TOWARDS REAL-TIME CONTINUOUS SYSTEM IDENTIFICATION USING MODIFIED HOPFIELD NEURAL NETWORKS

Daniel J. Stech
Aerospace Sciences Division
Frank J. Seiler Laboratory,
Air Force Materiel Command,
U.S. Air Force Academy, CO 80840

ABSTRACT
Solution of system identification problems on parallel analog networks is proposed. Recurrent neural networks whose dynamic equations have a Lyapunov function, relax to an equilibrium which is the minimum of the Lyapunov function. System identification problems are formulated in terms of a Lyapunov function and thus are solved using the recurrent networks. Convergence for linear and a set of nonlinear problems is assured. Identification of a simulated nonlinear system and of an experimental lightly damped structure demonstrate the practicality of the approach. Results show extremely fast solution times which are independent of the size of the problem.

NOMENCLATURE

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A$</td>
<td>system matrix</td>
</tr>
<tr>
<td>$B$</td>
<td>input matrix</td>
</tr>
<tr>
<td>$C^{-1}$</td>
<td>capacitance matrix</td>
</tr>
<tr>
<td>$E$</td>
<td>matrix of error functional Walsh coefficients</td>
</tr>
<tr>
<td>$e(t)$</td>
<td>error functional</td>
</tr>
<tr>
<td>$H$</td>
<td>Walsh matrix</td>
</tr>
<tr>
<td>$L(x)$</td>
<td>Lyapunov function</td>
</tr>
<tr>
<td>$N$</td>
<td>number of Walsh functions</td>
</tr>
<tr>
<td>$P$</td>
<td>Walsh integration matrix</td>
</tr>
<tr>
<td>$U$</td>
<td>matrix of control Walsh coefficients</td>
</tr>
<tr>
<td>$u(t)$</td>
<td>vector of controls</td>
</tr>
<tr>
<td>$W$</td>
<td>weight matrix</td>
</tr>
<tr>
<td>$X$</td>
<td>matrix of state Walsh coefficients</td>
</tr>
<tr>
<td>$x(t)$</td>
<td>vector of states</td>
</tr>
<tr>
<td>$\phi_n(t)$</td>
<td>$n^{th}$ Walsh function</td>
</tr>
</tbody>
</table>

INTRODUCTION
Solution of large scale system identification problems can be computationally intensive. Even relatively small scale system identification problems can result in long computation time compared to the time constant of the dynamic system of interest. The ability to do large scale system identification in real time would have numerous applications, for example, real time adaptation in vibration control of large flexible structures, or rapid reconfiguration of flight control systems in damaged aerospace vehicles.

Neural networks have been used for system identification. In the most common application, the neural network is “taught” to behave like the dynamical system of interest. These networks are typically of the recurrent or feedforward type and are trained with gradient based learning algorithms. They have the advantage of being able to learn nonlinear systems; however, the learning algorithm is difficult to implement in real time and tends to converge to a solution very slowly. Also, the resulting model tends to be a “black box” which does not give much insight into the physical system being modeled. [1, 2, 3]

A different approach to system identification using neural networks is to use the network to find some physical parameters of interest in a dynamical system. For example, some researchers have used Hopfield networks to identify the coefficients of a linear set of differential equations which model a dynamical system. [4] [5]. In this case, the system identification problem is posed as a least squares estimation problem which is solved with
a Hopfield network. The intent of this form of system identification is not to generate a "black box" model of the system, but rather to use the neural network as a computational tool to speed up the system identification to real-time using a network form which is easy to implement in hardware.

In this paper, an approach similar to Shoureshi [4] and Cetinkunt [5] is taken to solve the system identification problem. However, it is shown that for linear system identification problems a nonlinear Hopfield network is not necessary, a linear network will suffice. Also, unlike previous approaches, the method is directly extendable to a class of nonlinear system identification problems. The method of Doan and Rick [6] is extended to formulate the system identification as the minimization of a Lyapunov function. A network is then designed which has dynamics for which a Lyapunov function can be found. In this case, the network is guaranteed to relax to a stable equilibrium point. The state of the network at the equilibrium point minimizes the Lyapunov function. The form of the Lyapunov function suggests the architecture for the minimizing network. Also, it is shown that solution time is decoupled from the size of the problem to be solved, suggesting that large system identification problems can be solved in real-time. Finally, the practicality of the method is demonstrated by system identification of the modal frequencies of an experimental laboratory structure.

MINIMIZATION USING PARALLEL NETWORKS

Hopfield and Tank showed how highly interconnected networks of simple analog processors can collectively compute good solutions to difficult optimization problems [7]. Hopfield networks are classified as recurrent networks. Recurrent networks are dynamical in nature and as such can be modeled by ordinary differential equations. Unlike recurrent networks which are "trained" to match some input/output mapping, Hopfield networks are concerned with the fixed point solution of the network dynamic equations.

Hopfield gave a Lyapunov function for his network and called it the "energy function". Since the "energy function" is a Lyapunov function for the network dynamic equations, given some initial starting conditions, the states of the network will evolve to some equilibrium condition (fixed point) which is a minimum of the "energy" or Lyapunov function. To demonstrate this, Tank and Hopfield [8] solved a number of optimization problems by posing the problem in terms of an "energy function" and then constructing a network whose dynamic equations were associated with that particular Lyapunov function. Chua [9] gave a more general network formulation for solving nonlinear programming problems. Kennedy and Chua [10] gave the conditions which guaranteed the existence of a Lyapunov function for the more general network.

In general, it is desired to construct a network which has a Lyapunov function that corresponds to the optimization problem to be solved. Consider the generalized network shown in Figure 1. The nonlinear integrating amplifiers are typically referred to as neurons, and $W$ is referred to as the weight matrix. A fully populated weight matrix results in a fully connected network. The dynamic equations which describe the evolution of the states are given by,

\[ \dot{z} = f(W, z) \quad z(t_0) = z_0 \]  

where $z$ is a vector of the states (voltages) of the network. Dynamical behavior of the network voltages is dependent on the initial conditions, the weights, and the nonlinearity present in the integrators. Assume the weights are fixed and let

\[ \dot{z} = f(w, z) = -C^{-1} \nabla z L(z) \]  

where $C^{-1}$ is a positive $n \times n$ diagonal self capacitance matrix resulting from the capacitance of each neuron. If,

1. At least one solution to the problem exists. (i.e. the cost function is bounded from below)
2. \( L \) is continuous and its first and second partial derivatives are continuous.

then Kennedy and Chua showed \( L \) is a Lyapunov function for Equation (2) [10]. Maa and Shanblatt gave rigor to the results of Kennedy and Chua using optimization theory [11].

The key then is to pose the system identification problem as the minimization of a function \( C \). This Lyapunov function will determine the network dynamics which in turn will imply the network architecture. In the next section, the optimal control problem is posed as the minimization of a Lyapunov function.

SYSTEM IDENTIFICATION BY OPTIMIZATION

System identification is a form of dynamic optimization. This section outlines a method based on the work of Frick and Stech [12] and Doan and Frick [6] of formulating the system identification problem as a static optimization. It is then shown that the static optimization can be arranged into a Lyapunov or "energy function" whose minimum is found using a parallel analog network that has dynamics corresponding to the Lyapunov function.

Consider first the set of nonlinear dynamic systems described by,

\[
\dot{z}(t) = af(z(t), u(t)) \quad z(0) = z_0 \quad (3)
\]

where \( z(t) \in \mathbb{R}^n \) is the measured state of the system, \( u(t) \in \mathbb{R}^m \) is the input to the system, and \( \alpha \) is some constant vector of coefficients. The identification problem is to find the unknown coefficients \( \alpha \) which result in a system that best matches the input/output data \( z(t) \), and \( u(t) \).

Consider the integral form of (3) and form the error functional,

\[
e(t) = z(t) - z_0 - a \int_0^t f(z(\tau), u(\tau)) d\tau \quad (4)
\]

Minimizing \( e(t) \) given \( z(t) \) and \( u(t) \) is the dynamic optimization which solves the system identification problem. The dynamic optimization must be converted into a static optimization. This conversion is accomplished by expanding the time varying functions in the error functional into Walsh series consisting of \( N \) Walsh functions where \( N = 2^k, k > 0 \) (for a more detailed discussion of Walsh functions, see Chen [13]),

\[
x(t) \approx X \Psi(t) \quad u(t) \approx U \Psi(t) \quad (5)
\]

where \( X \) and \( U \) are matrices of coefficients and \( \Psi(t) = [\phi_0(t), \phi_1(t), \ldots \phi_{N-1}(t)]^T \) is a vector of the Walsh functions. Given these approximations, the error functional can be approximated,

\[
e(t) \approx E \Psi(t) \quad (6)
\]

where \( E \) is a matrix of Walsh coefficients. Using several properties of the Walsh functions,

\[
E = X - X_0 - a F(X, U) P \quad (7)
\]

where \( F(X, U) \) denotes some nonlinear operation on the given Walsh coefficients. For more details see [14]. Equation (7) can be written in the vec notation (see Brewer [15]) as,

\[
\text{vec}(E) = \text{vec}(X) - \text{vec}(X_0) - ((F(X, U) P)^T \otimes I_n) a \quad (8)
\]

where \( P \) is the Walsh integration matrix (see Chen [13]). To minimize \( \text{vec}(E) \) in the least squares sense, let

\[
J = (\text{vec}(E))^T \text{vec}(E) \quad (9)
\]

Assume \( J \) is a Lyapunov function, (i.e. \( J(\alpha) = L(\alpha) \)), then

\[
\nabla J(\alpha) = Ka - D \quad (10)
\]

where

\[
K = M^T M \quad (11)
\]

and

\[
M = ((F(X, U) P)^T \otimes I_n) \quad (12)
\]

and

\[
D = M^T (\text{vec}(X) - \text{vec}(X_0)) \quad (13)
\]

The subscripts on the identity matrices indicates their dimensions. The system identification problem is now posed as the minimization of a quadratic Lyapunov or "energy function". As shown in the previous section, the network which minimises this Lyapunov function will have the network dynamics,

\[
\dot{a} = -C^{-1} Ka + D \quad (14)
\]

This suggests the form of the network to be used. Figure 2 gives the basic architecture. The voltages \( V_n \) are the states of the network, each will represent a coefficient to be identified. The triangles represent integrators and \( K \) and \( D \) are conductances which are determined from (11) - (13). Given some initial voltages, the network states (voltages) will converge to an equilibrium condition which is a minimum of the Lyapunov function. This will be the solution of the system identification problem. Since \( K = M^T M \), it is at least positive semi definite. Also since (9) is quadratic, its first
Figure 2: Parallel analog neural network for solving system identification problems

and second derivatives are continuous. As a result, the network is guaranteed to converge to a solution. Also, notice that the relaxation time (solution time) of the network is a function of $C^{-1}$, not the size of the network. This effectively decouples the size of the problem from its solution time.

EXAMPLES

Nonlinear System ID

Consider the following simple system identification problem. Given the input/output data for the system modeled by the following scalar differential equation,

$$\dot{x}(t) = -ax^2(t) + u(t), x(0) = 0 \quad (15)$$

identify the parameter $a$. Stech and Frick [14] showed that it is straightforward to perform nonlinear operations on vectors of Walsh coefficients. This simply involves obtaining the Walsh series by multiplication with the Walsh matrix $H$, performing the nonlinear operation elementwise on the Walsh series, and then converting back by multiplication with the inverse Walsh matrix. (see [14] for more details) Using this procedure on the squared term gives

$$\text{vec}(E) = \text{vec}(X) - \text{vec}(X_0)$$

$$+ [((XH)^2H^{-1}P)^T \otimes I_n] a$$

$$- \text{vec}(UP) \quad (16)$$

From this it is evident that

$$M = ((XH)^2H^{-1}P)^T \otimes I_n \quad (17)$$

so

$$K = M^T M \quad (18)$$

and

$$D = M^T(\text{vec}(X) - \text{vec}(X_0) - \text{vec}(UP)) \quad (19)$$

The input/output data was obtained by setting $a = 10$ and simulating the system using a 4th order Runge Kutta solver with a unit step input. The output was then corrupted with 10% noise to simulate sensor noise (see Figure 3). This input/output data was then used to identify the parameter $a$. The capacitance of the network was assumed to be $0.01 \mu F$ and the initial voltages were assumed to be zero. The input and output time series was approximated using 32 Walsh functions. Once the Walsh coefficients for the input/output data are obtained, the $K$ and $D$ for this example can be easily calculated and the parallel network simulated. Figure 4 shows convergence of the network $a$ parameter to 9.983 in less than 1 millisecond. Identification of the parameter is with a few percent of the actual value.

Figure 3: Simulated nonlinear system - measured state

Figure 4: Simulated neural network dynamics
Structural System Identification

The previous example demonstrated the possibility of using neural networks to identify nonlinear systems. In this example, the effectiveness of the method is tested by identifying the modal frequencies of a laboratory test structure. The actual structure is shown in Figure 5.

Figure 5: Experimental structure

The structure was originally proposed by Potter [16]. It has two low frequency modes of interest, a torsional mode and bending mode. The torsional mode can be adjusted by changing the positions of the two masses. For this experiment, the masses were adjusted to give a bending mode at 5 Hz and a torsional mode at 7 Hz. These frequencies were verified by modal analysis using a Textronix 2630 Fourier Analyzer. An accelerometer was placed at each end of the structure and a force gauge at one end of the structure. A modified impulse accelerations measured for approximately .5 seconds. The accelerations were filtered to give position and velocity at each end of the beam. The goal was to use the modified Hopfield network to identify a state space model of the form

\[ \dot{z}(t) = Az(t) + Bu(t) \]  

(20)

where the modal frequencies of the identified \( A \) matrix matched the modal frequencies of the structure. Figure 7 shows the resulting position and velocity at each end of the beam. The input/output data was approximated using 128 Walsh functions and a parallel network was designed. The network was simulated by solving the network dynamical equations. Figure 8 shows convergence of the states of the network in less than 1 millisecond. Table 1 shows the identified frequencies obtained by solving for the eigenvalues of the system identified \( A \) matrix. These values are compared to the values obtained by time averaging an extensive amount of data.

Figure 6: Force input to the structure

(see figure 6) was imparted on the structure and the ac-

Figure 7: Position and velocity of each end of the structure

Figure 8: Time history of the states of the network

(Coefficients of the \( A \) and \( B \) matrices)
with the Fourier analyzer. The results indicate there is potential for modified Hopfield networks to perform real-time, accurate system identification.

**SUMMARY**

System identification problems can be converted into static optimization problems. If the function to be optimized is the Lyapunov function for a recurrent neural network, then the network solves the problem. Nonlinear networks are not required to solve linear or a class of nonlinear identification problems, linear networks are sufficient. These networks can be guaranteed to stably converge to a solution. Simulation of a parallel analog network to solve an experimental system identification problem showed solution times of less than a millisecond. Since the solution times can be shown to be independent of the size of the problem, significant headway has been made toward the real time identification of large systems.

**REFERENCES**


