Learning Control of a Flight Simulator Stick

An important part of a flight simulator is its control loading system, which is the part that emulates the behaviour of an aircraft as experienced by the pilot through the stick.

Such a system consists of a model of the aircraft that is to be simulated and a stick that is driven by an electric motor. To make the simulation as realistic as possible, the simulator stick should behave in the same way as the stick in the real aircraft.

However, due to the properties of the motor and the stick, small irregularities can be felt when the stick is moved, which do not occur in a real aircraft. Probable causes of these irregularities are cogging in the motor and small imperfections in the transmission.

Both disturbances have a reproducible nature. Because the disturbances are reproducible, feedback error learning control is used for control. The learning controller consists of two neural networks. One neural network is used to compensate the unknown friction and is operated in feed-forward. The other neural network compensates cogging and imperfections in the transmission and is operated in feedback. Experimental results showed that the learning controller is able to compensate the disturbances.

Keywords: intelligent control, neural control, adaptation, spline networks, feedback error learning, flight simulator

Wubbe J.R. Velthuis
Theo J.A. de Vries

University of Twente
Drebbel Institute for Systems Engineering

Koen H.J. Vrielink

Philips Research Laboratories, Eindhoven - NL

G.J. Wierda
André Borghuis

Fokker Control Systems, Schipho - NL

University of Twente
Drebbel Institute – EL-RT
PO Box 217 – NL-7500 AE Enschede
Tel: +31.53.489 28 17 - Fax:+31.53.489 22 23
e-mail: icontrol@rt.el.utwente.nl
http: www.rt.el.utwente.nl/icontrol/

1 Introduction

A flight simulator is a cost-effective and safe way to train a pilot in handling an aircraft. The simulator enables the pilot to experience a broad range of flight situations without running the risks involved when using a real aircraft.
An important part of a flight simulator is the so-called control loading system. This system comprises the command stick of the simulator and the hard- and software connected to this stick which emulate the behaviour of an aircraft as experienced by the pilot through the stick. In this research we consider a specific realisation of such a control loading system.

The plant part of the set-up consists of the following components (fig. 1):
- The *stick*, which has one rotational degree of freedom. Its angular position φ is measured by a potentiometer.
- A *spindle* and a *ball-screw* that together form the transmission between motor and stick.
- A *PMDC electric motor*. A tachometer is mounted on the motor for measuring its angular velocity. As the transmission between the motor and the stick is considered stiff, the angular velocity of the stick, ω, depends on the angular velocity of the motor in a linear way.
- An electric *power amplifier* (not shown in fig. 1).
- A *force sensor* that measures the force F_{ext} applied on the stick by the pilot.

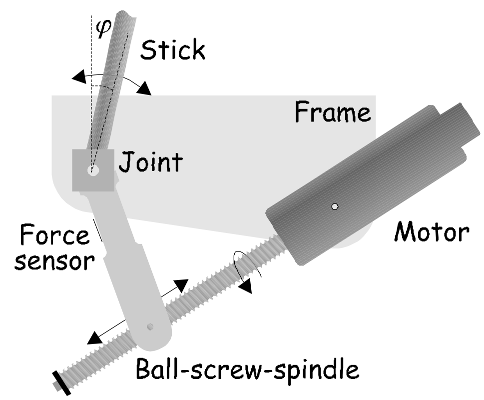

Fig. 1. Plant part of the set-up

By controlling the plant appropriately, the 'mechanical impedance' the pilot experiences at the stick can be controlled.
This is done as shown in fig. 2.
The motor is contained in a velocity loop with a PI controller, that is incorporated in the off-the-shelf amplifier. This loop is called the inner loop. The commanded stick angular velocity w_r for the inner loop is calculated by an aircraft controls model. Inputs to this model are the force F_{ext}, exerted on the stick by the pilot and the angular position of the simulator stick, φ.
This forms a secondary MISO loop, called the outer loop.

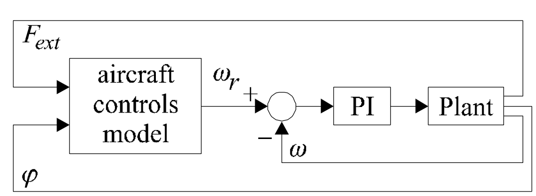

Fig. 2. Electronic Control Loading set-up

To make the simulation as realistic as possible, the impedance that one feels when manipulating the stick should be determined completely by the aircraft model. To achieve this, the inner loop has to compensate for all dynamic properties of the plant; only some (small) rotational inertia is acceptable, as any aircraft stick will feature this phenomenon. However, in the set-up used in this research the inner loop is unable to fully realise this. When the user moves the stick, small irregularities (ticks) can be felt.
Cogging in the motor and/or imperfections in the transmission probably cause these irregularities. In previous research, cogging in a linear motor was successfully compensated using a feedback error learning control system (Otten et al., 1997).
The learning control system consisted of a conventional feedback controller and a neural network that was operated in feed-forward. In this paper, we evaluate whether a feedback error learning control system can be applied in order to improve the behaviour of the control loading system.
In section 2, we motivate the application of feedback error learning in the form of a learning feed-forward controller. The theoretical background of the technique is presented in section 3, and the design choices are discussed in section 4.
The results of the experiments are given in section 5; conclusions follow in section 6.

2 Motivation for feedback error learning control

The most important dynamic properties of the plant that the inner loop has to deal with are:

- *Inertia* of the motor, the transmission and the stick.
- *Cogging force*. The cogging force is the magnetic force between the permanent magnets and the iron in the motor. The magnitude of the cogging force depends on the position of the motor and therefore also on the position of the simulator stick. The relation between the cogging force and angular position of the simulator can be approximated by a sine function. The pilot will experience cogging as a ripple.
- *Friction* in the motor and the ball-screw. It is known that the friction can be approximated by a Stribeck curve.
- *Imperfections* in the transmission.
- *Measurement noise.*

Of these effects, only measurement noise is stochastic in nature; all other effects are reproducible, that is, their momentary value is related to the momentary state of the plant. Inertia is the only linear effect; cogging, friction and transmission imperfections are highly non-linear. Hence, these latter effects and possibly measurement noise are responsible for the irregularities felt by the pilot. The hypothesis formulated at the start of the project was that cogging was the dominant cause for the undesirable effects.
When reproducible non-linear effects in a plant are known accurately, incorporating an inverse model in a feed-forward path can compensate them. In principle, this approach is applicable in the control loading system.
However, as often, it is a difficult and time-consuming undertaking to obtain an inverse model that is accurate enough to guarantee good control performance.
Moreover, each particular set-up will have somewhat different properties, and hence requires a specific inverse model.

To overcome these difficulties, a so-called feedback error learning controller (Kawato et al, 1988; Ng 1997) can be used (fig. 3). In such a controller, an adaptable mapping that is to become the inverse model is placed in parallel with a standard feedback controller.
The mapping is not designed prior to operation, but is trained during on-line control. The training signal is the output of the feedback controller. When the adaptable mapping is implemented as a B spline neural network (Brown and Harris, 1994) the learning control system is known as Learning Feed-Forward Control (LFFC) (Starrenburg et al, 1996).
In previous research (Otten et al., 1997), it has been shown that LFFC can effectively and efficiently be applied to deal with cogging and friction in a linear motor motion system. Therefore, application of LFFC in the inner loop of the electronic control loading system seems attractive in order to improve the system's performance.

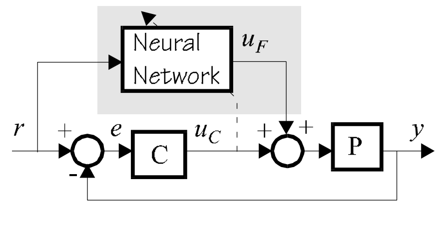

Fig. 3. Learning Feed-Forward Control
(P = plant, C = feedback controller)

3 Learning Feed-Forward Control

As shown in figure 3 the learning feed-forward controller consists of a conventional feedback controller and a neural network that is operated in feed-forward. The *feedback controller* has to compensate stochastic disturbances and has to provide the learning signal for the feed-forward controller. Since the stochastic disturbances are assumed to be much smaller than the reproducible disturbances, the feedback controller does not have to provide a high performance. The feedback controller can be designed for robust stability only, using the normal design procedures for linear controllers.
The *feed-forward* controller is a B spline Network (BSN) (Brown and Harris, 1994). A BSN is a neural network that uses B spline basis functions to store an input-output mapping. A B spline of order *n* consists of piecewise polynomial functions of order *n*-1. In this research only 2nd order B splines will be considered. The function evaluation of a B spline is generally called the *membership* and is denoted as μ (fig 4). That part of the input space for which μ is unequal to zero is called its *support*. The position at which the B spline evaluation equals 1 is known as the B spline knots.

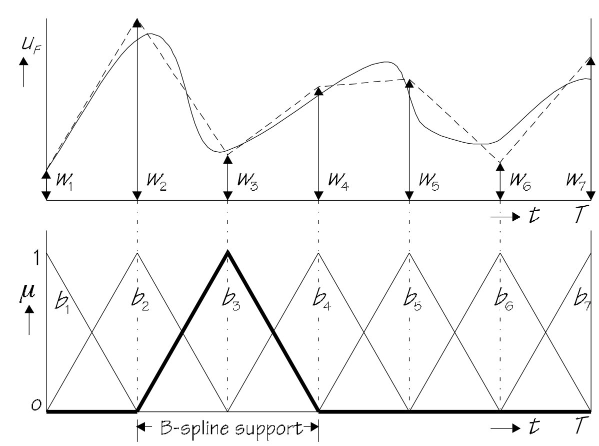

Fig. 4. 1-dimensional B spline Network
To create an i/o mapping, B splines are placed on the domain of the input of the BSN, in such a way that at each input value the sum of all

memberships equals 1. The output of the BSN is a weighted sum of the B spline evaluations:

$$u_F(r) = \sum_{i=1}^{N} \mu_i(r)\, w_i \tag{1}$$

In which w_i is the weight associated to the *i*-th B spline and *N* is the number of B splines. Training the network, in other words adapting the i/o mapping in such way that it comes closer to the desired i/o mapping, is done by adjusting the weights of the network. In LFFC the weights are adjusted on the basis of the output of the feedback controller. This signal is supposed to indicate how the feed-forward signal should be adapted in order to increase the tracking performance. The learning mechanism according to which the weights are adapted is now given by:

$$\Delta w_i = \gamma \mu_i(r) u_C(r) \tag{2}$$

In which Δw_i is the adaptation of the *i*-th B spline and γ is the learning rate. The learning rate γ determines how fast the weights of the BSN are adapted, $0 \leq \gamma \leq 1$. A large learning rate implies that the weights are adapted strongly. In case of a small learning rate ($\gamma \approx 0$) the weights are adapted slowly.

4 Design of the Learning Control System

The first part in the design of the LFFC, is the design of the feedback controller. In this research the feedback controller that was designed by Fokker is used for control.
Next the inputs and the structure of the LFFC have to be chosen. The structure of the LFFC depends on the nature of the disturbances the LFFC has to compensate. In case the disturbances depend on the position of the plant, the reference position must be used as input of the LFFC. When the disturbances do not only depend on the position of the plant, but also on the velocity, both the reference position and the reference velocity should be used as inputs of the LFFC. When the LFFC has to compensate several disturbances that each depend on one specific plant state, additive networks can be used (De Vries et al., 1998;Brown and Harris, 1994). That is, the feed-forward controller consists of several BSN's that each compensate one specific disturbance. In the simulator set-up the plant is expected to suffer from:

– Unknown friction: velocity dependent
– Cogging forces: position dependent
– Imperfections in the spindle: position dependent

It can be concluded that these disturbances can be compensated using a BSN that has the reference velocity as input to compensate the unknown friction, and a BSN that has the reference position as input to compensate the cogging forces and the effect of the imperfections in the spindle. The LFFC that results is depicted in figure 5.

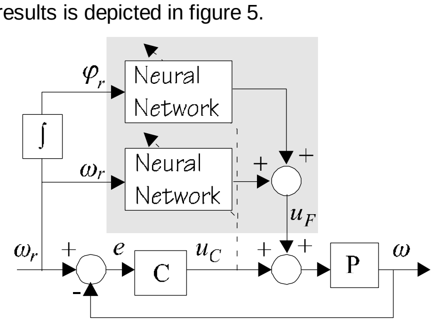

Fig. 5. LFFC structure

Simulation studies showed that the LFFC was able to compensate the unknown friction but not the cogging force and the effects of imperfections in the transmission. This can be understood as follows.
In the control loading system, the angular velocity of the simulator stick determines whether the simulated aircraft feels like a real aircraft. During simulation, the pilot does not interpret small errors in the angular position of the simulator stick as unrealistic. Therefore, the commanded angular velocity as calculated by the aircraft model is tracked tightly, at the cost of relatively large errors in angular position. Angular position errors of approximately 0.05 degrees occur frequently while in our set-up the period of the cogging force is 0.045 degrees.
So the error in the angular position of the simulator stick is of the same order of magnitude as the period of the cogging force. When this is the case, compensating the cogging force using a feed-forward controller it is not possible. This is illustrated in figure 6, where the magnitude of the cogging force is shown as a function of the position of the stick.
At a certain point in time, *t*, the reference angle is $\varphi_r(t)$. The cogging force that the feed-forward controller compensates is $F_{\text{cogg}}(\varphi_r(t))$. However, due to fact that the reference velocity is tracked in stead of the reference position, the actual value of the stick angle is $\varphi(t) \in [\varphi_r - 0.05, \varphi_r + 0.05]$ deg. It can be seen that in this interval the actual cogging force, $F_{\text{cogg}}(\varphi(t))$, may take any arbitrary value. Because $F_{\text{cogg}}(\varphi(t))$ does not resemble $F_{\text{cogg}}(\varphi_r(t))$, the cogging cannot be compensated by a feed-forward controller with input $\varphi_r(t)$.

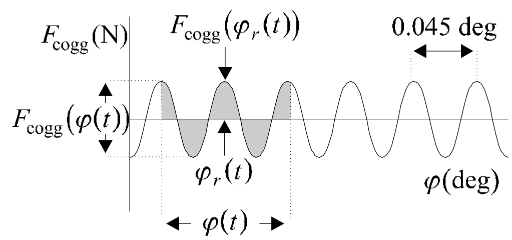

Fig. 6. Cogging force

To overcome this the measured angular position of the stick can be used as input of the BSN in stead of the reference angular position (Gomi and Kawato, 1993).
In other words, as far as position-related effects are concerned, an inverse model is contained in a feedback loop instead of a feed-forward path. This results in the system shown in figure 7.

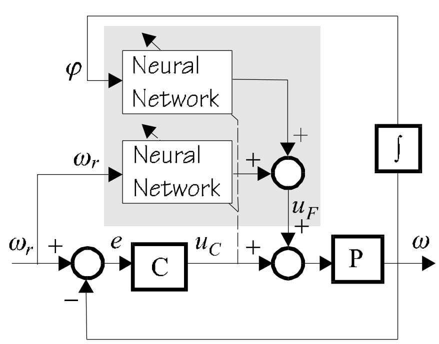

Fig. 7. Learning controller

Now that the structure of the learning controller has been selected, the distribution of the B splines has to be chosen. The distribution of the B splines is chosen by rule of thumb, based on the characteristics of the disturbances the BSN's have to compensate. In case of the BSN for ω_r, we known that BSN has to compensate the unknown friction, which can be approximated by the Stribeck curve. For this 60 B splines are used which are uniformly distributed over the domain of the angular velocity, which is [-30 deg/s, 30 deg/s]. In case of the BSN that has to compensate the cogging force, we know the period of the cogging force, namely 0.045 deg. The B splines are distributed in such way that each cogging period is covered by 10 B splines. The learning rate of both BSN's is chosen equal to 0.1.

Finally, a training strategy for the learning controller has to be designed. When both BSN's are trained simultaneously, it is not guaranteed that the feed-forward signals end up in the right BSN's. It might be such that the BSN with input φ learns to compensate disturbances that depend on the velocity of the stick. To prevent this, the BSN's are trained consecutively. In a stage 1 the BSN with input ω_r is trained. This is based on the assumption that a specific angular velocity is realized at a range of angular positions, which should cancel out the effects of the position dependent disturbances. After the BSN with input ω_r is fully trained, its weights are (temporarily) fixed. In stage 2 the BSN with input φ is trained. After convergence, the weights of this network are fixed permanently, as the position dependent effects are not time-variant. However, during operation the friction in the motor and the ball-screw may vary. Therefore, the learning of the weights of the BSN with input ω_r is done continuously (stage 3).

5 Experiments

The learning controller described in the previous section is used to con-trol the simulator stick. In the experiments the stick has to track a repeti-tive reference path as accurately as possible. The reference angular vel-ocity and the reference angular position are presented in figure 8 and 9.

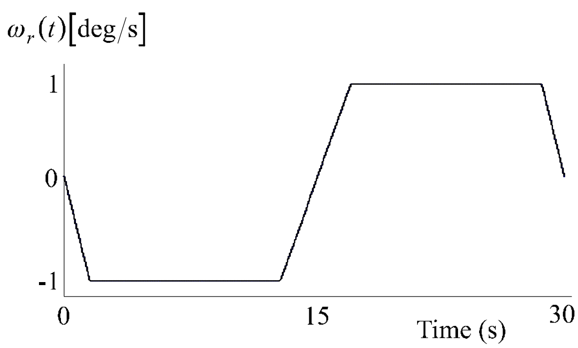

Fig. 8. Reference angular velocity

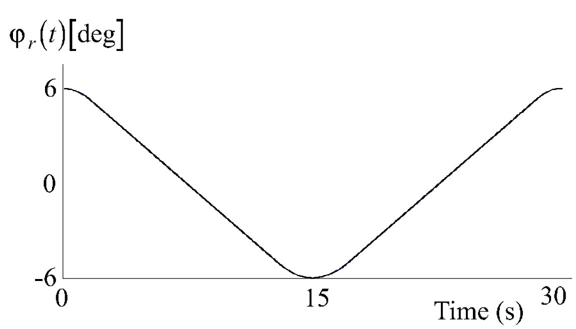

Fig. 9. Reference angular position

In the first experiment the stick is controlled by the PI- controller only. In figure 10 the power spectral density of the error *e* in the angular velocity of the stick is shown.

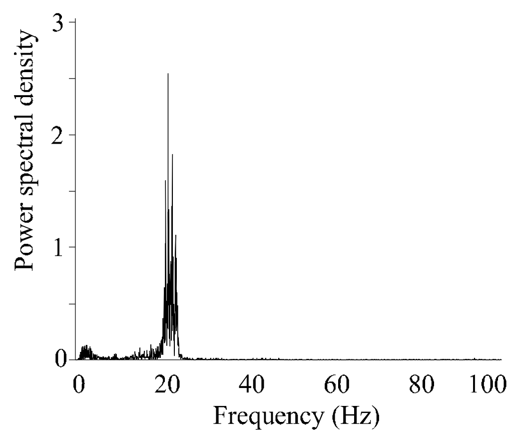

Fig. 10. Power spectral density of *e*, PI-control

The power spectral density of *e* shows a small peak near 0 Hz, which is probably caused by friction. Furthermore a peak exists at approximately 20 Hz. From the number of poles of the electro-motor and the long period of constant angular velocity it can be derived that this peak is caused by cogging forces.
The angular velocity of the motor is:

$$\omega_{motor} = \frac{\omega_r}{n} = \frac{1}{0.004} = 250 \text{ deg/s} \tag{3}$$

Where n=0.004 is the transmission coefficient of the spindle-ballscrew-stickjoint transmission. It is known that there are 32 poles in the motor and therefor 32 cogging periods per 360 deg motor rotation.
The frequency of the cogging will thus be:

$$f_{cogg} = \omega_{motor}\frac{32}{360°} \approx 22\text{ s}^{-1} \tag{4}$$

Next, learning control is applied. As stated in the previous section, training the learning controller consists of three stages. In the first stage the BSN with input ω_r learns to compensate the unknown friction. The reference angular velocity that is used in this training experiment should cover all angular velocities that occur during operation. In this way it is guaranteed that for each angular velocity that occurs in the simulation of the aircraft, the BSN knows how to compensate the unknown friction. The reference angular velocity the stick has to track during this straining stage is given in figure 11.

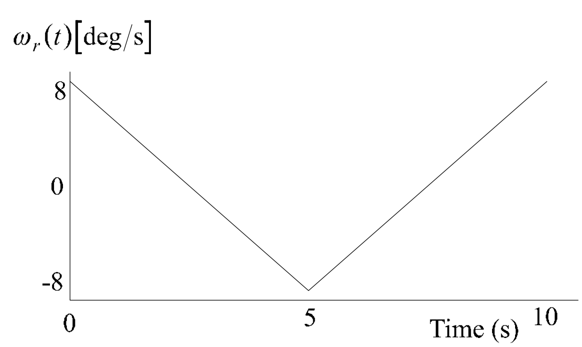

Fig. 11. Reference angular velocity used during training of the velocity network

Training the BSN is continued until the weights do no change anymore.
Now that the BSN with input ω_r is fully trained, its weights are fixed and the BSN with input φ is trained. The reference angular velocity and reference angular position of figure 8 and 9 are used in the training experiments. After the BSN with input φ is fully trained the power spectral density of the error *e* in the angular velocity of the stick is calculated (fig 12).

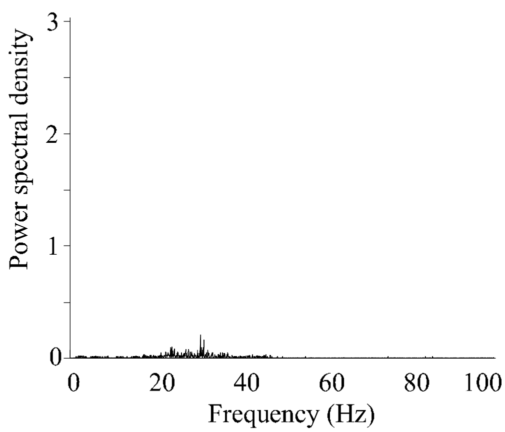

Fig. 12. Power spectral density of *e*, learning control

In the power spectral density of *e* it can be seen that both the peak that was caused by friction and the peak that was caused by cogging forces have decreased drastically. It may thus be concluded that the learning controller is able to reduce the influence of friction and the cogging disturbance. However, a small part is not compensated. This is probably due to the quantisation that is present in the measurement of φ. This signal is used as the input of the BSN that has to compensate the cogging forces. The quantisation resolution of φ is 0.008º, while the period of the cogging force is 0.045º. The consequence is that within one cogging period only 6 input values are presented to the BSN. The output of the BSN is thus a signal which consists of 6 piecewise constant parts, which is a rather bad approximation of the 'ideal' u_F that would fully compensate the cogging disturbance, see figure 13.

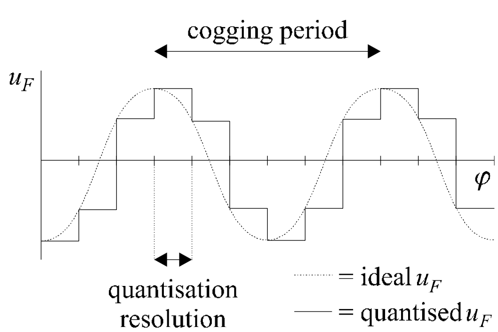

Fig. 13. Quantisation of the angular position signal

To further enhance the performance of the learning controller, the quantisation level has to be decreased. However, even when this is done the noise in the measurement of φ could prevent further increase in performance. Hence, the performance of the learning controller with feedback input depends on the quality of the measurement.

This is a disadvantage as compared to the pure learning feed-forward controller, where the input of the BSN is the reference path variable that is known exactly.

6 Conclusions

The probable causes of the irregularities experienced by the pilot when moving the stick are cogging, imperfections in the transmission and unknown friction. These disturbances have a reproducible nature.
A feedback error learning controller can be used to compensate for this. The proposed learning controller consists of 2 B spline networks (BSN's): one that compensates angular velocity dependent disturbances (i.e., the unknown friction) and one that compensates disturbances related to the angular position (i.e., cogging and imperfections in the transmission). The BSN's are trained consecutively by using the output of the feedback controller.
The BSN's may be contained either in the feed-forward or in the feedback path.
In this case, the BSN compensating effects that depend on the angular position cannot be contained in the feed-forward path, as this variable is not tracked accurately in the set-up.
The BSN compensating effects that depend on the angular velocity can be contained in the feed-forward path.
Experiments showed that cogging is the main cause of tracking error in the inner loop. The proposed learning controller can, after proper training, compensate for this and for the other effects. Also with the learning controller a small part of the tracking error remains; this is mainly due to the quantisation that is present in the measurement of the angular position.
When the quantisation level is decreased, measurement noise will probably limit the achievable performance.

References

1. Brown, M., and C. Harris, (1994), Neurofuzzy adaptive modelling and control, ISBN 0-13-134453-6.
2. Gomi, H., and M. Kawato, (1993), Neural Network Control for a Closed-Loop System Using Feedback-Error-Learning, Neural Networks, 6(7), ISSN 0893-6080, pp. 933-946.
3. Kawato, M, Y. Uno, M. Isobe and R. Suzuki, (1988), Hierarchical neural network model for voluntary movement with application to robotics, IEEE Control Systems Magazine, 8(2), pp 8-16.
4. Ng, G.W., (1997), Application of Neural Networks to Adaptive Control of Nonlinear Systems, ISBN 0-86380-214-1.
5. Otten, G., T.J.A. de Vries, J. van Amerongen, A.M. Rankers, and E.W. Gaal, (1997), Linear motor motion control using a learning feedforward controller, IEEE/ASME Tran. Mechatronics, 2(3), pp. 179-18.
6. Starrenburg, J.G., W.T.C. van Luenen, W. Oelen and J. van Amerongen, (1996), Learning feed forward controller for a mobile robot, Control Eng. Practice 4 (9), pp. 1221–1230.
7. Vries, T.J.A. de, L.J. Idema and W.J.R. Velthuis (1998), Parsimonious Learning Feed-Forward Control, o appear in Proc. European Symposium on Artificial Neural Networks ’98, Brugge, Belgium.