A Dynamical System Perspective of Structural Learning with Forgetting

Damon A. Miller, *Member, IEEE,* and Jacek M. Zurada, *Fellow, IEEE*

***Abstract*—Structural learning with forgetting is an established method of using Laplace regularization to generate skeletal artificial neural networks. In this paper we develop a continuous dynamical system model of regularization in which the associated regularization parameter is generalized to be a time-varying function. Analytic results are obtained for a Laplace regularizer and a quadratic error surface by solving a different linear system in each region of the weight space. This model also enables a comparison of Laplace and Gaussian regularization. Both of these regularizers have a greater effect in weight space directions which are less important for minimization of a quadratic error function. However, for the Gaussian regularizer, the regularization parameter modifies the associated linear system eigenvalues, in contrast to its function as a control input in the Laplace case. This difference provides additional evidence for the superiority of the Laplace over the Gaussian regularizer.**

***Index Terms*— Dynamical system, Gaussian, Laplace, neural network, pruning, regularization, rule extraction, structural learning.**

I. Introduction

SELECTION of an appropriate multilayer feedforward neural network (MFNN) architecture for a specific application remains an open issue. Overly complex networks typically provide small training errors at the expense of generalization performance. At the other extreme, MFNN's with insufficient complexity levels are not able to learn. This is a consequence of the well known bias-variance dilemma [1]. One common approach to the problem of MFNN architecture selection is to include a complexity penalty term in the training error criterion [2]. These methods begin with an initially oversized network and rely on the penalty term to suppress unnecessary weights during learning. Hence both the network structure and weights are determined at the conclusion of training. Limiting the network complexity is assumed to improve generalization performance. Smaller networks also have the advantage of providing reduced implementation costs and may enable the discovery of relationships in the training data, e.g., rule extraction.

Thus we consider the regularized error function $E_1(\mathbf{w}) = \beta E_d(\mathbf{w}) + \alpha E_w(\mathbf{w})$, where $E_d(\mathbf{w})$ is a measure of the network mapping error over the training set, $E_w(\mathbf{w})$ is a complexity penalty term, α and β are positive regularization parameters which determine the relative importance of each term, and the vector $\mathbf{w}$ contains all bias and nonbias weights [3], [4]. Here $E_d(\mathbf{w})$ is chosen to be the standard sum-of-squares error

$$E_d(\mathbf{w}) = \tfrac{1}{2}\sum_{p=1}^{P} \|\mathbf{d}^{\langle p\rangle} - \mathbf{o}^{\langle p\rangle}(\mathbf{x}^{\langle p\rangle}, \mathbf{w})\|^2 \tag{1}$$

where $\mathbf{d}^{\langle p\rangle}$ is the desired output for the pth MFNN input $\mathbf{x}^{\langle p\rangle}$ while $\mathbf{o}^{\langle p\rangle}$ is the actual MFNN output. If only the minima of $E_1(\mathbf{w})$ are of interest, we may instead use the error function

$$E(\mathbf{w}) = E_d(\mathbf{w}) + \varepsilon E_w(\mathbf{w}) \tag{2}$$

where $\varepsilon = \alpha/\beta$. The familiar gradient descent learning rule

$$\Delta\mathbf{w} = -\eta\nabla_{\mathbf{w}} E_d(\mathbf{w}) - \eta\varepsilon\nabla_{\mathbf{w}} E_w(\mathbf{w}) \tag{3}$$

where $\eta > 0$ is a learning constant, may be used to adapt the weights beginning at an initial condition $\mathbf{w}^{\langle 0\rangle}$ [5]. Selection of the regularization parameter ε is critical. For ε too large, the penalty term is over-emphasized, and the network mapping error will be large. For ε too small, the network complexity is under-emphasized, and an overly complex network architecture will result.

A. Paper Organization

The objective of this paper is to develop a continuous dynamical system model of learning subject to regularization to analyze and compare Laplace and Gaussian regularization and to investigate the utility of an adaptive ε. In this model ε is generalized to be a time-varying function. This paper is arranged as follows. We will first describe the *structural learning with forgetting* algorithm [6] which uses a Laplace regularizer and whose analysis provided the original motivation for this paper. Previous results from a Bayesian perspective of regularization and an error sensitivity analysis are then presented to justify a preference for the Laplace over the Gaussian regularizer [4]. We then develop and apply our dynamical system model to these two regularizers and offer a comparison based on these results. This includes the development of analytic solutions for a quadratic error surface. This paper is an extension of the work first presented in [7]. It is important to note that we are analyzing two already widely used approaches to MFNN regularization.

B. Structural Learning with Forgetting

Structural learning with forgetting (SLF) is an effective penalty term method which simultaneously trains and regularizes an initially oversized artificial neural network. Though this

Manuscript received January 28, 1997; revised January 2, 1998.

D. A. Miller is with the Department of Electrical and Computer Engineering, Western Michigan University, Kalamazoo, MI 49008 USA.

J. M. Zurada is with the Department of Electrical Engineering, The University of Louisville, Louisville, KY 40292 USA.

Publisher Item Identifier S 1045-9227(98)02769-6.

method is also applicable to recurrent networks, we consider only MFNN's. The primary goal of SLF is to obtain a skeletal network conducive to the discovery of input–output relations in the training data. SLF consists of three separate algorithms: learning with forgetting (LF), hidden units clarification, and learning with selective forgetting [6]. The complexity penalty term for LF and learning with selective forgetting is the Laplace regularizer

$$E_w(\mathbf{w}) = \sum_{i \in C} |w_i| \tag{4}$$

where the regularization class C consists of N weights. For LF C contains all of the MFNN weights and the gradient of (4) is considered to be

$$\nabla_{\mathbf{w}} E_w(\mathbf{w}) = \operatorname{sgn}(\mathbf{w}) \tag{5}$$

where $\operatorname{sgn}(w_i)$ is 1 for $w_i > 0$ and -1 otherwise, even though the partial derivative of (4) with respect to w_i is not defined for $w_i = 0$. Substitution into (3) yields the LF learning rule

$$\Delta \mathbf{w} = -\eta \nabla_{\mathbf{w}} E_d(\mathbf{w}) - \eta\varepsilon \quad \operatorname{sgn}(\mathbf{w}) \tag{6}$$

where the regularization parameter ε is considered to be constant. Ishikawa uses the term "forgetting" since the Laplace regularizer reduces weight magnitudes by $\eta\varepsilon$ at each training step; hence, we may refer to ε as the "forgetting rate." In selective learning with forgetting C is restricted to contain small magnitude weights. LF is itself capable of providing substantial MFNN reduction [8]. For details on SLF, see [6].

Although information theoretic measures may be used to select a model after they have undergone regularization with various values of ε [6], there are no guidelines for its *a priori* selection. A heuristic method for adapting ε has been developed in which the time available for regularization is specified and ε is used as a control parameter to provide the desired learning convergence rate [7]. Bayesian methods provide a probabilistic interpretation of regularization and techniques for adapting the associated parameters. We next introduce several key results provided by this approach.

C. A Bayesian Perspective of Regularization

The effects of Laplace and Gaussian regularization have been previously considered from a Bayesian perspective [4]. Here we provide a brief summary of these results for a single output MFNN. Minimization of $E_d(\mathbf{w})$ alone provides the maximum likelihood estimate of the network weights under the assumptions of a uniform weight prior and an identical and independent additive Gaussian noise model with zero mean and variance $1/\beta$ at the MFNN output for each training pattern. Minimization of $E_1(\mathbf{w})$ corresponds to maximization of a posterior weight density where the prior weight density is determined by $\alpha E_w(\mathbf{w})$.

The Laplace regularizer specifies that each weight w_i is subject to an identical and independent Laplace prior which is proportional to $e^{-\alpha|w_i|}$. Note that for this prior the mean value of each $|w_i|$ is $1/\alpha$. The estimates $\hat{\alpha}$ and $\hat{\beta}$ given by

$$\frac{1}{\hat{\alpha}} = \frac{1}{N} \sum_{i \in C} |w_i|$$

and

$$\frac{1}{\hat{\beta}} = \frac{1}{P} \sum_{p=1}^{P} [d_1^{\langle p \rangle} - o_1^{\langle p \rangle}(\mathbf{x}^{\langle p \rangle}, \mathbf{w})]^2 \tag{7}$$

may be used during training. Thus $E_w(\mathbf{w})$ is emphasized less as the mean absolute weight value increases and $E_d(\mathbf{w})$ is emphasized less as the noise estimate increases. Refer to [4] for details.

For comparison, consider the Gaussian regularizer

$$E_w(\mathbf{w}) = \tfrac{1}{2} \sum_{i \in C} w_i^2 \tag{8}$$

which specifies an identical and independent normal prior with zero mean and variance $1/\alpha$ for each weight. Reference [9] describes an adaptation strategy in which the eigenvalues of the Hessian of $E_d(\mathbf{w})$ are used to estimate the effective number of weights. This number may in turn be used to estimate α and β during training.

Reference [4], using the principles of transformation groups and maximum entropy, argues that the Laplace prior is particularly suited and thus is preferable to the Gaussian prior for internal MFNN weights, i.e., those weights on connections which lead to or from hidden neurons. This argument does not apply to weights on direct connections between the MFNN inputs and outputs, where a Gaussian prior is considered to be a reasonable choice. The Bayesian perspective also indicates that bias weights should not be subject to regularization. What follows is only applicable to those N weights considered to be in C. For an extensive discussion of Bayesian techniques for MFNN's, refer to [3], [4], and [9].

D. Sensitivity Calculations

Examining the conditions for a minimum of the regularized error provides additional evidence for a preference for a Laplace over a Gaussian regularizer [4]. For the Gaussian regularizer

$$\nabla_{\mathbf{w}} E_w(\mathbf{w}) = \mathbf{w} \tag{9}$$

and thus at a minimum of $E(\mathbf{w})$ each weight must satisfy

$$\left| \frac{\partial E_d(\mathbf{w})}{\partial w_i} \right| = \varepsilon |w_i|. \tag{10}$$

Sufficient conditions for the stability of a weight for the Laplace regularizer are

$$\left| \frac{\partial E_d(\mathbf{w})}{\partial w_i} \right| = \varepsilon \quad \text{if} \quad |w_i| > 0$$

and

$$\left| \frac{\partial E_d(\mathbf{w})}{\partial w_i} \right| < \varepsilon \quad \text{if} \quad |w_i| = 0 \tag{11}$$

which indicate that at a minimum of $E(\mathbf{w})$ the magnitude of the unregularized error sensitivity to all nonzero weights is ε. Weights for which this sensitivity is less than ε are zeroed. Contrast (11) with (10), where the sensitivity of $E_d(\mathbf{w})$ to a given weight at a minimum of $E(\mathbf{w})$ is proportional to its magnitude. In this case for a given weight to be exactly zeroed the sensitivity of the unregularized error to this

weight must itself be zero. "This is the same condition as for an unregularized network so that [G]aussian weight decay contributes nothing toward pruning in the strictest sense" [4, p. 121]. The Laplace regularizer has the advantage of establishing a pruning threshold which may be adapted during training [4]. In the next section we develop a dynamical system model of regularization to study and compare the Laplace and Gaussian regularizers and to consider the utility of an adaptive ε.

II. A Dynamical System Perspective of Regularization

As $\eta \to 0$ the discrete-time regularized learning rule (3) may be viewed as an approximation of a continuous-time dynamical system

$$\frac{d\mathbf{w}}{dt} = -\nabla_{\mathbf{w}} E_d(\mathbf{w}) - \varepsilon(t)\nabla_{\mathbf{w}} E_w(\mathbf{w}), \qquad \mathbf{w}(0) = \mathbf{w}^{\langle 0\rangle} \tag{12}$$

where ε is now considered to be a nonnegative time-varying function. Although $\varepsilon(t)$ is a function of time only, and is thus independent of the system state, we may assume that $\varepsilon(t)$ coincides with a function generated by an adaptive regularization strategy. Weight adaptation is now described by a continuous trajectory beginning at an initial condition $\mathbf{w}^{\langle 0\rangle}$ and ending at a fixed point $\mathbf{w}^{\langle f\rangle}$, which corresponds to a local minimum of $E(\mathbf{w})$. This trajectory in the weight space is determined by the negative superposition of the gradient fields of $E_d(\mathbf{w})$ and $\varepsilon(t)E_w(\mathbf{w})$. The second term of (12) corresponds to a variable regularization force which moves the weight trajectory toward regions of the weight space corresponding to a simplified MFNN architecture. We will now use this model to investigate Gaussian and Laplace regularization for a time-varying regularization parameter.

A. Gaussian Regularizer Dynamics

We first consider (12) for a Gaussian regularizer (8) and the quadratic error surface

$$E_d(\mathbf{w}) = \tfrac{1}{2}\mathbf{w}^T\mathbf{H}\mathbf{w} + \mathbf{b}^T\mathbf{w} + E_{d0} \tag{13}$$

where $\mathbf{w}$ is an N dimensional state vector. The $N \times N$ Hessian matrix $\mathbf{H}$, the N dimensional vector $\mathbf{b}$, and E_{d0} are constants which characterize the error surface. Here we assume that $\mathbf{H}$ is at least positive semidefinite. Thus $E_d(\mathbf{w})$ has a minimum value at all vectors $\hat{\mathbf{w}}$ which satisfy $\mathbf{H}\hat{\mathbf{w}} + \mathbf{b} = \mathbf{0}$. In general a MFNN error surface will, of course, not be quadratic. We have chosen this error surface to demonstrate the utility of our dynamical system perspective and to obtain useful results for a comparison of Laplace and Gaussian regularization. Furthermore, an equation of the form (13) may be used as a local approximation of a general $E_d(\mathbf{w})$ near a minimum.

Due to its symmetry $\mathbf{H}$ may be diagonalized using a unitary transformation $\Lambda = \mathbf{U}^T\mathbf{H}\mathbf{U}$ where the N column vectors $\mathbf{u}^{\langle i\rangle}$ of $\mathbf{U}$ comprise an orthonormal set of eigenvectors corresponding to the N nonnegative eigenvalues λ_i of $\mathbf{H}$ contained in the diagonal matrix Λ. The linear transformation $\mathbf{w}' = \mathbf{U}^T\mathbf{w}$ yields a coordinate system in which the weight space axes are the eigenvectors of $\mathbf{H}$. The prime denotes the representation

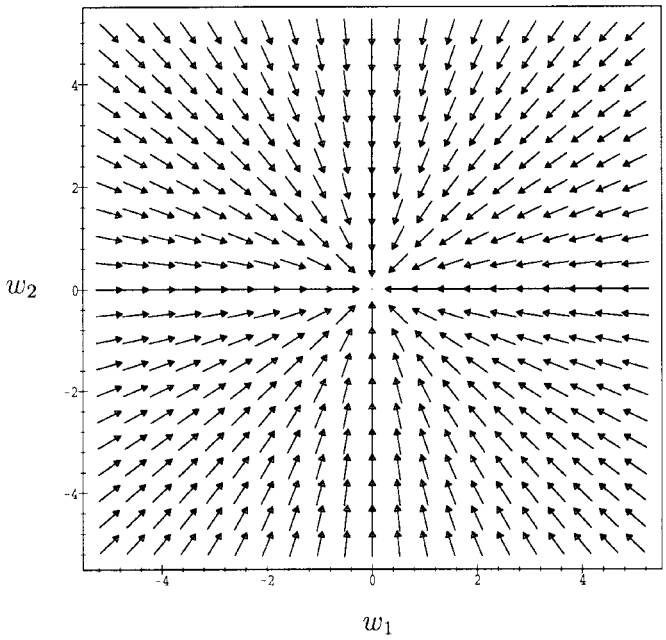

Fig. 1. Vector field showing the direction of the negative gradient of the Gaussian regularizer as described in (9) for $N = 2$.

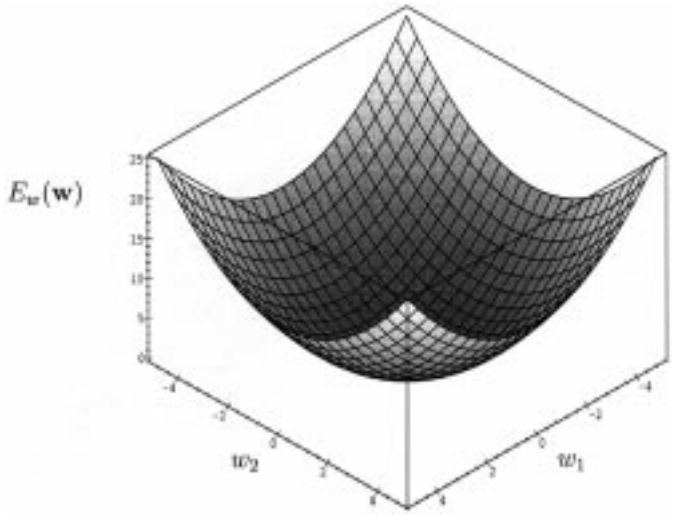

Fig. 2. Error surface for the Gaussian regularizer of (8) for $N = 2$.

of a vector in this new basis. This transformation preserves both length and angle as $\mathbf{U}$ is unitary [10].

Figs. 1 and 2 show that the Gaussian regularizer creates a regularization force directed toward the origin. Using (12) we obtain the linear system

$$\frac{d\mathbf{w}}{dt} = -[\mathbf{H} + \varepsilon(t)\mathbf{I}]\mathbf{w} - \mathbf{b}, \qquad \mathbf{w}(0) = \mathbf{w}^{\langle 0\rangle} \tag{14}$$

with a state transition matrix $\boldsymbol{\Phi}(t, t_0) = \mathbf{U}\,\mathrm{diag}[\mathbf{r}(t, t_0)]\mathbf{U}^T$ where

$$r_i(t, t_0) = \exp\left\{-\lambda_i(t - t_0) - \int_{t_0}^{t} \varepsilon(\tau)\,d\tau\right\}. \tag{15}$$

Thus the integral of $\varepsilon(t)$ is added to the system eigenvalues.

The solution of system (14) is

$$\mathbf{w}(t) = \{\mathbf{U}\,\mathrm{diag}[\mathbf{r}(t, t_0)]\}\,\mathbf{U}^T\mathbf{w}^{\langle 0\rangle} - \left\{\mathbf{U}\int_{t_0}^{t}\mathrm{diag}[\mathbf{r}(t, \phi)]\,\mathrm{d}\phi\right\}\mathbf{U}^T\mathbf{b}. \quad (16)$$

The effect of $\varepsilon(t)$ is most readily considered by using the eigenvectors of $\mathbf{H}$ as the basis of $\mathbf{R}^N$. We consider the weight trajectory $\mathbf{w}'(t) = \mathbf{U}^T\mathbf{w}(t)$ given by

$$\mathbf{w}'(t) = \{\mathrm{diag}\,[\mathbf{r}(t, t_0)]\}\,\mathbf{w}'^{\langle 0\rangle} - \left\{\int_{t_0}^{t}\mathrm{diag}[\mathbf{r}(t, \phi)]\,\mathrm{d}\phi\right\}\mathbf{b}'. \quad (17)$$

The regularization parameter $\varepsilon(t)$ increases the decay rate of $r_i(t, t_0)$ and decreases the value of its integral. Thus $\varepsilon(t)$ has a suppressive effect on $w_i'(t)$ which becomes stronger as $\varepsilon(t)$ increases. This effect is more significant for weights w_i' which are relatively less important for minimization of $E_d(\mathbf{w})$ as indicated by a small value of λ_i. For a larger value of λ_i, which indicates a relatively more important weight, this effect is reduced. Note that by varying ε we are moving the location of the system fixed point(s). For any $\varepsilon(t)$ which is not zero for a sufficient period of time at the conclusion of training, we will not optimize $E_d(\mathbf{w})$, illustrating the inherent tradeoff between pruning and unregularized error minimization. These observations are apparent in the case of a constant $\varepsilon(t) = \varepsilon$ where

$$w_i'(t) = \exp\{-(\lambda_i + \varepsilon)(t - t_0)\}w_i^1\langle 0\rangle - \frac{1}{\lambda_i + \varepsilon}[1 - \exp\{-(\lambda_i + \varepsilon)(t - t_0)\}]\,b_i' \quad (18)$$

which approaches $-b_i'/(\lambda_i + \varepsilon)$. This corresponds to the ith entry of a minimum of $E(\mathbf{w})$ as expressed in the new basis specified by $\mathbf{U}$ and denoted as $\tilde{\mathbf{w}}'$. The quadratic error (13) has a minimum in this basis at $\hat{\mathbf{w}}'$ where $\hat{w}_i' = -b_i'/\lambda_i$ for a positive definite $\mathbf{H}$. The ratio between the ith element of these weights at a minimum of the regularized and unregularized error functions is [9]

$$\frac{\tilde{w}_i'}{\hat{w}_i'} = \frac{\lambda_i}{\lambda_i + \varepsilon}. \quad (19)$$

Reference [9] relates the following observations on this ratio. For an important direction in the weight space, $\lambda_i \gg \varepsilon$, and the ratio (19) is near one. Thus $\hat{w}_i'$ is relatively unaffected by ε and $\tilde{w}_i' \approx \hat{w}_i'$. For a small value of λ_i, in which case the ratio (19) is also small, $\tilde{w}_i'$ is suppressed. Any reduction in $||\tilde{\mathbf{w}}'||$ corresponds to an equal reduction in $||\tilde{\mathbf{w}}||$ as $\mathbf{U}$ is unitary.

B. Laplace Regularizer Dynamics

We now consider (12) for a Laplace regularizer (4). The resulting system is

$$\frac{d\mathbf{w}}{dt} = -\nabla_{\mathbf{w}}E_d(\mathbf{w}) - \varepsilon(t)\mathrm{sgn}(\mathbf{w}), \qquad \mathbf{w}(0) = \mathbf{w}^{\langle 0\rangle} \quad (20)$$

with the caveat that dw_i/dt does not exist for $w_i = 0$. A stability analysis may be used to define a value of this derivative [4]. We employ a different approach and use the approximate Laplace regularizer

$$E_w(\mathbf{w}) = \left\{\frac{1}{\theta}\sum_{i\in C}\ln[\cosh(\theta w_i)]\right\} \quad (21)$$

which is differentiable for all w_i. This approximation of (4) improves as θ grows large. The corresponding gradient of (21) is

$$\nabla_{\mathbf{w}}E_w(\mathbf{w}) = \tanh(\theta\mathbf{w}) \quad (22)$$

which is approximately $\mathrm{sgn}(\mathbf{w})$ for θ large [11].

Substituting (22) for $\mathrm{sgn}(\mathbf{w})$ in (20) yields a smoothed dynamical system given by

$$\frac{d\mathbf{w}}{dt} = -\nabla_{\mathbf{w}}E_d(\mathbf{w}) - \varepsilon(t)\tanh(\theta\mathbf{w}), \qquad \mathbf{w}(0) = \mathbf{w}^{\langle 0\rangle}. \quad (23)$$

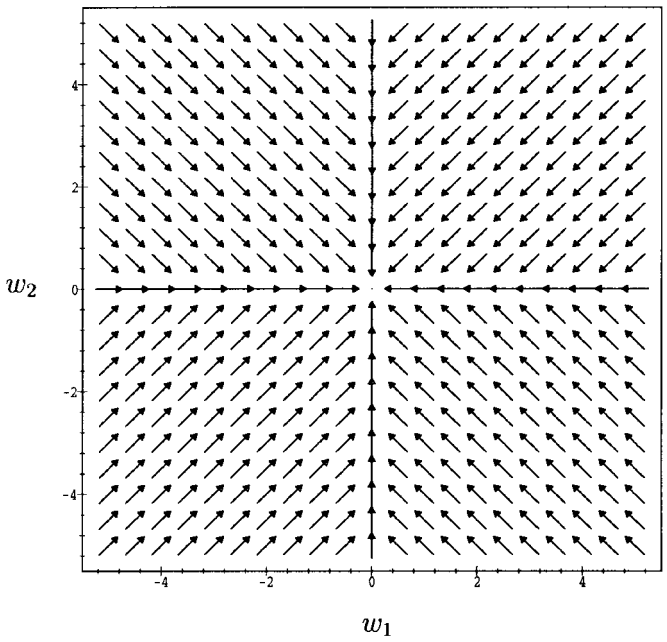

Fig. 3. Vector field showing the direction of the negative gradient of the approximate Laplace regularizer as described in (22) for $\theta = 10$ and $N = 2$.

Refer to Figs. 3 and 4. The gradient of the penalty term $\nabla_{\mathbf{w}}E_w(\mathbf{w}) = \tanh(\theta\mathbf{w})$ creates a regularization force directed toward the weight space axes. The strategy of LF is to scale this force by a constant amount ε. However, by considering ε as a variable function of time, more flexible behavior may be obtained. For example, a relatively large $\varepsilon(t)$ at the beginning of training will move the architecture toward a simplified configuration. Once redundant weights have been sufficiently suppressed, the value of $\varepsilon(t)$ may be reduced, since a relatively small value is able to restrict redundant weight growth as the regularization force dominates the movement of such weights. This reduction of $\varepsilon(t)$ enables nonredundant weights to settle near values necessary for local minimization of $E_d(\mathbf{w})$ for the simplified MFNN architecture. This type of behavior is apparent in the simulation results of [12].

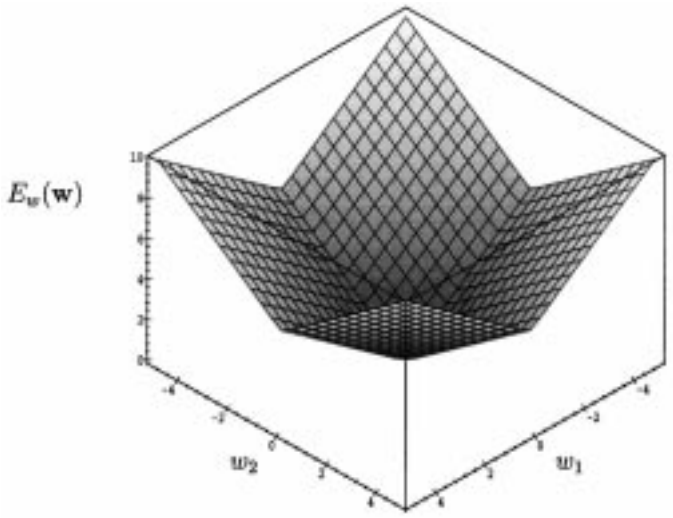

Fig. 4. Error surface for the approximate Laplace regularizer of (21) for $\theta = 10$ and $N = 2$.

1) Approximate Solution: In this section we develop an approximation of (23) valid in different regions of the weight space. This enables use of linear system techniques to study the effect of Laplace regularization for the quadratic error case (13). To this end we employ the approximation

$$\tanh(\theta w_i) \approx \begin{cases} -1, & \text{if } w_i \le -1/\theta \\ \theta w_i, & \text{if } |w_i| < 1/\theta \\ 1, & \text{if } w_i \ge 1/\theta. \end{cases} \tag{24}$$

In order to incorporate this approximation into (23) the vectors $\mathbf{h}$ and $\mathbf{p}$ are defined as follows. The magnitude of the ith element of the vector $\mathbf{h}$ is one for all weights considered to be large, i.e., all weights such that $|w_i| \ge 1/\theta$, and zero otherwise. Furthermore, $\text{sgn}(h_i) = \text{sgn}(w_i)$ for a large weight w_i. For those weights with $h_i = \pm 1$, we consider $\tanh(\theta w_i) \approx \pm 1$. The ith element of $\mathbf{p}$ is one for all small weights, i.e., all weights which satisfy $|w_i| < 1/\theta$. Small weights fall into the region where $\tanh(\theta w_i) \approx \theta w_i$. Thus

$$h_i \triangleq \begin{cases} -1, & \text{if } w_i \le -1/\theta \\ 0, & \text{if } |w_i| < 1/\theta \\ 1, & \text{if } w_i \ge 1/\theta \end{cases} \tag{25}$$

and

$$p_i \triangleq \begin{cases} 0, & \text{if } |w_i| \ge 1/\theta \\ 1, & \text{if } |w_i| < 1/\theta. \end{cases} \tag{26}$$

This yields a piecewise linear approximation to (22)

$$\nabla_{\mathbf{w}} E_w(\mathbf{w}) = \tanh(\theta \mathbf{w}) \approx \mathbf{h} + \theta \text{diag}(\mathbf{p})\, \mathbf{w}. \tag{27}$$

Using (27) we may now approximate (23) for the quadratic error (13) as

$$\begin{aligned} \frac{d\mathbf{w}}{dt} &= -\nabla_{\mathbf{w}} E_d(\mathbf{w}) - \varepsilon(t)[\mathbf{h} + \theta \text{diag}(\mathbf{p})\mathbf{w}] \\ &= -[\mathbf{H} + \theta \text{diag}(\mathbf{p})\varepsilon(t)]\mathbf{w} - [\mathbf{b} + \varepsilon(t)\mathbf{h}]. \end{aligned} \tag{28}$$

It is important to note that $\mathbf{h}$ and $\mathbf{p}$ are *constant* within each region under consideration and that $\text{sgn}(h_i) = \text{sgn}(w_i^{\langle 0 \rangle})$ for large weights. Once a weight transitions to a different region as defined in (25) and (26), a different linear system must be used. The linear system given in (28) has the solution

$$\begin{aligned} \mathbf{w}(t) = \mathbf{\Phi}(t, t_0)\mathbf{w}^{\langle 0 \rangle} &- \left[\int_{t_0}^{t} \mathbf{\Phi}(t, \phi)\, \mathrm{d}\phi\right]\mathbf{b} \\ &- \left[\int_{t_0}^{t} \mathbf{\Phi}(t, \phi)\varepsilon(\phi)\, \mathrm{d}\phi\right]\mathbf{h} \end{aligned} \tag{29}$$

where the state transition matrix is

$$\mathbf{\Phi}(t, t_0) = \exp\left\{-\mathbf{H}(t - t_0) - \theta \text{diag}(\mathbf{p}) \int_{t_0}^{t} \varepsilon(\tau)\, \mathrm{d}\tau\right\}. \tag{30}$$

We next consider (29) for the large weight case.

2) Large Weight Case: In this section we examine our model (28) of LF in regions of the weight space where all MFNN weights have a large magnitude as defined in (25) i.e., $|w_i| \ge 1/\theta$ for all i. The large weight case is the most important since the criterion for small weights (26) becomes vanishingly small as we improve approximation (21) to (4) by letting θ grow large. In fact, the results in this section could have been obtained by considering system (20) without approximation (23). However, we wish to use a continuous model of LF as developed in our previous work [11] and to consider the effects of θ.

For the large weight case $\mathbf{p} = \mathbf{0}$ and from (28)

$$\frac{d\mathbf{w}}{dt} = -\mathbf{H}\mathbf{w} - [\mathbf{b} + \varepsilon(t)\mathbf{h}], \qquad \mathbf{w}(0) = \mathbf{w}^{\langle 0 \rangle}. \tag{31}$$

Equation (31) is valid within a region of the weight space specified by $\mathbf{h}$ and denoted as $\mathbf{R}^{\langle \mathbf{h} \rangle}$. The vector $\mathbf{h}$ forms an angle of $\arccos(1/\sqrt{N})$ with the half of each weight space axis which roughly borders $\mathbf{R}^{\langle \mathbf{h} \rangle}$. Thus the last term of (31) forces the system state toward the weight space axes as is evident in a particular quadrant of Fig. 3. The regularization parameter acts as an input to control the magnitude of this force.

The state transition matrix for the large weight case is given by

$$\mathbf{\Phi}(t, t_0) = \mathbf{U} \exp\{-\mathbf{\Lambda}(t - t_0)\}\mathbf{U}^T \tag{32}$$

and (29) reduces to

$$\begin{aligned} \mathbf{w}(t) = \mathbf{\Phi}(t, t_0)\mathbf{w}^{\langle 0 \rangle} &- \{\mathbf{U}\, \text{diag}[\mathbf{q}(t, t_0)]\mathbf{U}^T\}\mathbf{b} \\ &- \{\mathbf{U}\, \text{diag}[\mathbf{f}(t, t_0)]\mathbf{U}^T\}\mathbf{h} \end{aligned} \tag{33}$$

where the N elements of $\mathbf{f}$ and $\mathbf{q}$ are

$$f_i(t, t_0) = \int_{t_0}^{t} \exp\{-\lambda_i(t - \phi)\}\varepsilon(\phi)\, \mathrm{d}\phi \tag{34}$$

and

$$q_i(t, t_0) = \frac{1 - \exp\{-\lambda_i(t - t_0)\}}{\lambda_i}. \tag{35}$$

The weight trajectory $\mathbf{w}'(t) = \mathbf{U}^T\mathbf{w}(t)$ is

$$\begin{aligned} \mathbf{w}'(t) = \exp\{-\mathbf{\Lambda}(t - t_0)\}\mathbf{w}'^{\langle 0 \rangle} &- \{\text{diag}[\mathbf{q}(t, t_0)]\}\mathbf{b}' \\ &- \{\text{diag}[\mathbf{f}(t, t_0)]\}\mathbf{h}' \end{aligned} \tag{36}$$

where $\varepsilon(t)$ only affects the last term. Equation (36) is valid only as long as $\mathbf{w}(t) = \mathbf{U}\mathbf{w}'(t)$ remains in $\mathbf{R}^{\langle \mathbf{h} \rangle}$.

The last term of (36) determines the effect of the Laplace regularizer within $\mathbf{R}^{\langle \mathbf{h} \rangle}$. The vector $\mathbf{h}'$ is the representation of $\mathbf{h}$ in the basis specified by $\mathbf{U}$. The direction of movement along the $\mathbf{u}^{\langle i \rangle}$ axis is determined by the sign of this projection as $f_i(t, t_0)$ is nonnegative. The magnitude of this projection is maximal for $\mathbf{h}$ and $\mathbf{u}^{\langle i \rangle}$ parallel or antiparallel and zero for $\mathbf{h}$ and $\mathbf{u}^{\langle i \rangle}$ perpendicular. The nonnegative $f_i(t, t_0)$ further scales this last term. For a small value of λ_i, $\varepsilon(t)$ has a greater effect on $f_i(t, t_0)$, while for a large λ_i this effect is reduced. Thus $\varepsilon(t)$ forces the system state toward the weight space axes primarily in directions which are less important for minimization of $E_d(\mathbf{w})$.

For a constant $\varepsilon(t) = \varepsilon$ the weight trajectory is

$$w_i'(t) = \exp\{-\lambda_i(t - t_0)\} w_i^{1\langle 0 \rangle} - q_i(t, t_0)[b_i' + \varepsilon h_i'] \quad (37)$$

which approaches $\tilde{w}_i' = -(1/\lambda_i)(b_i' + \varepsilon h_i')$. Consider $\tilde{\mathbf{w}}$ and $\hat{\mathbf{w}}$ in $\mathbf{R}^{\langle \mathbf{h} \rangle}$. The ratio between the elements of the corresponding vectors is

$$\frac{\tilde{w}_i'}{\hat{w}_i'} = 1 + \varepsilon \frac{h_i'}{b_i'}. \quad (38)$$

Since $\mathbf{b}' = -\mathbf{U}^T \mathbf{H} \hat{\mathbf{w}} = -\mathbf{\Lambda} \mathbf{U}^T \hat{\mathbf{w}}$, (38) may be expressed as

$$\frac{\tilde{w}_i'}{\hat{w}_i'} = 1 - \frac{\varepsilon}{\lambda_i} \frac{[\mathbf{u}^{\langle i \rangle} \cdot \mathbf{h}]}{[\mathbf{u}^{\langle i \rangle} \cdot \hat{\mathbf{w}}]}. \quad (39)$$

Thus the movement of the ith component of the regularized error function minimum $\tilde{\mathbf{w}}'$ along the basis vector $\mathbf{u}^{\langle i \rangle}$ is directly proportional to ε and inversely proportional to the importance of this direction as indicated by λ_i. This shift is linear in ε.

3) Small Weight Case: In this section we consider the LF model (28) for the case in which all of the MFNN weights have a small magnitude, i.e., $|w_i| < 1/\theta \; \forall i$. Thus $\mathbf{h} = \mathbf{0}$ and

$$\frac{d\mathbf{w}}{dt} = -[\mathbf{H} + \theta \varepsilon(t) \mathbf{I}]\mathbf{w} - \mathbf{b}, \qquad \mathbf{w}(0) = \mathbf{w}^{\langle 0 \rangle} \quad (40)$$

which corresponds to the Gaussian case (14) with $\theta = 1$. The weight trajectory is given by

$$\mathbf{w}'(t) = \{\mathrm{diag}[g_i(t, t_0)]\} \mathbf{w}'^{\langle 0 \rangle} - \left\{ \int_{t_0}^{t} \mathrm{diag}[g_i(t, \phi)] \, \mathrm{d}\phi \right\} \mathbf{b}' \quad (41)$$

where

$$g_i(t, t_0) = \exp\left\{ -\lambda_i(t - t_0) - \theta \int_{t_0}^{t} \varepsilon(\tau) \, \mathrm{d}\tau \right\}. \quad (42)$$

Note that an increase in the slope θ of the linear approximation increases the decay rate of $g_i(t, t_0)$ for a nonzero $\varepsilon(t)$.

4) Mixed Weight Case: This section describes the behavior of the LF model (28) for a MFNN with both small and large weights as defined in (25) and (26). For simplicity we consider $N = 2$, $\mathbf{H}$ diagonal, w_1 small, and w_2 large and positive. Thus $\mathbf{p} = [1, 0]^T$, $\mathbf{h} = [0, 1]^T$, and

$$\mathbf{w}(t) = \mathbf{w}'(t) = \begin{bmatrix} g_1(t, t_0) \, w_1^{\langle 0 \rangle} \\ \exp\{-\lambda_2(t - t_0)\} \, w_2^{\langle 0 \rangle} \end{bmatrix} - \begin{bmatrix} \int_{t_0}^{t} g_1(t, \phi) \, \mathrm{d}\phi \, b_1 \\ \int_{t_0}^{t} \exp\{-\lambda_2(t - \phi)\} \, \mathrm{d}\phi \, b_2 \end{bmatrix} - \begin{bmatrix} 0 \\ f_2(t, t_0) \end{bmatrix}. \quad (43)$$

For this case $\varepsilon(t)$ acts both as a parameter of the plant dynamics and as a system input. We will further consider these results in the section in which we compare the Gaussian and Laplace regularizers. Before proceeding, we provide a simple numerical example of our continuous model of Laplace regularization.

5) Numerical Example: In this section we consider a simple example to illustrate the use of the developed approximations for Laplace regularization and the effects of an adaptable $\varepsilon(t)$. In Fig. 5 $E_d(\mathbf{w}) = 0w_1 + (w_2 - 2)^2$. In this case w_1 is completely redundant and both weights are subject to regularization. Fig. 5 shows three trajectories for an initial condition $\mathbf{w}^{\langle 0 \rangle} = [-4, 4]^T$ along with constant error contours of $E_d(\mathbf{w})$. The first corresponds to the case of $\varepsilon(t) = 0$ in which system (23) simply descends for 3.2 time units to $\mathbf{w}^{\langle f, 1 \rangle} \approx [-4, 2]^T$ which corresponds to an unpruned solution. The second case is the solution of (23) for an *ad hoc* variable regularization function $\varepsilon(t) = t^2 \exp\{-0.7t\}$ and $\theta = 10$, which causes the system to settle near $\mathbf{w}^{\langle f, 2 \rangle} = [0, 2]^T$ after 15 time units. Note that a small $\varepsilon(t)$ at the end of training enabled the system to converge to a point near the optimal solution. The third trajectory is a portion of the approximate solution for the large weight case (33) with $\mathbf{p} = \mathbf{0}$ and $\mathbf{h} = [-1, 1]^T$, also for 15 time units. The approximation closely matches the second trajectory until it reaches the neighborhood of $w_1 = 0$.

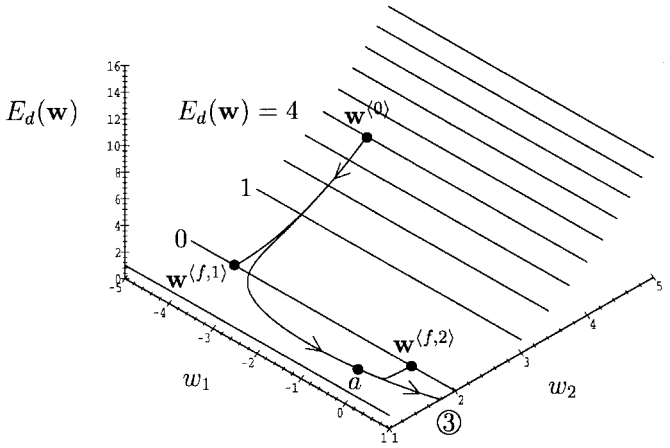

Fig. 5. Weight space trajectories for the approximate LF system (21) with $\varepsilon(t) = 0$ ($\mathbf{w}^{\langle f, 1 \rangle}$), $\varepsilon(t) = t^2 \exp\{-0.7t\}$ ($\mathbf{w}^{\langle f, 2 \rangle}$), and the large weight case approximation (33) with $\varepsilon(t) = t^2 \exp\{-0.7t\}$ (trajectory 3). Point a is used in the description of Fig. 7.

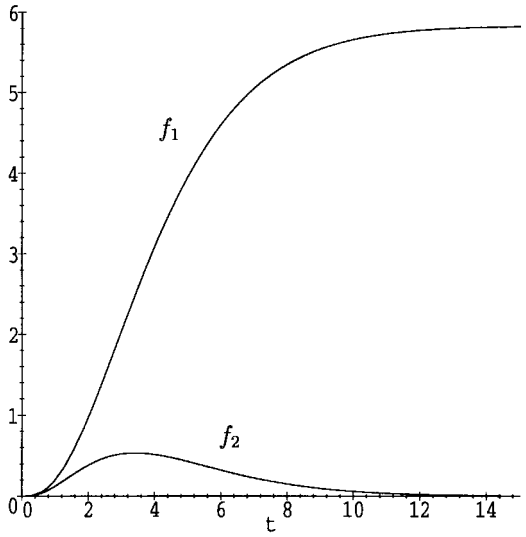

Fig. 6. Functions $f_1(t, 0)$ and $f_2(t, 0)$ for the large weight case of Fig. 5.

Fig. 6 shows $f_1(t, 0)$ compared with $f_2(t, 0)$ for $\lambda_1 = 0$ and $\lambda_2 = 2$. These functions measure the effect of $\varepsilon(t)$ on the weight trajectory as discussed in Section II-B2. Thus $\varepsilon(t)$ has a much greater effect on w_1 than w_2. Indeed, w_1 would not move from its initial state without a nonzero $\varepsilon(t)$. Furthermore, regularization required approximately three times as long as the unregularized minimization to reach the optimal point, providing some rationale for the heuristic method of [12] in which ε is adapted to slow the learning convergence rate in order to provide MFNN regularization. As demonstrated in the divergence of the third trajectory of Fig. 5, the mixed weight approximation must be used for w_1 close to zero.

Fig. 7 shows a magnified view of the second trajectory of Fig. 5 corresponding to the solution of (23) for t ranging from 4.94 to 15 time units along with constant error contours of $E_d(\mathbf{w})$. The trajectory for the mixed weight case as described by (43) is also shown. Note the close agreement. The rapid decay of $g_1(t, 4.94)$ due to $\varepsilon(t)$ is illustrated in Fig. 8—$g_1(t, 4.94)$ would have a constant value of one without $\varepsilon(t) \neq 0$. Function $f_2(t, 4.94)$ is shown for completeness. This simple example illustrates both the benefits of an regularization parameter and the utility of the presented approximation.

C. Comparison of the Gaussian and Laplace Regularizers

We have previously compared Gaussian and Laplace regularizers from a Bayesian perspective (Section I-C) and by comparing the sensitivity of the error to each weight at an error minimum (Section I-D). These results indicate that the Laplace regularizer has superior characteristics for use in regularizing the internal weights of MFNN's. Here we compare these regularizers using the results derived by considering learning as a dynamical system.

Even though both the Gaussian and Laplace regularizers reduce the norm of the weight solution, the manner in which this norm is reduced differs significantly. For the Gaussian regularizer and a quadratic error, $\varepsilon(t)$ provides weight regularization by modifying the system eigenvalues. This is

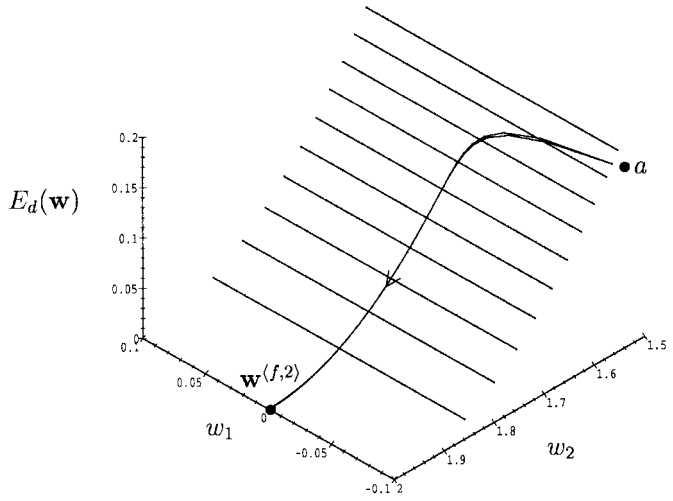

Fig. 7. Weight space trajectories for the approximate LF system (23) and the mixed weight case approximation (43) for $\varepsilon(t) = t^2 \exp\{-0.7t\}$. Note the orientation of the axes and the position of reference point a as compared to Fig. 5.

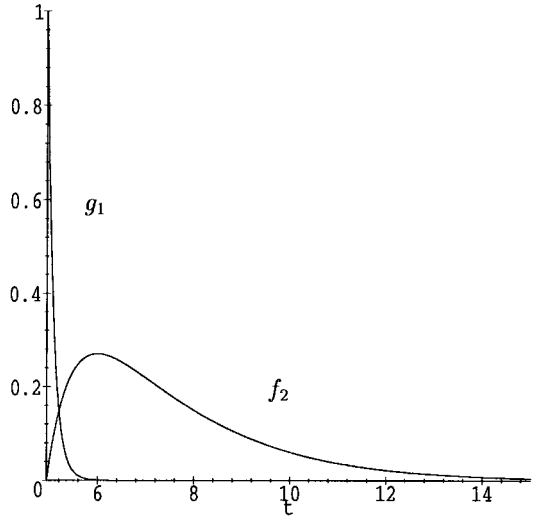

Fig. 8. Functions $g_1(t, 4.94)$ and $f_2(t, 4.94)$ for the mixed weight case of Fig. 7.

evident for the case of a constant ε as described in (19), which indicates that the Gaussian regularizer shifts the system minimum by an amount dependent upon the ratio between the eigenvalues of the regularized and unregularized systems. The Gaussian regularizer requires an unbounded ε to exactly zero a component of the weight $\mathbf{w}$ along the $\mathbf{u}^{\langle i\rangle}$ axis for $\lambda_i \neq 0$.

As θ grows large the large weight case of Section II-B2 describes with greater accuracy the effect of the Laplace regularizer. These results show that for the Laplace regularizer the regularization parameter $\varepsilon(t)$ acts as a control input, providing a regularization effect with a magnitude dependent upon the increased sensitivity of the weight trajectory to $\varepsilon(t)$ in directions less important for minimization of the unregularized error function $E_d(\mathbf{w})$. Equation (39) indicates that a constant ε may be used to linearly shift the system minimum within $\mathbf{R}^{\langle \mathbf{h}\rangle}$ toward the weight space axes. These axes are increasingly close to $\mathbf{R}^{\langle \mathbf{h}\rangle}$ as θ grows large. The small and mixed weight

cases of Sections II-B3 and II-B4 exhibit behavior similar to that of the Gaussian regularizer which vanishes as θ grows large, indicating the importance of the behavior of the Laplace regularizer at the origin as noted in [4].

These differences are also evident in Figs. 1–4 and in the stability conditions (10) and (11). Disregarding the effect of $E_d(\mathbf{w})$, the Gaussian regularizer forces the system state directly toward the origin. The magnitude of this regularization force is greater at points in the weight space farther from the origin as indicated in (9). In contrast, the Laplace regularizer is capable of forcing the system state toward points in the weight space where only one or more weights are zero. As shown in (5), the magnitude of this regularization force is independent of the system state and thus is the same for small and large weights. These observations provide a dynamical system perspective of the superiority of the Laplace over the Gaussian regularizer as previously reported [4].

III. Conclusions

We have developed a continuous dynamical system model of multilayer feedforward neural-network regularization in order to analyze the effects of Laplace regularization. For the case of a quadratic error surface, this approach provided analytic results which demonstrate that a generalized time-varying regularization parameter $\varepsilon(t)$ acts as a system input which controls the movement of weights toward the axes of the weight space. This regularization force has a greater effect in directions which are relatively less important for minimization of the quadratic error function. These analytic results also provide a partial explanation of the experimental results presented in [11], which demonstrated the effect of ε on the stability of fixed points in the weight space, a rationale for the method of [12], and additional insights into the use of adaptive regularization parameters. A similar analysis indicates that the Gaussian regularizer provides a regularization force which is directed toward the origin with a magnitude dependent upon the system state, in contrast to the Laplace regularizer. This provides a dynamical system perspective of the superiority of the Laplace regularizer as reported in [4]. We are working to extend this framework to more complex error surfaces and other penalty terms and are further investigating the utility of using a time-varying regularization parameter.

Acknowledgment

The authors would like to thank the reviewers for their helpful suggestions.

References

[1] P. D. Wasserman, *Advanced Methods in Neural Computing.* New York: Van Nostrand Reinhold, 1993.
[2] R. Reed, "Pruning algorithms—A survey," *IEEE Trans. Neural Networks*, vol. 4, pp. 740–747, Sept. 1993.
[3] D. J. C. MacKay, "A practical Bayesian framework for backpropagation networks," *Neural Computa.*, vol. 4, pp. 448–472, 1992.
[4] P. M. Williams, "Bayesian regularization and pruning using a Laplace prior," *Neural Computa.*, vol. 7, pp. 117–143, 1995.
[5] J. M. Zurada, *Introduction to Artificial Neural Systems.* Boston, MA: PWS, 1992.
[6] M. Ishikawa, "Structural learning with forgetting," *Neural Networks*, vol. 9, pp. 509–521, Apr. 1996.
[7] D. A. Miller and J. M. Zurada, "Dynamics of structural learning with an adaptive forgetting rate," in *Proc. Int. Conf. Neural Networks*, Houston, TX, June 9–12, 1997, vol. 3, pp. 1827–1832.
[8] A. Malinowski, D. A. Miller, and J. M. Zurada, "Reconciling training and weight suppression: New guidelines for pruning-efficient training," in *Proc. World Congr. Neural Networks*, Washington, DC, July 17–21, 1995, vol. 1, pp. 724–728.
[9] C. M. Bishop, *Neural Networks for Pattern Recognition.* New York: Oxford Univ. Press, 1995.
[10] B. Noble and J. W. Daniel, *Applied Linear Algebra*, 2nd ed. Englewood Cliffs, NJ: Prentice-Hall, 1977.
[11] A. Lozowski, D. A. Miller, and J. M. Zurada, "Dynamics of error backpropagation learning with pruning in the weight space," in *Proc. IEEE Int. Symp. Circuits Syst.*, Atlanta, GA, May 12–15, 1996, vol. 3, pp. 449–452.
[12] D. A. Miller, J. M. Zurada, and J. H. Lilly, "Pruning via dynamic adaptation of the forgetting rate in structural learning," in *Proc. Int. Conf. Neural Networks*,, Washington, DC, June 3–6, 1996, vol. 1, pp. 448–452.

Damon A. Miller (M'86) was born in Louisville, KY, in 1966. He received the B.S. degree in engineering science, the M.Eng. degree with specialization in electrical engineering, and the Ph.D. degree in computer science and engineering from the University of Louisville in 1988, 1989, and 1997, respectively.

From 1989 to 1993 he worked for General Electric Aerospace (now Lockheed Martin) in Valley Forge, PA. He is presently an Assistant Professor of Electrical and Computer Engineering at Western Michigan University, Kalamazoo. His research interests include artificial and biological neural networks and dynamical systems.

Dr. Miller received the General Electric Astro-Space General Manager's Award in 1992 and was selected as a University of Louisville Doctoral Fellow in 1993. He is a member of Eta Kappa Nu and Tau Beta Pi.

Jacek M. Zurada (M'82–SM'83–F'96), for photograph and biography, see p. 1 of the January 1998 issue of this TRANSACTIONS.