown by Figure 9. The NARMA(5,2,6) achieves better control of the IPNC-GIP plant when compared to the initial 5 seconds control under NARMA(5,2,1).

VI. CONCLUSION

Nonlinear identification of a GIP in closed loop has been performed and discussed in the paper. The two-directional low speed operation of the system magnified the effects of the nonlinearities, and the significance of the nonlinear approach to the identification problem. The GIP plant introduces a novel way of balancing an inverted pendulum by the gyroscopic action of a flywheel, and its real time control using NARMA-L2 neural network is achieved using Matlab-Simulink toolbox [18]. A strategy identifying successive models with increasing complexity is implemented. The learning controller displays generalisation ability since it is trained on fixed set point (zero) yet it is capable to keep the GIP under control when presented with a varying set point reference signal.

Experiments were conducted in closed loop under linear regulator for initial training data collection. Various NARMA-L2 controllers were designed and results were graphically and numerically compared at various set point conditions. The NARMA approach exhibits a much better identification (lower MSE) around the vertical where the nonlinearities are more effective (friction and low speed). The overall accuracies of the NARMA-L2 models were compared using the MSE criterion. The results show that the learning approach is more accurate not only at around zero set point but also around the vertical position of the pendulum (±10 degrees).

The experimental study given in the present paper is intended to constitute a basis for the ongoing study on adaptive control of mechanical systems using a nonlinear approach. The nonlinear nature of neural networks gives them an advantage over linear models in the prediction of non-linear systems. The initial closed loop identification required that the GIP be stabilized using a PID control law. The control law removes some of the nonlinearities from the plant, so a detuned control law is used which allows the GIP to exhibit more of its dynamics. This improves the quality of the data used in the system identification of NARMA models.

Further improvements are being considered, namely training of more robust non-linear controller using genetic algorithm as a network parameter search technique. Designing an effective controller for a weak system which can balance the GIP in other positions around the vertical position is a challenging area of control research. Furthermore, two more degrees of freedom can be realized by adding a controlled movable neck at the DC motor and pendulum body joint, so that the resulting pendulum will have three degrees of freedom, making suitable for tackling MIMO systems.

REFERENCES

[1] T. Sugie, and K. Fujimoto, "Controller design for an inverted pendulum based on approximate linearization," *Int. J. of robust and nonlinear control*, vol. 8, no 7, pp. 585-597, 1998.

[2] P.L. Kapitsa, "Dynamical stability of a pendulum with an oscillating suspension point," *Zh. Eksp. Teor. Fiz.*, vol. 24, # 5, pp. 588-597, 1951.

[3] A. Stephenson, "On a new type of dynamical stability," *Mem. Proc. Manch. Lit. Phil. Soc.*, vol. 52, no 8, pp. 1-10, 1908.

[4] I. Fantoni, and R. Lozani, "Nonlinear control for underactuated mechanical systems," *Appl. Mech. Rev.*, vol. 55, # 4, pp. 67-68, 2002.

[5] E.S. Saznov, P. Klinkhachorn, and R.L. Klein, "Hybrid LQG-Neural controller for inverted pendulum system," in *2003 Proc. 35th Southeastern Symposium on System Theory*, Morgantown, WV, 2003.

[6] A. Asgarie-Raad, "Intelligent control of two dimensional inverted pendulum," M.S. thesis, Sharif University of Technology, Iran, 1998.

[7] M. Esmailie-Khatier, "Construction and control of a two-dimensional inverted pendulum," M.S. thesis, Sharif University of Technology, Iran, 1994.

[8] A. Ghanbarie, "Neuro-fuzzy control of two dimensional inverted pendulums," M.S. thesis, Iran University of Science and Technology, Iran, 2000.

[9] P.J. Larcombe, "On the control of a two-dimensional multi-link inverted pendulum: the form of the dynamic equations from choice of co-ordinate system," *Int. J. Syst. Sci.*, vol. 23, no 12, pp. 2265-2289, 1992.

[10] R. Boudreau, S. Darenfed, and C. Gosselin, "On the computation of the direct kinematics of parallel manipulators using polynomial networks, " *IEEE transactions on Systems, Man, and Cybernetics*, vol. 28, no 2, pp. 213-220, March 1998.

[11] R. Boudreau, G. Levesque, and S. Darenfed, "Parallel manipulator kinematics using holographic neural models, " *Journal of Robotics and Computer-Integrated Manufacturing*, vol. 14, no 1, pp. 37-44, February 1998.

[12] R. Boudreau, S. Darenfed, and E.C. Biden, "Application of time series and polynomial learning networks to robot trajectory error control," *Journal of Robotics and Computer-Integrated Manufacturing*, vol. 12, no 1, pp. 73-79, March 1996.

[13] K.S. Narendra, and S.M. Mukhopadhyay, "Adaptive control using neural networks and approximate models," *IEEE Transactions on neural networks*, vol. 8, no 3, pp. 475-485, May, 1997.

[14] K.S. Narendra, and K. Parthasarthy, "Identification and control of dynamical systems using neural networks," *IEEE Transactions on neural networks*, vol. 1, no 1, pp. 4-27, March, 1990.

[15] Extra Dimension Technologies (http://www.xdtech.com), Inverted Pendulum New Class: operating manual, 2001.

[16] U. Forssell, and L. Ljung, "Closed-loop identification revisited," *Automatica*, vol. 35, pp.1215–1241, 1999.

[17] J. Lampinen, "Advances in neural network modeling," in *1997 Proc. the Tool environments and development methods for intelligent systems*, Oulu, Finland, pp. 28-36.

[18] Neural Networks Toolbox User's Guide, Version 3.0, The Mathworks.