Copyright © IFAC 9th Triennial World Congress
Budapest, Hungary, 1984

A NEW CONCEPT FOR LEARNING CONTROL INSPIRED BY BRAIN THEORY

E. Ersü and H. Tolle

Institut für Regelungstechnik, Fachgebiet Regelsystemtheorie, Technische Hochschule Darmstadt, Schlossgraben 1, D-6100 Darmstadt, Federal Republic of Germany

Abstract. The paper explains an unconventional learning control method based on assumptions in the literature about human problem solving and information storage in neuronal networks. The on-line learning comprises two stages: The dynamic input-output behaviour of the process to be controlled is stored stepwise in a neuron-like manner into an associative memory as a predictive process model, the control strategy planned via this model by optimization of a goal oriented performance index is then trained in the same way into a second associative memory. As a general mapping the learned behaviour is in both cases in general nonlinear,and by this such a control design is especially suited for strongly nonlinear processes. Simulations demonstrate the applicability of the new control concept.

Keywords. Associative memory systems; adaptive control; artificial intelligence; biocybernetics; brain models; learning systems; neuronal networks; nonlinear control.

I. INTRODUCTION

In recent years the application of learning, adaptive or self-tuning methods to automatic control systems has become an important area of research (Tsypkin, 1973; Åström and others, 1977; Saridis, 1979). This is due to the fact that most of the complex industrial processes are of nonlinear nature, but control theory deals mostly with linear systems. The problem lies not only in the difficulty of modelling nonlinear systems but also in finding some classes of nonlinear systems to which then a somehow standard control design algorithm can be applied. The learning or adaptive control schemes have been proposed to bridge this gap of missing mathematically prestructured description of nonlinear complex systems.

This paper describes a learning control method motivated by neurobiological and psychological research on the human brain, trying to make some of its tremendous abilities to learn, to adapt, to associate and to prepare signals to influence and/or change its environment applicable to man-made systems. Actually the method is a synthesis of the *macro-intelligence* given by human problem solving capabilities (section II) and the *micro-intelligence* to be found in the basic working structure of neuronal networks (section III). The main features of the concept which has been developed and implemented by E. Ersü (1980, 1982, 1983) are that a *predictive model* of the unknown environment and the control strategy evaluated via a goal oriented performance index working with the predictive model are both treated as *general mathematical mappings* which are represented and stored on-line in *associative memory systems* similar to information storage in neuronal networks.

II. HUMAN PROBLEM SOLVING: AN INTELLIGENT CONTROL STRATEGY

The *human problem solving behaviour* as a higher mental process is an intelligent *connection* from *perception* to *action* via *reasoning* on the basis of past and present experiences. The most important properties that contribute to such an intelligent behaviour are - see Arbib, 1972; Dörner, 1974; Newell and Simon, 1972 -:

- . a modifiable predictive model of the unknown environment
- . the ability to plan actions and action sequences which represent solutions to upcoming problems by using the predictive model
- . the ability of generalizing the past experience during perception, action and, of course, planning for similar problem situations.

The act of perception and the act of planning result in memorization of what is perceived and planned, that means in two corresponding memories: one for the predictive model, one for the strategy selected by the planning process. The ability of generalization is to be handled by the way of memorization and will be dealt with in section III, therefore.

In terms of control theory the perception corresponds to learning of the predictive model of the process at hand, the reasoning to weighting possible actions by using the predictive model together with a goal oriented performance index, the generalization to suppressing of planning activities when for a certain situation the best action is already learnt and now a very similar situation arises.

Fig.II.1. illustrates this basic idea in form of a control system. The short term memory is needed for the environmental information of the near past as unique characterization of the actual situation. The shaded associative memories (long term memories LTM) store the predictive model for planning and the selected good actions for reaction in "similar" situations without reference to the planning level, "similar" being dependent on the generalizing capabilities in the information storage cycle.

III. INFORMATION STORAGE IN NEURONAL NETWORKS: ASSOCIATIVE STIMULUS-RESPONSE MAPPING AS GENERAL SYSTEM REPRESENTATION

Fig.II.1. is an assumption on the macrostructure of learning in the human brain. The supporting microstructure is given by the neuronal networks, and huge progress has been made since the beginning of this century - especially in the last two decades - in understanding their functioning by theoretical and experimental research. The general result is that incoming information pattern represented by dendral activity of the specific neuronal network is mapped through weighted synaptic connections as a coding onto an output information pattern, which is the axonal activity of the network. This mapping of afferent information (stimulus) onto an efferent one (response) can be modelled from a technical point of view by an associative memory system - (Kohonen, 1977; Amari, 1977; Palm, 1982) - and is also one of the basic elements of the psychological association theory.

An associative memory system is by the way an information storage in which the location of the output information is addressed by the content of the input information only (Kohonen, 1977). This type of information retrieval - which is an outstanding feature of the associative neuronal network models - avoids long searching procedures as they are necessary in general in the conventional memory systems.

In classical system theory input- output descriptions are based on an assumed or predetermined mathematical structure, normally a set of linear differential equations. Replacement of these predetermined structures by learned stimulus-response type of associative memory mappings lead to more general system representations by (in general nonlinear) mathematical mappings between n-dimensional input-vectors $\underline{I}$ and m-dimensional output-vectors $\underline{O}$:

$$S : \underline{I} \rightarrow \underline{O} \qquad \underline{I} \in R^n , \quad \underline{O} \in R^m \tag{1}$$

The utilization of such mappings for system representation is the most fundamental idea underlying the learning control concept introduced in this paper.

Given a time-discrete representation of a time-invariant deterministic system

$$\underline{y}(k+1) = \underline{f}_s[\underline{x}(k) , \underline{u}(k) , \underline{v}(k)] \tag{2}$$

with input vector $\underline{u} \in U \subseteq R^{n_u}$

output vector $\underline{y} \in Y \subseteq R^{n_y}$

measurable disturbance vector $\underline{v} \in V \subseteq R^{n_v}$

vector of the "inner state" $\underline{x}^T(k)=(\underline{y}^T(k),\ldots,\underline{y}^T(k-\alpha_s), \underline{u}^T(k-1),\ldots,\underline{u}^T(k-\beta_s), \underline{v}^T(k-1),\ldots,\underline{v}^T(k-\gamma_s))$

$$\underline{x} \in X \subseteq R^{n_y x n_u x n_v}$$

scalars $\alpha_s, \beta_s, \gamma_s$

sampled time $t = k \cdot T_s$

and sampling period T_s

a general mathematical mapping for (2) can be defined by

$$S : [\underline{x}(k), \underline{u}(k), \underline{v}(k)] \rightarrow \underline{y}(k+1) \tag{3}$$

The mathematical expression (3) memorized in an associative neural network model, i.e. an associative memory system, does not require any a-priori structural information about the system at hand except the scalars α_s, β_s, γ_s, representing the amount of history to be used in the model.

As a result of a comparative study of various neuronal network models (Ersü, 1981; Ersü and Militzer, 1982; Falb, 1981) the associative memory system called AMS and discussed below was found to be a suitable system for real-time control applications (Ersü, 1982). AMS is conceptually based on CMAC (Cerebellar Model Articulation Controller) conceived by Albus as a neuronal network model of the human cerebellar cortex (Albus, 1972, 1975). CMAC itself is a modified version of the Perceptron, a trainable pattern classifier proposed by Rosenblatt as a model of the human visual system (Rosenblatt, 1962).

In terms of computer science AMS is a distributed associative memory system by which a function or a mapping of several variables can be evaluated via simple content addressing mechanisms rather than by complex mathematical operations.

The AMS information processing structure is illustrated in Fig. III.1. Mathematically AMS can be presented by the overall mapping

$$H : \underline{s} \rightarrow \underline{p} \tag{4}$$

by which according to the neurophysiological notation (Albus, 1972) the sensory input $\underline{s}=(s_1,\ldots,s_n)^T$ of n variables s_i with varying resolutions R_i, i.e. each s_i may have R_i distinguishable values, is mapped onto the m-dimensional output vector p representing the axons of Purkinje cells leaving the cerebellar cortex.

Depending on the contents of the input information $\underline{s}$ a constant number r^* of association cells (pointers for locations in the physical memory marked by * in Fig. III.1.) are activated by a special encoding procedure out of r memory locations. Thus the incoming information is subdivided onto r^* information elements which are linked together to form the output information. The concatenation of these information elements, i.e. the sum of weights w_{ij} of these selected memory locations, represents the value of each output channel p_j.

Learning, i.e. training occurs by adjusting the selected weights w_{ji} for an input $\underline{s}$ depending on the error $\underline{e}$ between the desired $\hat{\underline{p}}$ and the actual output $\underline{p}$

$$\underline{e} = \hat{\underline{p}} - \underline{p} \tag{5}$$

which is distributed over the selected weights.

The encoding procedure by which similar inputs share some of the r^* memory locations - the number of shared locations being dependent on the degree of similarity - is responsible for the AMS's most fundamental feature of *generalization*, i.e. similar inputs tend to produce similar outputs even if they are not trained before. This ability - which is of great importance during learning - means mathematically that for untrained input variables $\underline{s}_x$ the memory system evaluates an output $\underline{p}_x$ the value of which depends on the similarity of the corresponding input $\underline{s}_x$ to all other already trained $\underline{s}_i$ in a specific nearest neighbourhood. For a single $\underline{s}_i$ a generalization expression can be given by

$$|\underline{p}_i - \underline{p}_x| = \lambda \cdot |\underline{s}_i - \underline{s}_x| \tag{6}$$

The scalar λ, the degree of generalization can be pretuned by the *generalization variable* r^*.

By the encoding implemented in the AMS generalization also reduces the theoretically necessary number of memory locations due to the possible input combinations ($r_t = \prod_{i=1}^{n} R_i$) as memory locations are shared by similar inputs.

Further this distributed type of storage has the advantage that memory damages and disturbances have comparably small influence on the output values. Additionally a pseudorandom addressing technique in the encoding procedure performs a location independent memorization of information. (For further details see Ersü and Militzer, 1982)

The following example will demonstrate the basic properties of the system: A reference function $\hat{p} = h(\underline{s})$ shown in Fig. III.2. with two inputs s_1 and s_2 each defined on the interval [0,256] with a resolution of unity, i.e. $s_i = j$; $j = 0,\ldots,256$; $i=1,2$ is trained on the memory system by a single training cycle in 289 of 66049 possible points (0,4 %) with a root-mean-square error of 2,1 %. (Fig. III.3.). This interesting result is due to the generalization capability of the memory system. The generalization variable r^* was chosen to 16 which is optimum for this reference function.

IV. THE IMPLEMENTED LEARNING CONTROL SYSTEM

Based on the basic ideas of the two preceding sections a control system has been proposed and designed by E. Ersü, 1980 for technical control tasks. As mentioned before the concept incorporates two associative memory systems of type AMS, one for the predictive model of the unknown process and one for the control strategy. Furtheron the following pre-assumptions are made:

a) The unknown multivariable process at hand is deterministic, time-invariant or weakly time-variant.

b) Time is sampled with a sampling period T_s ($k \hat{=} kT_s$) to allow time for learning and planning.

c) The control input $\underline{u} \in U \subset R^{n_u}$ is quantized and the control input space U is finite.

d) The process output $\underline{y} \in Y \subset R^{n_y}$ is quantized and the output space Y is finite.

e) The measurable disturbances $\underline{v} \in V \subset R^{n_v}$ are quantized and out of a finite space V.

f) The process can be described mathematically by equation (3) with some α_s, β_s and γ_s.

g) The overall performance index

$$I_G = \sum_{k=0}^{k_e-1} L_G[\underline{y}(k+1), \underline{w}(k+1), \underline{u}(k)] \tag{7}$$

with $\underline{w} \in W \subset R^{n_w}$

can be represented by a 1-step ahead subgoal

$$I_S(k) = \sum_{i=1}^{l} L_S[\underline{y}(k+i), \underline{w}(k+i), \underline{u}(k-1+i)] \tag{8}$$

i.e. minimizing the subgoal in each step also minimizes the overall performance index, so that the subgoal directs the learning control toward the optimum with respect to the global goal.

The concept based on AMS uses an output predictive algorithm scheme with a 1-step ahead control strategy where according to the equation (2) and (3) the predictive process model

$$M: [\underline{\xi}_M(k), \underline{u}(k), \underline{v}(k)] \rightarrow \hat{\underline{y}}(k+1) \tag{9}$$

with some α_m, β_m and γ_m

and the control strategy

$$C: [\underline{\xi}_C(k), \underline{v}(k), \underline{w}(k)] \rightarrow \underline{u}(k) \tag{10}$$

with some α_c, β_c and γ_c

are represented by two general mappings stored in two different AMSs.

The algorithm scheme in each time cycle is as follows[1] (Fig. II.1.)[2]:

i. the predictive model is updated by the measured prediction error

$$\underline{e}(k) = \underline{y}(k) - \hat{\underline{y}}(k) \tag{11}$$

where $\hat{\underline{y}}(k)$ is obtained by (9):

$$M: [\underline{\xi}_M(k-1), \underline{u}(k-1), \underline{v}(k-1)] \rightarrow \hat{\underline{y}}(k) \tag{12}$$

ii. an optimization (decision or planning) scheme is activated, if necessary for calculating an optimal control decision $\underline{u}^*(k)$ for the subgoal (8).

As for the time instant $t=t_k$ $\underline{y}(k+1)$ can only be predicted by (9), $\hat{\underline{y}}(k+1)$ is used in (8) to calculate the expected costs

$$\hat{I}_S(k) = L_S[\hat{\underline{y}}(k+1), \underline{w}(k+1), \underline{u}(k)] \tag{13}$$

thus

$$I_S(k) = \hat{I}_S(k) \quad \text{and} \quad u_{opt}(k) = \underline{u}^*(k) \tag{14}$$

where $\underline{u}_{opt}(k)$ minimizes $I_S(k)$,

is valid only for a trained region G_T of the input space of the predictive model memory representing the region in which training has already occured, and the model is reliable to some degree specified by the *generalization variable* r^* (s. preceding section).

Hence $\hat{I}_S(k)$ is minimized under the constraint

$$\underline{u}(k) \in G_T \tag{15}$$

To speed up the optimization a starting approximation for evaluating $\underline{u}^*(k)$ can be obtained by the past decision experience

$$C: [\underline{\xi}_C(k), \underline{v}(k), \underline{w}(k)] \rightarrow \underline{u}^o(k) \tag{16}$$

iii. the control decision $\underline{u}^*(k)$ optimized by (ii.) is memorized in control memory to be used as the best decision making (ii.) superfluous in the long range, and giving either an excellent optimization starting point (see (16)) or being used from a time on to be decided by the user without further inclusion of the predictive learning loop.

---

[1] For the sake of simplicity the algorithm will be discussed for l=1.

[2] In Fig. II.1. $\underline{v}$ is neglected.

iv. the last operation before applying a control input to the process is to look for a suboptimal control input $\underline{\tilde{u}}(k) \neq \underline{u}^*(k)$

with

$$\underline{\tilde{u}}(k) - \underline{u}^*(k) < \varepsilon \tag{17}$$

but

$$\underline{\tilde{u}}(k) \not\in G_T \tag{18}$$

for some specified ε.

$\underline{\tilde{u}}(k)$ applied to the process excites it to further untrained information for the model AMS and, so, enlarges G_T. This exploratory procedure called active learning is speeding up the learning.

V. RESULTS

The concept was successfully tested by simulations on several nonlinear examples, single-input-single output, as well as multi-input-multi-output. The principle learning behaviour will be demonstrated by a real-time implemented first order single-input-single-output example. A second example (simulation) with a highly complex nonlinear multi-input-multi-output process will show the efficiency of the concept.

Example I: For the first order process

$$\dot{z}(t) = \frac{1}{10}\,[u(t) - z(t)] \tag{19}$$

the input and output variables for the AMS's are defined as:

$$u(k) = g(k) \quad ; \quad y(k) = z(k)$$

$$s_M(k) = s_C(k) = y(k) \quad ; \quad w=5 \tag{20}$$

The one-step ahead performance index is of the form

$$I_S(k) = (y(k) - w)^2 \tag{21}$$

and the overall performance index

$$I_G = \sum_{k=0}^{k_e-1} I_S(k) \tag{21}$$

The control input is bounded

$$0 \le u(k) \le 10. \tag{22}$$

With a sampling period of T_S=6s Fig. V.1.a and Fig. V.1.b demonstrate the evolution of learning by successive learning trials. Neither the model nor the control-AMS had any a-priori information before the first run. Fig. V.1.b illustrates the control history. Obviously the system can generate a very useful control strategy after a few trials. Fig. V.1.c shows the learning curve which is the performance index (21) for each learning trial plotted down over the number of trials. In Fig. V.1.d the learning behaviour for set-point changes is demonstrated.

Example II: The nonlinear MIMO-process simulating a chemical reaction is given by

$$\dot{y}_1 = \frac{5-y_1}{12.5}\,u_1 - 18.828\cdot 10^{33}\cdot y_1 \cdot e^{-75.2315/y_2} \tag{23}$$

$$\dot{y}_2 = \frac{1}{400}[(24-32y_2)u_1 + 242.88\cdot 10^{33}\cdot y_1\cdot e^{-75.2315/y_2} + 28.8\,\frac{3.73-4y_2}{5+0.92u_2}\,u_2]$$

The one-step ahead performance index is given as

$$I_S(k) = \underline{y}(k+1) - \underline{w} \tag{24}$$

with

$$\underline{w}^T = (1.4 \quad , 0.9225) \; .$$

The vector of the inner state is defined by

$$\underline{s}_M(t) = \underline{s}_C(k) = \underline{y}(k).$$

The control input is bounded:

$$0 \le u_i \le 1 \; ; \quad i=1,2 \tag{25}$$

As model AMS was chosen 16 KByte of memory, and for control AMS 2 KByte. A sampling period of 2s in real-time has been used. The simulation example illustrated in Fig. V.2.a and Fig, V.2.b shows the learning behaviour of the control system by successive training trials on the same initial condition. 50 runs with 50 sample steps each (k_e=50) were carried out with initially untrained memory systems. The "learning curve" in Fig. V.2.c demonstrates the learning convergence. The unmonotonous decrease of I_G is due to the exploratory active learning procedure.

The examples above discuss only some principal features of the concept. Ersü and Mao, 1983 shows a more detailed example of a waste-water neutralization process. As mentioned above the test examples (simulated and real-time) did not show any principal problems. Even in the case of a robot arm which does not fulfill the conditions of BIBO (Bounded-Input-Bounded-Output)-stability required for learning control systems (Saridis, 1979) learning converged and the stability problem was overcome after a few learning trials.

VI. CONCLUSIONS

A new, unconventional control concept is introduced which is motivated by models of the information processing elements and loops in the human brain. The fundamental ideas underlying the concept are:

i. Representing systems by general mathematical mappings stored in associative memory systems.

ii. Utilizing this kind of system description for the predictive model of the unknown process and for the controller in a learning control system.

The basic properties of the system are:

- It does not distinguish between linear or nonlinear processes, as well as between linear and nonlinear control functions, a very fundamental feature due to the mathematically general way of system description by associative mappings.

- There is no need for an off-line structural model pre-assumption or modelling respectively. Off-line engineering efforts can be reduced to a degree which is necessary for determining the parameters α_m, β_m, γ_m, α_c, β_c, γ_c and r^*.[3)]

3) In practice one will choose $\alpha_m = \alpha_c$, $\beta_m = \beta_c$, $\gamma_m = \gamma_c$

- Learning the model and evaluating the control function occurs in discrete points of the corresponding input space. In contrary to the conventional adaptive control by which a global generalization for the whole working space occurs due to the structural model pre-assumption the proposed learning control approach has local generalization properties in a certain neighbourhood of the trained points of the corresponding input and output space (generalization region specified by r^*).

- Due to the implemented active learning procedure and the generalizing capabilities of the used associative memory system AMS the control system has fast learning convergence. Active learning is not a prerequisite for systems learning behaviour, but it speeds up learning and excites the process to additional information around the optimal control function.

- Learning mainly occurs around the optimal control path; thus the information inflow is optimized to an extent which is sufficient for gaining the goal state.

- A priori information about the process and the control function can be used as a priori training for the memory systems which will additionally speed up the learning process.

Though the approach presented is very effective, further research is needed to broaden its generality. The latter is at present limited by several theoretical and practical assumptions. Missing are especially theoretical results regarding stability, and regarding classes of processes and classes of performance criteria to which the concept will apply successfully. However, all examples attacked up to now, worked satisfactorily. A practical difficulty arises when $\underline{\ }(k)$, $\underline{y}(k)$, $\underline{y}(k)$, $\underline{u}(k)$ are of high order with fine resolutions. In principle the method in this paper still applies but the "curse of dimensionality" is a handicap which can result in huge memory requirements and heavy computational efforts. The recent developments in VLSI-technology signal solutions for these problems, however.

ACKNOWLEDGEMENT

The work was partly supported by the DFG (Deutsche Forschungsgemeinschaft).

REFERENCES

Albus, J.S. (1972). Theoretical and experimental aspects of a cerebellar model. PhD Thesis, Univ. of Maryland.

Albus, J.S. (1975). The cerebellar model articulation controller. Trans. ASME, series G, Vol. 97, No. 3

Amari, S.I. (1977). Neural theory of association and concept formation. Biol. Cybernetics, Vol. 26, pp. 175-185.

Arbib, M.A. (1972). The metaphorical brain. Wiley & Sons, New York.

Åström, K.J., Borrison, U., Ljung, L. and Wittenmark, B. (1977). Theory and applications of self-tuning regulators. Automatica, 13.

Bruhm, H. (1982). Implementation of a learning control concept with associative memory systems. Study Project. (directed by E. Ersü in German). Inst. of Control Engineering, Dep. Control Theory, Techn. Univ. of Darmstadt.

Dörner, D. (1974). Die kognitive Organisation beim Problemlösen. Hans Huber Verlag, Bern.

Ersü, E. (1980). Self-organizing control with associative memories. Internal Report at the Inst. of Control Engineering, Techn. Univ. of Darmstadt (in German).

Ersü, E. and Mao, X. (1983). Control of pH using a self-organizing control concept with associative memories. Int. IASTED Conf. on "Applied Control and Identification", Copenhagen, Denmark.

Ersü, E. (1983). On the application of associative neural network models to technical control problems. Proc. in Life Sciences: Localization and orientation in biology and engineering ed. Varjú and Schnitzler, Berlin, Heidelberg, New York.

Ersü, E. and Militzer, J. (1982). Software implementation of a neuron-like associative memory system for control applications. Proc. of the ISMM 8th Int. Symp. MIMI, Davos, Switzerland.

Falb, A. (1981). Implementation of a learning associative algorithm with distributed memories. Master Thesis at Inst. of Control Engineering, Techn. Univ. of Darmstadt, (directed by E. Ersü, in German).

Kohonen, T. (1977). Associative Memory. Springer, Berlin, Heidelberg, New York.

Newell, A. and Simon, H.A. (1972). Human problem solving. Englewood Cliffs, N. J. Prentice Hall.

Palm, G. (1982). Neural Assemblies. Springer, Berlin, Heidelberg, New York.

Rosenblatt, F. (1962). Principles of Neurodynamics. Spartan Books, Washington.

Saridis, G.N. (1979). Self-organizing control of stochastic systems. Marcel Dekker, New York.

Tsypkin, Ya.Z. (1973). Foundations of the theory of learning systems. Academic Press, New York.

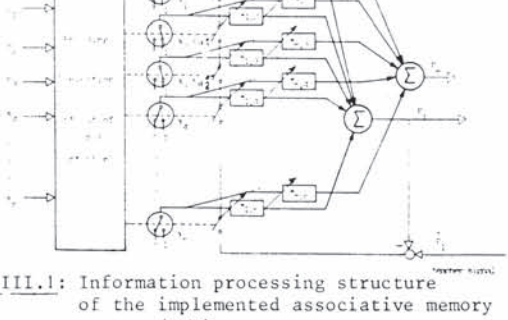

Fig. III.1: Information processing structure of the implemented associative memory system (AMS)

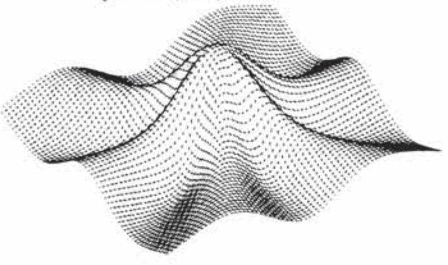

Fig. III.2: Reference function $\hat{\underline{p}} = H(\underline{s})$

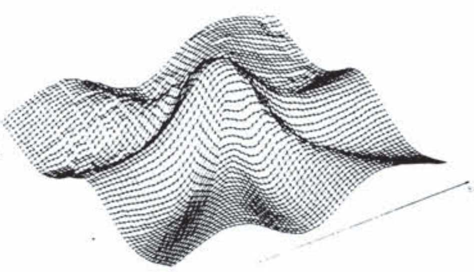

Fig. III.3: Response of AMS after training on 289 points (0,4 %)

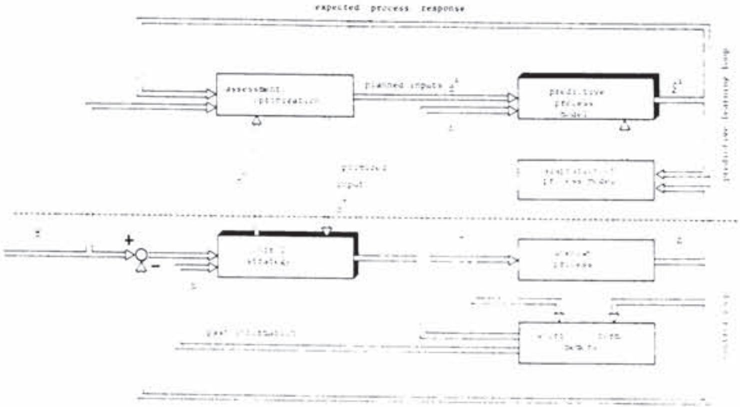

Fig. II.1: Learning control system by Ersü (Shaded blocks indicate the associative memory systems)

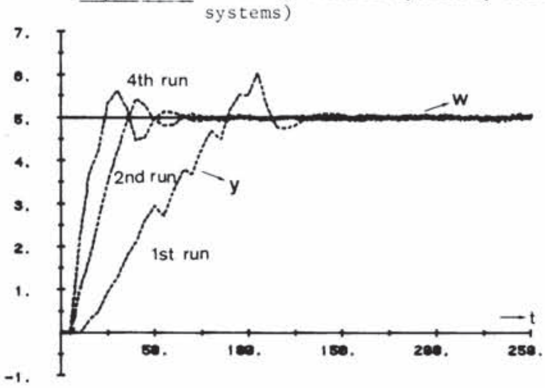

Fig. V.1.a: Real-time learning control behaviour by successive trials

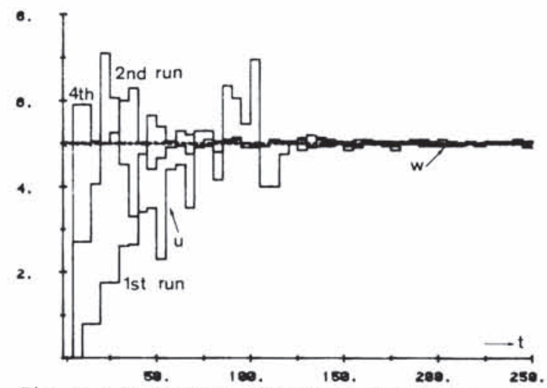

Fig. V.1.b: Control history of Fig. V.1.a

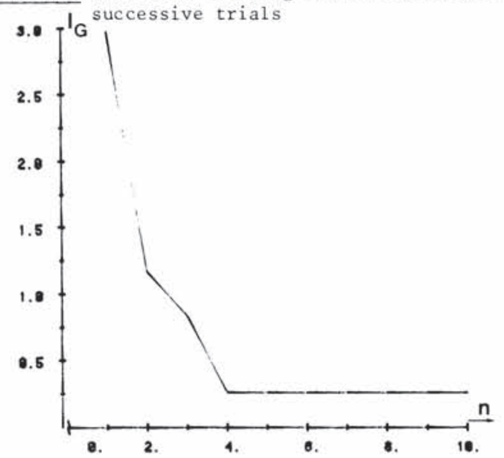

Fig. V.1.c: Learning curve for the example of Fig. V.1.a (n=number of learning runs)

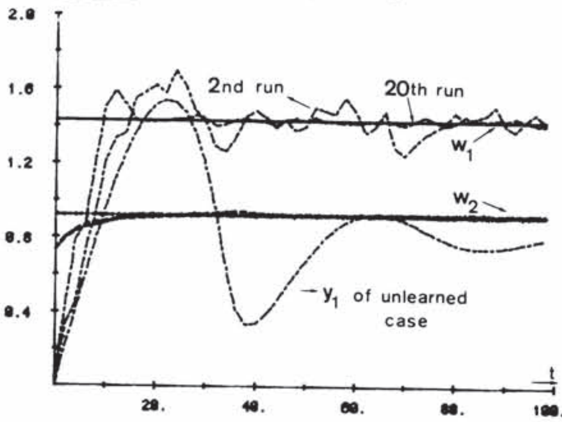

Fig. V.2.a.b: Evolution of learning by successive trials for the chemical process

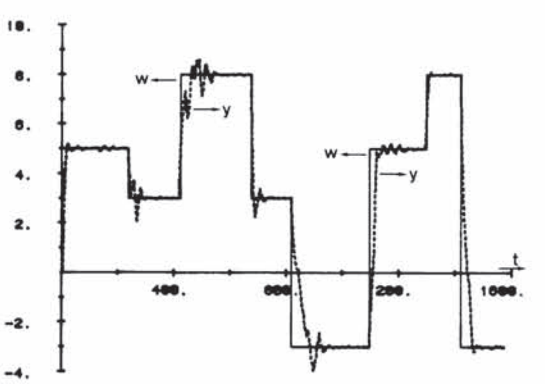

Fig. V.1.d: Learning control behaviour for new steady state values w

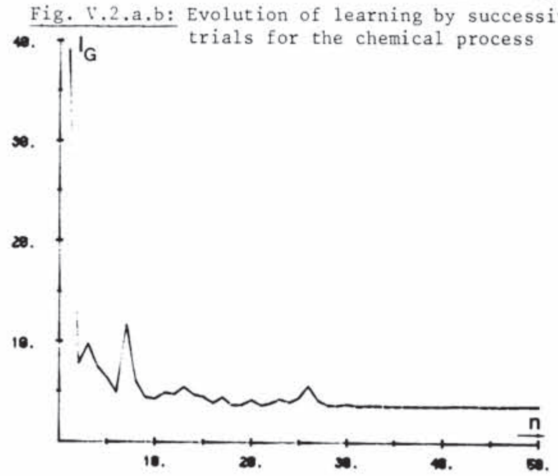

Fig. V.2.c: Learning curve for the example of Fig. V.2.a