INTELLIGENT CONTROL OF AN INVERTED PENDULUM - TRAINING AND EVOLUTION

José Homero Feitosa Cavalcanti
UFPB/CCT/DEE
Av. Aprigio Veloso, 882
58.109-970 Campina Grande - PB
Fone: (083) 333-1000 Ramal 442
E-mail: homero@dee.ufpb.br

Edilson Ferneda
UFPB/CCT/DSC
Av. Aprigio Veloso, 882
58.109-970 Campina Grande - PB
Fone/Fax: (083) 333-1698
E-mail: edilson@dsc.ufpb.br

ABSTRACT

In this paper, the characteristics of a intelligent control system - ICS, based in fuzzy rules and neural networks controllers, and the use of d.c. motor real time control are described. The ICS is considered the main one and can evolve. The neural network controller is the slave and the fuzzy rules are used to help to adapt the neural controller parameters during the learning phase. The experimental results obtained with the pendulum arm direct and oscillating movement for its equilibrium in the up right position, is also shown. Actually, this ICS is one testbed for the AI research group of the Federal University of Paraíba (UFPB/CCT/DSC-DEE).

1 - Introduction

The intelligence may be defined as the ability of a system to act in a suitable way in uncertain environment, where an appropriate action increases the probability of success, and success is the achievement of behavioral sub goals that support the ultimate goal of the system. Intelligence can be observed to grow and evolve, through increased computational power, and through accumulation of knowledge of how to sense, decide, and act in a complex and changing world. In natural systems, intelligence grows, over the lifetime of an individual through maturation and learning, and over intervals spanning generations, through evolution.

There are problems of control which can not be formulated and studied by using differential/difference equation mathematical framework. Sometimes there are incomplete, uncertain and partially correct information about the system state that can be used and reevaluated using new information or after the system evolution.

An Intelligent Control System can autonomously achieve a high level goal, while its components, such as control goal, plant models and control laws are not completely defined. ICS can address control problems that can not be formulated in the language of conventional control. The area of ICS is in fact interdisciplinary, and it attempts to combine and extend theories and methods from areas such as control, computer science and expert systems, planning, learning, searching algorithms, Petri Nets, neural nets and fuzzy logic. ICS can emulate human mental faculties such as adaptation and learning. The ability to adapt to changing conditions is necessary in an intelligent system. Although, adaptation does not necessarily requires the ability to learn, for systems to be able to adapt to a wide variety of unexpected changes, learning is essential. The
increase in intelligence is achieved by successive introduction of additional intelligent processes which improves the performance of the system by making alterations to the plan and control strategy.

Intelligent control systems using neural network are been used to adaptive control of non-linear dynamic systems. The Expert System in the shell of the ICS can be used to set up initial values for weights and parameters structure of the neural network. It reasons symbolically to produce a control strategy by using "a priori" the knowledgement and the process state using fuzzy rules and the data base systems.

The principal characteristic of a neural controller lies in its inherent capacity for interpolation, requiring a large number of iterations for its training with the plants dynamics. Cavalcanti et alii [1993][1994] have demonstrated that under certain conditions a direct neural controller, based on multi-layer artificial neural network (MLANN) can be trained and it can be adapted on line in real time without any previous off line training. In the controllers based on fuzzy logic and using memberships function, the control variables is calculated employing the fuzzy rules which in general are defined a priori.

Physical systems like an inverted pendulum are frequently utilized as a benchmark process to experimentally demonstrate and test the control algorithms for non-linear systems. The choice of a rigid inverted pendulum arm is basically due to its inherent nonlinear behavior and its stable and unstable positions. Cavalcanti et alii [1994] have presented experimental results for a neural fuzzy controller to position the inverted pendulum by moving it continuously from the most stable position (vertically down) to an unstable position (vertically upright position).

In this paper, the characteristics of a ICS, based in fuzzy rules and neural networks controllers, and the use of a d.c. motor real time hybrid control are described. The experimental results obtained with the pendulum arm direct movement and equilibrium in the up right position is showed. The relation between the terms "learning" and "evolve" in the AI field, with neural network training and fuzzy control strategy in the intelligent control field, are showed. Using these terms, the evolution of the ICS, with new control strategy, to permit free error movement of the arm in a oscillating form is also described.

2 - Neural Controller

In the design of a neural controller for an inverted pendulum the following notation conventions are adopted: $\theta(k+1)$ is the desired reference position for the pendulum arm mounted on the shaft of a d.c. motor; $\theta(k+1)$ is any read angular position of the pendulum arm at any instant; $\Omega(k)$ is the motor shaft speed; $U(k)$ is the voltage of the motor armature; and $D(k+1)$ is the error between the desired reference position and the angular position of the pendulum arm which is given by:

$$D(k+1) = P_d[\sin(\theta(k+1)-\theta(k))]$$  \(1\)
The configuration of a direct neural position controller for the d.c. motor shaft is shown in Figure 1 where \( X_1 = U(k-1), \ X_2 = \Omega(k), \ X_3 = D(k+1), \ X_4 = \sin[2\pi \theta(k+1)] \) and \( X_5 = \sin[2\pi \theta(k+1)] \) are the inputs to the MLANN (all values are on the normalized p.u. basis).

The adaptation performance index for the MLANN with the d.c. motor and the inverted pendulum arm is defined by:

\[
E = \frac{1}{2}(e_{k+1})^2 = \frac{1}{2}[\theta(k+1) - \theta(k+1)]^2
\]  

(2)

The neurons used in the MLANN are the sigmoid (S) and hyperbolic tangent (T) types as given by the following equations:

\[
S(x) = \Theta/[1 + \exp(-\beta(\sum X_{in} W_{in} + T))]
\]

and

\[
T(x) = 2 S(x)/\Theta - 1
\]  

(3)

For the MLANN of Figure 1 to adapt in real time, it is important the knowledge of Jacobean of the plant (i.e. the d.c. motor and the pendulum arm). Cavalcanti et alii [1993] have shown that the parameters of the neural direct controller of the d.c. motor shaft speed can be adapted in real time by making \( \theta \) constant and:

\[
U(k+1) = U(k) + \Delta U(k)
\]  

(4)

and \( \Delta U(k) \) can be calculated using generalized delta rule with:

\[
\Delta U(k) = \eta e(k+1) \partial \theta(k)/\partial U(k) = \mu e(k+1)
\]  

(5)

In equation (5) \( \partial \theta(k)/\partial U(k) \) represents the d.c. motor Jacobean. In the present implementation, the Jacobean of the d.c. motor plus the pendulum it is not known, and thus an approximate value is used.

![Figure 1: Neural controller for d.c. motor shaft position.](image)

For the design of the neural control algorithm it is assumed that the d.c. motor represented by \( x = f(x, u) \), has at least one equilibrium state and that, without loss of generality, the origin may be considered to be in equilibrium state, and the following definition is used:
Definition - A dynamic system is in passive state when it is at equilibrium point with $x_e = f(x_e, u_e)$ and the neural controller guarantees the value $u_e$ of input [Lima et alii, 1994].

As an example, in the passive state with $x=0$, when the inputs for neural controller of figure 1 are zero i.e., $[U(k-1)=\theta r(0)=\theta(0)=D(0)=\Omega(0)=0.]$, its MLANN gives the output $U(k)=0$. In the passive state of the overall system, the following conditions must be satisfied: $|U(k+1)-U(k)| < \xi |\theta r(k)-\theta(k)| < \varepsilon$, when $\xi$ and $\varepsilon$ are sufficiently small.

3 - Positions control of an inverted pendulum

Many authors have investigated the positioning of pendulum arm in the upright position (i.e., stabilizing the inverted pendulum). Batur & Kasparian [1991] have reported some simulation results obtained with a fuzzy controller employed to stabilize an inverted pendulum on a moving car with small variations imposed on the angular deviation of the pendulum from the normal upright position. Ishida etali [1991] have also shown the use of an MLANN for stabilizing the inverted pendulum mounted on a car for small angular perturbations from the normal upright position. Lin & Sheu [1992] presented experimental results with a fuzzy and conventional hybrid controller, for here above type of system. Associating the angular position of the pendulum arm with a particular quadrant, the fuzzy controller rules were defined and additionally a conventional feedback controller is employed for fine tuning the position of the pendulum when it is close to the desired inverted position.

The general schematic of simple pendulum is shown in figure 2. The torque $T_l(k)$ which tends to bring the pendulum arm back to the stable position is given as in equation 6.

$$T_l(k) = (P/g) L^2 \frac{d^2 \theta(k)}{dt^2} + P L \sin(\theta(k))$$

(6)

Figure 2: The schematic of a simple pendulum.

The schematic of the mounting of a rigid arm on the motor shaft (large circle) is shown in figure 3. There is a mass $M$ (small circle) attached to the pendulum arm. A pendulum is said to be an inverted one if it is stable around $\theta=0$ (or $3\pi/2<\theta<\pi/2$).

The arm length $L$ of the pendulum arm is chosen to be 10cm and $P=0.1Kg$ in such way that for certain values of $\theta$, one can make $T_l(k)$ larger then the viscous friction of the motor shaft. Referring to figures 3 and 4, and considering the pendulum to be in a state of rest (stable state), one can make the following observations:
1) With the pendulum in the unstable state with $\theta = 0$ rd (figure 3.a), the smallest ever displacement will make $T_l(k) \neq 0$.

2) With the $\theta = \pm \pi/2$ rd, the $|T_l(k)|$ is maximum (figures 3.b and 3.d).

3) With $\theta = \pm \pi$ rd (figure 3.c), the pendulum is in a stable state. With small displacement around $\theta = \pm \pi$ rd, the pendulum tends to return to the stable state.

4) The absolute value of the torque developed by the motor $T_m(k)$ should be crescent to stabilize the arm in a position between $\theta = 0$ rd and $\theta <= \pm \pi/2$ rd.

5) The absolute value of the torque developed by the motor $T_m(k)$ should be decrescent to stabilize the arm in a position between $\theta >= \pm \pi/2$ rd and $\theta = \pm \pi$ rd.

In figure 4, one can observe the trajectory of the displacement of the pendulum arm has been divided in four quadrant and the sign of the load torque of the pendulum for each of these quadrants is indicated [Lin & Sheu, 1992]. The torque developed in the motor, $T_m(k)$ (directly proportional to the armature voltage) for stabilizing the pendulum arm may have the same magnitude as $T_l(k)$ but opposite in sign. The membership functions for fuzzy levels of $\theta$ in the four different quadrants are given in terms of $\theta_m$.

On the basis of the d.c motor shaft $\theta(k)$ must follows the reference position $\theta_r(k)$ (see the MLANN training algorithm). The value of the armature voltage $U(k)$ is modified in concordance with equations (4) and (5). For the inverted pendulum the exact value of $\Delta U(k)$ is not known. In this present implementation only the sign $\Delta U(k)$ is assumed to be known (based on fuzzy logic).

Based on the principal characteristics of the inverted pendulum (presented in section 4), the torque circle of figure 4a and the $\theta$ membership function figure 4.b, the following fuzzy rules are developed for the MLANN training, followed by the control action ($\theta_m$ represents the membership function of the quadrant of $\theta(k)$):

1) if $\theta_m = Q1$ or $\theta_m = Q2$ then $U(k)>0$;
2) if $\theta_m = Q3$ or $\theta_m = Q4$ then $U(k)<0$;
3) if $\theta_m = Q2$ then $\Delta U(k)<0$;
4) if $\theta_m = Q2$ then $\Delta U(k)>0$;
5) if $\theta_m = Q3$ then $\Delta U(k)<0$;
6) if $\theta_m = Q4$ then $\Delta U(k)<0$;
During the training phase of fuzzy/neural controller, some additional rules were developed in relation to the circle torque of figure 4.a.

1) $\theta r(k) = \pm \pi/2$ rd were not employed since these positions represent maximum values of $T_m(k)$.

2) The MLANN of the neural controller was trained for reference positions close to the stable positions $\theta r(k) = \pm \pi$ rd (figure 5a).

3) The values of $\theta r(k)$ are developed to guarantee the passive states during the training phase of the MLANN.

The training of the MLANN in the quadrant Q2 is equivalent to its training in quadrant Q1. Similarly, the training of the MLANN in the quadrant Q3 is equivalent to its training in quadrant Q4. The use of the function $\sin(\theta)$ at the inputs i.e., $X3 = P_d[\sin(\theta r(k+1) - \theta r(k))]$, $X4 = \sin(2\pi \theta r(k+1))$ and $X5 = \sin(2\pi \theta r(k+1))$ of the MLANN, establishes the equivalence in the training between the positions shown in figure 5.a and 5.b, without the need for development of new fuzzy rules.

The MLANN of figure 1 was trained during 20 times using phase 2 of the proposed algorithm for $\theta(k)$ varying for $\pi - \pi/10$ rd to $\pi + \pi/10$ rd.
MLANN Training Algorithm

PHASE 1 - Initialize the MLANN parameters.
1) The weights Winp and Wout are defined and initial values of neural networks parameters are attributed (θ = β = 1 and T = 0). The MLANN is trained for the zero passive state.

PHASE 2 - MLANN training around the stable point θ=π.
2) Define the values of θr(k) over a small range.
3) Use θr(k) so that it should be possible to calculate new values of U(k) for the passive state of the MLANN.
4) Repeat steps 2 and 3 until |θ(k) - θr(k)|<ε, and θr(k) remains constant and ε is sufficiently small.

The use of the concept of a passive state (refer to algorithm) allows the training of the MLANNs with the dynamics of the pendulum without considering the acceleration term [ML^2 \ d^2\theta(k)/dt^2] of equation 6. For position control the fuzzy variable θp is defined with a membership function θp = G when θ(k) > 2π, and θp = N when θ(k) < -2π. For the velocity control, the fuzzy variable Ωp is defined with a membership function Ωp = G when Ω(k) > 1rps, and Ωp = N when Ω(k) < -1rps (refer to figure 4 c).

7) if θp==G then θ(k) = θ(k) - 2π
8) if θp==N then θ(k) = θ(k) + 2π
9) if Ωp==G then U(k)=0
10) if Ωp==N then U(k)=0

5 - Direct positioning of the inverted pendulum in the upright position

The general schematic of the ICS used by neural controller is showed in figure 6. This ICS uses the algorithm 1 to supervise the on line position control. The initial strategy for stabilizing the pendulum arm, coupled to d.c. motor shaft in the upright position, is to train the controller MLANN with the pendulum arm oscillating around the stable position (θr(k)=π) by making θr(k)=π±V. This is done in such way that the pendulum arm, when moved from θ(k)=π to θ(k)=0, should attain sufficient velocity to overcome the barrier of the maximum load torque Ti(k) for θr(k)=π/2.
In Figure 7 are shown the curves \( (U(k), \Omega(k), \theta_r(k) \text{ and } \theta(k)) \) obtained with the ICS during the on line position control of the pendulum arm. Initially, the pendulum arm is at \( p0 \) with \( \theta(k) = \pi \), it is moved around \( \theta(k) = 0 \) (positions \( p2, p3, p4 \) and \( p5 \) as shown in figure 8). The pendulum arm rests in a position close to \( \theta(k) = 0 \) with error \( E \) as shown in figure 8.

During the experiment, some times, the value of final error \( E \) in the \( p3 \) position was near to the point of maximum load torque for \( \theta(k) = \pm \pi/2 \), and the pendulum arm returned for position close to \( \theta(k) = \pi \). This fail in positioning the pendulum arm is considered an error and must be corrected. This problem is observed after a change in the mass attached to the pendulum arm. Next section shows the evolution of the controller strategy to guarantee positioning the pendulum arm with success.
6 - Positioning the inverted pendulum in the upright position with oscillation

The new positioning strategy, evolved from the direct strategy, should guarantee that: 1) the MLANN learn the actual mass attached to the pendulum arm; 2) the pendulum arm velocity must be sufficient to overcome the barrier of the maximum load torque; 3) the pendulum arm rests close to \( \theta(k)=0 \).

The new strategy for stabilizing the pendulum coupled to d.c. motor shaft in the upright position, evolved from the strategy to direct positioning the pendulum arm, is to train and positioning the controller MLANN with the pendulum arm oscillating around the stable position \( \theta(k)=0 \) by making \( \theta_r(k)=\pi/2 \). This is done in such way that pendulum arm should attain sufficient oscillation to overcome the barrier of the maximum load torque \( T_l(k) \) for \( \theta(k)=\pi/2 \), and get close to the position \( \theta(k)=0 \). At this stage the \( \theta_r(k) \) is making constant and equals to zero. This strategy is considered evolved from the direct positioning strategy.

In Figure 9a, \( \theta(k) \) and \( U(k) \) curves for the oscillating motion of the motor shaft and consequently the pendulum arm are shown with \( \theta(k)=\pi \) [or \( \theta(k)=0.5p.u. \)] until \( \theta(k) \) gets close to zero. In Figure 9b the zoom schematic of the oscillating motion of the pendulum around \( \theta(k)=\pi \) is shown. In figure 5 the real time positions of the pendulum arm are shown. Using this strategy, the final error \( E \) always will be zero.

7 - Conclusion

The design consideration, implementation details and experimental results for the neural control of an integrated d.c. motor with a pendulum arm coupled to it have been presented. The neural controller employs an MLANN and fuzzy rules were used to control the actuator. The MLANN has been trained using the back propagation algorithm for positioning the pendulum arm from stable down position to an unstable upright position. The algorithm for direct positioning of the pendulum arm in up right position has been verified to perform satisfactory. The oscillating positioning strategy of the pendulum arm, evolved after the analysis of the final error between the pendulum arm position and the target obtained with the direct positioning strategy, has shown to be possible to guarantee the successfully positioning of the pendulum arm after a change in its load.
Currently, the Intelligence Artificial group of Federal University of Paraiba studies models and computational techniques (Artificial Intelligence and Software Engineering) for the design, development and testing of systems which have, on one hand, the mechanisms that incorporates new objects (new concepts) treated by the system (language evolution - learning as function) and, on the other hand, the introspective ability with the objective to make evolution of problem resolution methods which are inherent to the domain treated by the system (learning to learning), and to the user interaction (heuristic evolution - learning as property). In [Ferneda at ali, 1992] and [Ferneda, 1992] symbolic methods of automatic learning are presented and a knowledge acquisition protocol and knowledge representation model are proposed. These learning methods are integrated with the objective to make evolution of a knowledge base. Considering the systems where learning is not a function, but a feature of their behavior [Sainte Marie], we believe that the application of conexionist techniques could be useful due to its incremental learning based on examples, adaptatively and generalization properties.

REFERENCES


[Sainte Marie] Sainte Marie C. de "The necessity of learning while doing", Laboratoire d'Informatique Fondamentale et d'Intelligence Artificielle - IMAG, St. Martin d’Heres (França).