



ROBUST CONTROL USING INVERSE DYNAMICS NEUROCONTROLLERS

CSABA SZEPESVÁRI^{†‡} and ANDRÁS LŐRINCZ[†]

[†] Department of Adaptive Systems, Institute of Isotopes of the
Hungarian Academy of Sciences, Budapest, P.O. Box 77, Hungary, H-1525

[‡] Bolyai Institute of Mathematics, University of Szeged,
Szeged, Aradi vrt. tere 1, Hungary, H-6720

Key words and phrases: stable control, inverse dynamics, inverse dynamics controller, dynamic feedback, ultimate uniform boundedness, Liapunov's second method, neural networks, non-linear control, over-controlled plants, control of chemical plants

1. INTRODUCTION

Neurocontrollers typically realise static state feedback control where the neural network is used to approximate the inverse dynamics of the controlled plant [1]. In practice it is often unknown *a priori* how precise such an approximation can be. On the other hand, it is well known that in this control mode even small approximation errors can lead to instabilities [2]. The same happens if one is given a precise model of the inverse dynamics, but the plant's dynamics changes. The simplest example of this kind is when a robot arm grasps an object that is heavy compared to the arm. This problem can be solved by increasing the stiffness of the robot, i.e., if one assumes a "strong" controller. Industrial controllers often meet this assumption, but recent interest has grown towards "light" controllers, such as robot arms with air muscles that can be considerably faster [3]. There are well-known ways of neutralising the effects of unmodeled dynamics, such as the σ -modification, signal normalisation, (relative) dead zone, and projection methods, being widely used and discussed in the control literature (see for example, [2]). Here a novel architecture that does direct identification of the inverse dynamics and a new method that utilizes this inverse dynamics controller in two copies are described. The result is robust controller of high precision put on a firm mathematical bases. The capabilities of the controller will be demonstrated on a chaotic bioreactor. The attractive learning properties will be discussed.

2. THE PDA MAP

The architecture of the feedforward inverse dynamics controller is shown in Fig. 1. It is a fully self-organizing architecture that forms a Position-and-Direction-to-Action (PDA) map introduced in [4]. It has four layers: a sensory layer (SL), a geometry discretizing layer (GDL), an interneural layer (INL), and a control layer (CL). The layers are connected as follows. Spatially tuned feedforward connections bring the the sensory information to the GDL. GDL has recurrent self-organised intralayer geometrical connections (GCs) that connect neighbouring nodes of the GDL and govern a diffusive equation. The sustained input activities of the SL give rise to steady state diffusion field

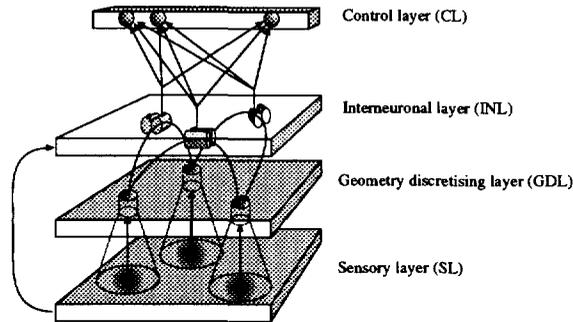


Fig. 1. Architecture of the feedforward neurocontroller

(SSDF). It is assumed that sensory inputs corresponding to start, target and obstacle entities are recognized by some higher order system and the start, target and obstacle nodes of the GDL are identified. The start and target inputs serve as sources and sinks of the diffusion, respectively on the GDL. The obstacle inputs form the forbidden zone: diffusion through GCs of nodes receiving obstacle inputs is inhibited. Under these conditions the diffusion has its maximum at the start node(s) its minimum at target node(s) and avoids the obstacle regions. The model is an extension of previous works [5,6,7,8]. The diffusing activities are sensed by the interneurons. These are simple linear I/O units that serve as sensors and also as associative feedforward connection points to neurons of CL. The network is fully self-organizing with Hebbian-learning, the only exception being the said higher order recognition module that initializes the formation of the steady state GDL activities. The INL to CL connections (the control connections (CC)) form the direct, associative identification of the inverse dynamics in the form of a PDA map. The free space learnt CCs allow move the plant along the gradient of the SSDF if the CCs belonging to interneurons of start node GCs are controlling the motions. The obstacle avoidance without further training is a natural consequence of the structure. Learning properties are discussed elsewhere [9].

Since the controller should move the plant along the SSDF, the control problem may be formalized as follows. The equation of motion is given as [10]

$$\dot{\mathbf{q}} = \mathbf{b}(\mathbf{q}) + \mathbf{A}(\mathbf{q}) \mathbf{u} (= \kappa \nabla \sigma^*), \quad (2.1)$$

where $\mathbf{q} \in \mathbf{R}^n$ is the state vector of the plant, $\dot{\mathbf{q}}$ is the time derivative of \mathbf{q} , $\mathbf{u} \in \mathbf{R}^m$ is the control signal, $\mathbf{b}(\mathbf{q}) \in \mathbf{R}^n$, $\mathbf{A}(\mathbf{q}) \in \mathbf{R}^{n \times m}$ that admits a generalised inverse, $\mathbf{A}^{-1}(\mathbf{q})$, $\kappa > 0$ is a proportional, and σ^* is the SSDF. We assume that the domain (denoted by D) of the state variable \mathbf{q} is compact and is simply connected; that $n \leq m$, and for each $\mathbf{q} \in D$ the rank of matrix $\mathbf{A}(\mathbf{q})$ is equal to n ; that is, the matrix is non-singular. As a consequence the plant is strongly controllable. In this case the inequality $n < m$ means that there are more independent actuators than state vector components, i.e., the control problem is redundant.

Let $\mathbf{v} = \mathbf{v}(\mathbf{q})$ be a fixed n dimensional vector field over D . The *speed field tracking task* is to find the static state feedback control $\mathbf{u} = \mathbf{u}(\mathbf{q})$ that solves the equation

$$\mathbf{v}(\mathbf{q}) = \mathbf{b}(\mathbf{q}) + \mathbf{A}(\mathbf{q})\mathbf{u}(\mathbf{q}). \quad (2.2)$$

Conventional tasks, such as the *point to point control* and the *trajectory tracking* tasks cannot be exactly rewritten in the form of speed field tracking and speed field tracking is more robust against

noise then these conventional tasks. Speed fields must be carefully designed if $A(q)$ is singular (as in the case of a robot arm).

Given the plant's dynamics by Equation (2.1) the main value of the inverse dynamics of the plant being modelled by the PDA map may be written as follows:

$$p(q, \dot{q}) = A^{-1}(q)(\dot{q} - b(q)), \tag{2.3}$$

The control signal $u(q) = p(q, v(q))$ solves the speed field tracking control task.

3. DYNAMIC STATE FEEDBACK CONTROL

The approximation errors of the inverse dynamics can be viewed as permanent perturbation to the plant's dynamics. Thus we assume that instead of Equation (2.1) the plant follows

$$\dot{q} = \hat{b}(q) + \hat{A}(q)u, \tag{3.4}$$

where $\hat{A}(q)$ is another nonsingular matrix field. Let us seek feedback compensatory control signal, $w = w(t)$, such that the control signal $u(q) + w$ solves the original speed field tracking problem for the perturbed plant. The simplest corresponding error-feedback law is to let w change until $\dot{q} = v(q)$ or, equivalently, $u(q) = p(q, \dot{q})$:

$$\begin{aligned} \dot{w} &= \Lambda(u(q) - p(q, \dot{q})) \\ \dot{q} &= \hat{b}(q) + \hat{A}(q)(u(q) + w), \end{aligned} \tag{3.5}$$

where Λ is a fixed positive number, the gain coefficient of dynamic feedback. If the speed of the

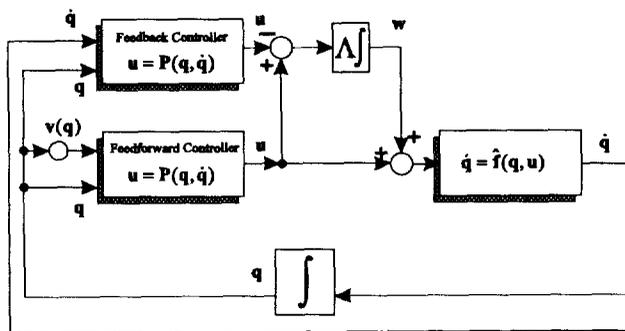


Fig. 2. SDS Control by doubling the role of the inverse dynamics controller

plant is measurable then Equation System (3.5) can be realised by a compound control algorithm. The block diagram of the compound controller is given in Fig. 2. The compound controller will be called as the Static and Dynamic State Feedback Controller (SDS Controller).

If S is a symmetric real matrix then let $\lambda_{\min}(S)$ denote by the minimal eigenvalue of S . Let $\|\cdot\|$ denote the Euclidean norm and let $z = \dot{q} - v(q)$ be the error of tracking. Robustness of control means that the error of tracking is small. Conditions are given in the following theorem:

THEOREM 3.1 Assume that the perturbation of $\mathbf{A}(\mathbf{q})$ can be decomposed as $\hat{\mathbf{A}}(\mathbf{q}) = \mathbf{D}(\mathbf{q})\mathbf{A}(\mathbf{q})$. Suppose that $\mathbf{A}(\mathbf{q})$, $\mathbf{b}(\mathbf{q})$, $\mathbf{v}(\mathbf{q})$ and $\mathbf{D}(\mathbf{q})$, $\hat{\mathbf{b}}(\mathbf{q})$ have continuous first derivatives and the constants $a = \inf\{\|\mathbf{A}(\mathbf{q})\| \mid \mathbf{q} \in D\}$, $d = \inf\{\|\mathbf{D}(\mathbf{q})\| \mid \mathbf{q} \in D\}$, and $\lambda = \inf\{\lambda_{\min}(\mathbf{D}(\mathbf{q}) + \mathbf{D}^T(\mathbf{q})) \mid \mathbf{q} \in D\}$ are positive. Then for all $\epsilon > 0$ there exists a gain Λ and an absorption time $T > 0$ such that for all $\mathbf{z}(0)$ that satisfy $\|\mathbf{z}(0)\| < K\Lambda$ it holds that $\|\mathbf{z}(t)\| < \epsilon$ provided that $t > T$ and the solution can be continued up to time t . Here K is a fixed positive constant and $\mathbf{z}(0)$ denotes the initial value of \mathbf{z} . Further, $\Lambda \sim O(1/\epsilon)$.

For convenience $\mathbf{D}(\mathbf{q}) + \mathbf{D}^T(\mathbf{q})$ will be called the *Symmetrised Perturbation matrix* and will be abbreviated to *SP-matrix*. We say that a perturbation of Equation (2.1) is *non-invertive* or *uniformly positive definite* if λ is positive. The proof is based on a modification of Liapunov's Second Method with the semi-Liapunov function $V(\mathbf{x}) = \mathbf{z}^T \mathbf{z}$. If \mathbf{z} satisfies the conclusions of Theorem 3.1 then it is said to admit the property of uniform ultimate boundedness (UUB).

4. COMPUTER SIMULATIONS OF A CHAOTIC PLANT

Chemical systems can be relatively simple in that they have few variables, but still troublesome to control due to strong nonlinearities which are difficult to model accurately. A prime example is the bioreactor. In its simplest form, a bioreactor is simply a tank containing water and cells (e.g., yeast or bacteria) which consume nutrients ("substrate") and produce products (both desired and undesired) and more cells. The simplest version of the bioreactor is a continuous flow stirred tank reactor (CFSTR) in which cell growth depends only on the nutrient being fed to the system. The target values to be controlled are the cell mass yield and the nutrient concentration. A basic set of equations for such a bioreactor is:

$$\begin{aligned} \dot{c}_1 &= -c_1(u+v) + c_1(1-c_2)e^{\frac{c_2}{\gamma}} \\ \dot{c}_2 &= -c_2(u+v) + Sv + c_1(1-c_2)e^{\frac{c_2}{\gamma}} \frac{1+\beta}{1+\beta-c_2}, \end{aligned} \tag{4.6}$$

where c_1 and c_2 are, respectively, dimensionless mass and substrate conversions [11]. The control parameter, u , is the flow rate through the reactor, while the control parameter, v , is the flow rate used to raise the substrate concentration. The constants β and γ determine the rate of cell growth and nutrient consumption, while S defines the dimensionless form of substrate concentration.

This problem has proved challenging for conventional controllers and was suggested as a control benchmark problem in [12]. The system is difficult to control for several reasons: the uncontrolled equations are highly nonlinear and exhibit limit cycles. Optimal behaviour occurs in near an unstable region. Note that for $\beta = 0.02$, $\gamma = 0.48$ and $S = 1$ a Hopf bifurcation occurs at $u = 0.829$, $v = 0.0004$.

Our experiments with this plant consisted of two phases corresponding to two sets of target values for c_1^* and c_2^* . First, the system was brought to a steady state at $(c_1^*, c_2^*) = (0.0737, 0.8760)$ and then the target values were changed to $(c_1^*, c_2^*) = (0.1287, 0.8688)$ which corresponds to a stable fixed point of the reactor with flow rates $(u^*, v^*) = (0.8012, 0.0004)$. Then the target cell mas was increased again, now to $(c_1^*, c_2^*) = (0.1737, 0.7978)$ with equilibrium control values $(u^*, v^*) = (1.0652, 0.0004)$. Control was made at intervals of 0.5 s, while simulating the bioreactor with Euler-methods and with $dt = 0.01$ s. One phase lasted 50 sconds. The said change in set-point is sufficient to shift from a stable regime into the domain of attraction of a limit cycle. Even if the correct model is known, small errors in parameters give rise to very inaccurate control when a feedforward controller is used: an error of 2% in γ leads to 50% error in the target cell mass [13].

We tested SDS Control using the inverse dynamics corresponding to Equation (4.6) with $\hat{\gamma} = 1.1\gamma$.¹ Since the change of γ results in an additive perturbation of the inverse dynamics, thus SDS Control can be applied without any restrictions and the theory predicts that SDS Control will yield bounded tracking error. The speed field was given by $v(c_1, c_2) = \lambda(c_1^* - c_1, c_2^* - c_2)$, where λ was determined so that with ideal control $e(t^*)/e(0) = 1/e$ for $t^* = 10/3$ s, where $e(t)$ is the error of either c_1 or c_2 at time t . The error of tracking is shown in Fig. 3a. The first of the three subfigures correspond to the case when the inverse dynamics model was perfect, while the second and third subfigures correspond to the cases when the inverse dynamics was imperfect with and without dynamic feedback, respectively. Note that there is almost no difference between the first and second subfigures, meaning that SDS Feedback could very efficiently compensate for the mismatched inverse dynamics model. However, when there was no feedback the error percentage of c_1 was larger than 60%. This figure suggests that SDS Control is able to *perfectly* compensate the perturbation, although the perturbation is highly nonlinear. This also follows from the theory since it can be shown that provided that the plant reaches a sufficiently small neighborhood of the designated equilibrium state and can be linearized then the feedback can compensate in a perfect fashion. Fig. 3b shows the control variables and the ideal control values in the three cases. This figure reinforces the impression that SDS Control is efficient. In these experiments A, the feedback gain, was 1.

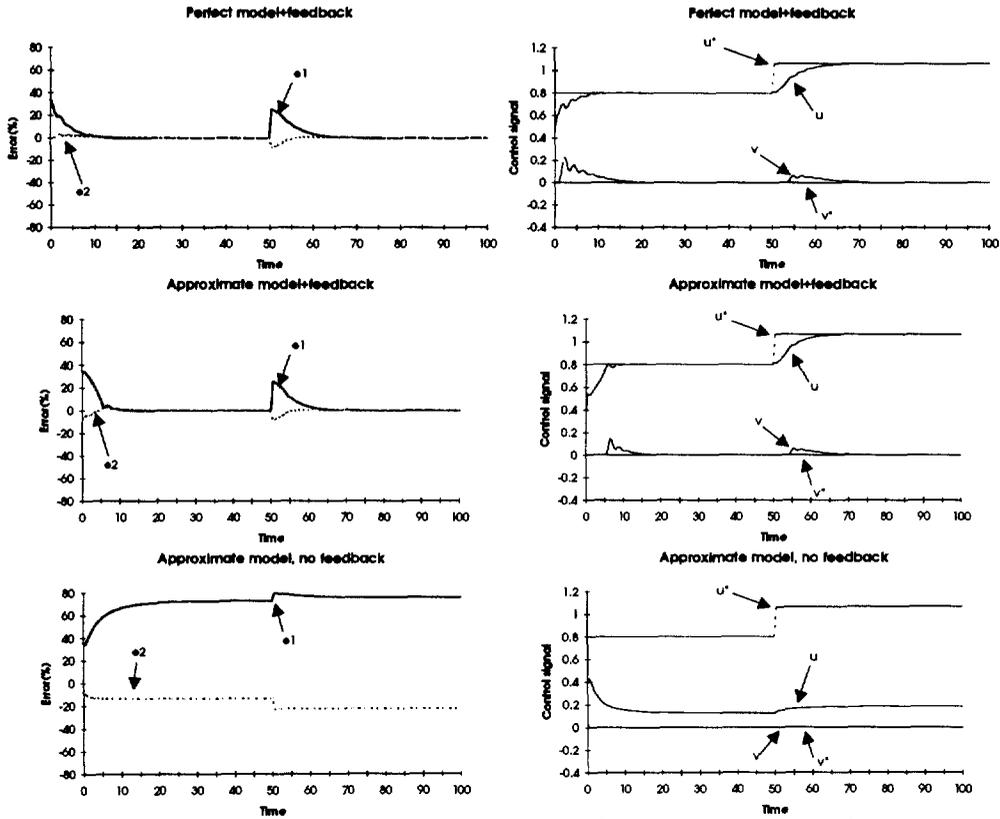
5. DISCUSSION

It is an important question whether the theory can be applied to real world situations. Along these lines it can be shown that the SDS Control is stable when an idealised robot arm grasps or releases an idealised object [14]. Here the scheme was applied to control a chaotic bioreactor with a severely mismatched inverse dynamics model.

In neurocontrol the inverse dynamics is modelled by a neural network, e.g., by the PDA map that may be inherently incapable of realising the inverse dynamics with zero error (i.e., the structural approximation error is non-zero). Can then the boundedness of tracking error be guaranteed if one applies SDS Control? Assume that we approximate $\mathbf{A}^{-1}(\mathbf{q})$ and $\mathbf{b}(\mathbf{q})$ by $\mathbf{P}(\mathbf{q})$ and $\mathbf{s}(\mathbf{q})$, respectively. Then, of course, $\mathbf{A}(\mathbf{q})$ “approximates” $\mathbf{P}^{-1}(\mathbf{q})$. By the same token one could also say that the inverse dynamics of our controller is “exact”, i.e., the plant’s equation is given by $\dot{\mathbf{q}} = \mathbf{P}^{-1}(\mathbf{q}) + \mathbf{s}(\mathbf{q})$ and Equation (2.1) could be considered as a perturbed system. In both cases one can apply Theorem 3.1 and obtain stability provided that besides some smoothness conditions $\inf_{\mathbf{q}} \lambda_{\min}(\mathbf{D}^T(\mathbf{q}) + \mathbf{D}(\mathbf{q})) > 0$ holds, where $\mathbf{D}(\mathbf{q}) = \mathbf{A}(\mathbf{q})\mathbf{P}(\mathbf{q})$. If $\lambda = \inf_{\mathbf{q}} \lambda_{\min}(\mathbf{D}^T(\mathbf{q}) + \mathbf{D}(\mathbf{q})) > 0$, we say that *the controller represents the inverse dynamics of the plant signproperly*. Under this condition we have that for large enough gains the tracking error is UUB and the ultimate bound on error is proportional to $1/\Lambda$. This also means that *no matter how imprecise the controller is the error of tracking may be arbitrarily decreased by choosing large enough Λ s*. The positivity of the symmetrised perturbation matrix follows if \mathbf{P} approximates \mathbf{A}^{-1} sufficiently closely. Consequently, the number of parameters of the estimator can be reduced, which may revitalise the use of fast learning local approximator based regularization neural networks as direct inverse controllers. Besides fast learning, the advantage of local approximators is that *a priori* knowledge of the control manifold can be used in training, e.g. when using geometry representing networks one may introduce neighbour training without the loss of generality provided that the inverse dynamics function is smooth.

Another attractive property of the scheme lies in its learning properties. The GDL can be formed in a self-organizing fashion [15,16,17]. The PDA map is formed via an *associative learning* method when the training data are given in the form of action-response pairs. That is, it is possible to apply SDS Control *during learning*, if the effective (SDS) control signal is used for association. Note, that

¹Control with the self-organised PDA map is under development.



Part a: Error of control in percent vs. time

The error percentage of control is given as $e_i = 100(c_i - c_i^*)/c_i^*$, $i = 1, 2$, where (c_1^*, c_2^*) is the desired set point of the bioreactor. After 50 s a new set-point is designated that corresponds to an unstable equilibrium of the plant. When feedback is in effect the error quickly reduces to zero. Without feedback the error remains high even if the inverse dynamics is moderately imprecise. For details of the experiment see the text.

Part b: Control values vs. time

(u^*, v^*) are the ideal equilibrium control values, while (u, v) are the actual control values. After 50 s a new set-point is designated that corresponds to an unstable equilibrium of the plant. Thus the ideal control values can not be assumed from the beginning. For details of the experiment see the text.

Fig. 3. Bioreactor control

during learning the tracking of the desired trajectory is more precise with *SDS* Control than without it. This means that neither stability, nor the identification process is affected by *SDS* Control. This is not the case for most *variational learning schemes*.

In the proofs we assumed that the perturbation is stationary, i.e., it does not change with time, which is unrealistic. It may be shown, however, that almost the same method can be applied to nonstationary perturbations. This means that if the changes are slow, i.e., these terms are bounded, then for large enough Λ one may retain the ultimate boundedness of the error signal. However, it should be mentioned that in real world applications these changes are usually fast (e.g., when a robot arm grasps a heavy object). In such a case the error signal may become very large and the system may leave the stable region. This can be handled for example by the projection method.

Noise can be treated just like non-stationary perturbations provided that it has bounded amplitude and bandwidth. Note, that as opposed to linear controllers the present method is capable of compensating such noise that enters the system at the inputs of the controller. However, if the noise enters the system just before the point where the compensatory vector is integrated, i.e., the noise affects $\dot{\mathbf{w}}$, then the system may easily become unstable. This problem, however, is not peculiar to our system, but is shared by every *dynamic* state feedback controllers.

6. CONCLUSIONS

PDA map for direct, associative inverse system identification was described. The utilization of two copies of the inverse dynamics controllers, e.g., the PDA map, in *SDS* Control Mode were proposed to compensate inhomogeneous, non-linear, non-additive perturbations of non-linear plants that admit inverse dynamics. Such perturbations arise, for example, when a robot arm grasps or releases a heavy object. One copy acts as the original closed loop controller while the other identical copy is used to develop the compensatory signal. The advantage of this compound controller is that it can develop a control signal for compensating unseen perturbations and structural approximation errors and thus can control the plant more precisely than the closed loop feedforward controller alone. Also the associative nature of the PDA map allows learning while controlling. The scheme relaxes the number of parameters required to achieve a given precision in control and thus may enable the use of fast learning local approximator based neural networks, that are otherwise known to suffer from combinatorial explosion in the dimension of the state.

7. ACKNOWLEDGEMENTS

We are grateful to Prof. András Krámli for his invaluable comments and suggestions. This work was partially founded by OTKA grants T017110, T014330, T014566, and US-Hungarian Joint Fund Grant 168/91-A 519/95-A.

REFERENCES

- [1] W.T. III Miller, R.S. Sutton, and P.J. Werbos, editors. *Neural Networks for Control*. MIT Press, Cambridge, Massachusetts, 1990.
- [2] R. Ortega and T. Yu. Theoretical results on robustness of direct adaptive controllers: a survey. In *Proc. 10th IFAC World Congress*, pages 26–31, Munich, 1987.
- [3] P. van der Smagt and K. Schulten. Control of pneumatic robot arm dynamics by a neural network. In *Proc. of the 1993 World Congress on Neural Networks*, pages 180–183, Lawrence Erlbaum Associates, Inc., Hillsdale, NJ., 1993.
- [4] T. Fomin, Cs. Szepesvári, and A. Lőrincz. Self-organizing neurocontrol. In *Proc. of IEEE WCCI/ICNN'94*, pages 2777–2780, IEEE Inc., Orlando, Florida, 1994.

- [5] G. Lei. A neural model with fluid properties for solving labyrinthian puzzle. *Biological Cybernetics*, 64(1):61–67, 1990.
- [6] C.I. Connolly and R.A. Grupen. On the application of harmonic functions to robotics. *Journal of Robotic Systems*, 10(7):931–946, 1993.
- [7] R. Glasius, A. Komoda, and S. Gielen. Neural network dynamics for path planning and obstacle avoidance. *Neural Networks*, 1994.
- [8] G.F. Marshall and L. Tarassenko. Robot path planning using vlsi resistive grids. In *IEEE Proc., Vision, Image and Signal Processing*, pages 267–272, 1994.
- [9] Cs. Szepesvári and A. Lőrincz. Neurocontrol I: self-organizing speed-field tracking. *Neural Network World*, 1995. submitted.
- [10] A. Isidori. *Nonlinear Control Systems*. Springer-Verlag, Berlin, 1989.
- [11] P. Agrawal, C. Lee, H.C. Lim, and D. Ramkrishna. Theoretical investigations of dynamic behavior of isothermal continuous stirred tank biological reactors. *Chemical Engineering Science*, 37, 1982.
- [12] L.H. Ungar. A bioreactor benchmark for adaptive network-based process control. In *Neural Networks for Control*, pages 387–402, MIT Press, Cambridge, 1992.
- [13] D.D. Brengel and W.D. Seider. A multi-step nonlinear predictive controller. *Ind. Eng. Chem. Res.*, 28:1812–1822, 1989.
- [14] Cs. Szepesvári, Sz. Cimmer, and A. Lőrincz. Dynamic state feedback neurocontroller for compensatory control. *IEEE Trans. on Neural Networks*, 1995. (submitted).
- [15] Cs. Szepesvári, L. Balázs, and A. Lőrincz. Topology learning solved by extended objects: a neural network model. *Neural Computation*, 6(3):441–458, 1994.
- [16] T. Martinetz. Competitive Hebbian learning rule forms perfectly topology preserving maps. In *Proc. of ICANN'93*, pages 427–434, Springer-Verlag, London, Amsterdam, The Netherlands, 1993.
- [17] Cs. Szepesvári and A. Lőrincz. Approximate geometry representation and sensory fusion. *Neurocomputing*, 1996. (in press).