Proc. 1st IFAC Conf. on Mechatronic Systems (Darmstadt, Germany, 18–20 Sept. 2000), R. Isermann (ed.), VDI/VDE Gesellschaft Mess-und Automatiserungstechnik GMA, Düsseldorf, Germany.

LEARNING FEED-FORWARD CONTROL: A SURVEY AND HISTORICAL NOTE

Theo J.A. de Vries, Wubbe J.R. Velthuis, Job van Amerongen

Control Laboratory, C.J. Drebbel Institute, University of Twente
PO Box 217, 7500 AE Enschede, The Netherlands. Phone: +31 53 489 2788,
Fax: +31 53 489 2223, icontrol@el.utwente.nl, http://www.rt.el.utwente.nl/icontrol/

Abstract: From a mechatronic point of view, the performance of electro-mechanical motion systems can be improved by changing both the mechanical design and the controller. The design of a controller is generally based on a model of the plant. Thus, to improve the controller, a more accurate model of the plant is required. When the structure is not known or when many parameters cannot be determined, learning control may be considered. A simple yet powerful learning control scheme that is suitable for electro-mechanical motion systems is Learning Feed-Forward Control. In this paper an overview is given of applications that have been reported concerning this scheme. Also, relations are listed with alternative learning control schemes that are in some sense alike.

Keywords: Learning control, Neural networks, Intelligent control, Feedback Error Learning, Learning Feed-Forward Control, Iterative Learning Control, Motion control

1 INTRODUCTION

Our research is concerned with electro-mechanical motion systems, i.e., electrically actuated mechanical plants that require servo-control of the position of an end-effector. From a mechatronic point of view, the performance of electro-mechanical motion systems can be improved by changing both the mechanical design and the controller. In this paper, we will focus on increasing the performance by means of control, thereby assuming that the mechanical design is being optimized at the same time.
The design of a controller is generally based on a model of the plant. Thus, to improve the controller, a more accurate model of the plant is required. When modeling a plant, the following problems can be encountered [Harris *et al.*, 1993]:

- The system is too complex to understand or to represent in a simple way.
- The model is difficult or expensive to evaluate. The characteristics of some (non-linear) effects may be hard to obtain, e.g. Coulomb friction.
- The plant may be subject to large environmental disturbances, which are difficult to predict.
- The plant parameters may be time varying.

Adaptive control [Astrom and Wittenmark, 1989; Mareels and Polderman, 1996] can offer a solution when the structure of the dynamics plant and the disturbances that act on it are known, but the values of some parameters cannot be determined. When the structure is not known or when many parameters cannot be determined, learning control may be considered.

A simple yet powerful learning control scheme that is suitable for electro-mechanical motion systems is Learning Feed-Forward Control. The purpose of this paper is to provide an overview of applications that have been reported concerning this scheme. Also, relations are listed with alternative learning control schemes that are in some sense alike.

2 FEEDBACK ERROR LEARNING

The origin of Learning Feed-Forward Control can be traced back to a learning control scheme that is known as Feedback Error Learning (FEL), see Figure 1. This scheme was proposed by [Kawato *et al.*, 1987; Miyamoto *et al.*, 1988], and reported an application to a robot manipulator that had to track non-repetitive paths. The learning control system consists of two parts:

- The (standard) *feedback controller*.
- The *feed-forward controller*. In stead of designing a feed-forward controller on the basis of a model, [Kawato *et al.*, 1987] proposed to implement the feed-forward controller as a function approximator. The type of function approximator that [Kawato *et al.*, 1987] used is a Multi Layer Perceptron (MLP) neural network [Hertz, 1991; Haykin, 1994]. During control, the input-output relation of the function approximator is adapted. A main difficulty is now to select a learning signal, which indicates how the input-output relation of the function approximator must be adapted. The learning signal can be obtained in a number of ways [Er and Liew, 1997]. [Kawato *et al.*, 1987] proposed to use the output of the feedback controller as a learning signal.

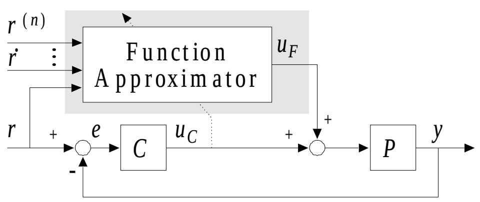

Figure 1: Originally proposed Feedback Error Learning controller

When the plant is not subject to disturbances and the feed-forward controller becomes about equal to the inverse plant, $F = P^{-1}$, then the output of the plant, y, will equal the reference, r. However, the plant will always be subject to disturbances. These disturbances can either have a stochastic or a reproducible nature.

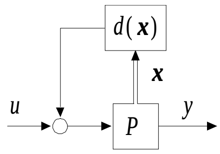

Figure 2: System and reproducible disturbances

Reproducible disturbances reoccur in the same way when a specific motion is repeated. This means that they can be regarded as a function of the state of the plant, $\boldsymbol{x}$, see Figure 2. An example is position dependent friction. Like the process dynamics, a feed-forward controller can compensate such reproducible disturbances (as long as it has the proper inputs).

Hence, the compensator (C) 'only' has the following functions:

- It provides the learning signal for the feed-forward controller.
- It determines the minimum tracking performance at the beginning of learning.
- It compensates stochastic disturbances

The important point is that as long as disturbances are mostly reproducible and hardly stochastic, the compensator does not determine the performance. For mechatronic motion systems, this situation is often present, as can be concluded from the applications of the FEL controller that have been reported:

- an automatic braking system for automobiles [Ohno et al., 1994];
- control of a camera system [Bruske et al., 1997];
- control of robot manipulators [Kim and Lee, 1996];
- welding [Tzafestas et al., 1997].

The applications showed that the FEL controller considerably improved upon the performance of the feedback controller and that it is able to obtain a high tracking performance without extensive modelling.
Much of the practical value of a learning controller depends on the type of function approximator that is used. The MLP has the following principal properties:

- *Small memory requirement*. One of the nice properties of the MLP is that it is able to approximate high dimensional target functions with a low number of parameters. Therefore, the amount of computer memory required for implementation is small.
- *Computationally expensive*. Calculating the output of the MLP and adapting its weights involves a large number of complex computations. For some real time control

applications, this type of neural network may therefore not be suited.

- *Slow converges and local minima*. The learning mechanism converges slowly and can easily get trapped in a local minimum. Which local minimum the network weights end up in, depends on the initial weights of the network. Therefore, it is necessary to perform several training experiments with different initial weights, to obtain an acceptable tracking accuracy.
- The input-output relation is *not locally adaptable*. Adapting one of the parameters of the MLP changes the input-output relation over the entire input domain. This implies that an MLP network has problems with learning highly correlated data which is presented consecutively [Hrycej, 1997]. In these situations, the network tends to fit the last data, resulting in a poor generalising ability. When a motion system has to perform low-velocity motions, the FEL controller will have a poor performance.
- *Good generalising ability*. An advantage of the fact that the input-output relation can only be adapted globally is that the MLP tends to have a good generalising ability.
- *No control over the smoothness* of the approximation. The number of parameters of an MLP determines the maximum accuracy of the approximation. It does not guarantee certain smoothness. By learning, the MLP may approximate the target function very rough in one part of the input domain and very accurately in another.

In spite of the fact that the FEL controller finally achieves a high tracking performance, the learning behaviour is not optimal due to the MLP network [Dean et al., 1991; Katic and Vukobratovic, 1995; Er and Liew, 1997].

3 IMPROVEMENTS

Looking at the properties above, we may conclude that, in case its learning behavior is improved, FEL control has much potential. Different approaches exist to overcome the problems of the FEL controller. In the following we will briefly present three methods. The first two methods alter the structure of the learning controller, whereas in the last approach another type of function approximator is used.

Firstly, one can improve the learning behavior by selecting different inputs for the function approximator. In [Gomi and Kawato, 1993; Er and Liew, 1997] the error signal was added as an input for the function approximator, changing the learning controller from a pure feed-forward controller into a feed-forward / feedback controller. From a stability point of view, this is a radical change. Experiments showed that this learning controller overcomes the problems of the original FEL controller. Analogous results have been reported for the case where the plant output y was added as an input to the function approximator, see e.g. [Miller, 1987]; many more researches have worked on this scheme.

A second method is to use multiple feed-forward controllers, each trained to perform a specific task [Jacobs and Jordan, 1993]. A supervisory neural network learns which feed-forward controller is used for which task. This learning controller was tested on a manipulator that had to perform motions with objects of different weight. After learning, each of the feed-forward controllers had learned to move a specific object. The supervisory network had learned which feed-forward controller to use for which object.

Since the MLP network mainly causes the difficulties of FEL control, an obvious approach is to look for different function approximators. In [Kraft and Campagna, 1990; Ananthraman and Grag, 1993] the MLP network is replaced by a CMAC network. The CMAC network belongs to the class of neural networks that employ basis functions. The basis functions of CMAC consist of piecewise polynomial functions and have a value unequal to zero on a compact part of the input space only. The basis functions are distributed such that at each point in the input space, ρ basis functions overlap. The parameter ρ is known as the generalization parameter and can be chosen by the designer. The output of the CMAC network is a weighted sum of the basis function evaluations. Adapting the weights of the network, not the basis functions themselves, performs learning. All this yields the following improvements:

- Faster convergence. Since learning takes place locally, only a small number of weights are adapted, which results in a fast convergence.
- Ability to learn correlated data. The locations of the basis functions are fixed, which is beneficial for the learning of correlated data.
- No local minima. The learning mechanism does not suffer from local minima.

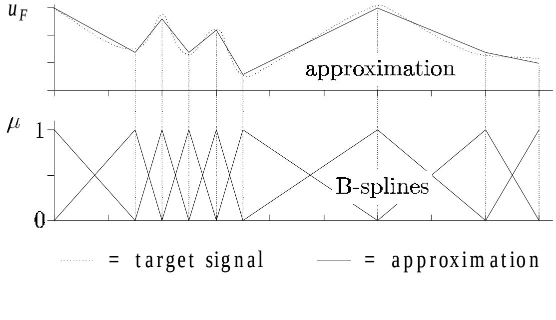

Figure 3: BSN mapping

However, a disadvantage is that the controller designer has to choose the distribution of the basis functions. This requires some prior knowledge of the desired input-output mapping and tuning of the basis function distribution may be necessary before an acceptable performance is obtained. Experiments showed that replacing the MLP network by a CMAC network gives a superior learning behavior and more accurate tracking.

3.1 LEARNING FEED-FORWARD CONTROL

The function approximator incorporated in FEL can also be implemented as a B-spline neural network (BSN) [Brown and Harris, 1994] instead of an MLP or CMAC. This type of FEL control has been named Learning Feed-Forward Control (LFFC) and was originally proposed by [Van Luenen, 1993]. The BSN approach is similar to the CMAC approach. Like the CMAC network, the BSN uses basis functions, known as B-splines, for approximation (see figure 3).

The BSN is much alike a CMAC; it has somewhat less modelling capabilities, but is easier designed as it has fewer design parameters and is more transparent [Brown and Harris, 1994].
The following applications of LFFC have been reported:

- Path tracking of a mobile robot vehicle [Starrenburg *et al.*, 1996]
- Positioning of a synchronous permanent magnet linear motor [Otten et al., 1997; Velthuis et al., 1998b]
- Admittance control of a flight simulator stick [Velthuis et al., 1998a]

These applications showed the benefits of using a BSN as function approximator. Also, they highlight that LFFC is an attractive means to solve tough mechanical design issues (e.g., friction), and hence should be considered during mechatronic design of motion systems.
The principal drawback however also is clear: a BSN (or any other neural network exploiting basis functions) is that the number of network weights grows exponentially with the dimension of the input space. Since accuracy considerations may require many B-splines, e.g. when a highly non-linear function is to be mapped, these networks are impractical when the number of inputs is large. This is known as the *curse of dimensionality* [Brown and Harris, 1994; Bossley *et al.*, 1996].

3.2 ITERATIVE LEARNING CONTROL

When the plant is to track a certain path repeatedly for many times, it is beneficial to view reproducible disturbances and plant dynamics as a function of the (periodic) motion time. This implies that it suffices to let this periodic motion time be the only input of the function approximator, thereby bypassing the curse of dimensionality. Then, the scheme of figure 4 is obtained.

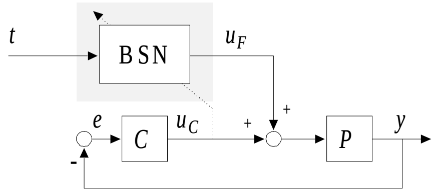

Figure 4: Time-indexed LFFC

The structure of the time-indexed LFFC (figure 4) resembles the structure of the well-known Iterative Learning Control scheme (ILC) [Arimoto *et al.*, 1984; Kawamura *et al.*, 1988; Moore, 1992; Arimoto, 1998; Moore, 1999] when $L=C$, see figure 5.

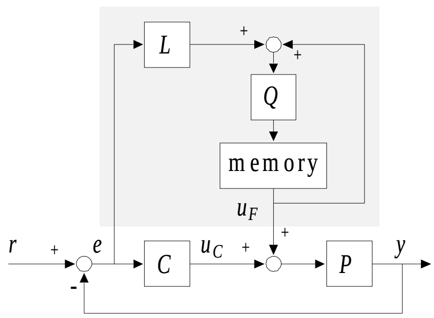

Figure 5: ILC (with low-pass Q-filter)

The importance of this is that robustness analyses and constructive stability proofs exist for ILC. Using these results, a constructive stability proof has been obtained for both time-indexed [Velthuis *et al.*, 2000] and path-indexed LFFC [Velthuis, 2000]. The time-indexed stability proof has been validated experimentally as well [Velthuis et al., 1998b].

4 CONCLUSION

In this paper, we have given an overview of applications in which a Learning Feed-Forward Controller or a closely related scheme controls a mechatronic system. From these publications, it can be concluded that it is attractive to have such learning control schemes at one's disposal during mechatronic design of motion systems. Using the overview, the interested researcher or designer may find out about the current status of this field of research.

REFERENCES

Ananthraman, S. and D.P. Grag (1993). Training Backpropagation and CMAC Neural Networks for Control of a SCARA Robot, *Engineering Applications of Artificial Intelligence*, **6** (2), 105-115.

Arimoto, S. (1998). A Brief History of Iterative Learning Control. In: *Iterative Learning Control: Analysis, Design, Integration and Applications*, 3-7. Kluwer Academic Publish-ers.

Arimoto, S., S. Kawamura and F. Miyazaki (1984). Bettering operation of robots by learning, *Journal of Robotic Systems*, **1**, 123-140.

Astrom, K.J. and B. Wittenmark (1989). *Adaptive Control,* Addison Wesley, Reading, Mass., U.S.A.

Bossley, K.M., M. French, E. Rogers, C.J. Harris and M. Brown (1996). Recent advances in neurofuzzy control system theory and design, Proc. ViCAM, EFAM, Guimarães, Portugal, 117-133.

Brown, M. and C.J. Harris (1994). *Neurofuzzy adaptive modelling and control*, Prentice Hall, Hemel-Hempstead, UK.

Bruske, J., M. Hansen, L. Riehn and G. Sommer (1997). Biologically inspired calibration-free adaptive saccade control of a binocular camera-head, *Biological Cybernetics*, **77**, 433-446.

Dean, P., J.E.W. Mayhew, N. Thacker and P.M. Langdon (1991). Saccade control in a simulated robot camera-head system: neural net architectures for efficient learning of inverse kinematics, *Biological Cybernetics,* **66** , 27-36.

Er, M.J. and K.C. Liew (1997). Control of Adept One SCARA Robot Using Neural Networks, *IEEE Trans. Industrial Electronics*, **44** (6), 762-768.

Gomi, H. and M. Kawato (1993). Neural Network Control for a Closed-Loop System Using Feedback-Error-Learning, *Neural Networks*, **6** (7), 933-946.

Harris, C.J., C.G. Hoore and M. Brown (1993). *Intelligent Control: Aspects of Fuzzy Logic and Neural Nets*, World Scientific Publishing Co. Pte. Ltd. , Singapore.

Haykin, S. (1994). *Neural Networks: A Comprehensive Foundation,* Toronto, Canada, Macmillan College Publishing Company.

Hertz, J., A. Krogh and R.G. Palmer (1991). *Introduction to the Theory of Neural Computation,* Addison-Wesley.

Hrycej, T., (1997). *Neurocontrol: Toward an Industrial Control Methodology*, John Wiley & Sons, Inc. , New York, USA.

Jacobs, R.A. and M.I. Jordan, (1993). Learning Piecewise Control Strategies in a Modular Neural Network Architecture, *IEEE Trans. Systems, Man, and Cybernetics*, **23** (2), 337-345.

Katic, D.M. and K. Vukobratovic, 1995, Highly Efficient Robot Dynamcis Learning by Decomposed Connectionist Feedforward Control Structure, *IEEE Trans. Systems, Man and Cybernetics*, **25** (1), 145-158.

Kawamura, S., F. Miyazaki and S. Arimoto (1988). Realization of robot motions based on learning method, *IEEE Transactions on Systems, Man and Cybernetics*, **18** (1), 125-134.

Kawato, M., K. Furukawa and R. Suzuki (1987). A Hierarchical Neural-Network Model for Control and Learning of Voluntary Movement, *Biological Cybernetics*, **57** , 169-185.

Kim, S.W. and J.J. Lee (1996). Filtered-Error-Learning Neural Networks for Stable Trajectory Tracking Control of Robot Manipulators, *Mechatronics*, **6** (2), 181-192.

Kraft, L.G. and D.P. Campagna (1990). A Comparison Between CMAC Neural Network Control and Two Traditional Adaptive Control Systems, *IEEE Control Systems Magazine*, , 36-43.

Mareels, I. and J.W. Polderman (1996). *Adaptive Systems: an Introduction*, Birkhaeuser, Boston, USA.

Miller, W.T. (1987). Sensor based control of robotic manipulators using a general learning algorithm, *IEEE J. Robotics and Automation,* **3**, 157–165

Miyamoto, H., M. Kawato, T. Setoyama and R. Suzuki (1988). Feedback-Error-Learning Neural Network for Trajectory Control of a Robotic Manipulator, *Neural Networks*, **1**, 251-265.

Moore, K.L., (1999). *Iterative Learning Control*: *an Expository Overview*. http: //www.engineering.usu.edu/ece/faculty/mo orek/mooreweb.html

Moore, K.L., M. Dahleh and S.P. Bhattacharyya (1992). Iterative Learning Control: A Survey and New Results, *J. Robotic Systems*, **9** (5), 563-584.

Ohno, H., T. Suzuki, K. Aoki, A. Takashi and G. Sugimoto (1994). Neural Network Control for Automatic Braking Control System, *Neural Networks*, **7** (8), 1303-1312.

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 Trans. Mechatronics*, **2** (3), 179-187.

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. Practise*, **4** (9), 1221-1230.

Tzafestas, S.G., G.G. Rigatos and E.J. Kyriannakis (1997). Geometry and Thermal Regulation of GMA Welding via Conventional and Neural Adaptive Control, *Journal of Intelligent and Robotic Systems*, **19**, 153-186.

Van Luenen, W.T.C. (1993). *Neural networks for control*, PhD-thesis, University of Twente, Enschede, Netherlands, ISBN 90-9006645-4.

Velthuis, W.J.R., T.J.A. de Vries, P. Schaak, E. Gaal (2000). Stability Analysis of Learning Feed-Forward Control, *Automatica*, *to appear*.

Velthuis, W.J.R. (2000), *Learning Feed-Forward Control – Theory, Design and Applications*, PhD thesis, University of Twente, Enschede, Netherlands, ISBN 90-36514126.

Velthuis, W.J.R., T.J.A. de Vries, K.H.J. Vrielink, G.J. Wierda and A. Borghuis (1998a). Learning control of a flight simulator stick, *Journal A*, **39**, 29-34.

W.J.R. Velthuis, T.J.A. de Vries and E. Gaal, (1998b). Experimental verification of the stability analysis of Learning Feed-Forward Control, *Proc. 37th IEEE Conference on Decision and Control*, (Tampa, Florida, USA), 1225-1229.