7-27-1995

A Synthesis of Gradient and Hamiltonian Dynamics Applied to Learning in Neural Networks

J.W. Howse

C.T. Abdallah

G.L. Heileman

Department of Electrical and Computer Engineering

School of Engineering

University of New Mexico

A Synthesis of Gradient and Hamiltonian Dynamics Applied to Learning in Neural Networks

James W. Howse Chaouki T. Abdallah Gregory L. Heileman

Department of Electrical and Computer Engineering
University of New Mexico
Albuquerque, NM 87131

UNM Technical Report Number: EECE95–003

Current Date: July 27, 1995

Abstract

The process of model learning can be considered in two stages: model selection and parameter estimation. In this paper a technique is presented for constructing dynamical systems with desired qualitative properties. The approach is based on the fact that an n-dimensional nonlinear dynamical system can be decomposed into one gradient and $(n-1)$ Hamiltonian systems. Thus, the model selection stage consists of choosing the gradient and Hamiltonian portions appropriately so that a certain behavior is obtainable. To estimate the parameters, a stably convergent learning rule is presented. This algorithm is proven to converge to the desired system trajectory for all initial conditions and system inputs. This technique can be used to design neural network models which are guaranteed to solve certain classes of nonlinear identification problems.

Key Words

Dynamical systems, System Identification

Acknowledgements

This research was supported by a grant from Boeing Computer Services under Contract W–300445. The authors would like to thank Vangelis Coutsias, Tom Caudell, Don Hush, and Bill Horne for stimulating discussions and insightful suggestions.

Chapter 1

Introduction

A fundamental problem in mathematical systems theory is the identification of dynamical systems. System identification is a dynamic analogue of the pattern recognition problem. A set of input-output pairs $(u(t), y(t))$ is given over some time interval $t \in [\tau_i, \tau_f]$. The problem is to find a model which for the given input sequence returns an approximation of the given output sequence. Broadly speaking, solving an identification problem involves two steps. The first is choosing a class of identification models which are capable of emulating the behavior of the actual system. For non-linear identification, a common model choice is a recurrent neural network. One reason for this is that it was shown by Sontag (1992) and Funahashi and Nakamura (1993) that certain classes of recurrent networks can approximate an arbitrary dynamical system over a compact set for a finite time interval. Several recurrent models for system identification were proposed in Narendra and Parthasarathy (1990). In a similar vein, a set of constructive recurrent models were introduced in Cohen (1992). While the expressed purpose of these models was associative memory, they can be modified for use in system identification by including an appropriate term for the system inputs.

The second step in system identification involves selecting a method to determine which member of the class of models best emulates the actual system. In Narendra and Parthasarathy (1990) the model parameters are learned using a variant of the back-propagation algorithm. No learning algorithm is proposed for the models in Cohen (1992). Similar to the problem of learning model parameters for system identification is the problem that is often referred to in the literature as "trajectory following". Algorithms to solve this problem for continuous time systems have been proposed by Pearlmutter (1989), Sato (1990), and Saad (1992) to name only a few. One problem with all of these algorithms is that to our knowledge, no one has ever proven that the error between the learned and desired trajectories vanishes. The difference between system identification and trajectory following is that in system identification one wants to obtain an approximation which is good for a broad class of input functions. Conversely, in trajectory following one is often concerned only with the system performance on the small number of specific inputs (i.e. trajectories) that are used in learning. Nevertheless these trajectory following algorithms could be applied to parameter estimation for system identification.

In this paper we present a class of nonlinear models and an associated learning algorithm. The learning algorithm guarantees that the error between the model output and the actual system vanishes. Our class of models is based on those in Cohen (1992), with an appropriate system input. We show that these systems are one instance of the class of models generated by decomposing the dynamics into a component normal to some surface and a set of components tangent to the same surface. Conceptually this formalism can be used to design dynamical systems with a variety of desired qualitative properties. Our learning procedure is related to one discussed in Narendra and Annaswamy (1989) for use in linear system identification. This learning procedure allows the parameters of Cohen's models to be learned from examples rather than being programmed in advance. We prove that this learning algorithm is

convergent in the sense that the error between the model trajectories and the desired trajectories is guaranteed to vanish.

This paper is organized as follows. In Section 2 the decomposition of dynamics into a component normal to some surface and a set of components tangent to the same surface is discussed. Section 3 is a brief review of one of the potential design techniques introduced in Cohen (1992). The learning algorithm and some theorems about its behavior are given in Section 4. In Section 5 the results of some computer simulations are presented. The proofs for all of the theorems are given in the Appendix.

Chapter 2

Constructing the Model

First some terminology will be defined. For a system of n first order ordinary differential equations, the *phase space* of the system is the n-dimensional space of all state components. A solution *trajectory* is a curve in phase space described by the differential equations for one specific starting point. At every point on a trajectory there exists a tangent vector. The space of all such tangent vectors for all possible solution trajectories constitutes the *vector field* for this system of differential equations. The operation $\|\boldsymbol{x}\|$ denotes the p-norm $(|x_1|^p + |x_2|^p + \cdots + |x_n|^p)^{\frac{1}{p}}$ of the n-dimensional vector $\boldsymbol{x}$ for some p such that $1 \leq p < \infty$, and where $|\cdot|$ is the absolute value. For the purposes of the theorems any p-norm may be chosen. In the simulations $p = 2$ has been chosen so that the norm is the Euclidean distance.

The identification models in this paper are systems of first order ordinary differential equations. The form of these equations will be obtained by considering the system dynamics as motion relative to some surface. At each point in the state space an arbitrary system trajectory will be decomposed into a component normal to this surface and a set of components tangent to this surface. This approach was suggested to us by the results of Mendes and Duarte (1981), where it is shown that an arbitrary n-dimensional vector field can be decomposed locally into the sum of 1 gradient vector field and $(n-1)$ Hamiltonian vector fields. The concept of a potential function will be used to define these surfaces. A *potential function* $V(\boldsymbol{x})$ is any scalar valued function of the system states $\boldsymbol{x} = [x_1, x_2, \ldots, x_n]^\dagger$ which is at least twice continuously differentiable (i.e. $V(\boldsymbol{x}) \in C^r : r \geq 2$). The operation $[\cdot]^\dagger$ denotes the transpose of the vector. If there are n components in the system state, the function $V(\boldsymbol{x})$, when plotted with respect all of the state components, defines a surface in an $(n+1)$-dimensional space, which is called the *graph* of $V(\boldsymbol{x})$. There are two curves passing through every point on the graph which are of interest in this discussion, they are illustrated in Figure 2.1. The dashed curve is referred to as a *level surface*, it is a surface along which $V(\boldsymbol{x}) = k$ for some constant k. Note that in general this level surface is an $(n-1)$-dimensional manifold in $\mathbb{R}^n$. The solid curve moves "downhill" along $V(\boldsymbol{x})$ following the path of steepest descent through the point $\boldsymbol{x}_0$. The vector which is tangent to this curve at $\boldsymbol{x}_0$ is normal to the level surface at $\boldsymbol{x}_0$. The system dynamics will be designed as motion relative to the level surfaces of $V(\boldsymbol{x})$. Any point where $\nabla_{\boldsymbol{x}} V(\boldsymbol{x}) = \boldsymbol{0}$ is called a *critical point* of $V(\boldsymbol{x})$. The three critical points of the potential function in Figure 2.1 are labeled $\mathcal{A}_1$, $\mathcal{A}_2$, and $\mathcal{B}_1$. The points $\mathcal{A}_1$ and $\mathcal{A}_2$ are minima of the potential surface, and $\mathcal{B}_1$ is a saddle point. The results in Mendes and Duarte (1981) require n different local potential functions to achieve arbitrary dynamics. However, the results of Cohen (1992) suggest that a considerable number of dynamical systems can be realized using only a single global potential function.

A system capable of traversing any downhill path along a given potential surface $V(\boldsymbol{x})$, can be constructed by decomposing each element of the vector field into a vector normal to the level surface of $V(\boldsymbol{x})$ which passes through each point $\boldsymbol{x}$, and a set of vectors tangent to the level surface of $V(\boldsymbol{x})$ at $\boldsymbol{x}$. So the potential function $V(\boldsymbol{x})$ is used to partition the n-dimensional phase space into two subspaces.

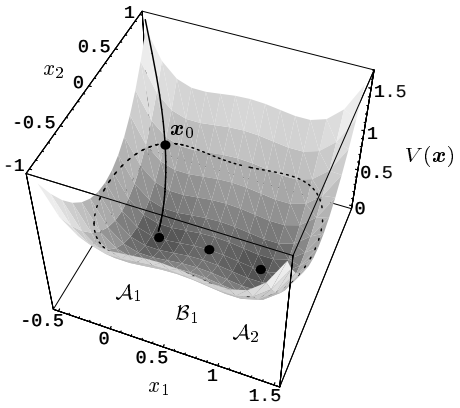


Figure 2.1: The graph of the potential function $V(\boldsymbol{x}) = x_1^2\,(x_1-1)^2 + x_2^2$ plotted versus its two dependent variables x_1 and x_2. The dashed curve is called a level surface and is given by $V(\boldsymbol{x}) = 0.5$. The solid curve follows the path of steepest descent through $\boldsymbol{x}_0$. The points $\mathcal{A}_1$ and $\mathcal{A}_2$ are minima of this surface, and $\mathcal{B}_1$ is a saddle point. All three of these points are critical points.

The first contains a vector field normal to some level surface $V(\boldsymbol{x}) = k$ for $k \in \mathbb{R}$, while the second subspace holds a vector field tangent to $V(\boldsymbol{x}) = k$. The subspace containing all possible normal vectors to the $\mathfrak{n}$-dimensional level surface at a given point, has dimension 1. This is equivalent to the statement that every point on a smooth surface has a unique normal vector. Similarly, the subspace containing all possible tangent vectors to the level surface at a given point has dimension $(\mathfrak{n}-1)$. An example of this partition in the case of a 3-dimensional system is shown in Figure 2.2. Since the space of all tangent

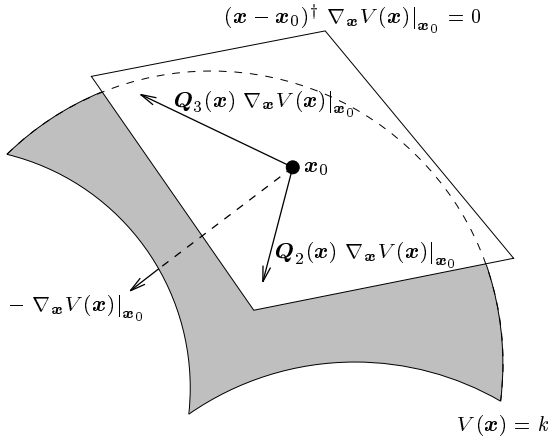


Figure 2.2: The partitioning of a 3-dimensional vector field at the point $\boldsymbol{x}_0$ into a 1-dimensional portion which is normal to the surface $V(\boldsymbol{x}) = k$ and a 2-dimensional portion which is tangent to $V(\boldsymbol{x}) = k$. The vector $(-\nabla_{\boldsymbol{x}} V(\boldsymbol{x})|_{\boldsymbol{x}_0})$ is the normal vector to the surface $V(\boldsymbol{x}) = k$ at the point $\boldsymbol{x}_0$. The plane $(\boldsymbol{x}-\boldsymbol{x}_0)^\dagger\,\nabla_{\boldsymbol{x}} V(\boldsymbol{x})|_{\boldsymbol{x}_0} = 0$ contains *all* of the vectors which are tangent to $V(\boldsymbol{x}) = k$ at $\boldsymbol{x}_0$. Two linearly independent vectors are needed to form a basis for this tangent space, the pair $\boldsymbol{Q}_2(\boldsymbol{x})\,\nabla_{\boldsymbol{x}} V(\boldsymbol{x})|_{\boldsymbol{x}_0}$ and $\boldsymbol{Q}_3(\boldsymbol{x})\,\nabla_{\boldsymbol{x}} V(\boldsymbol{x})|_{\boldsymbol{x}_0}$ that are shown are just one possibility.

vectors at each point on a level surface is $(\mathfrak{n}-1)$-dimensional, $(\mathfrak{n}-1)$ linearly independent vectors are required to form a basis for this space.

Mathematically, there is a straightforward way to construct dynamical systems which either move downhill along $V(\boldsymbol{x})$ or remain at a constant height on $V(\boldsymbol{x})$. In this paper, dynamical systems which

always move downhill along some potential surface are called *gradient-like systems*. These systems are defined by differential equations of the form

$$\dot{\boldsymbol{x}} = -\boldsymbol{P}(\boldsymbol{x})\,\nabla_{\boldsymbol{x}} V(\boldsymbol{x}), \tag{2.1}$$

where $\boldsymbol{P}(\boldsymbol{x})$ is a matrix function which is symmetric (i.e. $\boldsymbol{P}^\dagger = \boldsymbol{P}$) and positive definite at every point $\boldsymbol{x}$, and where $\nabla_{\boldsymbol{x}} V(\boldsymbol{x}) = [\frac{\partial V}{\partial x_1}, \frac{\partial V}{\partial x_2}, \ldots, \frac{\partial V}{\partial x_n}]^\dagger$. These systems are a special case of the Morse gradient flows discussed in Franks (1982). Since the matrix function $\boldsymbol{P}(\boldsymbol{x})$ is symmetric, it is positive definite if all of the eigenvalues are positive. This matrix defines a way to measure distance, or a *Riemannian metric*, which may change at each point. Relative to this distance measure, the components of the vector field formed by Equation (2.1) are always normal to some level surface of $V(\boldsymbol{x})$. This implies that the trajectories of the system formed by Equation (2.1) always move downhill along the potential surface defined by $V(\boldsymbol{x})$. This can be shown by taking the time derivative of $V(\boldsymbol{x})$ which is $\dot{V}(\boldsymbol{x}) = -[\nabla_{\boldsymbol{x}} V(\boldsymbol{x})]^\dagger\, \boldsymbol{P}(\boldsymbol{x})\, [\nabla_{\boldsymbol{x}} V(\boldsymbol{x})] \leq 0$. Because $\boldsymbol{P}(\boldsymbol{x})$ is positive definite, $\dot{V}(\boldsymbol{x})$ can only be zero where $\nabla_{\boldsymbol{x}} V(\boldsymbol{x}) = \boldsymbol{0}$, elsewhere $\dot{V}(\boldsymbol{x})$ is negative. This means that the trajectories of Equation (2.1) always move toward a level surface of $V(\boldsymbol{x})$ formed by "slicing" $V(\boldsymbol{x})$ at a lower height (i.e. smaller k), as pointed out in Khalil (1992). It is also easy to design systems which remain at a constant height on $V(\boldsymbol{x})$. Such systems will be denoted *Hamiltonian-like systems*. They are specified by the equation

$$\dot{\boldsymbol{x}} = \boldsymbol{Q}(\boldsymbol{x})\,\nabla_{\boldsymbol{x}} V(\boldsymbol{x}), \tag{2.2}$$

where $\boldsymbol{Q}(\boldsymbol{x})$ is a matrix function which is skew-symmetric (i.e. $\boldsymbol{Q}^\dagger = -\boldsymbol{Q}$) at every point $\boldsymbol{x}$. These systems are similar to the Hamiltonian systems defined in Arnold (1989), except that the matrix $\boldsymbol{Q}(\boldsymbol{x})$ is not required to satisfy the Jacobi identity (i.e. $q_{li}\frac{\partial q_{jk}}{\partial x_l} + q_{lj}\frac{\partial q_{ki}}{\partial x_l} + q_{lk}\frac{\partial q_{ij}}{\partial x_l} = 0$). The elements of the vector field defined by Equation (2.2) are always tangent to some level surface of $V(\boldsymbol{x})$. Hence the trajectories of this system remain at a constant height (i.e. at $V(\boldsymbol{x}) = k$) on the potential graph of $V(\boldsymbol{x})$. Again this is indicated by the time derivative of $V(\boldsymbol{x})$, which in this case is $\dot{V}(\boldsymbol{x}) = [\nabla_{\boldsymbol{x}} V(\boldsymbol{x})]^\dagger\, \boldsymbol{Q}(\boldsymbol{x})\, [\nabla_{\boldsymbol{x}} V(\boldsymbol{x})] = 0$. This indicates that the trajectories of Equation (2.2) always remain on the level surface on which the system starts. So a model which can follow an arbitrary downhill path along the potential surface $V(\boldsymbol{x})$ can be designed by combining the dynamics of Equations (2.1) and (2.2). The dynamics in the subspace normal to the level surfaces of $V(\boldsymbol{x})$ can be defined using 1 equation of the form in Equation (2.1). Similarly the dynamics in the subspace tangent to the level surfaces of $V(\boldsymbol{x})$ can be defined using $(n-1)$ equations of the form in Equation (2.2). Hence the total dynamics for the model are

$$\dot{\boldsymbol{x}} = -\boldsymbol{P}(\boldsymbol{x})\,\nabla_{\boldsymbol{x}} V(\boldsymbol{x}) + \sum_{i=2}^{n} \boldsymbol{Q}_i(\boldsymbol{x})\,\nabla_{\boldsymbol{x}} V(\boldsymbol{x}). \tag{2.3}$$

For this model the number and location of equilibria is determined by the function $V(\boldsymbol{x})$, while the manner in which the equilibria are approached is determined by the matrices $\boldsymbol{P}(\boldsymbol{x})$ and $\boldsymbol{Q}_i(\boldsymbol{x})$. The critical points of $V(\boldsymbol{x})$ are the only equilibria of this system.

If the graph of the potential function $V(\boldsymbol{x})$ is ① bounded below (i.e. $V(\boldsymbol{x}) > \mathcal{M}\ \ \forall\ \ \boldsymbol{x} \in \mathbb{R}^n$, where $\mathcal{M}$ is a constant), ② radially unbounded (i.e. $\lim_{\|\boldsymbol{x}\|\to\infty} V(\boldsymbol{x}) \to \infty$) , and ③ has only a finite number of isolated critical points (i.e. in some neighborhood of every point where $\nabla_{\boldsymbol{x}} V(\boldsymbol{x}) = \boldsymbol{0}$ there are no other points where the gradient vanishes), then the system in Equation (2.3) satisfies the conditions of Theorem 10 in Cohen (1992). Therefore the system will converge to one of the critical points of $V(\boldsymbol{x})$ for all initial conditions. Note that this system is capable of all downhill trajectories along the potential surface only if the $(n-1)$ vectors $\boldsymbol{Q}_i(\boldsymbol{x})\,\nabla_{\boldsymbol{x}} V(\boldsymbol{x})\ \ \forall\ \ i = 2, \ldots, n$ are linearly independent at every point $\boldsymbol{x}$. This means that the rank of the $n \times (n-1)$ matrix $(\boldsymbol{Q}_2\,\nabla V\ \ \boldsymbol{Q}_3\,\nabla V\ \ \cdots\ \ \boldsymbol{Q}_n\,\nabla V)$ is $(n-1)$ for all $\boldsymbol{x}$. If the number of states n is even, then it is always possible to construct a system of $(n-1)$ linearly independent vectors which vanish at $\nabla_{\boldsymbol{x}} V(\boldsymbol{x}) = \boldsymbol{0}$. This is due to the following reason. If $V(\boldsymbol{x})$ satisfies the 3 criteria given above, then there is some closed and bounded region which contains all of the critical points. Outside this region, the level surfaces of $V(\boldsymbol{x})$ can be smoothly transformed

into the sphere $\mathcal{S}^{n-1}$ (i.e. for sufficiently large k, there is a homeomorphism from $V(\boldsymbol{x}) = k$ to $\mathcal{S}^{n-1}$). According to Milnor (1965), a result due to Brouwer states that $\mathcal{S}^{n-1}$ has a smooth field of non-zero tangent vectors if and only if $(n-1)$ is odd, which implies that n is even.

Chapter 3

Designing the Potential

This chapter gives a brief overview of the method for potential construction presented as Theorem 4 and Proposition 8 in Cohen (1992). This construction guarantees that the resulting potential graph is bounded below, radially unbounded, and has a finite number of isolated critical points. The construction in Theorem 4 is schematized for a 3-dimensional example in Figure 3.1. First choose a set of points $\mathcal{A}$,

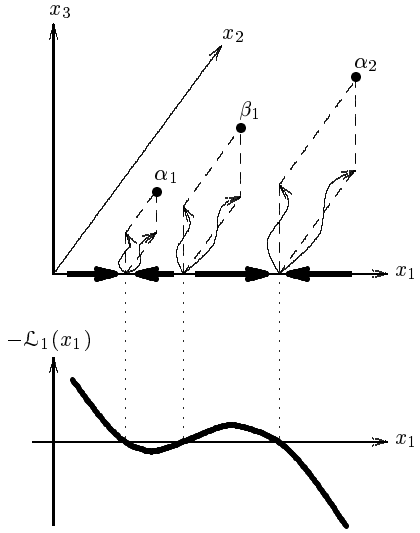


Figure 3.1: The scheme described in Theorem 4 of Cohen (1992). The points labeled α_1 and α_2 are the desired locations for local minima of the potential function, while that labeled β_1 is the desired location for a local maxima. The dashed lines indicate the projections of each of these points onto the x_1 coordinate axis. The function $-\mathcal{L}_1(x_1)$ is a polynomial which vanishes at the x_1 coordinates of the 3 points.

whose coordinates define the locations of the desired point attractors, these points will be local minima of the potential function. In Figure 3.1 these are the two points labeled α_1 and α_2. Project each of these points onto the x_1 coordinate axis. Then choose a second set of points $\mathcal{B}$ which will be the saddle points of the potential function. The x_1 coordinate value of each point in $\mathcal{B}$ must lie in between the x_1 coordinate values of two adjacent points from $\mathcal{A}$. In Figure 3.1 this is the one point labeled β_1. Now construct a polynomial function of x_1 which vanishes at the x_1 coordinate values of all of the points in both $\mathcal{A}$ and $\mathcal{B}$. This is the polynomial labeled $-\mathcal{L}_1(x_1)$ in Figure 3.1. For each of the other coordinates x_i, $i = 2, \ldots, \mathfrak{n}$, construct a Lagrange interpolation polynomial $\mathcal{L}_i(x_1)$ such that the value

of this polynomial at the x_1 coordinate of any point in $\mathcal{A}$ or $\mathcal{B}$ is the projection of that point onto the x_i coordinate axis. The dynamics of this system in the x_1 direction are merely given by the polynomial $-\mathcal{L}_1(x_1)$. In the other directions, the system exponentially decays to the value of $\mathcal{L}_i(x_1)$ corresponding to the value of x_1. So the overall dynamics are

$$\begin{aligned} \dot{x}_1 &= -\mathcal{L}_1(x_1) \\ \dot{x}_i &= -x_i + \mathcal{L}_i(x_1) \;\; \forall \;\; i = 2, \ldots, \mathfrak{n}. \end{aligned} \tag{3.1}$$

Notice that since $\dot{x}_i$ in this equation depends only on x_i and x_1, each $\dot{x}_1$-$\dot{x}_i$ pair can be considered as an independent 2-dimensional system for every i. In fact, because $\mathcal{L}_i(x_1)$ is an interpolation polynomial that depends only on x_1, the entire behavior in the x_i dimension is merely a transformation of the behavior in the x_1 dimension, with $\mathcal{L}_i(x_1)$ defining the transformation. This means that the dynamics of the entire system is "lifted" out of the x_1 dimension by the combination of all $(\mathfrak{n}-1)$ polynomial transformations $\mathcal{L}_i(x_1)$, $i = 2, \ldots, \mathfrak{n}$. Based on these dynamics in Equation (3.1), Proposition 8 of Cohen (1992) presents a potential function $V(\boldsymbol{x})$ whose only critical points are the members of the set $\mathcal{A} \cup \mathcal{B}$. The resulting potential function has the form

$$V(\boldsymbol{x}) = \mathcal{K} \int_{\chi_1}^{x_1} \mathcal{L}_1(\gamma)\, d\gamma + \sum_{i=2}^{\mathfrak{n}} \left[\frac{1}{2} \left(x_i - \mathcal{L}_i(x_1) \right)^2 + \frac{1}{2} \int_{\chi_i}^{x_1} \mathcal{L}_1(\gamma) \left[\mathcal{L}_i'(\gamma) \right]^2 \, d\gamma \right], \tag{3.2}$$

where $\mathcal{K}$ is a real positive constant, $\chi_i \; \forall \; i = 1, \ldots, \mathfrak{n}$ are real constants chosen so that the integrals are positive valued, and $\mathcal{L}_i'(x_1) \equiv \frac{d\mathcal{L}_i}{dx_1}$.

Chapter 4

The Learning Rule

In this chapter we introduce a learning rule for systems similar to those in Equation (2.3). The only change made to the system is the addition of a term for the system inputs. In Equation (2.3) the number and location of equilibria can be controlled using the potential function $V(\boldsymbol{x})$, while the manner in which the equilibria are approached can be controlled with the matrices $\boldsymbol{P}(\boldsymbol{x})$ and $\boldsymbol{Q}_i(\boldsymbol{x})$. If it is assumed that the locations of the equilibria are known, then a potential function which has these critical points can be constructed using Equation (3.2). The problem of system identification is thereby reduced to the problem of parameterizing the matrices $\boldsymbol{P}(\boldsymbol{x})$ and $\boldsymbol{Q}_i(\boldsymbol{x})$ and finding the parameter values which cause this model to best emulate the actual system. If the elements $\boldsymbol{P}(\boldsymbol{x})$ and $\boldsymbol{Q}_i(\boldsymbol{x})$ are correctly chosen, then a learning rule can be designed which makes the model dynamics converge to that of the actual system.

Specifically, choose each element of these matrices to have the form

$$P_{rs} = \sum_{j=1}^{\mathtt{n}} \sum_{k=0}^{\mathtt{l}-1} \xi_{rsjk}\, \vartheta_k(x_j) \qquad \text{and} \qquad Q_{rs} = \sum_{j=1}^{\mathtt{n}} \sum_{k=0}^{\mathtt{l}-1} \lambda_{rsjk}\, \varrho_k(x_j), \tag{4.1}$$

where $\{\vartheta_0(x_j), \vartheta_1(x_j), \ldots, \vartheta_{\mathtt{l}-1}(x_j)\}$ and $\{\varrho_0(x_j), \varrho_1(x_j), \ldots, \varrho_{\mathtt{l}-1}(x_j)\}$ are a set of $\mathtt{l}$ orthogonal polynomials which depend on the state x_j. There is a set of such polynomials for every state x_j, $j = 1, 2, \ldots, \mathtt{n}$. The constants ξ_{rsjk} and λ_{rsjk} determine the contribution of the kth polynomial which depends on the jth state to the value of P_{rs} and Q_{rs} respectively. In this case the dynamics in Equation (2.3) become

$$\dot{\boldsymbol{x}} = \sum_{j=1}^{\mathtt{n}} \sum_{k=0}^{\mathtt{l}-1} \left\{ \boldsymbol{\Xi}_{jk} \left[\vartheta_k(x_j)\, \nabla_{\boldsymbol{x}} V(\boldsymbol{x}) \right] + \sum_{i=2}^{\mathtt{n}} \boldsymbol{\Lambda}_{ijk} \left[\varrho_{ik}(x_j)\, \nabla_{\boldsymbol{x}} V(\boldsymbol{x}) \right] \right\} + \boldsymbol{\Upsilon}\, \boldsymbol{g}(\boldsymbol{u}(t)) = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\xi}, \boldsymbol{\lambda}, \boldsymbol{\upsilon}, t) \tag{4.2}$$

where $\boldsymbol{\Xi}_{jk}$ is the $(\mathtt{n} \times \mathtt{n})$ matrix of all values ξ_{rsjk} which have the same value of j and k. Likewise $\boldsymbol{\Lambda}_{ijk}$ is the $(\mathtt{n} \times \mathtt{n})$ matrix of all values λ_{rsjk}, having the same value of j and k, which are associated with the ith matrix $\boldsymbol{Q}_i(\boldsymbol{x})$. This system has $\mathtt{m}$ inputs, which may explicitly depend on time, that are represented by the $\mathtt{m}$-element vector function $\boldsymbol{u}(t)$. The $\mathtt{m}$-element vector function $\boldsymbol{g}(\cdot)$ is a smooth, possibly nonlinear, transformation of the input function. The matrix $\boldsymbol{\Upsilon}$ is an $(\mathtt{n} \times \mathtt{m})$ parameter matrix which determines how much of input $s \in \{1, \ldots, \mathtt{m}\}$ effects state $r \in \{1, \ldots, \mathtt{n}\}$. So the dynamics depend on the system states $\boldsymbol{x}$ and all of the parameters $\boldsymbol{\xi} = [\xi_{rsjk}]^\dagger : r, s, j = 1, \ldots, \mathtt{n},\ k = 0, \ldots, \mathtt{l}-1$, $\boldsymbol{\lambda} = [\lambda_{rsjk}]^\dagger : r, s, j = 1, \ldots, \mathtt{n},\ k = 0, \ldots, \mathtt{l}-1$ and $\boldsymbol{\upsilon} = [\upsilon_{rs}]^\dagger : r = 1, \ldots, \mathtt{n},\ s = 1, \ldots, \mathtt{m}$.

The dynamics given by Equation (4.2) are a model of the actual system dynamics. Using this model and samples of the actual system states, an estimator for the states of the actual system can be designed. The dynamics of this state estimator are

$$\dot{\hat{\boldsymbol{x}}} = \boldsymbol{\mathcal{R}}_s\, (\hat{\boldsymbol{x}} - \boldsymbol{x}) + \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\xi}, \boldsymbol{\lambda}, \boldsymbol{\upsilon}, t) \tag{4.3}$$

where $\boldsymbol{x}$ is a sample of the actual system states. The term $\boldsymbol{\mathcal{R}}_s$ is a matrix of real constants whose eigenvalues must all be negative. This means that $\hat{\boldsymbol{x}}$ is an estimate of the actual system states which depends on the form of the model $\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{\xi}, \boldsymbol{\lambda}, \boldsymbol{\upsilon}, t)$. The goal is to find a set of parameters $\boldsymbol{\xi}$, $\boldsymbol{\lambda}$ and $\boldsymbol{\upsilon}$ which cause the error $(\hat{\boldsymbol{x}} - \boldsymbol{x})$ to vanish. The dynamics of a parameter estimator which accomplishes this are

$$\begin{aligned}
\dot{\boldsymbol{\Xi}}_{jk} &= -\boldsymbol{\mathcal{R}}_p\,(\hat{\boldsymbol{x}} - \boldsymbol{x})\left[\vartheta_k(x_j)\,\nabla_{\boldsymbol{x}} V(\boldsymbol{x})\right]^{\dagger} \;\;\forall\;\; j = 1, \ldots, \mathfrak{n},\;\; k = 0, \ldots, \mathfrak{l} - 1 \\
\dot{\boldsymbol{\Lambda}}_{ijk} &= -\boldsymbol{\mathcal{R}}_p\,(\hat{\boldsymbol{x}} - \boldsymbol{x})\left[\varrho_{ik}(x_j)\,\nabla_{\boldsymbol{x}} V(\boldsymbol{x})\right]^{\dagger} \;\;\forall\;\; i, j = 1, \ldots, \mathfrak{n},\;\; k = 0, \ldots, \mathfrak{l} - 1 \\
\dot{\boldsymbol{\Upsilon}} &= -\boldsymbol{\mathcal{R}}_p\,(\hat{\boldsymbol{x}} - \boldsymbol{x})\left[\boldsymbol{g}(\boldsymbol{u}(t))\right]^{\dagger},
\end{aligned} \tag{4.4}$$

where $\boldsymbol{\mathcal{R}}_p$ is a matrix of real constants which is symmetric and positive definite. Note that the term $(\hat{\boldsymbol{x}} - \boldsymbol{x})\left[\vartheta_k(x_j)\,\nabla_{\boldsymbol{x}} V(\boldsymbol{x})\right]^{\dagger}$ is the outer product of $\mathfrak{n}$-dimensional vectors, hence the result is an $(\mathfrak{n} \times \mathfrak{n})$ matrix. Likewise the terms $(\hat{\boldsymbol{x}} - \boldsymbol{x})\left[\varrho_{ik}(x_j)\,\nabla_{\boldsymbol{x}} V(\boldsymbol{x})\right]^{\dagger}$ and $(\hat{\boldsymbol{x}} - \boldsymbol{x})\left[\boldsymbol{g}(\boldsymbol{u}(t))\right]^{\dagger}$ are also outer products. The following theorem shows that the system of differential equations defined by Equations (4.2), (4.3) and (4.4) converge to a set of parameters such that the error $(\hat{\boldsymbol{x}} - \boldsymbol{x})$ between the estimated and target trajectories vanishes.

Theorem 4.1. *Given the model system*

$$\dot{\boldsymbol{x}} = \sum_{i=1}^{\mathsf{k}} \boldsymbol{A}_i\,\boldsymbol{f}_i(\boldsymbol{x}) + \boldsymbol{B}\,\boldsymbol{g}(\boldsymbol{u}(t)) \tag{4.5}$$

where $\boldsymbol{A}_i \in \mathbb{R}^{\mathfrak{n}\times\mathfrak{n}}$ *and* $\boldsymbol{B} \in \mathbb{R}^{\mathfrak{n}\times\mathfrak{m}}$ *are unknown matrices, and* $\boldsymbol{f}_i : \mathbb{R}^{\mathfrak{n}} \to \mathbb{R}^{\mathfrak{n}}$, $\boldsymbol{f}_i \in C^1$ *and* $\boldsymbol{g} : \mathbb{R}^{\mathfrak{m}} \to \mathbb{R}^{\mathfrak{m}}$, $\boldsymbol{g} \in C^1$ *are known functions such that* $\|\boldsymbol{u}(t)\| \leq \mathcal{U}$ *for some* $\mathcal{U} > 0$ *implies* $\|\boldsymbol{x}(t)\| \leq \mathcal{S}_u$ *for some* $\mathcal{S}_u > 0$ *(i.e. bounded inputs imply bounded solutions). Choose a state estimator of the form*

$$\dot{\hat{\boldsymbol{x}}} = \boldsymbol{\mathcal{R}}_s\,(\hat{\boldsymbol{x}} - \boldsymbol{x}) + \sum_{i=1}^{\mathsf{k}} \hat{\boldsymbol{A}}_i\,\boldsymbol{f}_i(\boldsymbol{x}) + \hat{\boldsymbol{B}}\;\boldsymbol{g}(\boldsymbol{u}(t)) \tag{4.6}$$

where $\boldsymbol{\mathcal{R}}_s \in \mathbb{R}^{\mathfrak{n}\times\mathfrak{n}}$ *is a matrix of real constants whose eigenvalues must all be negative, and* $\hat{\boldsymbol{A}}_i$ *and* $\hat{\boldsymbol{B}}$ *are the estimates of the actual parameters. Choose parameter estimators of the form*

$$\begin{aligned}
\dot{\hat{\boldsymbol{A}}}_i &= -\boldsymbol{\mathcal{R}}_p\,(\hat{\boldsymbol{x}} - \boldsymbol{x})\left[\boldsymbol{f}_i(\boldsymbol{x})\right]^{\dagger} \;\;\forall\;\; i = 1, \ldots, \mathsf{k} \\
\dot{\hat{\boldsymbol{B}}} &= -\boldsymbol{\mathcal{R}}_p\,(\hat{\boldsymbol{x}} - \boldsymbol{x})\left[\boldsymbol{g}(\boldsymbol{u}(t))\right]^{\dagger}
\end{aligned} \tag{4.7}$$

where $\boldsymbol{\mathcal{R}}_p \in \mathbb{R}^{\mathfrak{n}\times\mathfrak{n}}$ *is a matrix of real constants which is symmetric and positive definite, and* $(\hat{\boldsymbol{x}} - \boldsymbol{x})\left[\cdot\right]^{\dagger}$ *denotes an outer product. For these choices of state and parameter estimators* $\lim_{t\to\infty}(\hat{\boldsymbol{x}} - \boldsymbol{x}) = 0$ *for all initial conditions. Furthermore, this remains true if* any *of the elements of* $\hat{\boldsymbol{A}}_i$ *or* $\hat{\boldsymbol{B}}$ *are set to* 0*, or if any of these matrices are restricted to being symmetric or skew-symmetric.*

The proof of this theorem appears in the Appendix. Note that convergence of the parameter estimates to the actual parameter values is not guaranteed by this theorem. Since Equations (4.2), (4.3), and (4.4) are in the form of Equations (4.5), (4.6), and (4.7) respectively, Theorem 4.1 implies that the parameter estimates produced by Equation (4.4) cause the state estimates in Equation (4.3) to converge to the actual state values.

Theorem 4.1 is based on the assumption that the state vector in Equation (4.5) is bounded if the input $\boldsymbol{u}(t)$ is bounded (i.e. BIBS stability). If $\boldsymbol{f}_i(\cdot)$ and $\boldsymbol{g}(\cdot)$ are linear functions, the resulting linear system is BIBS stable if it is asymptotically stable when $\boldsymbol{u}(t) = \boldsymbol{0}$, as shown by Willems (1970). However, it was shown by Varaiya and Liu (1966) that asymptotic stability of the zero input case alone does not guarantee BIBS stability for nonlinear systems. This means that in order to determine the boundedness of the solutions $\boldsymbol{x}(t)$ of Equation (4.2), a non-autonomous nonlinear system must be considered. In general this can be quite difficult, but for systems of this form, results in LaSalle and Lefschetz (1961) can be used to prove the following theorem.

Theorem 4.2. *Given the dynamical system*

$$\dot{\boldsymbol{x}} = -\boldsymbol{P}(\boldsymbol{x})\,\nabla_{\boldsymbol{x}} V(\boldsymbol{x}) + \sum_{i=2}^{\mathrm{n}} \boldsymbol{Q}_i(\boldsymbol{x})\,\nabla_{\boldsymbol{x}} V(\boldsymbol{x}) + \boldsymbol{h}(\boldsymbol{u}(t)), \tag{4.8}$$

where $V : \mathbb{R}^{\mathrm{n}} \to \mathbb{R}$, $V \in C^2$ *is the potential function,* $\boldsymbol{h} : \mathbb{R}^{\mathrm{m}} \to \mathbb{R}^{\mathrm{n}}$, $\boldsymbol{h} \in C^1$, *and* $\boldsymbol{u} : \mathbb{R} \to \mathbb{R}^{\mathrm{m}}$, $\boldsymbol{u} \in C^1$ *is a time varying input function. The matrix function* $\boldsymbol{P} : \mathbb{R}^{\mathrm{n}} \to \mathbb{R}^{\mathrm{n}\times\mathrm{n}}$, $\boldsymbol{P} \in C^1$ *is symmetric positive definite, and* $\boldsymbol{Q}_i : \mathbb{R}^{\mathrm{n}} \to \mathbb{R}^{\mathrm{n}\times\mathrm{n}}$, $\boldsymbol{Q}_i \in C^1\ \forall\ i = 2, \ldots, \mathrm{n}$ *are skew-symmetric. Furthermore,* $V(\boldsymbol{x}) \geq 0$ *for all* $\boldsymbol{x}$, *and there exists an* $\mathcal{F}_u > 0$, *such that for* $\|\boldsymbol{x}\| > \mathcal{F}_u$, $\|\nabla_{\boldsymbol{x}} V(\boldsymbol{x})\| \geq \mathcal{K}_u$ *for some* $\mathcal{K}_u > 0$ (*i.e. the length of* $\nabla_{\boldsymbol{x}} V(\boldsymbol{x})$ *has a non-zero lower bound*). *Also there exists a* $\mathcal{U} > 0$ *such that* $\|\boldsymbol{u}(t)\| \leq \mathcal{U}$. *If all of the above conditions are satisfied, then there exists* $\mathcal{S}_u > 0$ *such that corresponding to each solution* $\boldsymbol{x}(t)$ *of Equation* (4.8) *there is a* $\mathcal{T} > 0$ *with the property that* $\|\boldsymbol{x}(t)\| \leq \mathcal{S}_u$ *for all* $t > \mathcal{T}$ (*i.e. the solutions* $\boldsymbol{x}(t)$ *of Equation* (4.8) *are ultimately bounded*).

For the proof of this see the Appendix. This theorem states that if there is a region outside which the length of $\nabla_{\boldsymbol{x}} V(\boldsymbol{x})$ has a non-zero lower bound, then all solutions to Equation (4.8) are ultimately bounded provided that the norm of the input signal $\|\boldsymbol{u}(t)\|$ is bounded. Note that the system has n states and m inputs. It turns out that $\mathcal{K}_u$ depends on $\mathcal{U}$, the upper bound on $\|\boldsymbol{u}(t)\|$ (see the proof). So if the system is to accommodate arbitrarily large inputs, there must be a region $\|\boldsymbol{x}\| > \mathcal{F}_m$ in which $\|\nabla_{\boldsymbol{x}} V(\boldsymbol{x})\|$ is strictly increasing (i.e. $\|\boldsymbol{x}_1\| > \|\boldsymbol{x}_2\| \Rightarrow \|\nabla_{\boldsymbol{x}} V(\boldsymbol{x}_1)\| > \|\nabla_{\boldsymbol{x}} V(\boldsymbol{x}_2)\|$). If this is the case, then for any $\mathcal{K}_u$, and hence any $\mathcal{U}$, there exists a region $\|\boldsymbol{x}\| > \mathcal{F}_u \geq \mathcal{F}_m$ in which $\|\nabla_{\boldsymbol{x}} V(\boldsymbol{x})\| \geq \mathcal{K}_u$. The condition $\|\nabla_{\boldsymbol{x}} V(\boldsymbol{x})\| \geq \mathcal{K}_u$ implies that $V(\boldsymbol{x}) \geq \mathcal{K}_u \|\boldsymbol{x}\|$ which means that $V(\boldsymbol{x})$ is radially unbounded, but not necessarily convex or even increasing. It is not obvious what condition on $V(\boldsymbol{x})$ implies $\|\nabla_{\boldsymbol{x}} V(\boldsymbol{x})\| \geq \mathcal{K}_u$, for instance $V(\boldsymbol{x}) \geq \mathcal{K}_u \|\boldsymbol{x}\| \not\Rightarrow \|\nabla_{\boldsymbol{x}} V(\boldsymbol{x})\| \geq \mathcal{K}_u$. An interesting converse to this theorem can also be proven. If $V(\boldsymbol{x})$ is continuous, lower bounded, and has some region $\|\boldsymbol{x}\| > \mathcal{F}_u$ where $\|\nabla_{\boldsymbol{x}} V(\boldsymbol{x})\| \geq \mathcal{K}_u$, then there exists some region (or possibly regions) $\|\boldsymbol{x} - \boldsymbol{\mathcal{C}}\| < \mathcal{F}_l$ wherein $\|\nabla_{\boldsymbol{x}} V(\boldsymbol{x})\| \leq \mathcal{K}_l$ for some $\mathcal{F}_l, \mathcal{K}_l > 0$. In this region it can be shown that $\dot{V}(\boldsymbol{x})$ is always positive, hence this region is unstable and the system will eventually leave it. Therefore the solutions of Equation (4.8) have both an ultimate upper bound and an ultimate lower bound, so for $t > \mathcal{T}$, $\mathcal{S}_l \leq \|\boldsymbol{x}(t)\| \leq \mathcal{S}_u$ for some $\mathcal{S}_u \geq \mathcal{S}_l > 0$.

As previously stated, Theorem 4.1 does not guarantee the convergence of the parameter estimates to the actual parameter values. This issue has been widely addressed in the adaptive identification and control literature, as discussed in Narendra and Annaswamy (1987). It was determined that if the signals within the adaptive system possessed certain properties, then the origin of the system was globally uniformly asymptotically stable. This guarantees the convergence of the parameter estimates. Signals with these properties are said to be *persistently exciting* by Narendra and Annaswamy (1987). Intuitively, persistent excitation means that the input is rich enough to excite all the modes of the system being considered. For linear systems persistent excitation becomes a condition on the input signal alone, since a linear system can not generate frequency modes. For a nonlinear system the condition must be on both the input signal and the internal signals of the system, since nonlinear systems can generate new frequency modes. Using results from Morgan and Narendra (1977) the following theorem can be proven for the identification system defined by Equations (4.5), (4.6) and (4.7).

Theorem 4.3. *Given the model system*

$$\dot{\boldsymbol{x}} = \sum_{i=1}^{\mathrm{k}} \boldsymbol{A}_i\,\mathcal{F}_i(\tilde{\boldsymbol{x}})\,\nabla_{\boldsymbol{x}} V(\boldsymbol{x}) + \boldsymbol{B}\,\boldsymbol{g}(\boldsymbol{u}(t)), \tag{4.9}$$

where $\mathcal{F}_i : \mathbb{R}^{\mathrm{d}} \to \mathbb{R}$, $\mathrm{d} \leq \mathrm{n}$ (*i.e.* $\tilde{\boldsymbol{x}} \subseteq \{x_1, x_2, \ldots, x_{\mathrm{n}}\}$), $\mathcal{F}_i \in C^1$. *Let all* $\boldsymbol{A}_i \in \mathbb{R}^{\mathrm{n}\times\mathrm{n}}$ *be either symmetric positive definite or skew-symmetric and let Equation* (4.9) *satisfy all of the conditions in Theorems* 4.1 *and* 4.2. *Define the error functions* $\boldsymbol{e} = \hat{\boldsymbol{x}} - \boldsymbol{x}$, $\boldsymbol{\Phi}_i = \hat{\boldsymbol{A}}_i - \boldsymbol{A}_i$, *and* $\boldsymbol{\Psi} = \hat{\boldsymbol{B}} - \boldsymbol{B}$. *From Equations* (4.6)

and (4.7) *the state and parameter error dynamics are*

$$\begin{aligned}
\dot{\boldsymbol{e}} &= \dot{\hat{\boldsymbol{x}}} - \dot{\boldsymbol{x}} = \boldsymbol{\mathcal{R}}_s\, \boldsymbol{e} + \sum_{i=1}^{\mathrm{k}} \boldsymbol{\Phi}_i\, \mathcal{F}_i(\tilde{\boldsymbol{x}})\, \nabla_{\boldsymbol{x}} V(\boldsymbol{x}) + \boldsymbol{\Psi}\, \boldsymbol{g}(\boldsymbol{u}(t)), \\
\dot{\boldsymbol{\Phi}}_i &= \dot{\hat{\boldsymbol{A}}}_i - \dot{\boldsymbol{A}}_i = -\boldsymbol{\mathcal{R}}_p\, \boldsymbol{e}\, \left[\mathcal{F}_i(\tilde{\boldsymbol{x}})\, \nabla_{\boldsymbol{x}} V(\boldsymbol{x})\right]^{\dagger}, \\
\dot{\boldsymbol{\Psi}} &= \dot{\hat{\boldsymbol{B}}} - \dot{\boldsymbol{B}} = -\boldsymbol{\mathcal{R}}_p\, \boldsymbol{e}\, \boldsymbol{g}(\boldsymbol{u}(t))^{\dagger},
\end{aligned} \tag{4.10}$$

Let $\|\dot{\boldsymbol{u}}(t)\| \leq \mathcal{D}$ *for some* $\mathcal{D} > 0$*, and let there exist positive constants* t_0*,* $\mathcal{T}$*, and* ϵ *such that for every unit vector* $\boldsymbol{w} \in \mathbb{R}^{\mathrm{n}+\mathrm{m}}$

$$\frac{1}{\mathcal{T}} \int_t^{t+\mathcal{T}} \left\| \left\{ \left[\mathcal{F}_1(\tilde{\boldsymbol{x}}(\tau)) + \mathcal{F}_2(\tilde{\boldsymbol{x}}(\tau)) + \cdots + \mathcal{F}_{\mathrm{k}}(\tilde{\boldsymbol{x}}(\tau))\right] \nabla_{\boldsymbol{x}} V(\boldsymbol{x}(\tau))^{\dagger}\, \boldsymbol{g}(\boldsymbol{u}(\tau))^{\dagger} \right\} \boldsymbol{w} \right\| d\tau \geq \epsilon \quad \forall \quad t \geq t_0. \tag{4.11}$$

Then the equilibrium point $\boldsymbol{e} = \boldsymbol{0}$*,* $\boldsymbol{\Phi}_i = \mathbf{O}$*,* $\boldsymbol{\Psi} = \mathbf{O}$ *is globally uniformly asymptotically stable.*

See the Appendix for a proof of this Theorem. Note that Equation (4.10) is non-autonomous due to the input term. Also, the choice of parameter error dynamics is dictated by the fact that the actual parameters $\boldsymbol{A}_i$ and $\boldsymbol{B}$ are assumed to be unknown constants. This theorem gives a condition on the internal signals and inputs of the system in Equation (4.10) which guarantee convergence of the parameter estimates to their actual values. The intuitive meaning of this condition is far from obvious. In part it means that there is a time interval $\mathcal{T}$ over which the vector $\left\{\left[\sum_{i=1}^{\mathrm{k}} \mathcal{F}_i(\tilde{\boldsymbol{x}})\right]\, \nabla_{\boldsymbol{x}} V(\boldsymbol{x})\, \boldsymbol{g}(\boldsymbol{u}(t))\right\}$ points in all directions with sufficient length as t takes on values in the interval. Notice that in Equation (4.2) the form of $\mathcal{F}_i(\tilde{\boldsymbol{x}})$ is $\omega_k(x_j)$, $k = 0, 1, \ldots, \mathrm{l} - 1$, $j = 1, 2, \ldots, \mathrm{n}$ where $\omega_k(\cdot)$ is the kth member of a set of l orthogonal polynomials, and $x_j \in \{x_1, x_2, \ldots, x_{\mathrm{n}}\}$.

Chapter 5

Simulation Results

Now an example is presented in which the parameters of the model in Equation (4.2) are learned, using the training rule in Equations (4.3) and (4.4), on one input signal and then are tested on a different input signal. The actual system has three equilibrium points, two stable points located at $(1,3)$ and $(3,5)$, and a saddle point located at $(2-\frac{\sqrt{3}}{3}, 4+\frac{\sqrt{3}}{3})$. In this example the dynamics of both the actual system and the model are given by

$$\begin{pmatrix}\dot{x}_1\\ \dot{x}_2\end{pmatrix} = -\begin{pmatrix}\mathcal{A}_1+\mathcal{A}_2\,x_1^2+\mathcal{A}_3\,x_2^2 & 0\\ 0 & \mathcal{A}_4+\mathcal{A}_5\,x_1^2+\mathcal{A}_6\,x_2^2\end{pmatrix}\begin{pmatrix}\dfrac{\partial V}{\partial x_1}\\ \dfrac{\partial V}{\partial x_2}\end{pmatrix} + \begin{pmatrix}0 & -\{\mathcal{A}_7+\mathcal{A}_8\,x_1+\mathcal{A}_9\,x_2\}\\ \mathcal{A}_7+\mathcal{A}_8\,x_1+\mathcal{A}_9\,x_2 & 0\end{pmatrix}\begin{pmatrix}\dfrac{\partial V}{\partial x_1}\\ \dfrac{\partial V}{\partial x_2}\end{pmatrix} + \begin{pmatrix}\mathcal{A}_{10}\\ 0\end{pmatrix} u, \qquad (5.1)$$

where $V(\boldsymbol{x})$ is defined in Equation (3.2) and u is a time varying input. For the actual system the parameter values were $\mathcal{A}_1 = \mathcal{A}_4 = -4$, $\mathcal{A}_2 = \mathcal{A}_5 = -2$, $\mathcal{A}_3 = \mathcal{A}_6 = -1$, $\mathcal{A}_7 = 1$, $\mathcal{A}_8 = 3$, $\mathcal{A}_9 = 5$, and $\mathcal{A}_{10} = 1$. In the model the 10 elements $\mathcal{A}_i$ are treated as the unknown parameters which must be learned. Note that the first matrix function is positive definite if the parameters $\mathcal{A}_1$–$\mathcal{A}_6$ are all positive valued. The second matrix function is skew-symmetric for all values of $\mathcal{A}_7$–$\mathcal{A}_9$. For this particular system $\nabla_{\boldsymbol{x}} V(\boldsymbol{x})$ is

$$\begin{pmatrix}\dfrac{\partial V}{\partial x_1}\\ \dfrac{\partial V}{\partial x_2}\end{pmatrix} = \begin{pmatrix}576\,x_1^5 - 5379.45\,x_1^4 + 19742.3\,x_1^3 - 35767.5\,x_1^2 + 31999.2\,x_1 - 24\,x_1\,x_2 + 47\,x_2 - 11239.5\\ -12\,x_1^2 + 47\,x_1 + x_2 - 38\end{pmatrix}. \qquad (5.2)$$

It is relatively easy to show that for this example, $\|\nabla_{\boldsymbol{x}} V(\boldsymbol{x})\|$ is eventually strictly increasing as illustrated in Figure 5.1. The function is actually increasing in the X-shaped trough seen in the fig-

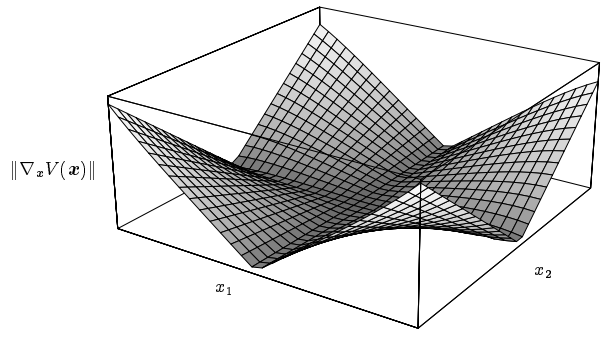


Figure 5.1: The graph of the norm of the gradient $\|\nabla_{\boldsymbol{x}} V(\boldsymbol{x})\|$ for the system defined in Equation (5.2).

ure, but at a much slower rate than in the surrounding areas. This means that for *any* bounded input, the system defined by Equations (5.1) and (5.2) satisfies the conditions in Theorem 4.2 and therefore has ultimately bounded solutions. The two input signals used for training and testing were $u_1(t) = 10000\left(\sin\frac{1}{3}1000\,t + \sin\frac{2}{3}1000\,t\right)$ and $u_2(t) = 5000\,\sin 1000\,t$. The phase space responses of the actual system to the inputs u_1 and u_2 are shown by the solid curves in Figures 5.4(b) and 5.4(a) respectively. Note that both of these inputs produce a periodic attractor in the phase space of Equation (5.1).

In order to evaluate the effectiveness of the learning algorithm the Euclidean distance between the actual and learned state and parameter values was computed and plotted versus time. The results are shown in Figure 5.2. Figure 5.2(a) shows these statistics when training with input u_1, while Figure 5.2(b) shows the same statistics for input u_2. The solid curves are the Euclidean distance between the learned and actual system states, and the dashed curves are the distance between the learned and actual parameter values. These statistics have two noteworthy features. First, the error between the learned and desired states quickly converges to very small values, regardless of how well the actual parameters are learned. This result was guaranteed by Theorem 4.1. Second, the minimum error between the learned and desired parameters is much lower when the system is trained with input u_1. Specifically the minimum parameter error for input u_1 is 1.65, while for input u_2 it is 6.47. Intuitively this is because input u_1 excites more frequency modes of the system than input u_2. Notice that the parameter error curve in Figure 5.2(a) appears to be eventually monotonically decreasing. So it seems reasonable to conclude that for input u_1 the parameter estimates eventually converge to the actual parameter values. The same conclusion also seems to justified for input u_2 since the envelope of the parameter error curve in Figure 5.2(b) decreases with time. These observations illustrate the relationship between parameter convergence and persistent excitation that was addressed in Theorem 4.3. Recall that in a nonlinear system the frequency modes excited by a given input do not depend solely on the input because the system can generate frequencies not present in the input. These conclusions are further supported by the plots of the power spectrum of state $x_1(t)$ for each input, shown in Figure 5.3. Figure 5.3(a) shows the power spectrum for input $u_1(t)$, while Figure 5.3(b) shows it for input $u_2(t)$. The dashed lines show the frequencies present in the input signal. Note that the DC peak in both power spectra is due to the fact that neither of the periodic structures generated by these inputs is centered at the origin. These plots have two features of note. First, input u_1 clearly excites more system modes than input u_2. This partially explains why the parameter convergence for u_1 is better than for u_2, as shown in Figure 5.2.

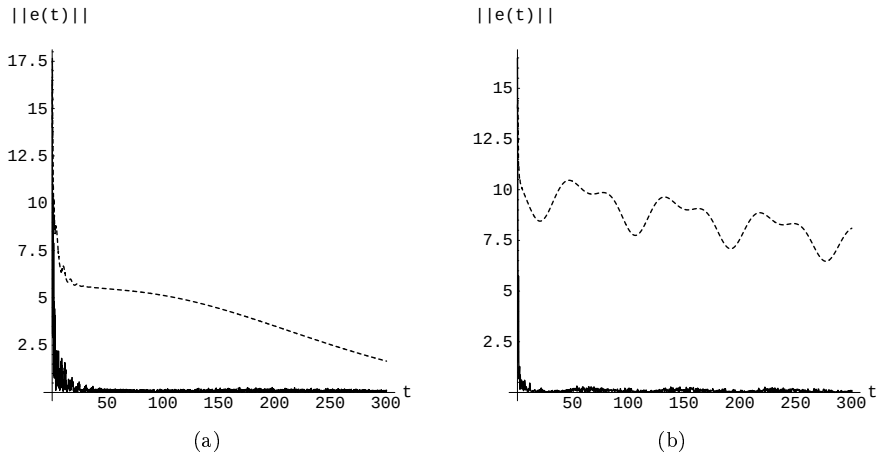
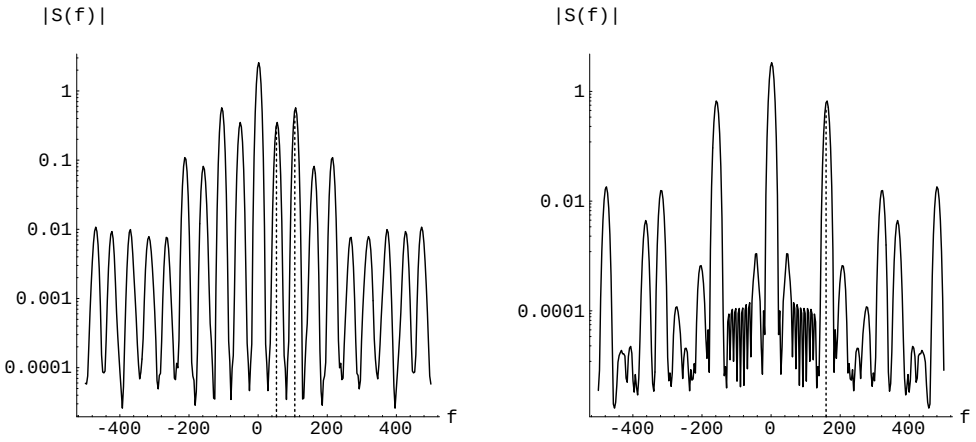


Figure 5.2: (a) The state and parameter errors for training using input signal u_1. The solid curve is the Euclidean distance (i.e. $\sqrt{\sum_{i=1}^{2}\hat{x}_i - x_i}$) between the state estimates and the actual states as a function of time. The dashed curve shows the distance (i.e. $\sqrt{\sum_{i=1}^{10}\mathcal{A}_i - \mathcal{A}_i}$) between the estimated and actual parameter values versus time.
(b) The state and parameter errors for training using input signal u_2.

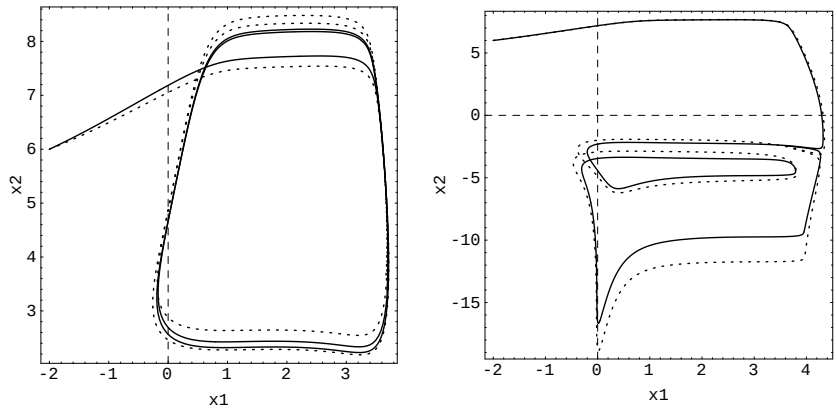


(a) (b)

Figure 5.3: (a) The power spectral density versus frequency of the state $x_1(t)$ when Equation (5.1) is driven by input $u_1(t)$. The dashed lines represent the two frequencies present in the input. (b) The power spectral density of $x_1(t)$ for input $u_2(t)$. The dashed line represents the one frequency present in the input.

Second, both inputs excite modes in the system which are at frequencies *not* present in the input. This is a result of the nonlinearities in Equation (5.1). The large number of spectral components supports the conclusion that for this particular system both u_1 and u_2 are persistently exciting.

The quality of the learned parameters can be qualitatively judged by comparing the phase plots using the learned and actual parameters for each input, as shown in Figure 5.4. In Figure 5.4(a) the



(a) (b)

Figure 5.4: (a) A phase plot of the system response when trained with input u_1 and tested with input u_2. The solid line is the response to the test input using the actual parameters. The dotted line is the system response using the learned parameters.
(b) A phase plot of the system response when trained with input u_2 and tested with input u_1.

system was trained using input u_1 and tested with input u_2, while in Figure 5.4(b) the situation was reversed. The solid curves are the system response using the actual parameter values, and the dashed curves are the response using the final values of the learned parameters. The Euclidean distance between the target and test trajectories in Figure 5.4(a) is in the range $(0, 0.64)$ with a mean distance of 0.21 and a standard deviation of 0.14. The distance between the the target and test trajectories in Figure 5.4(b)

is in the range $(0, 4.53)$ with a mean distance of 0.98 and a standard deviation of 1.35. Qualitatively, both sets of learned parameters give an accurate response for non-training inputs. Note that even when the error between the learned and actual parameters is large, the periodic attractor resulting from the learned parameters appears to have the same "shape" as that for the actual parameters.

Chapter 6

Conclusion

We have presented a conceptual framework for designing dynamical systems with specific qualitative properties by decomposing the dynamics into a component normal to some surface and a set of components tangent to the same surface. We have presented a specific instance of this class of systems which converges to one of a finite number of equilibrium points. By parameterizing these systems, the manner in which these equilibrium points are approached can be fitted to an arbitrary data set. We also presented a learning algorithm to estimate these parameters which is guaranteed to converge to a set of parameter values for which the error between the learned and desired trajectories vanishes.

Bibliography

Arnold, V. (1989). *Mathematical methods of classical mechanics* (2nd edition)., Vol. 60 of *Graduate Texts in Mathematics*. Springer-Verlag, Inc., New York, NY.

Cohen, M. (1992). The construction of arbitrary stable dynamics in nonlinear neural networks. *Neural Networks*, *5*(1), 83–103.

Franks, J. (1982). *Homology and dynamical systems*, Vol. 49 of *Regional Conference Series in Mathematics*. American Mathematical Society, Providence, RI.

Funahashi, K.-I., & Nakamura, Y. (1993). Approximation of dynamical systems by continuous time recurrent neural networks. *Neural Networks*, *6*(5), 801–806.

Khalil, H. (1992). *Nonlinear Systems*. Macmillan Publishing Co., New York, NY.

LaSalle, J., & Lefschetz, S. (1961). *Stability by Lyapunov's direct method, with applications* (1st edition)., Vol. 4 of *Mathematics in Science and Engineering*. Academic Press, Inc., New York, NY.

Mendes, R., & Duarte, J. (1981). Decomposition of vector fields and mixed dynamics. *Journal of Mathematical Physics*, *22*(7), 1420–1422.

Milnor, J. (1965). *Topology from a Differentiable Viewpoint*. University Press of Virginia, Charlottesville, VA.

Morgan, A., & Narendra, K. (1977). On the stability of non-autonomous differential equations $\dot{x} = [A + B(t)]x$ with skew-symmetric matrix $B(t)$. *SIAM Journal of Control and Optimization*, *15*(1), 163–176.

Narendra, K., & Annaswamy, A. (1987). Persistent excitation in adaptive systems. *International Journal of Control*, *45*(1), 127–160.

Narendra, K., & Annaswamy, A. (1989). *Stable Adaptive Systems*. Prentice-Hall, Inc., Englewood Cliffs, NJ.

Narendra, K., & Parthasarathy, K. (1990). Identification and control of dynamical systems using neural networks. *IEEE Transactions on Neural Networks*, *1*(1), 4–27.

Pearlmutter, B. (1989). Learning state space trajectories in recurrent neural networks. *Neural Computation*, *1*(2), 263–269.

Saad, D. (1992). Training recurrent neural networks via trajectory modification. *Complex Systems*, *6*(2), 213–236.

Sato, M.-A. (1990). A real time learning algorithm for recurrent analog neural networks. *Biological Cybernetics*, *62*(2), 237–241.

Sontag, E. (1992). Neural nets as system models and controllers. In *Proceedings of the 7th Yale Workshop on Adaptive and Learning Systems*, pp. 73–79. Yale University Press.

Varaiya, P., & Liu, R. (1966). Bounded-input bounded-output stability of nonlinear time-varying differential systems. *Journal of SIAM, Series A: Control*, *4*(4), 699–704.

Willems, J. (1970). *Stability theory of dynamical systems*. John Wiley & Sons, Inc., New York, NY.

Appendix A

Proofs

***Proof of Theorem* 4.1**

The following lemma, which is proved in Narendra and Annaswamy (1989), is used in the proof of this theorem.

Lemma A.1 (Barbalat). *If $\dot{\boldsymbol{f}}(t)$ is bounded and $\left(\lim_{t\to\infty}\int_{t_0}^{t}|\boldsymbol{f}(\tau)|^2 d\tau\right)^{\frac{1}{2}}$ exists and is finite, then $\lim_{t\to\infty}\boldsymbol{f}(t)=\boldsymbol{0}$.*

Proof. To show that the equilibrium point $\boldsymbol{e}=\boldsymbol{0}$, $\boldsymbol{\Phi}_i=\boldsymbol{O}$, $\boldsymbol{\Psi}=\boldsymbol{O}$ in Equation (4.10) is globally stable, choose the Lyapunov function

$$W(\boldsymbol{e},\boldsymbol{\Phi}_i,\boldsymbol{\Psi})=\boldsymbol{e}^\dagger\,\boldsymbol{\mathcal{R}}_p\,\boldsymbol{e}+\mathrm{Tr}\left(\sum_{i=1}^{\mathrm{k}}\boldsymbol{\Phi}_i^\dagger\,\boldsymbol{\Phi}_i+\boldsymbol{\Psi}^\dagger\,\boldsymbol{\Psi}\right),$$

where $\mathrm{Tr}(\cdot)$ is the trace of the matrix in the argument. Since $\boldsymbol{\mathcal{R}}_p$ is positive definite, $W(\cdot)$ is positive definite because the trace of a sum of matrix inner products is a quadratic form. Since both terms of $W(\cdot)$ are quadratic forms, $W(\cdot)$ is radially unbounded. The time derivative of $W(\cdot)$ is

$$\dot{W}=\dot{\boldsymbol{e}}^\dagger\,\boldsymbol{\mathcal{R}}_p\,\boldsymbol{e}+\boldsymbol{e}^\dagger\,\boldsymbol{\mathcal{R}}_p\,\dot{\boldsymbol{e}}+\mathrm{Tr}\left(\sum_{i=1}^{\mathrm{k}}\dot{\boldsymbol{\Phi}}_i^\dagger\,\boldsymbol{\Phi}_i+\sum_{i=1}^{\mathrm{k}}\boldsymbol{\Phi}_i^\dagger\,\dot{\boldsymbol{\Phi}}_i+\dot{\boldsymbol{\Psi}}^\dagger\,\boldsymbol{\Psi}+\boldsymbol{\Psi}^\dagger\,\dot{\boldsymbol{\Psi}}\right).$$

Using the fact that $\mathrm{Tr}(\boldsymbol{A}+\boldsymbol{B})=\mathrm{Tr}(\boldsymbol{A})+\mathrm{Tr}(\boldsymbol{B})$, $\mathrm{Tr}(\boldsymbol{A}^\dagger\boldsymbol{B})=\mathrm{Tr}(\boldsymbol{B}^\dagger\boldsymbol{A})$, and for $\boldsymbol{A}$ symmetric $\boldsymbol{d}^\dagger\boldsymbol{A}\,\boldsymbol{c}=\boldsymbol{c}^\dagger\boldsymbol{A}\,\boldsymbol{d}$, the time derivative can be rewritten as

$$\dot{W}=\boldsymbol{e}^\dagger\left(\boldsymbol{\mathcal{R}}_p\,\boldsymbol{\mathcal{R}}_s+\boldsymbol{\mathcal{R}}_s^\dagger\,\boldsymbol{\mathcal{R}}_p\right)\boldsymbol{e}+2\sum_{i=1}^{\mathrm{k}}\boldsymbol{e}^\dagger\,\boldsymbol{\mathcal{R}}_p\,\boldsymbol{\Phi}_i\,\boldsymbol{f}_i(\boldsymbol{x})+2\boldsymbol{e}^\dagger\,\boldsymbol{\mathcal{R}}_p\,\boldsymbol{\Psi}\,\boldsymbol{g}(\boldsymbol{u}(t))+\mathrm{Tr}\left(2\sum_{i=1}^{\mathrm{k}}\dot{\boldsymbol{\Phi}}_i^\dagger\,\boldsymbol{\Phi}_i+2\,\dot{\boldsymbol{\Psi}}^\dagger\,\boldsymbol{\Psi}\right). \tag{A.1}$$

Substituting in the adaptive laws gives

$$\begin{aligned}\dot{W}=&\,\boldsymbol{e}^\dagger\left(\boldsymbol{\mathcal{R}}_p\,\boldsymbol{\mathcal{R}}_s+\boldsymbol{\mathcal{R}}_s^\dagger\,\boldsymbol{\mathcal{R}}_p\right)\boldsymbol{e}+2\sum_{i=1}^{\mathrm{k}}\boldsymbol{e}^\dagger\,\boldsymbol{\mathcal{R}}_p\,\boldsymbol{\Phi}_i\,\boldsymbol{f}_i(\boldsymbol{x})+2\boldsymbol{e}^\dagger\,\boldsymbol{\mathcal{R}}_p\,\boldsymbol{\Psi}\,\boldsymbol{g}(\boldsymbol{u}(t))\\&+\mathrm{Tr}\left(-2\sum_{i=1}^{\mathrm{k}}\boldsymbol{f}_i(\boldsymbol{x})\,\boldsymbol{e}^\dagger\,\boldsymbol{\mathcal{R}}_p\,\boldsymbol{\Phi}_i-2\,\boldsymbol{g}(\boldsymbol{u}(t))\,\boldsymbol{e}^\dagger\,\boldsymbol{\mathcal{R}}_p\,\boldsymbol{\Psi}\right).\end{aligned}$$

Using the fact that $\mathrm{Tr}(\boldsymbol{d}\,\boldsymbol{c}^\dagger)=\boldsymbol{c}^\dagger\boldsymbol{d}$, the time derivative can be reduced to

$$\dot{W}=\boldsymbol{e}^\dagger\left(\boldsymbol{\mathcal{R}}_p\,\boldsymbol{\mathcal{R}}_s+\boldsymbol{\mathcal{R}}_s^\dagger\,\boldsymbol{\mathcal{R}}_p\right)\boldsymbol{e}.$$

Since $\boldsymbol{\mathcal{R}}_s$ has strictly negative eigenvalues, the solution to the equation $\boldsymbol{\mathcal{R}}_p\,\boldsymbol{\mathcal{R}}_s+\boldsymbol{\mathcal{R}}_s^\dagger\,\boldsymbol{\mathcal{R}}_p=-\boldsymbol{Q}_0$ for any symmetric positive definite matrix $\boldsymbol{Q}_0$ is a symmetric positive definite matrix $\boldsymbol{\mathcal{R}}_p$.

$$\therefore\ \dot{W}=-\boldsymbol{e}^\dagger\boldsymbol{Q}_0\,\boldsymbol{e}\leq 0$$

Therefore the equilibrium state of the error functions is globally uniformly stable. This implies that $\boldsymbol{e}(t)$, $\boldsymbol{\Phi}_i(t)$, and $\boldsymbol{\Psi}(t)$ are bounded for all $t \geq t_0$. Since $\boldsymbol{f}_i(\boldsymbol{x})$ and $\boldsymbol{g}(\boldsymbol{u}(t))$ are bounded for bounded inputs, $\dot{\boldsymbol{e}}$ as defined in Equation (4.10), is bounded. Since $\boldsymbol{Q}_0$ is positive definite and $\dot{W} \leq 0$,

$$0 \leq \int_{t_0}^{\infty} \boldsymbol{e}(\tau)^\dagger \boldsymbol{Q}_0 \, \boldsymbol{e}(\tau) \, d\tau < \infty.$$

Therefore by Barbalat's Lemma $\lim_{t \to \infty} \boldsymbol{e}(t) = \boldsymbol{0}$.

The following proof segments outline the technique used to prove that this result continues to be true if the matrix $\boldsymbol{\Psi}$ is restricted to being symmetric or skew-symmetric. It is apparent that these arguments follow if *any* of the parameter matrices are restricted in this manner.

Case 1) $\boldsymbol{\Psi}$ is symmetric (i.e. $\boldsymbol{\Psi}^\dagger = \boldsymbol{\Psi}$)
This can be insured by writing the adaptive law as

$$\dot{\boldsymbol{\Psi}} = -\frac{1}{2}\left(\boldsymbol{\mathcal{R}}_p \, \boldsymbol{e} \, \boldsymbol{g}(\boldsymbol{u}(t))^\dagger + \left(\boldsymbol{\mathcal{R}}_p \, \boldsymbol{e} \, \boldsymbol{g}(\boldsymbol{u}(t))^\dagger\right)^\dagger\right).$$

In this case Equation (A.1) for $\dot{W}$ becomes

$$\begin{aligned}\dot{W} &= \cdots + 2\boldsymbol{e}^\dagger \, \boldsymbol{\mathcal{R}}_p \, \boldsymbol{\Psi} \, \boldsymbol{g}(\boldsymbol{u}(t)) + \mathrm{Tr}\left(2 \left[-\frac{1}{2}\left\{\boldsymbol{\mathcal{R}}_p \, \boldsymbol{e} \, \boldsymbol{g}(\boldsymbol{u}(t))^\dagger + \left(\boldsymbol{\mathcal{R}}_p \, \boldsymbol{e} \, \boldsymbol{g}(\boldsymbol{u}(t))^\dagger\right)^\dagger\right\}\right]^\dagger \boldsymbol{\Psi}\right) + \ldots, \\ &= \cdots + 2\boldsymbol{e}^\dagger \, \boldsymbol{\mathcal{R}}_p \, \boldsymbol{\Psi} \, \boldsymbol{g}(\boldsymbol{u}(t)) - \boldsymbol{e}^\dagger \, \boldsymbol{\mathcal{R}}_p \, \boldsymbol{\Psi}^\dagger \, \boldsymbol{g}(\boldsymbol{u}(t)) - \boldsymbol{e}^\dagger \, \boldsymbol{\mathcal{R}}_p \, \boldsymbol{\Psi} \, \boldsymbol{g}(\boldsymbol{u}(t)) + \ldots \overset{\boldsymbol{\Psi}^\dagger = \boldsymbol{\Psi}}{=} 0.\end{aligned}$$

Case 2) $\boldsymbol{\Psi}$ is skew-symmetric (i.e. $\boldsymbol{\Psi}^\dagger = -\boldsymbol{\Psi}$)
This can be insured by writing the adaptive law as

$$\dot{\boldsymbol{\Psi}} = -\frac{1}{2}\left(\boldsymbol{\mathcal{R}}_p \, \boldsymbol{e} \, \boldsymbol{g}(\boldsymbol{u}(t))^\dagger - \left(\boldsymbol{\mathcal{R}}_p \, \boldsymbol{e} \, \boldsymbol{g}(\boldsymbol{u}(t))^\dagger\right)^\dagger\right).$$

In this case Equation (A.1) for $\dot{W}$ becomes

$$\begin{aligned}\dot{W} &= \cdots + 2\boldsymbol{e}^\dagger \, \boldsymbol{\mathcal{R}}_p \, \boldsymbol{\Psi} \, \boldsymbol{g}(\boldsymbol{u}(t)) + \mathrm{Tr}\left(2 \left[-\frac{1}{2}\left\{\boldsymbol{\mathcal{R}}_p \, \boldsymbol{e} \, \boldsymbol{g}(\boldsymbol{u}(t))^\dagger - \left(\boldsymbol{\mathcal{R}}_p \, \boldsymbol{e} \, \boldsymbol{g}(\boldsymbol{u}(t))^\dagger\right)^\dagger\right\}\right]^\dagger \boldsymbol{\Psi}\right) + \ldots, \\ &= \cdots + 2\boldsymbol{e}^\dagger \, \boldsymbol{\mathcal{R}}_p \, \boldsymbol{\Psi} \, \boldsymbol{g}(\boldsymbol{u}(t)) + \boldsymbol{e}^\dagger \, \boldsymbol{\mathcal{R}}_p \, \boldsymbol{\Psi}^\dagger \, \boldsymbol{g}(\boldsymbol{u}(t)) - \boldsymbol{e}^\dagger \, \boldsymbol{\mathcal{R}}_p \, \boldsymbol{\Psi} \, \boldsymbol{g}(\boldsymbol{u}(t)) + \ldots \overset{\boldsymbol{\Psi}^\dagger = -\boldsymbol{\Psi}}{=} 0.\end{aligned}$$

The following proof segment outlines the technique used to prove that this result continues to be true if any elements of the matrix $\boldsymbol{\Psi}$ are set to 0. Again, these arguments follow if *any* of the parameter matrices are restricted in this manner.

Case 3) Any elements of $\boldsymbol{\Psi}$ are set to 0
This can be achieved by writing $\boldsymbol{\Psi}$ as

$$\boldsymbol{\Psi} = \sum_{i=1}^{\mathfrak{g}} \boldsymbol{R}_i \, \boldsymbol{\Psi}_f \, \boldsymbol{C}_i,$$

where $\mathfrak{g} < \mathfrak{n}^2$ and $\boldsymbol{\Psi}_f$ is the matrix containing all $\mathfrak{n}^2$ possible parameters. The leading matrix $\boldsymbol{R}_i$ has a single 1 in the diagonal position corresponding to the *row* of the desired element of $\boldsymbol{\Psi}_f$. All other elements of $\boldsymbol{R}_i$ are 0. The trailing matrix $\boldsymbol{C}_i$ has a 1 in the diagonal position corresponding to the *column* of the desired element of $\boldsymbol{\Psi}_f$, with all other elements being 0. The appropriate adaptive law in this case is

$$\dot{\boldsymbol{\Psi}} = -\sum_{i=1}^{\mathfrak{g}} \boldsymbol{R}_i \left(\boldsymbol{\mathcal{R}}_p \, \boldsymbol{e} \, \boldsymbol{g}(\boldsymbol{u}(t))^\dagger\right) \boldsymbol{C}_i.$$

In this case Equation (A.1) for $\dot{W}$ becomes

$$\dot{W} = \cdots + 2\,\boldsymbol{e}\,\boldsymbol{\mathcal{R}}_p\,\left(\sum_{i=1}^{\mathfrak{g}} \boldsymbol{R}_i\,\boldsymbol{\Psi}_f\,\boldsymbol{C}_i\right)\,\boldsymbol{g}(\boldsymbol{u}(t)) + \mathrm{Tr}\left(-2\left[\sum_{i=1}^{\mathfrak{g}} \boldsymbol{R}_i\,\left(\boldsymbol{\mathcal{R}}_p\,\boldsymbol{e}\,\boldsymbol{g}(\boldsymbol{u}(t))^\dagger\right)\,\boldsymbol{C}_i\right]^\dagger\left[\sum_{i=1}^{\mathfrak{g}} \boldsymbol{R}_i\,\boldsymbol{\Psi}_f\,\boldsymbol{C}_i\right]\right) + \ldots.$$

From the distributive property of matrix multiplication $\boldsymbol{a}^\dagger \sum_{i=1}^{\mathfrak{g}} \boldsymbol{B}_i\,\boldsymbol{c} = \sum_{i=1}^{\mathfrak{g}} \boldsymbol{a}^\dagger\,\boldsymbol{B}_i\,\boldsymbol{c}$ and $\left(\sum_{i=1}^{\mathfrak{g}} \boldsymbol{A}_i\right)^\dagger \left(\sum_{i=1}^{\mathfrak{g}} \boldsymbol{B}_i\right) = \sum_{i=1}^{\mathfrak{g}}\sum_{j=1}^{\mathfrak{g}} \boldsymbol{A}_i^\dagger\,\boldsymbol{B}_j$. So the expression for $\dot{W}$ becomes

$$\dot{W} = \cdots + 2\sum_{i=1}^{\mathfrak{g}} \boldsymbol{e}\,\boldsymbol{\mathcal{R}}_p\,\left(\boldsymbol{R}_i\,\boldsymbol{\Psi}_f\,\boldsymbol{C}_i\right)\,\boldsymbol{g}(\boldsymbol{u}(t)) + \mathrm{Tr}\left(-2\sum_{i=1}^{\mathfrak{g}}\sum_{j=1}^{\mathfrak{g}}\left[\boldsymbol{R}_i\,\left(\boldsymbol{\mathcal{R}}_p\,\boldsymbol{e}\,\boldsymbol{g}(\boldsymbol{u}(t))^\dagger\right)\,\boldsymbol{C}_i\right]^\dagger\left[\boldsymbol{R}_j\,\boldsymbol{\Psi}_f\,\boldsymbol{C}_j\right]\right) + \ldots. \tag{A.2}$$

If these two terms can be shown to cancel for a single value of i, then they will cancel for *any* sum over different i values. For a single i value the matrix $\boldsymbol{R}_i\,\boldsymbol{\Psi}_f\,\boldsymbol{C}_i$ contains *only one* non-zero value, located in the uth row and the wth column. The result of the product $\boldsymbol{\mathcal{R}}_p\,(\boldsymbol{R}_i\,\boldsymbol{\Psi}_f\,\boldsymbol{C}_i)$ is to select the uth column of $\boldsymbol{\mathcal{R}}_p$. The result of the product $\boldsymbol{\mathcal{R}}_p\,(\boldsymbol{R}_i\,\boldsymbol{\Psi}_f\,\boldsymbol{C}_i)\,\boldsymbol{g}(\boldsymbol{u}(t))$ is to select the wth row of $\boldsymbol{g}(\boldsymbol{u}(t))$. Hence the first term is

$$2\,\boldsymbol{e}\,\boldsymbol{\mathcal{R}}_p\,\left(\boldsymbol{R}_i\,\boldsymbol{\Psi}_f\,\boldsymbol{C}_i\right)\,\boldsymbol{g}(\boldsymbol{u}(t)) = 2\sum_{k=1}^{\mathfrak{n}} e_k\,\mathfrak{r}_{ku}\,\psi_{uw}\,g_w. \tag{A.3}$$

Similarly, for a single i value, the matrix $\boldsymbol{R}_i\,\left(\boldsymbol{\mathcal{R}}_p\,\boldsymbol{e}\,\boldsymbol{g}(\boldsymbol{u}(t))^\dagger\right)\,\boldsymbol{C}_i$ contains only one non-zero value located in the uth row and the wth column. Consideration of the form of $\boldsymbol{\mathcal{R}}_p\,\boldsymbol{e}\,\boldsymbol{g}(\boldsymbol{u}(t))^\dagger$ leads to the conclusion that the single entry has the form

$$\boldsymbol{R}_i\,\left(\boldsymbol{\mathcal{R}}_p\,\boldsymbol{e}\,\boldsymbol{g}(\boldsymbol{u}(t))^\dagger\right)\,\boldsymbol{C}_i = \sum_{k=1}^{\mathfrak{n}} e_k\,\mathfrak{r}_{uk}\,g_w.$$

The transpose of this matrix contains the above entry in the wth row and uth column. Likewise the matrix $\boldsymbol{R}_i\,\boldsymbol{\Psi}_f\,\boldsymbol{C}_i$ contains a single non-zero value for each value of j. When $j = i$ this value occurs in the uth row and the wth column and the result of the product $\left[\boldsymbol{R}_i\,\left(\boldsymbol{\mathcal{R}}_p\,\boldsymbol{e}\,\boldsymbol{g}(\boldsymbol{u}(t))^\dagger\right)\,\boldsymbol{C}_i\right]^\dagger\left[\boldsymbol{R}_i\,\boldsymbol{\Psi}_f\,\boldsymbol{C}_i\right]$ is a matrix with a single non-zero entry in the uth position along the diagonal. When $j \neq i$ the single entry in $\boldsymbol{R}_i\,\boldsymbol{\Psi}_f\,\boldsymbol{C}_i$ occurs somewhere else and the result of the product $\left[\boldsymbol{R}_i\,\left(\boldsymbol{\mathcal{R}}_p\,\boldsymbol{e}\,\boldsymbol{g}(\boldsymbol{u}(t))^\dagger\right)\,\boldsymbol{C}_i\right]^\dagger\left[\boldsymbol{R}_i\,\boldsymbol{\Psi}_f\,\boldsymbol{C}_i\right]$ is the zero matrix. So the second term is

$$\mathrm{Tr}\left(-2\sum_{i=1}^{\mathfrak{g}}\sum_{j=1}^{\mathfrak{g}}\left[\boldsymbol{R}_i\,\left(\boldsymbol{\mathcal{R}}_p\,\boldsymbol{e}\,\boldsymbol{g}(\boldsymbol{u}(t))^\dagger\right)\,\boldsymbol{C}_i\right]^\dagger\left[\boldsymbol{R}_j\,\boldsymbol{\Psi}_f\,\boldsymbol{C}_j\right]\right) = -2\sum_{k=1}^{\mathfrak{n}} e_k\,\mathfrak{r}_{uk}\,\psi_{uw}\,g_w \overset{\boldsymbol{\mathcal{R}}_p^\dagger=\boldsymbol{\mathcal{R}}_p}{=} -2\sum_{k=1}^{\mathfrak{n}} e_k\,\mathfrak{r}_{ku}\,\psi_{uw}\,g_w. \tag{A.4}$$

Taking the sum of Equations (A.3) and (A.4) leads to the conclusion that the two terms in Equation (A.2) cancel for a single i value. Therefore they will also cancel for a sum over any set of i values. ♦

***Proof of Theorem* 4.2**

The following lemma, which is proved in the reference, is used in the proof of this theorem.

Lemma A.2 (LaSalle and Lefschetz (1961)). *Let $V(\boldsymbol{x})$ be a scalar function which for all $\boldsymbol{x}$ has continuous first partial derivatives with the property that $\lim_{\|\boldsymbol{x}\|\to\infty} V(\boldsymbol{x}) \to \infty$. If $\dot{V}(\boldsymbol{x}) \leq -\epsilon < 0$ for all $\boldsymbol{x}$ outside some closed and bounded set $\mathcal{M}$, then the solutions of $\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, t)$ are ultimately bounded.*

Proof. Since $\boldsymbol{h}(\cdot)$ is continuous, $\|\boldsymbol{u}(t)\| \leq \mathcal{U} \Rightarrow \|\boldsymbol{h}(\boldsymbol{u}(t))\| \leq \tilde{\mathcal{U}}$. It is given that $\|\nabla_{\boldsymbol{x}} V(\boldsymbol{x})\| \geq \mathcal{K}_u$. Choose $\mathcal{K}_u$ to be

$$\|\nabla V\| \geq \mathcal{K}_u = \frac{\tilde{\mathcal{U}} + \sqrt{\tilde{\mathcal{U}}^2 + 4\,\lambda_{min}\,\epsilon}}{2\,\lambda_{min}},$$

where ϵ is a positive constant and λ_{min} is the smallest eigenvalue of $\boldsymbol{P}(\boldsymbol{x})$ in the region where $\|\boldsymbol{x}\| > \mathcal{F}_u$. Since $\boldsymbol{P}$ is symmetric positive definite, the smallest eigenvalue λ_{min} is real and positive.

$$\therefore \lambda_{min}\,\|\nabla V\|^2 - \|\nabla V\|\,\tilde{\mathcal{U}} - \epsilon \geq 0$$

It is given that $\|\boldsymbol{h}\| \leq \tilde{\mathcal{U}} \Rightarrow -\|\boldsymbol{h}\| \geq -\tilde{\mathcal{U}}$

$$\therefore \lambda_{min} \|\nabla V\|^2 - \|\nabla V\| \|\boldsymbol{h}\| - \epsilon \geq \lambda_{min} \|\nabla V\|^2 - \|\nabla V\| \tilde{\mathcal{U}} - \epsilon \geq 0$$

By the Cauchy-Schwarz inequality $\|\nabla V\| \|\boldsymbol{h}\| \geq |\nabla V^\dagger \boldsymbol{h}|$

$$\therefore \lambda_{min} \|\nabla V\|^2 - |\nabla V^\dagger \boldsymbol{h}| - \epsilon \geq \lambda_{min} \|\nabla V\|^2 - \|\nabla V\| \|\boldsymbol{h}\| - \epsilon,$$

where $|\cdot|$ is the absolute value. For the absolute value $|\nabla V^\dagger \boldsymbol{h}| \geq \nabla V^\dagger \boldsymbol{h}$

$$\therefore \lambda_{min} \|\nabla V\|^2 - \nabla V^\dagger \boldsymbol{h} - \epsilon \geq \lambda_{min} \|\nabla V\|^2 - |\nabla V^\dagger \boldsymbol{h}| - \epsilon$$

Since λ_{min} is the smallest eigenvalue of the matrix $\boldsymbol{P}(\boldsymbol{x})$

$$\therefore \nabla V^\dagger \boldsymbol{P} \nabla V - \nabla V^\dagger \boldsymbol{h} - \epsilon \geq \lambda_{min} \nabla V^\dagger \nabla V - \nabla V^\dagger \boldsymbol{h} - \epsilon \geq 0$$
$$\Rightarrow -\nabla V^\dagger \boldsymbol{P} \nabla V + \nabla V^\dagger \boldsymbol{h} \leq -\epsilon$$

The quantity on the left side of the inequality is precisely $\dot{V}(\boldsymbol{x})$ for Equation (4.8). Therefore $\|\nabla_{\boldsymbol{x}} V(\boldsymbol{x})\| \geq \mathcal{K}_u \Rightarrow \dot{V}(\boldsymbol{x}) \leq -\epsilon$ for all $\boldsymbol{x}$ such that $\|\boldsymbol{x}\| > \mathcal{F}_u$.

It is well known from real analysis that $\int \|\nabla V\| \, d\boldsymbol{x} \geq \| \int \nabla V \, d\boldsymbol{x} \|$.

$$\therefore \int \|\nabla V\| \, d\boldsymbol{x} \geq \left\| \int \nabla V \, d\boldsymbol{x} \right\| > \left\| \int \mathcal{K}_u \, d\boldsymbol{x} \right\|$$
$$\Rightarrow \|V + c\| > \|\mathcal{K}_u \boldsymbol{x}\|$$

By Minkowski's inequality $\|V\| + \|c\| \geq \|V + c\|$,

$$\therefore \|V\| + \|c\| \geq \|V + c\| > \mathcal{K}_u \|\boldsymbol{x}\|$$

Since $V \geq 0$ and since V and c are scalars

$$\therefore V > \mathcal{K}_u \|\boldsymbol{x}\| - |c|$$

Hence $\|\nabla_{\boldsymbol{x}} V(\boldsymbol{x})\| \geq \mathcal{K}_u \Rightarrow V(\boldsymbol{x}) > \mathcal{K}_u \|\boldsymbol{x}\| \Rightarrow \lim_{\|\boldsymbol{x}\| \to \infty} V(\boldsymbol{x}) \to \infty$. Note that the converse of this implication is *not* true. Using these two results, it follows immediately from Lemma A.2 that the solutions $\boldsymbol{x}(t)$ of Equation (4.8) are ultimately bounded. ♦

***Proof of Theorem* 4.3**

Proof. Since Equation (4.9) satisfies all of the conditions of Theorem 4.2, its solutions $\|\boldsymbol{x}(t)\|$ are bounded. Let $\boldsymbol{f}_i(\boldsymbol{x}) = \mathcal{F}_i(\tilde{\boldsymbol{x}}) \nabla_{\boldsymbol{x}} V(\boldsymbol{x})$ and note that $\|\dot{\boldsymbol{f}}_i(\boldsymbol{x}(t))\| = \|\frac{\partial \boldsymbol{f}_i}{\partial \boldsymbol{x}} \dot{\boldsymbol{x}}\| \leq \|\frac{\partial \boldsymbol{f}_i}{\partial \boldsymbol{x}}\| \|\dot{\boldsymbol{x}}\|$. Since $\boldsymbol{f}_i$ and $\boldsymbol{g}$ are continuous, and $\|\boldsymbol{x}(t)\|$ and $\|\boldsymbol{u}(t)\|$ are bounded, $\|\boldsymbol{f}_i(\boldsymbol{x}(t))\|$ and $\|\boldsymbol{g}(\boldsymbol{u}(t))\|$ are bounded. Therefore $\|\dot{\boldsymbol{x}}\| = \|\sum_{i=1}^{\mathrm{k}} \boldsymbol{A}_i \boldsymbol{f}_i(\boldsymbol{x}) + \boldsymbol{B}\boldsymbol{g}(\boldsymbol{u}(t))\| \leq \sum_{i=1}^{\mathrm{k}} \|\boldsymbol{A}_i\| \|\boldsymbol{f}_i(\boldsymbol{x})\| + \|\boldsymbol{B}\| \|\boldsymbol{g}(\boldsymbol{u}(t))\|$ is bounded. Since $\boldsymbol{f}_i$ is continuously differentiable, $\frac{\partial \boldsymbol{f}_i}{\partial \boldsymbol{x}}$ is continuous. Since $\|\boldsymbol{x}(t)\|$ is bounded, $\|\frac{\partial \boldsymbol{f}_i}{\partial \boldsymbol{x}}\|$ is bounded. Therefore $\|\dot{\boldsymbol{f}}_i(\boldsymbol{x}(t))\|$ is bounded, since it is less than or equal to the product of bounded functions. Similarly $\|\dot{\boldsymbol{g}}(\boldsymbol{u}(t))\| = \|\frac{\partial \boldsymbol{g}}{\partial \boldsymbol{u}} \dot{\boldsymbol{u}}\| \leq \|\frac{\partial \boldsymbol{g}}{\partial \boldsymbol{u}}\| \|\dot{\boldsymbol{u}}\|$. Since $\boldsymbol{g}$ is continuously differentiable, $\frac{\partial \boldsymbol{g}}{\partial \boldsymbol{u}}$ is continuous. Since $\|\boldsymbol{u}(t)\|$ is bounded, $\|\frac{\partial \boldsymbol{g}}{\partial \boldsymbol{u}}\|$ is bounded. Since it is given that $\|\dot{\boldsymbol{u}}(t)\|$ is bounded, $\|\dot{\boldsymbol{g}}(\boldsymbol{u}(t))\|$ is bounded.

It is given that there exist positive constants t_0, $\mathcal{T}$, and ϵ such that for every unit vector $\boldsymbol{w} \in \mathbb{R}^{\mathrm{n}+\mathrm{m}}$

$$\frac{1}{\mathcal{T}} \int_t^{t+\mathcal{T}} \left\| \left\{ [\mathcal{F}_1(\tilde{\boldsymbol{x}}(\tau)) + \mathcal{F}_2(\tilde{\boldsymbol{x}}(\tau)) + \cdots + \mathcal{F}_{\mathrm{k}}(\tilde{\boldsymbol{x}}(\tau))] \, \nabla_{\boldsymbol{x}} V(\boldsymbol{x}(\tau))^\dagger \, \boldsymbol{g}(\boldsymbol{u}(\tau))^\dagger \right\} \boldsymbol{w} \right\| d\tau \geq \epsilon \quad \forall \quad t \geq t_0,$$
$$\Rightarrow \frac{1}{\mathcal{T}} \int_t^{t+\mathcal{T}} \left\| \left\{ \boldsymbol{f}_1(\boldsymbol{x}(\tau))^\dagger \, \boldsymbol{f}_2(\boldsymbol{x}(\tau))^\dagger \ldots \boldsymbol{f}_{\mathrm{k}}(\boldsymbol{x}(\tau))^\dagger \, \boldsymbol{g}(\boldsymbol{u}(\tau))^\dagger \right\} \boldsymbol{w}_{\mathrm{k}} \right\| d\tau \geq \epsilon \quad \forall \quad t \geq t_0.$$

where $\boldsymbol{w}_{\mathrm{k}} \in \mathbb{R}^{\mathrm{k}\mathrm{n}+\mathrm{m}}$. With this inequality and the fact that $\|\dot{\boldsymbol{f}}_i(\boldsymbol{x})\|$ and $\|\dot{\boldsymbol{g}}(\boldsymbol{u})\|$ are bounded, it follows immediately from Theorem 4 in Morgan and Narendra (1977) that Equation (4.10) is globally uniformly asymptotically stable. ♦