[3] C. G. Cassandras and G. Bao, “A stochastic comparison algorithm for continuous optimization with estimations,” in *Proc. 33rd Conf. Decision and Control*, Lake Buena Vista, FL, 1994.
[4] L.-Y. Dai, “Convergence properties of ordinal comparison in the simulation of discrete event dynamic systems,” *J. Optimization Theory Appl.*, vol. 91, no. 2, pp. 363–388, 1996.
[5] L.-Y. Dai and C.-H. Chen, “Rate of convergence for ordinal comparison of dependent simulations in discrete event dynamic systems,” *J. Optimization Theory and Appl.*, vol. 94, no. 1, July 1997.
[6] M. L. Ginsberg, “Do computers need common sense?,” in *Proc. Knowledge Representation and Reasoning Conf.*, 1996.
[7] W.-B. Gong, Y.-C. Ho, and W. Zhai, “Stochastic comparison algorithm for discrete optimization with estimations,” *SIAM J. Optimization*, to be published.
[8] Y.-C. Ho, “Heuristics—Rules of thumb, and the 80/20 proposition,” *IEEE Trans. Automat. Contr.*, vol. 39, pp. 1025–1027, May 1994.
[9] ——, “Overview of ordinal optimization,” in *Proc. 33rd IEEE Conf. Decision and Control*, Lake Buena Vista, FL, Dec. 1994.
[10] Y.-C. Ho and M. Deng, “Large search space problems in ordinal optimization,” in *Proc. Conf. Decision and Control*, 1994.
[11] Y.-C. Ho and M. E. Larson, “Ordinal optimization approach to rare event probability problems,” *J. Discrete Event Dynamic Syst.*, vol. 5, pp. 281–301, 1995.
[12] Y.-C. Ho, R. S. Sreenivas, and P. Vakili, “Ordinal optimization in DEDS,” *J. Discrete Event Dynamic Syst.*, vol. 3, pp. 61–68, 1992.
[13] H. J. Kushner and D. S. Clark, *Stochastic Approximation Methods for Constrained and Unconstrained Systems*. New York: Springer-Verlag, 1978.
[14] T. W. E. Lau and Y. C. Ho, “Universal alignment probability and subset selection in ordinal optimization,” *J. Optimization Theory and Appl.*, vol. 39, no. 3, pp. 455–490, June 1997.
[15] T. J. Santner and A. C. Tamhane, *Design of Experiments: Ranking and Selection*. New York: Marcel Dekker, 1984.
[16] P. Vakili, L. Mollamustaflaglu, and Y.-C. Ho, “Massively parallel simulation of a class of discrete event systems,” in *Proc. IEEE Frontiers MPC Symp.*, Washington, DC, 1992.
[17] X. L. Xie, “Dynamics and convergence rate of ordinal comparison of stochastic discrete event systems,” *IEEE Trans. Automat. Contr.*, vol. 42, pp. 586–590, Apr. 1997.

Curve Evolution Models for Real-Time Identification with Application to Plasma Etching

Jordan Berg, Anthony Yezzi, and Allen Tannenbaum

***Abstract*—It is desirable, in constructing an algorithm for real-time control or identification of free surfaces, to avoid representations of the surface requiring mesh refinement at corners or special logic for topological transitions. Level set methods provide a promising framework for such algorithms. In this paper we present: 1) A mathematical representation of free surface motion that is particularly well-suited to real-time implementation; 2) a technique for estimating an isotropic and homogeneous normal velocity based on a simple measurement; and 3) an application to a semiconductor etching problem.**

***Index Terms*—Curve evolution, level set methods, parameter estimation, plasma etching, process modeling, semiconductor manufacturing.**

I. INTRODUCTION

Algorithms for evolving free surfaces may be roughly separated into two categories. “Lagrangian” methods directly propagate a discrete representation of the boundary itself. *Front tracking* methods are Lagrangian. In contrast, “Eulerian” methods propagate a scalar function everywhere in the domain, from which the movement of the boundary may be inferred. *Level set* and *volume of fluid* methods are Eulerian. There are advantages and disadvantages associated with each of these alternatives, as well as the variety of subdivisions within them. We are concerned here with the development of a surface evolution model suitable for real-time identification and control of an industrial plasma etching process. In this context, Eulerian approaches in general, and level set methods in particular, are very promising. Level set models may be evolved on a fixed grid, in some cases without any iteration required. In contrast, most front-tracking methods require adaptive mesh refinement, especially if corners may form in the surface. Perhaps the biggest advantage of a real-time level set model is the ease with which topological transitions, such as splitting or merging of surfaces, are handled. Such transitions may cause significant difficulties for a front-tracking approach. A more detailed comparison of various curve evolution schemes may be found in [10].

The idea of applying the theory of curve evolution for modeling surface development in reactive etching processes has been considered by a number of authors [4], [12], [10], [9]. Sethian and Adalsteinsson [10], [9] present a powerful and flexible level set-based simulation framework that includes many important processes and mechanisms. Note that all previous work has focused on *simulation*. For the present paper—the first work we are aware of that applies the level set formulation to the *identification* of a free surface—we consider only the simplest possible case: isotropic etching. For further simplification we use a two-dimensional (2-D) planar approximation for the wafer surface features.

Manuscript received September 30, 1997. This work was supported in part by grants from the National Science Foundation ECS-9700588, by the Air Force Office of Scientific Research AF/F49620-94-1-0058DEF and AF/F49620-949190461, by the Army Research Office DAAH04-94-G-0054, DAAH04-93-G-0332s, by a MURI Grant, and by the Institute for Mathematics and Its Applications.

J. Berg is with the Department of Mechanical Engineering, Texas Tech University, Lubbock, TX 79409-1021 USA (e-mail: jberg@ttu.edu).

A. Yezzi and A. Tannenbaum are with the Department of Electrical Engineering, University of Minnesota, Minneapolis, MN 55455 USA.

Publisher Item Identifier S 0018-9286(99)00570-X.

Recent studies have incorporated real-time feedback control into the plasma etching process [2], [6], [13], [7], [14]. The emphasis of these studies has been on the plasma, rather than the wafer surface. The feature evolution estimation problem is complementary to the plasma control task. For example, one can imagine using the active surface area generated by the estimator as an input to the plasma model.

II. Curve Flows and Static Level Sets

With a 2-D planar representation of the surface, the moving interface is described by a family of parameterized curves, $C(t) = (x(s,t), y(s,t))$. The curve evolves according to $\partial C/\partial t = \hat{\beta}(s,t)\mathcal{N}$ where s parameterizes the curve, $\mathcal{N}$ is the normal vector, and $\hat{\beta}$ is the *velocity function*.

To represent the evolution of this curve, we apply the *static level set* approach. Consider the class of curves evolving so that the velocity function never changes sign. Once such a curve has passed through a point, it never returns. Thus each point in the plane may be assigned a unique value, equal to the time at which the curve passed through it. If the curve never reaches a particular point, a value of ∞ may be assigned there. The resulting *crossing-time* function $T(x,y)$ is a time-invariant level set function. The evolving interface at time t is given by

$$C(t) = \{(x,y) \in R^2 : T(x,y) = t\}. \tag{1}$$

The time-invariant PDE describing $T(x,y)$ is $\hat{\beta}\|\nabla T\| = 1$. Sethian has presented a *fast marching* (FM) algorithm for solving this equation. For details of the algorithm, as well as further discussion of static level sets, see [11].

Our estimation algorithm will require us to calculate the rate of change in the length of the curve. The appropriate expression for *smooth* curves is $\partial L/\partial t = \int_0^1 \hat{\beta}\kappa\rho\, ds$ [5]. Here, κ is the curvature and ρ is the usual metric [3]. However, evolving features can develop corners and hence will only be *piecewise* smooth. In the case to be considered, the normal velocity will be piecewise constant, that is, constant on a given segment, but possibly varying from segment to segment. In particular, it may take one of two values: either zero, or a positive value β. The underlying notion is that the curve is propagating through an "active" medium in which it has a uniform, isotropic, normal velocity β. This medium has embedded in it inert inclusions, through which the curve cannot pass.

Corners in the curve affect the value of $\partial L/\partial t$. Let θ be the *orientation* of the curve, that is, the angle between the tangent to the curve and the x-axis, moving along the curve in the positive sense [3]. At the ith corner there is a jump in orientation, $\Delta\theta_i$. The necessary correction terms have been calculated. A description of the analysis, and a listing of the terms themselves, may be found in [1]. The key feature concerning us here is that the shock correction terms are all of the form $\beta\chi(\Delta\theta_i)$. Let $X(t;\beta)$ denote the sum over all corners of the $\chi(\Delta\theta_i)$ terms. Further, let $K(t;\beta) := \int \kappa\rho\, ds$ denote the *total curvature*, where the integral is over the smooth segments only. The notation indicates that β is treated as a constant parameter. Then the corrected expression for the rate of change of length is

$$L_t(t;\beta) = \beta K(t;\beta) + \beta X(t;\beta). \tag{2}$$

We make the following observation: The shape of the evolving feature may be expressed as a function of the variable $R := \beta t$ only. This may be seen from Huygens' principle, which states that the front is the envelope of the set of circles with radii R centered on the initial active surface. Thus we can write $L = L(R)$, $K = K(R)$, and $X = X(R)$, and $L_t = L'(R)\beta$. Comparison with (2) gives $L'(R) = K(R) + X(R)$, and so

$$L_\beta = L'(R)t. \tag{3}$$

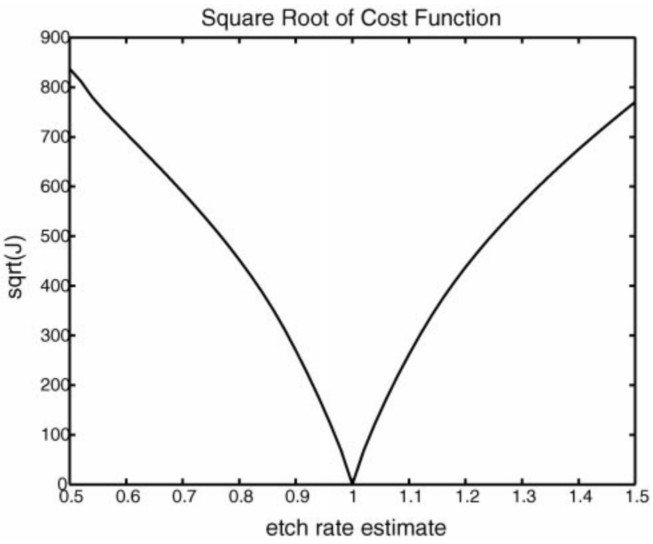


Fig. 1. Square root of J versus estimated β. True $\beta = 1$.

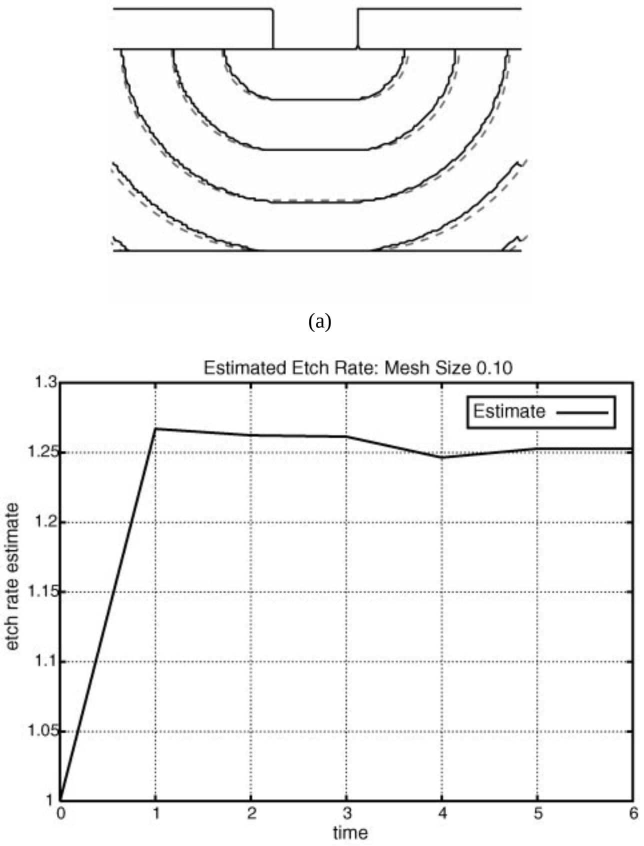
The obstacle to the practical application of this formula is that it is quite tricky to compute the change in tangent direction, $\Delta\theta$, accurately. Although the level set function may be used directly to generate this information through computation of the unit normal (which is simply $\nabla T/\|\nabla T\|$), these estimates are most precise when T is *smooth*. Of course, it is exactly when T is *nonsmooth* that we require the values.

Siddiqi *et al.* compare techniques for recovering contours from a level set function [8]. They present a method based loosely on contour tracing algorithms found in computer vision applications, but with two key differences. The first is the addition of *shock placement* logic, similar to essentially nonoscillatory interpolation schemes; the second is the use of geometric, rather than polynomial, interpolation. The scheme presented in [8] detects shocks by abrupt changes in orientation or curvature. Double shocks are "relieved" by a single interpolated shock, smoothing both orientation and curvature. We refer to [8] and the references therein for more details. The version implemented here is similar, but only first order. That is, shocks are detected by a large change in orientation only, and when consecutive shocks are relieved, only the orientation is smoothed.

III. Isotropic Etching of a Long Trench

This section describes a highly simplified model of a plasma etching process. A uniform layer of silicon sits on an inert substrate. The silicon is masked with a layer of resist, except for one or more narrow gaps. The simplifying assumptions are as follows: 1) The etch can be considered as 2-D planar; 2) The patterning is periodic; 3) The etch is isotropic; 4) The "active" material is homogeneous; and 5) The mask and substrate are perfectly inert.

It remains to define an appropriate measurement. As material is removed from the surface of the wafer, it enters the reactor chamber in the form of reaction by-products. The concentration of these by-products can be determined via mass spectroscopy. While not standard on industrial reactors, mass spectrometers are available in the laboratory [6]. This is the signal that we propose to use for reconstructing the etch rate and feature evolution. For the purposes of this preliminary study we use the following simplification: Assume that the rate of change of the concentration of etching by-products in the plasma is known and that this rate is due solely to removal of material from the wafer.



(a)

(b)

Fig. 2. $h = 0.10$. (a) Feature evolution—Solid line: Estimated; Dashed line: Exact. (b) Etch rate estimate. True value $= 1.25$.

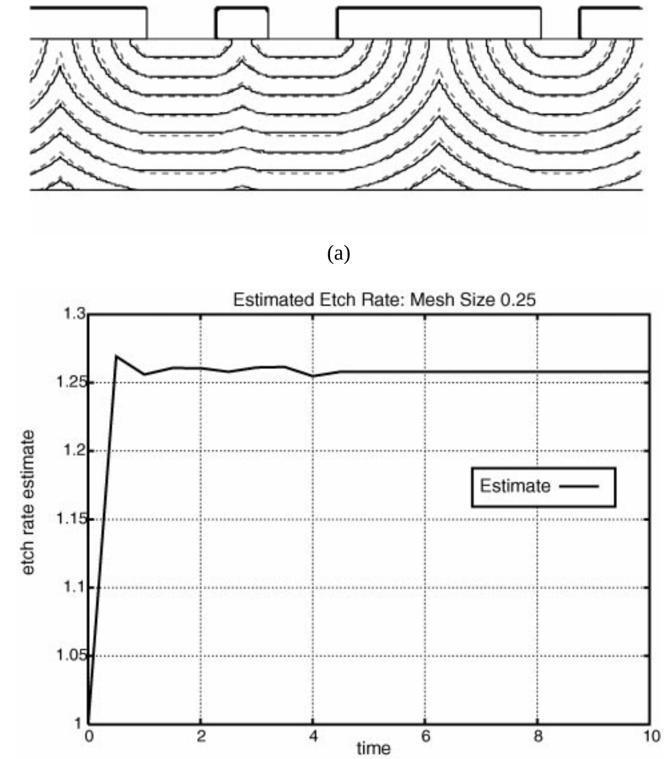
Under the above assumptions, the predicted measurement is just $y(t) = \beta L(t;\beta)$ where $L(t;\beta)$ is the length of the estimated surface and the dependence of the time history of L on β is explicitly indicated. It is also understood that $L(t;\beta)$ refers only to the portion of the surface that is exposed silicon, not resist or substrate. Note that it is impossible to relate $y(t)$ to the etch rate without knowledge of the feature shape.

The remainder of this paper concerns the construction of an estimator that will track the evolving feature morphology, based on the total etch rate measurement. The level set simulation plays the role of the plant model, and the etch rate β is used as an adjustable parameter. We assume that the initial geometry is known exactly, but that the etch rate, though constant or slowly varying, is not known. An algorithm is needed that solves the problem of "best" matching an etch rate estimate to the measured data. The estimated etch rate is then used to propagate the estimated feature.

We express the objective as a minimization problem

$$\min_{\beta} J(\beta) = \min_{\beta} \frac{1}{2} R(\beta)^t R(\beta) \tag{4}$$

where $[R(\beta)]_i = \beta L(t_i;\beta) - y(t_i)$. Generally, efficient solution of minimization problems requires analytical expressions for the first derivative of the cost function with respect to the unknown parameters. The problem in this case is taking derivatives through the set-valued function in (1). The derivative $J_\beta(\beta)$ is $J_\beta(\beta) = R(\beta)^T \nabla R(\beta)$. The necessary derivative is given by $[\nabla R(\beta)]_i = L(t_i;\beta) + \beta L_\beta(t_i;\beta)$. At any time t_i, given some value of β, $L(t_i;\beta)$ can be found using a forward solve. $L_\beta(t_i;\beta)$ is given by (3). The required functions of $R = \beta t$ are evaluated as described in Section II.



(a)

(b)

Fig. 3. (a) Three-trench feature evolution—Solid line: Estimate, $h = 0.25$; Dashed line: Truth model, $h = 0.10$. (b) Three-trench etch rate estimate. True value $= 1.25$.

With an expression for the first derivative available almost everywhere, it is natural to ask whether Gauss–Newton iteration may be used to find the minimizing β estimate. The answer is no; this cost function is not sufficiently smooth. To see this, consider a simple unpatterned planar etch. Here the evolving feature is a line segment of constant length L. The corresponding measurement at any time is simply $y = \gamma L$, where γ is the true etch rate. Let the etch go to completion, that is, until the measured signal is zero. Let the time required for this to occur be denoted T. For an etch rate estimate β, the cost function at completion may be found to be $J(\beta) = LM|\beta - \gamma|/2$ where $M := L\gamma T$ is the true total area of material to be etched. Thus Hessian-based methods are inappropriate. The isotropic etch exhibits similar behavior. Fig. 1 is a plot of the square root of J versus β. Again the cost function shows a corner at the solution. However, it is still possible to use the information contained in our calculation of the first derivative.

Fig. 1 suggests that J is smooth away from the solution and that there are no other local minimizers in a broad range of etch rates. Therefore, J decreases in the direction of the solution, and the sign of J_β changes only across the true etch rate. Then for the present case, where the parameter space is one-dimensional and the parameter must be positive, the following algorithm is suitable: 1) Make an initial guess at the etch rate $\beta^{(i)} = \beta_0$; 2) Evaluate $J_\beta^{(i)} := J_\beta(\beta^{(i)})$ for the current etch rate estimate; 3) If $J_\beta^{(i)}$ is negative, set $\beta^{(i+1)} = k\beta^{(i)}$, if $J_\beta^{(i)}$ is positive, set $\beta^{(i+1)} = \beta^{(i)}/k$, where $k > 1$; 4) Loop to 2) until $J_\beta^{(i)}$ changes sign; and 5) When $J_\beta^{(i)}$ changes sign, the solution has been bracketed. Proceed by bisection on J_β, as though searching for a root of J_β.

IV. SIMULATION RESULTS

When the periodic trench pattern consists of a single trench, the exact solution is easy to write down, based on Huygens' principle [1].

The exact solution was used to generate measurement data, which was input to the estimation scheme described above. Here the height of the active layer is 5 units, with an estimator mesh size of 0.1. The true etch rate was set to 1.25, and the initial guess given to the estimator was 1. The etch rate was estimated at intervals of 1 time unit, and the corresponding surface drawn. Fig. 2 shows the results compared to the exact solution.

The estimator was also tested using a mask pattern which repeats after three trenches. Again, the height of the active layer is 5 units. In this case the truth model used is an FM simulation with a mesh size of 0.1, while the estimator mesh size is 0.25. Again, a true etch rate of 1.25 is used, and an initial guess of 1.0 is supplied to the estimator, which updates an etch rate estimate at 0.5 time unit intervals. While no attempt was made to streamline the code for fast execution, the computation times are quite reasonable. Each surface profile estimate took under 20 s on a DEC AlphaStation 500. This compares favorably to plasma etch times. Fig. 3 shows the results.

ACKNOWLEDGMENT

The authors would like to thank P. Khargonekar for his help on the control literature in the area of semiconductor manufacturing. Also they would like to thank J. Higman and T. Higman for enlightening conversations on thin-film processing.

REFERENCES

[1] J. M. Berg, A. Yezzi, and A. R. Tannenbaum, “Phase transitions, curve evolution, and the control of semiconductor manufacturing processes,” *IMA Preprint 1454*, Inst. Math. Its Appl., Univ. Minnesota, 1997.

[2] S. W. Butler, K. J. McLaughlin, T. F. Edgar, and I. Trachtenberg, “Development of techniques for real-time monitoring and control in plasma etching, II: Multivariable control system analysis of manipulated, measured, and performance variables,” *J. Electrochem. Soc.*, vol. 138 no. 9, pp. 2727–2735, 1991.

[3] M. P. do Carmo, *Differential Geometry of Curves and Surfaces*. Englewood Cliffs, NJ: Prentice-Hall, 1976.

[4] I. Katardjiev, G. Carter, and M. Nobes, “The application of the Huygens principle to surface evolution in inhomgeneous, anisotropic and time-dependent systems,” *J. Phys. D: Appl. Phys.*, vol. 22, pp. 1813–1824, 1989.

[5] B. B. Kimia, A. Tannenbaum, and S. W. Zucker, “On the evolution of curves via a function of curvature, I: The classical case,” *J. Math. Anal. Appl.*, vol. 163, pp. 438–458, 1992.

[6] K. J. McLaughlin, T. F. Edgar, and I. Trachtenberg, “Real-time monitoring and control in plasma etching,” *IEEE Contr. Syst.*, pp. 3–10, Apr. 1991.

[7] B. A. Rashap, M. E. Elta, H. Etemad, J. P. Fournier, J. S. Freudenberg, M. D. Giles, J. W. Grizzle, P. T. Kabamba, P. P. Khargonekar, S. Lafortune, J. R. Moyne, D. Teneketzis, and F. L. Terry, “Control of semiconductor manufacturing equipment: Real-time feedback control of a reactive ion etcher,” *IEEE Trans. Semiconduct. Manufact.*, vol. 8, no. 3, pp. 286–297, 1995.

[8] K. Siddiqi, B. Kimia, and C.-W. Wang. “Geometric shock-capturing ENO schemes for subpixel interpolation, computation, and curve evolution,” Tech. Rep. LEMS-142, Brown Univ., Feb. 1995.

[9] J. A. Sethian, “An overview of level set methods for etching, deposition, and lithography development,” *IEEE Trans. Semiconduct. Manufact.*, vol. 10, no. 1, pp. 167–184, 1997.

[10] ——, *Level Set Methods: Evolving Interfaces in Geometry, Fluid Mechanics, Computer Vision, and Materials Science*. Cambridge, MA: Cambridge Univ. Press, 1996.

[11] ——, “A fast marching level set method for monotonically advancing fronts,” in *Proc. Nat. Acad. Sci.*, 1996, vol. 93, no. 4, pp. 1591–1595.

[12] E. Shaqfeh and C. Jurgensen, “Simulation of reactive ion etching pattern transfer,” *J. Appl. Phys.*, vol. 66, pp. 4664–4675, 1989.

[13] T. Vincent, P. P. Khargonekar, and F. L. Terry, “An extended Kalman filtering-based method of processing reflectometry data for fast *in-situ* etch rate measurements,” *IEEE Trans. Semiconduct. Manufact.*, vol. 10, no. 1, pp. 42–51, 1997.

[14] D. A. White, D. Boning, S. W. Butler, and G. G. Barna, “Spatial characterization of wafer state using principal component analysis of optical emission spectra in plasma etch,” *IEEE Trans. Semiconduct. Manufact.*, vol. 10, no. 1, pp. 52–61, 1997.

Robustifying Nonlinear Systems Using High-Order Neural Network Controllers

George A. Rovithakis

***Abstract*—A robustifying control methodology for affine in the control nonlinear dynamical systems is developed in this paper. A correction control signal is added to a nominal controller (designed to guarantee a desired performance for the corresponding nominal system), to render the actual system uniformly ultimately bounded. The control signal is smooth and does not require the *a priori* knowledge of an upper bound on the modeling error and/or optimal weight values. Simulations performed on a simple nonlinear system illustrate the approach.**

***Index Terms*—Neural networks, robust nonlinear adaptive control.**

I. INTRODUCTION

The need to deal with complex systems, that include uncertain and possibly unknown nonlinearities, operating in highly uncertain environments, has attracted a lot of research activity mainly through a neurocontrol approach [1]. Due to their massive parallelism, very fast adaptability and inherent approximation capabilities, neural networks have extensively been used mostly as approximation models of unknown nonlinearities. Initially, the application of neural networks in dynamical systems modeling and moreover in control has been demonstrated via numerous empirical studies employing gradient learning algorithms, in which besides the very interesting simulation results, no systematic design methodology was provided to guarantee at least stability of the overall system.

In order to find solution to stability and robustness issues, Lyapunov design techniques have been utilized [2]–[18]. The common trick used is that they exploit the approximation capabilities of neural networks, by substituting the actual system nonlinearities with neural network models, which are of known structure but contain a number of unknown constant parameters, called synaptic weights, plus a modeling error term. It is common in the literature and will also be used herein to denote as optimal these weight values that correspond to minimum modeling error.

However, most of the above works impose restrictions on the form of the allowable nonlinearities and furthermore require *a priori* knowledge of an upper bound on the modeling error and on the norm of the optimal weight values. Unfortunately, in many practical systems such bounds may not be available. Recently, an attempt to relax this restriction has been demonstrated in [19]. However, the results presented may ultimately be applied only to pure feedback systems.

Manuscript received October 18, 1996; revised April 11, 1997.

The author is with the Department of Electronic and Computer Engineering, Technical University of Crete, 73100 Chania, Crete, Greece.

Publisher Item Identifier S 0018-9286(99)00571-1.