Application-Level Robustness and Redundancy in Linear Systems

Cesare Alippi

***Abstract*—The paper quantifies the degradation in performance of a linear model induced by perturbations affecting its identified parameters. We extend sensitivity analyses available in the literature, by considering a generalization-based figure of merit instead of the inaccurate training one. Effective off-line techniques reducing the impact of perturbations on generalization performance are introduced to improve the robustness of the model. It is shown that further robustness can be achieved by optimally redistributing the information content of the given model over topologically more complex linear models of neural network type. Despite the additional robustness achievable, it is shown that the price we have to pay might be too high and the additional resources would be better used to implement a n-ary modular redundancy scheme.**

***Index Terms*—Linear computation, linear neural networks, perturbation analysis, robustness, sensitivity.**

I. Introduction

Application robustness, defined as the ability to provide a contained degradation in performance when the algorithm solving the application is perturbed in its structural parameters, has an immediate impact on the design of a reliable circuit. In fact, given two applications A_1 and A_2 (with A_1 more robust than A_2), a physical device, and perturbations having the same magnitude, then the experienced loss in performance of A_1 is smaller than that of A_2.

The robustness of a computational flow subject to perturbations affecting its parameters has been widely studied in the literature. Results can be used within an analysis framework [1], [2] to validate an architectural design (perturbations are applied to parameters and the loss in performance is evaluated at the device output) or to provide design guidelines for the subsequent implementation (synthesis phase) [1], [3], [4]. The analysis phase is generally embedded in the synthesis one to estimate the robustness of a candidate solution; for its interest, we focus the attention on the analysis phase.

Methods provided in the literature for robustness/sensitivity analyses of parameterized models are generally tailored to a specific computational model, i.e., linear/nonlinear, with known/unknown parameters while suitable hypotheses are envisioned to make the mathematics more amenable. When the coefficients of a linear model are known the perturbation analysis is simple and the perturbation/performance relationship can be easily derived in a closed form (e.g., see [4]). Conversely, when the model becomes nonlinear, a Taylor expansion for the model or the loss function is generally considered (e.g., see [1], [5] and [6]).

The robustness analysis becomes more complex when the coefficients of the model are unknown and need to be identified from a set Z^N of $N(input,\, output)$ pairs [7], [8]. In such a case, the presence of noise-affected data and a limited N reflect on the model coefficients which differ, in probability, from the unknown nominal ones. To address this relevant case some authors assume that the identified coefficients are the true ones while others opt for a statistical approach by assuming convenient distributions for the involved entities [3], [4].

Manuscript received February 1, 2001; revised January 29, 2002. This paper was recommended by Associate Editor G. R. Chen.

The author is with Dip. Elettronica e Informazione, Politecnico di Milano, Milan 20133, Italy.

Publisher Item Identifier 10.1109/TCSI.2002.800840.

In both cases, the sensitivity analysis is based on a training error function to quantify the performance loss of the perturbed model. As training figures of merit are inadequate to evaluate the performance of a model and generalization indexes must be considered instead, so an accurate sensitivity analysis must be related to validation figures of merit and not to training ones.

The involved entities can be formalized as follows. Denote by $\overline{y} = \Theta^o x$ the unknown reference model characterized by the d-dimensional column vector of inputs x and the d-dimensional row vector of parameters Θ^o. As with the classic identification theory [7], we assume that the available measured output y satisfies the $y = \overline{y} + \varepsilon$ model, where $\varepsilon = WN(0,\, \sigma_\varepsilon^2)$ is an additive, independent and identically distributed stationary noise.

Note that x can be a nonlinear function $x = g(z)$ in the external input vector z, independent from Θ^o: a polynomial expansion is therefore a linear model in its coefficients.

We consider the general case in which Θ^o is unknown and a model family M_0: $\hat{y}(\Theta) = \Theta x$ is selected for the parameter identification phase [3]. We observe that M_0 can also be interpreted as the simplest model of a nested linear neural network hierarchy $\mathbf{M}$: M_0; $M_1 \subseteq M_2 \subseteq M_3 \cdots \subseteq M_k \cdots$ (where M_k stands for a linear network with a single hidden layer of k units and a bias term only on the linear output neuron). M_0 can be intended as degenerate model (zero hidden units) of $\mathbf{M}$. For its nature, model M_k can degenerate to M_0 in force of the linear property by carrying out a multiplication between the weights associated with the two layers.

The aim of the paper is to extend and complete results given in the literature by:

- providing a novel approach to evaluate the robustness of model M_0 based on its generalization ability and not on its training error;
- introducing off line transformations to improve the robustness ability of M_0;
- investigating the relationships between robustness improvement and topological redundancy obtainable by considering a more complex model M_k instead of model M_0.

The structure of the paper is as follows. To characterize the robustness ability of M_0 we introduce, in Section II, an appropriate generalization-based performance loss function. Off-line transformations are then suggested in Section III to improve the robustness of M_0. The gain in robustness obtainable by redistributing the information content of M_0 over suitable models of $\mathbf{M}$ is also provided. An example showing how to improve the robustness of a linear filter is finally given in Section IV.

II. A Generalization-Based Figure of Merit for Robustness

In the following, we assume that parameter identification for model M_0 is based on a Least Mean Squared procedure (e.g., see [7]) minimizing the training function

$$J_{tr} = \frac{1}{2N} \sum_{i=1}^{N} (y_i - \hat{y}(\Theta,\, x_i))^2 \tag{1}$$

which provides the estimate $\hat{\Theta}$ of Θ^o having generalization performance $J_{\mathrm{val}}(\hat{\Theta})$. A generic perturbation $\delta\Theta$ affecting $\hat{\Theta}$ provides the perturbed vector $\hat{\Theta} + \delta\Theta$ of generalization performance $J_{\mathrm{val}}(\hat{\Theta} + \delta\Theta)$. A correct measure for the loss in performance is therefore $J_{\mathrm{val}}(\hat{\Theta} + \delta\Theta) - J_{\mathrm{val}}(\hat{\Theta})$ and not the training error discrepancy $J_{tr}(\hat{\Theta} + \delta\Theta) - J_{tr}(\hat{\Theta})$ as considered by several authors. It should be noted that cross-validation techniques are useless in our analysis since they provide the punctual loss estimate $J_{\mathrm{val}}(\hat{\Theta}+\delta\Theta) - J_{\mathrm{val}}(\hat{\Theta})$ but not the analytical function describing the relationships between model, perturbation and generalization performance loss. In addition, cross-validation estimates cannot be taken into account when a limited Z^N is available. Such problems can be overcome by following the analysis suggested in [9], [10] where a theory for estimating $J_{\mathrm{val}}(\Theta)$ valid also on a limited data set is presented. There, by suitably expanding with Taylor $J_{\mathrm{val}}(\Theta)$ and $J_{tr}(\Theta)$ around Θ^o, evaluating the expansion in $\hat{\Theta}$ and taking suitable averages, it is shown that the expected validation performance $J_{\mathrm{val}}(\Theta)$ is related to the expected training performance as

$$E\left[J_{\mathrm{val}}\left(\hat{\Theta}\right)\right] = \frac{N+\hat{p}}{N-\hat{p}}\, E\left[J''_{tr}\left(\hat{\Theta}\right)\right] \tag{2}$$

where expectation is taken over all possible sets of N training pairs Z^N; $\hat{p} = rank(J_{tr}(\hat{\Theta}))$ is an estimate of the effective number of parameters used by the model to fit the data ($p = d$ for M_0 if the rank of the Hessian form associated with the training error is full).

In reality, we have only one training data set and we cannot take the average required by (2). In such a case it can be shown [7], [9], [10] that

$$J_{\mathrm{val}}\left(\hat{\Theta}\right) = \frac{N+\hat{p}}{N-\hat{p}}\, J_{tr}\left(\hat{\Theta}\right) + f(\Theta^o) \tag{3}$$

where $f(.)$ is an unknown function. Fortunately, since we have to compute $J_{\mathrm{val}}(\hat{\Theta}+\delta\Theta) - J_{\mathrm{val}}(\hat{\Theta})$, the dependency of $f(.)$ disappears in the robustness analysis and the variation in generalization performance can be expressed as

$$\delta J = J_{tr}\left(\hat{\Theta}+\delta\Theta\right)\frac{N+\hat{p}+\delta p}{N-\hat{p}-\delta p} - J_{tr}\left(\hat{\Theta}\right)\frac{N+\hat{p}}{N-\hat{p}}. \tag{4}$$

δp models the possible variation in rank induced by the perturbation. By expanding with Taylor $J_{tr}(\hat{\Theta}+\delta\Theta)$ around $\hat{\Theta}$ and remembering that the gradient is null (the training procedure ends in a minimum for J_{tr}), we have that $J_{tr}(\hat{\Theta}+\delta\Theta) = J_{tr}(\hat{\Theta}) + (1/2)\delta\Theta^T J''_{tr}(\hat{\Theta})\delta\Theta$, which, inserted in the above, provides

$$\delta J = \frac{1}{2}\,\delta\Theta^T J''_{tr}\left(\hat{\Theta}\right)\delta\Theta\,\frac{N+\hat{p}+\delta p}{N-\hat{p}-\delta p} + \frac{\hat{\sigma}^2\delta p}{N-\hat{p}-\delta p} \tag{5}$$

where $\hat{\sigma}^2$ is an estimate of the noise variance (e.g., see [10]). We observe that if $\delta p < 0$ and $\delta J < 0$, the perturbation improves the performance of the model. This comment can be related to the Principal Component Pruning technique suggested in [11] and constitutes the bases for other pruning techniques such as Optimal Brain Damage [12] and Surgeon [13]. Since these perturbations improve the generalization ability of the model they should be always considered at the model optimization level. Without loss of generality we can, therefore, assume that model M_0 has been correctly dimensioned and, hence, that the probability of having a continuous perturbation modifying δp is null. We, therefore, have that $\delta p = 0$ and the (5) becomes

$$\delta J = \frac{1}{2}\,\delta\Theta^T J''_{tr}\left(\hat{\Theta}\right)\delta\Theta\,\frac{N+\hat{p}}{N-\hat{p}}. \tag{6}$$

After model optimization the Hessian $H = J''_{tr} = (1/N)\sum_{i=1}^{N} xx^T$ is definite positive by construction and hence, from (6), any perturbation introduces a loss in generalization performance. The facto, the δJ suggested by (6) constitutes an estimate of the generalization performance loss when the linear model is subject by a generic perturbation $\delta\Theta$.

In perturbation analyses we can identify two interesting cases. The worst case perturbation, which addresses the evaluation of the maximum amplification of δJ, and the average case perturbation, which quantifies the average value of δJ.

Worst Case Perturbation Analysis

The worst case perturbation occurs when $\delta\Theta = |\delta\Theta| u_{\lambda\_\max}$, i.e., the perturbation is parallel to the eigenvector associated with the maximum eigenvalue $\lambda_{\max}(H)$ of the Hessian. From (6)

$$\max \delta J = \lambda_{\max}(H)|\delta\Theta|^2\,\frac{1}{2}\,\frac{N+\hat{p}}{N-\hat{p}}. \tag{7}$$

An index for measuring the robustness of model M_0 is therefore

$$R_{\max}(H) = \lambda_{\max}(H)\,\frac{1}{2}\,\frac{N+\hat{p}}{N-\hat{p}}.$$

Average Case Perturbation Analysis

Likewise, we can easily compute the average case perturbation. If we assume that the components of the perturbation vector $\delta\Theta$ are independent and identically distributed with zero mean and variance $\sigma^2_{\delta\Theta}$ we have that

$$\begin{aligned} E_{\delta\Theta}[\delta J] &= \frac{1}{2}\,\frac{N+\hat{p}}{N-\hat{p}}\,\mathrm{tr}\left(\delta\Theta\delta\Theta^T J''_{tr}\left(\hat{\Theta}\right)\right) \\ &= \frac{1}{2}\,\frac{N+\hat{p}}{N-\hat{p}}\,\sigma^2_{\delta\Theta}\sum_{i=1}^{d}\lambda_i \end{aligned} \tag{8}$$

where tr is the trace operator and λ_i the ith eigenvalue of H. A criterion for robustness can be derived by neglecting the dependency of the perturbations

$$R_{\mathrm{avg}}(H) = \frac{1}{2}\,\frac{N+\hat{p}}{N-\hat{p}}\sum_{i=1}^{d}\lambda_i(H).$$

Note that expressions (7) and (8) decouple the performance loss in two contributions: the first refers to the strength of the perturbation (i.e., its magnitude or variance), the second depends only on the application (the eigenvalues).

As an example, consider a general-purpose digital implementation for M_0. As a perturbation source we consider truncation which neglects bits of weight below 2^q. Therefore, the maximum magnitude to be used in (7) is $|\delta\Theta|^2 = d2^{2q}$ while the variance associated with such a perturbation is bounded by $\sigma^2_{\delta\Theta} = 2^{2q}/3$ [1]. The synthesis phase will use such information to dimension the q granting an acceptable loss in performance as done in [1], [4].

Differently, the robustness indexes are affected by the application but they are not function of the perturbation magnitude (which is considered fixed).

III. Robustness Improvement and Structural Redundancy

From the previous section, it is obvious that we would like to minimize $R_{\max}$ and R_{avg} to keep under control the impact of perturbations on the device performance; this can be accomplished with off-line transformations and by exploiting structural redundancy.

A. Off-Line Transformations to Improve the Robustness of Model M_0

Lemma 1: Lossless Transformation: A transformation leading to zero mean inputs does not worsen the worst case perturbation and always improves the average case perturbation.

To prove the lemma we consider $M_0\colon \Theta = [W, b_\zeta]$ where ζ is the $d+1$ dimensional input vector having the nonnull mean vector μ_ζ

and b_ζ is the model bias. Denote by $\delta\Theta = [\delta W, \delta b]$ a generic continuous perturbation. From (6) we have that $\delta J = (1/2)((N+\hat{p})/(N-\hat{p}))(\delta W H_\zeta \delta W^T + \delta^2 b)$. The Hessian can be written as

$$H_\zeta = \frac{1}{N}\sum_{i=1}^{N} \zeta\zeta^T = \frac{1}{N}\sum_{i=1}^{N}(x+\mu_\zeta)(x+\mu_\zeta)^T = \frac{1}{N}\sum_{i=1}^{N} xx^T + \mu_\zeta\mu_\zeta^T = H_x + M$$

with $M = \mu_\zeta\mu_\zeta^T$. By invoking the Weyl theorem [15] we have that

$$\lambda_l(H_x) + \lambda_{\max}(M) \geq \lambda_l(H_\zeta) \geq \lambda_l(H_x) + \lambda_{\min}(M) \text{ holds } \quad \forall l$$

with $\lambda_{\min}(M) = 0$ since M is a semidefinite positive matrix by construction and $\lambda_{\max}(M) = \mathrm{tr}(M)$. In particular, we have that $\lambda_{\max}(H_\zeta) \geq \lambda_{\max}(H_x)$ from which $R_{\max}(H_x) \leq R_{\max}(H_\zeta)$ and the first part of the lemma is proved.

The average case perturbation always improves since the sum of eigenvalues equalizes the trace of M, which is always strictly positive if M differs from the zero matrix. The second part of the lemma follows by noting that $\mathrm{tr}(H_\zeta) = \mathrm{tr}(H_x) + \mathrm{tr}(M) > \mathrm{tr}(H_x)$ from which $R_{\mathrm{avg}}(H_x) \leq R_{\mathrm{avg}}(H_\zeta)$.

The main consequence of the lemma is that a simple off-line transformation, which transforms the input to be zero mean, improves the robustness of M_0. It is obvious that the transformation, which only modifies the bias term of M_0, does not change the generalization performance.

Lemma 2: Lossy Transformation: A tolerated pruning transformation improves the average case perturbation.

If we note that an extended pruning technique implies k connections removal, the Hessian H_x of order d becomes the reduced Hessian $H_{x,k}$ of order d–k. For its nature $H_{x,k} \subset H_x$ and, therefore, by invoking the interlacing-Cauchy theorem [15], which states that the eigenvalues of $H_{x,k}$ are suitably interleaved with those of H_x, the thesis follows.

B. Structural Redundancy Techniques to Improve the Robustness of Model M_0

The section investigates the possibility of improving the robustness of M_0 by considering a structural redundancy scheme which redistributes the information content of the application over more complex models. The class of models we consider for structural redundancy is the class of linear neural networks $\mathbf{M}$. By neglecting the bias contribution term, the kth model of $\mathbf{M}$ is characterized by $(1+d)k$ weight and, hence, with respect to M_0 (d weights), it possesses a potential structural redundancy. Denote by $\overline{\theta}$ the weight vector between hidden units and output unit and with $\tilde{\theta}_i$ the generic weight vector between the ith hidden unit and the input ones.

Lemma 3: Sufficient Conditions for Structural Redundancy:

- *Local improvement in robustness* a generic model M_k is locally more robust than M_0 with respect to generic perturbations of magnitude $|\delta\Theta|$ affecting:
  1) the $\overline{\theta}$ coefficients if $\|\tilde{\theta}\|_2 < 1$;
  2) the $\tilde{\theta}_i$ coefficients if $\overline{\theta}_i < 1$.

  The robustness index $R_{\max}$ improves when $\|\tilde{\theta}\|_2$ and $\overline{\theta}_i$ get closer to zero.
- *Global improvement in robustness*
  3) Given model M_0 of weights W and chosen an arbitrary value $\alpha < 1$, any model M_k with $k > |W|^2/\alpha^2$ grants $R_{\max}(H(M_k)) \leq R_{\max}(H(M_0))$ for any perturbation affecting the weights of generic linear neuron. Model M_k can be constructed by setting $\overline{\theta}_i = \alpha, \forall i = 1, k$ and $\tilde{\theta}_i = W/k\alpha, \forall i = 1, k$.

The detailed proof of the theorem is given in [19], here we provide only a sketch of it. In can be shown that δJ of (6) can be transformed in a canonical form in which the Hessian is split in matrices either dependent on $\tilde{\theta}$ or $\overline{\theta}$. It is then studied the effect of the perturbations on $\overline{\theta}$ (point 1 of the Lemma) and $\tilde{\theta}$ (point 2 of the Lemma). In the former case, by bounding δJ associated with M_k we obtain that $R_{\max}(H(M_k)) \leq \|\theta\|_2^2 R_{\max}(H(M_0))$: by requiring $\|\tilde{\theta}\|_2 < 1$ point 1 one the lemma is proved.

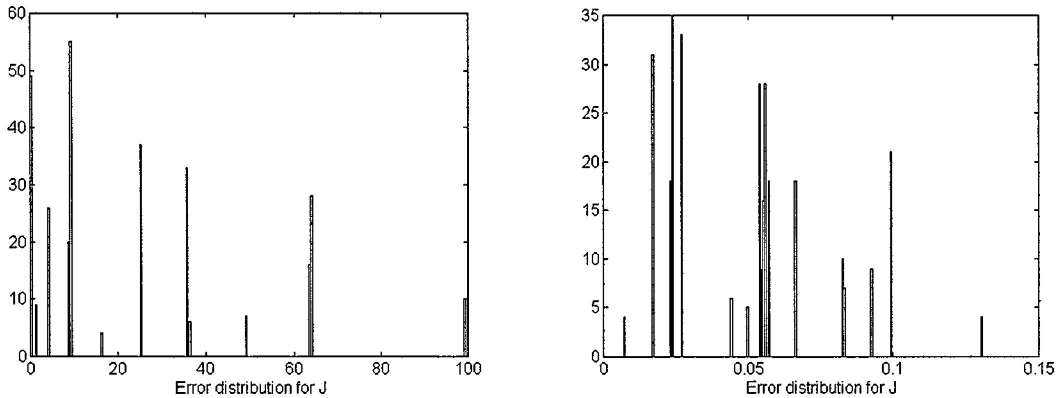
Similarly, in the latter case, we obtain that $R_{\max}(H(M_k)) \leq \overline{\theta}_1^2 R_{\max}(H(M_0))$ from which the thesis follows by requiring $\overline{\theta}_i < 1$. To prove the third point of the lemma we constrain model M_k to satisfy point 1 and 2, subject to the additional linear constraints $W = \overline{\theta}\tilde{\theta}$. Without loss of generality, by setting all coefficient of $\overline{\theta}$ to α and considering the Moore–Penrose pseudoinverse $\overline{\theta}^+$ we have that $\tilde{\theta} = \overline{\theta}^+ W$ from which the $\tilde{\theta}$ matrix is composed of identical rows of value $W/k\alpha$. Finally, by requiring the spectral radius $\|\tilde{\theta}\|_2 = |W|/\alpha\sqrt{k}$ to be smaller than one we obtain the minimum k granting the improvement in robustness.

The lemma shows that by spanning the hierarchy we can both improve locally and globally the robustness of the model with respect to the worst case perturbation case by considering perturbations affecting either $\overline{\theta}$ or $\tilde{\theta}_i$. Note that global robustness requires all neurons composing the linear neural network to be more robust than model M_0. In particular, lemma 3 states that the improvement is achievable with an arbitrary robustness value by acting on α and selecting an appropriate model M_k. To show the dependency in α, from the proof of Lemma 3 we have that $R_{\max}(H(M_k)) \leq \alpha^2 R_{\max}(H(M_0))$ for perturbations affecting $\tilde{\theta}_i$ and $R_{\max}(H(M_k)) \leq (|W|^2/k\alpha^2)R_{\max}(H(M_0))$ for perturbations affecting $\overline{\theta}$. We should compare the gain in robustness η obtained by considering model M_k instead of model M_0 with the increase in model complexity. From the above two relationships we have that, once suitably selected k, $R_{\max}(H(M_k)) \leq \eta R_{\max}(H(M_0))$ holds for a generic perturbation affecting a generic linear neuron of M_k with $\eta = \max(\alpha^2, |W|^2/k\alpha^2)$. Computational complexity C, here defined as the number of parameters of the model (and hence related to the number of multiplications and additions to be implemented) increases from the d weights of M_0 to the $n = k(d+1)$ of M_k. From lemma 3 $n \geq (|W|^2/\alpha^2)(d+1)$ and, therefore, $C(M_k) \geq (|W|^2/\alpha^2)((d+1)/d)C(M_0)$. Both the robustness gain and model complexity scale quadratically with α^2.

From the theory point of view, the result is surely appreciable by itself. Nevertheless, we have to compare the computational cost of structural redundancy with the one requested to implement a classic fault tolerance redundancy architecture. In such a case, by neglecting low order contributions, model M_k would require a complexity roughly k times that of M_0. We could have used instead the k units M_0 to implement a classic k-ary modular redundancy scheme based on a replica mechanism for M_0 and a voting unit at the end. Such a solution supports a $k-1$ error detection and correction of $k-2$ errors [16], [17]; here errors are induced by perturbations affecting the redundant units. With this schema we can provide a correct result whereas information distribution over model M_k always introduces an error (even if it can be made arbitrarily small). After these observations, Lemma 3 implicitly states that the improvement in robustness with spatial redundancy over $\mathbf{M}$ is too costly from the performance/computational point of view.

IV. Case Study: Constructing a Robust Convolution Module

In this section, we consider the linear "peak detection" filter $y = 0.25\zeta(t-2) + 1.25\zeta(t-1) + 0.25\zeta(t) - 1$. $N = 500$ inputs have been extracted from a nonzero mean Gaussian



(a) (b)

Fig. 1. (a) The original filter. (b) The transformed filter.

distribution $\zeta = N(4, 0.04)$. The eigenvalues of the Hessian are $eigen(H_\zeta) = [3.7868, 3.7278, 52.9061]$ and then $R_{\max}(H_\zeta) = 26.7724$ and $R_{\rm avg}(H_\zeta) = 30.5751$.

By applying Lemma 1, and hence transformation $\zeta = x + 4$, we have zero mean inputs [i.e., $x = N(0, 0.04)$] and the new filter becomes $y = 0.25x(t-2) + 1.25x(t-1) + 0.25x(t) + 6$. The new eigenvalues are $eigen(H_x) = [3.7834, 3.7259, 3.9388]$ with $R_{\max}(H_x) = 1.9932$ and $R_{\rm avg}(H_x) = 5.7931$. It is immediate to observe that the transformation significantly reduced the impact of the worst and the average perturbation cases.

To experimentally test the impact of physical perturbations on $J_{\rm val}$ we represented the coefficients of the filter in a fixed point truncation-based notation. We then applied perturbations affecting a randomly chosen bit for each coefficient of the filter (there are therefore 3 simultaneous faults within the filter). We then tested the impact of errors on $J_{\rm val}$ before and after the transformation on the same fault set. The histograms of the induced variation in $J_{\rm val}$ are given in Fig. 1(a) for the original filter and in Fig. 1(b) for the transformed one. We immediately see that after the transformation, the new filter is significantly more robust since the effect of the generalization ability $J_{\rm val}$ is reduced.

V. Conclusions

The paper investigates the robustness issue in linear models whose coefficients have been identified from a set of measured data. It is shown that the difficult problem of considering the generalization error as loss function can be tackled by following the statistical approach leading to the network information criterion and final prediction error criteria. Off-line transformations can be derived which improve the robustness of the computation. An additional gain in robustness can be achieved by considering more complex linear models implementing a sort of structurally redundancy. Despite the achievable gain in robustness, the computational complexity is not justified when compared with performance obtainable by a classic n-ary modular redundancy scheme requiring the same resources.

Acknowledgment

The author wishes to thank the reviewers, whose comments and hints significantly helped to improve the manuscript.

References

[1] J. Holt and J. Hwang, "Finite precision error analysis of neural network hardware implementations," *IEEE Trans. Comput.*, vol. 42, pp. 281–290, Mar. 1993.

[2] M. Stevenson, R. Winter, and B. Widrow, "Sensitivity of feedforward neural networks to weights errors," *IEEE Trans. Neural Networks*, vol. 1, pp. 71–80, Mar. 1990.

[3] S. Piché, "The selection of weights accuracies for madalines," *IEEE Trans. Neural Networks*, vol. 6, pp. 432–445, Mar. 1995.

[4] C. Alippi and L. Briozzo, "Accuracy vs. precision in digital VLSI architectures for signal processing," *IEEE Trans. Comput.*, vol. 47, pp. 472–477, Apr. 1998.

[5] C. Alippi, V. Piuri, and M. Sami, "Sensitivity to errors in artificial neural networks: A behavioral approach," *IEEE Trans. Circuits Syst. I*, vol. 42, pp. 358–361, June 1995.

[6] G. Dundar and K. Rose, "The effects of quantization on multilayer neural networks," *IEEE Trans. Neural Networks*, vol. 6, pp. 1446–1451, Nov. 1995.

[7] L. Ljung, *System Identification, Theory for the User*. Englewood Cliffs, NJ: Prentice-Hall, 1987.

[8] M. H. Hassoun, *Fundamentals of Artificial Neural Networks*. Cambridge, MA: MIT Press, 1995.

[9] N. Murata, S. Yoshizawa, and S. Amari, "Network information criterion—Determining the number of hidden units for an artificial neural network model," *IEEE Trans. Neural Networks*, vol. 5, pp. 865–872, Nov. 1994.

[10] C. Alippi, "FPE-based criteria to dimension feedforward neural topologies," *IEEE Trans. Circuits Syst. I*, vol. 46, pp. 962–973, Aug. 1999.

[11] A. Levin, T. Leen, and J. Moody, "Fast pruning using principal components," in *Proc. NIPS6*, vol. 6, Dec. 1994, pp. 35–42.

[12] Y. Le Cun, J. S. Denker, and S. A. Solla, "Optimal brain damage," in *Proc. NIPS2*, vol. 2, Feb. 1990, pp. 598–605.

[13] B. Hassibi and D. G. Stork, "Second order derivative for network pruning: Optimal brain surgeon," in *Proc. NIPS5*, vol. 2, Jan. 1993, pp. 164–173.

[14] W. K. Pratt, *Digital Image Processing*. New York: Wiley Interscience, 1978.

[15] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge, U.K.: Cambridge Univ. Press, , 1985.

[16] D. K. Pradhan, *Fault Tolerant Computing, Theory and Techniques*. Englewood Cliffs, NJ: Prentice Hall, 1986, vol. I and II.

[17] R. Negrini, M. Sami, and R. Stefanelli, *Fault Tolerance Through Reconfiguration in VLSI and WSI Array*. Cambridge, MA: MIT Press, 1989.

[18] G. W. Stewart and J. Sun, *Matrix Perturbation Theory*. New York: Academic, 1990.

[19] C. Alippi, "A parameters sensitivity analysis in identified linear systems," Dip. Elettronica e informazione, Politecnico di Milano, Milan, Italy, Internal Rep., 2001.