LINEAR ERROR MODELING OF ANALOG AND MIXED-SIGNAL DEVICES

Gerard N. Stenbakken and T. Michael Souders

National Institute of Standards and Technology
Gaithersburg, MD 20899
(301) 975-2406

Abstract

Techniques are presented for developing linear error models for analog and mixed-signal devices. Methods for choosing parameters and assuring the models are complete and well-conditioned, are included. Once established, the models can be used in a comprehensive approach for optimizing the testing of the subject devices.

1. Introduction

In a previous paper given at the 1990 International Test Conference, the authors presented a comprehensive approach for optimizing the testing of analog and mixed-signal devices [1]. The entire process described in that paper is carried out by performing algebraic operations on appropriate models of the devices to be tested. It was assumed in that paper that accurate models were available. In this paper, we address the development of the required models, and discuss their properties.

The approach described in [1] is based on an m × n linear coefficient matrix model, **A**, that relates the device's error response, **y**, (at all m candidate test conditions), to a set of n underlying variables, **x**:

$$\mathbf{y} = \mathbf{A}\,\mathbf{x} \qquad (1)$$

The error response vector, **y**, represents the actual device's deviation from the ideal response at each test condition. (If, for example, the device is a D/A converter, **y** might be the vector of output voltage deviations from the ideal, for each of the $m = 2^N$ code states, where N is the number of bits.)

As shown in [1], once an accurate model has been developed, algebraic operations on the model can be used to:

1. select an optimum set of test points which will minimize the test effort and maximize the test confidence,
2. estimate the parameters of the model (i.e., the error variables) from measurements made at the selected test points,
3. predict the response of the device at all candidate test points from measurements made at the selected test points, and
4. calculate the accuracy of the parameter estimates and response predictions, based on the random error in the measurement.

2. Model Development

A model serves two general purposes: It parameterizes the device response in terms of a set of n error parameters represented by **x**, and it provides the capability of accurately predicting the response for any test condition,

once estimates are available for the parameters. These two principles, parameterization and completeness, are the primary considerations when developing a model. Parameterization refers to the choice of basis functions that are taken to underlie the device's response, and completeness refers to the accuracy with which the basis can explain the response of any device for which the model is intended, i.e., the degree to which the selected basis spans the space of possible responses, **y**. The choice of parameters depends on the goals of the testing. If, for example, the goal is to identify and trim out-of-tolerance components based on functional tests of the devices, then the error signatures of the components to be trimmed should be chosen as model vectors. On the other hand, if there are no trimmable components and the primary goal is to correctly sort good and bad devices with the fewest required measurements, then it is desirable to parameterize the response with the most efficient basis, i.e., the smallest set of vectors which is still complete in the sense that it gives accurate predictions.

In a model, each column vector represents the error signature of a particular component or parameter of interest. Visualization often affords a better intuitive understanding of the underlying error processes and their contributions to the overall errors of the devices under test. Therefore, when developing a model, it is often quite useful to visualize candidate column vectors by plotting them vs. test condition index. For this reason, we will sometimes use pictorial matrix representations as in fig. 1, and will illustrate certain column vectors graphically.

2.1. Parameterization (choosing a candidate basis)

There are three basic approaches to parameterization: physical, *a priori*, and empirical.

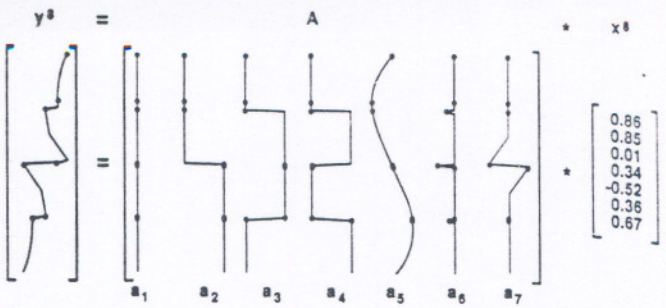

Fig. 1. Pictorial representation of Eq. 1 for a fictitious device. The dots represent locations of selected test points. Each vector is plotted vertically as magnitude vs. test condition index.

Physical Models

Physical models are developed when the parameters are known from design and correspond to conventional modeling parameters. For example, if an accurate equivalent circuit is known (including the connection matrix and nominal component values as in a SPICE model), a physical model can be computed as the normalized partial derivatives or sensitivities of the output response of the device with respect to the component parameters, evaluated at their nominal values:

$$\mathbf{A} = \begin{bmatrix} \frac{\partial Y_1}{\partial p_1}\cdot p_1 & \frac{\partial Y_1}{\partial p_2}\cdot p_2 & \cdots & \frac{\partial Y_1}{\partial p_n}\cdot p_n \\ \frac{\partial Y_2}{\partial p_1}\cdot p_1 & \frac{\partial Y_2}{\partial p_2}\cdot p_2 & \cdots & \frac{\partial Y_2}{\partial p_n}\cdot p_n \\ \vdots & \vdots & & \vdots \\ \frac{\partial Y_m}{\partial p_1}\cdot p_1 & \frac{\partial Y_2}{\partial p_2}\cdot p_2 & \cdots & \frac{\partial Y_m}{\partial p_n}\cdot p_n \end{bmatrix} \qquad (2)$$

Such models correspond to first order Taylor expansions of the circuit's response. For linear systems, the adjoint network approach provides a computationally efficient method of computing the required sensitivities [2]. Time-domain sensitivities can be computed efficiently for specific input waveforms using a modified nodal formulation and numerical integration techniques [3]. Versions of public domain as well as commercial software are currently available for computing these sensitivity

matrices. SPICE 3C for example, has this capability. Another technique for determining the sensitivity matrix is to experimentally perturb each component in turn by a small amount, and record the resulting change in output response for each test condition. While this is usually not feasible to perform for all of the components, it might be a reasonable choice for components that are trimmable and for which physical model vectors are important.

The strength of physical models lies in the direct correspondence between the model variables and measurable physical parameters, be it resistance, capacitance, transistor transconductance, open loop gain, etc. For this reason, physical models are useful in fault diagnosis and in alignment and trimming operations. Since the actual values of the parameters subsequently can be estimated from the test data, the results can be used to compute the amount of trimming that will be required to globally optimize the devices' performance.

Physical models, nevertheless, are not without problems. For complex devices, it becomes more and more difficult to obtain accurate circuit models, and the computational burdens they entail may be excessive. In some cases, detailed design knowledge of the device may simply be unavailable to the test engineer. It is often inadequate to model only the components which are intentionally designed into a circuit; the parasitic reactances, interconnection resistances, etc., must also be included, as well as dielectric absorption, self-heating or other nonideal effects which contribute to the overall response of the device. In other cases, a first order Taylor's approximation may be insufficient. This can occur when the devices' behavior has a nonlinear dependence on its components and the model parameters deviate substantially from their nominal values. It is possible to represent a higher order Taylor expansion with nonlinear terms as a linear model in higher dimensional space, but this quickly can become computationally intractable. Finally, the sensitivity vectors of different components in the same circuit are often identical, so that the column vectors of a physical model will rarely all be linearly independent. The existence of these "ambiguity groups" obscures the components and causes the sensitivity matrix to be rank deficient [4]. In such cases, the matrix must be reduced to full column rank by column pruning, as will be explained in a later section.

A Priori Models

A priori models are those that use vectors chosen to represent the device based on formal mathematical systems and engineering judgement. Often, they are comprised of a relatively small subset of basis functions that is capable of approximating the device performance to the required accuracy. For example, subsets of the Walsh functions have been used for modeling some A/D and D/A converter architectures [5]. As illustrated in fig. 2, the Rademacher subset of Walsh functions, square wave functions having powers of 2 periods over the defined interval, are an ideal representation of the "bit errors" of conventional D/A converter architectures using binary-weighted current or voltage sources, as well as of successive approximation A/D converters. These functions are not only linearly independent but are orthogonal, and they can represent any integral linearity errors that are due to weighting errors of the individual bits. However, for many converters the Rademacher functions are not complete in the sense that they cannot represent superposition errors; therefore, they must be augmented with other vectors to form a complete basis when superposition errors are present.

In other cases, *a priori* models afford a simple way to incorporate the benefits of engineering judgement into a model, minimizing the amount of empirical modeling that must be done. In fig. 3, five such examples are given.

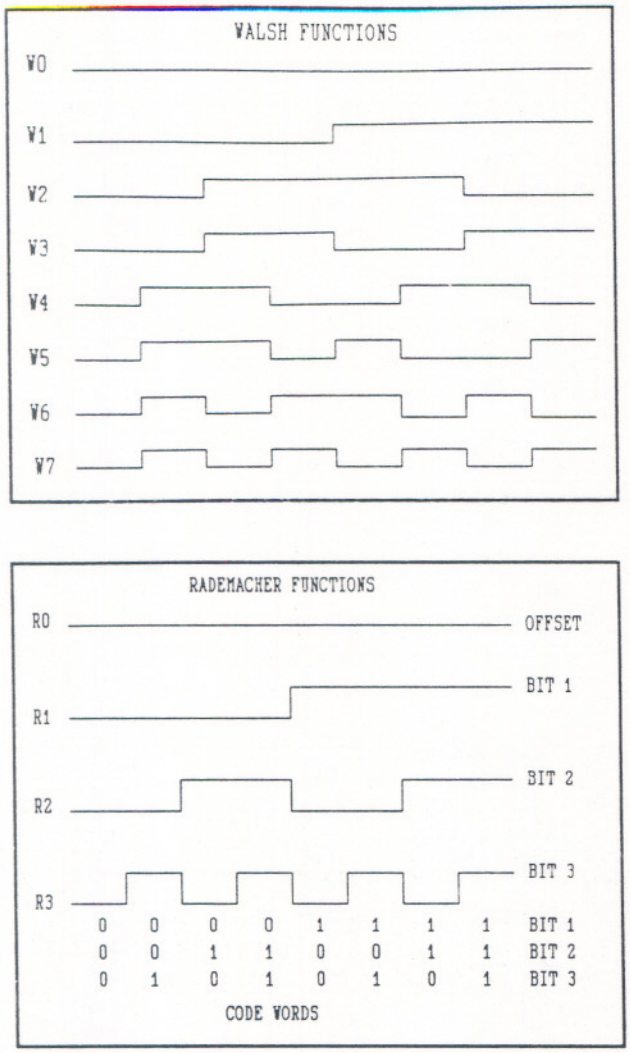

Fig. 2. The first eight Walsh functions (top) and the first four Rademacher functions (bottom) which are a subset of the Walsh functions. The correspondence of the Rademacher functions to the bit error signatures of data converters is illustrated.

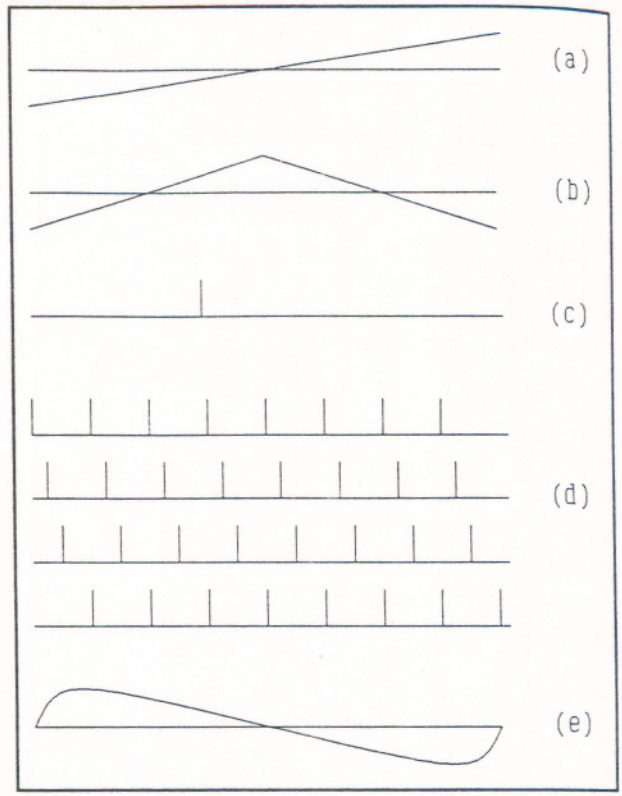

Fig. 3. Examples of a priori model vectors representing bipolar gain errors (a,b), glitches (c), repeating error patterns (d), and the error signature of device self-heating (e). The vertical scales are in arbitrary units and the horizontal scale is the candidate test condition index.

The vectors of fig. 3 (a) and (b) are useful in describing the gain errors of bipolar devices when there are different gain errors for each polarity. In (c), a vector with a single non-zero element (a Haar function) can be used to force a measurement at a specific point, e.g., where glitches are likely to occur. To efficiently incorporate the possibility of a repeating error pattern, comb functions as illustrated in (d) are useful. This example can represent any four-element pattern that repeats eight times across the range of candidate test points. Measurements at only four test points are necessary to determine the complete pattern. Finally, fig. 3 (e) illustrates a vector that has been generated analytically from physical reasoning, in this case representing the effects of device self-heating.

Empirical Models

Empirical models are learning-based, and are especially attractive for performance testing applications. They require no detailed design knowledge of the device, nor are they strictly limited to linear dependence on the variables. These models are obtained by numerically analyzing the data from exhaustive testing of

representative units coming off the production line. They are based on the premise that a selected lot of devices will manifest all of the degrees of freedom or variability of the manufacturing process. The response of each of the devices will vary somewhat from the others since the values of the underlying variables differ from unit to unit. (If this were not the case, then testing would be unnecessary.) The candidate set of basis vectors for the model is formed from k complete m-point response vectors that have been measured during exhaustive tests of k devices, where k is a statistically significant sample. Each column vector of the candidate basis, **A'**, is one of the k response vectors, as follows, where the superscripts designate the specific device, and the subscripts designate the test point index:

$$\mathbf{A}' = \begin{bmatrix} y^1 & y^2 & \cdots & y^k \end{bmatrix} = \begin{bmatrix} y_1^1 & y_1^2 & \cdots & y_1^k \\ y_2^1 & y_2^2 & \cdots & y_2^k \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ \cdot & \cdot & & \cdot \\ y_m^1 & y_m^2 & \cdots & y_m^k \end{bmatrix} \qquad (3)$$

If enough device responses have been included in **A'**, they will form a basis that spans the entire space of possible **y** vectors. On the other hand, such a candidate basis will almost certainly be rank deficient and will require column pruning to reduce the model to full column rank, as described in a later section.

This empirical approach not only eliminates the need for detailed design knowledge of the device under test, but it also minimizes the number of variables that are actually required. For example, the sensitivity vectors of many components of a device will be negligibly small and therefore need not be considered. Some other components may have vectors that are identical to each other as explained above, such as the components of cascaded gain stages. Still other components can have vectors that are different, but always track each other. This is common in integrated circuits where a single processing variable such as dopant level or exposure time during metalization will affect many components equally, causing them all to vary in fixed proportion. Similarly, small mask misalignments will affect many components in some proportional way. In these cases, the parameters of the many affected components are no longer uncorrelated, but are instead correlated by a few underlying process variables. The empirical modeling process is only sensitive to these functional process variables.

One disadvantage associated with empirical models is the noise that they incorporate. Since the vectors are derived from measurement data, they include noise as well as true model information, and the variance of predictions made from the model will necessarily increase. The effects can be minimized by minimizing the amount of measurement noise included in the empirical vectors. This can be accomplished by averaging repeated measurements of each candidate vector that is to be used in the empirical model.

Mixed Models

It is also possible, and often desirable, to combine the modeling approaches, physical, *a priori*, and empirical, to achieve the best features of each. This is accomplished by starting with the physical parameters that are known to be important and are perhaps trimmable. It is often rather simple to compute the sensitivity to these parameters, even when the overall circuit is unknown. Next, the sensitivity matrix comprising this physical model is augmented with *a priori* and empirical vectors. The result will be a model that can be useful not only for making accurate predictions, but also for estimating the actual values of critical parameters that can be trimmed to achieve compliance.

2.2. Column Pruning

Column pruning is a process for selecting a set of linearly independent column vectors from a larger candidate set of vectors. In general, there will be many such sets that are possible; however, what is most desirable is to find a set that is maximally independent so that the effects of random measurement noise are minimized when the model is used. The QR decomposition with column pivoting is well suited for this task and is available as a simple call in many linear algebra software packages [6,7,8]. When applied to the matrix of candidate vectors, **A'**, the QRD operation searches the candidate set, selects the vector with largest norm and pivots it to the first position, then orthogonalizes all remaining vectors to it. Next, the k-1 remaining orthogonalized vectors are searched, the one having the largest norm is pivoted into the second position, and the k-2 remaining vectors are orthogonalized to it (note that they are already orthogonal to the first). The process is repeated until all of the vectors have been ordered and orthogonalized. The process outputs the pivot vector which designates the ordering, and forms two factors, **Q** and **R**, of the original matrix:

$$\underset{m\times n}{\mathbf{A}} = \underset{m\times m}{\mathbf{Q}}\ \underset{m\times n}{\mathbf{R}} \qquad (4)$$

where **Q** is orthonormal and **R** is a right triangular matrix. The jth diagonal element of the triangular matrix, R_{jj}, is the norm of the jth orthogonalized column, after pivoting.

$$\mathbf{R} = \begin{bmatrix} r_{11} & r_{12} & \cdot & \cdot & \cdot & r_{1n} \\ & r_{22} & \cdot & & & r_{2n} \\ & & \cdot & & & \cdot \\ & & & \cdot & & \cdot \\ & & & & \cdot & \cdot \\ & & & & & r_{nn} \\ & & & 0 & & \end{bmatrix} \qquad (5)$$

As less and less independent information is contained in successive columns, the norms become smaller and smaller, until (in the case of empirical vectors) they are dominated by measurement noise. Beyond a given point, the inclusion of additional vectors reduces the efficiency of the model, i.e., requires a larger number of vectors, while providing negligible improvement in the model's ability to predict the device's performance. A rule of thumb for when to stop adding vectors can be quantified if the standard deviation, σ, of the measurement noise is known. The (j+1)th and successive vectors can be dropped from the model if

$$m^{-\frac{1}{2}} r_{j+1,j+1} < 1.6\,\sigma \qquad (6)$$

This relationship was determined empirically through computer simulations; an analytic derivation on a bound for how much error remains in the prediction is the subject of current research. The term $m^{-\frac{1}{2}}$ in Eq. 6 converts the norm (root-**sum**-squares) to the root-**mean**-square measure used for σ.

Once the cutoff point, j, has been determined, the matrix **A** is formed by selecting the j vectors of **A'** that correspond to the first j pivot locations indicated by the pivot vector.

It should be borne in mind that determining the correct cutoff point does not guarantee that the remaining (unselected) vectors contain no significant information, but only that the information cannot be used because it is too corrupted with measurement noise. Tests for completeness described in a later section should be used to determine the accuracy of the selected model and the amount of unmodeled information that remains.

2.3. Orthogonalizing

When mixed modeling is used, it becomes important to force certain vectors, either sensitivity or *a priori*, to be selected in the column pruning process. With some software packages, notably LINPACK, this is easily accomplished since the programmer can designate a forced pivot. However, such

flexibility is not available in many other common linear algebra software packages, so the following approach should be used with these. Partition the candidate matrix A' into A_1 which contains the vectors which must be selected, and A_2, which contains the remaining vectors. Orthogonalize the vectors of A_2 with respect to each of those of A_1. An orthogonalized vector, a_j', is formed when a_j is orthogonalized with respect to vector a_i, as follows:

$$a_j' = a_j - \frac{a_i (a_i^T a_j)}{(a_i^T a_i)} \qquad (7)$$

where superscript T designates transpose, and $a_i^T a_j$ is the scalar product of a_i and a_j.

When the orthogonalization is completed, the two subspaces, A_1 and A_2', will be orthogonal. The new candidate matrix, $[A_1 A_2']$, can then be column pruned as above, with the assurance that all of the vectors of A_1 will be selected, provided that they are themselves all linearly independent, and have norms large enough to pass the selection test.

2.4. Testing Completeness

Once a model has been column pruned, it should be tested for completeness. This can be done only by applying the model to another sample of devices that have been exhaustively tested, and checking the goodness of fit. The first step in this process (for the kth device) is to estimate the parameter vector, x^k, by fitting the model to the test data:

$$\hat{x}^k = (A^T A)^{-1} A^T y^k \qquad (8)$$

Next, the residuals of the fit may be computed:

$$r^k = y^k - A \hat{x}^k \qquad (9)$$

The vector of residuals can be examined for randomness and rms value. A model that is complete should produce residuals that are randomly distributed and whose rms value is no greater than the rms measurement noise. If the residuals are significantly larger than this value, then the model can undergo further improvement by the addition of more vectors. It may only be possible to do this efficiently, i.e., with a relatively small number of additional vectors, by using additional averaging to reduce the noise components in additional candidate vectors. Ultimately, the decision to augment the model is an economic one: if the residuals are greater than the random noise but still within the uncertainty limits that are desired for the testing process, then the addition of more vectors may not be considered worth the additional effort.

The completeness of a model can and should also be checked on-line to assure that it tracks possible changes in the manufacturing process. As described in [1], the procedure is very similar to that just outlined, but only measurement data at the selected test points is used.

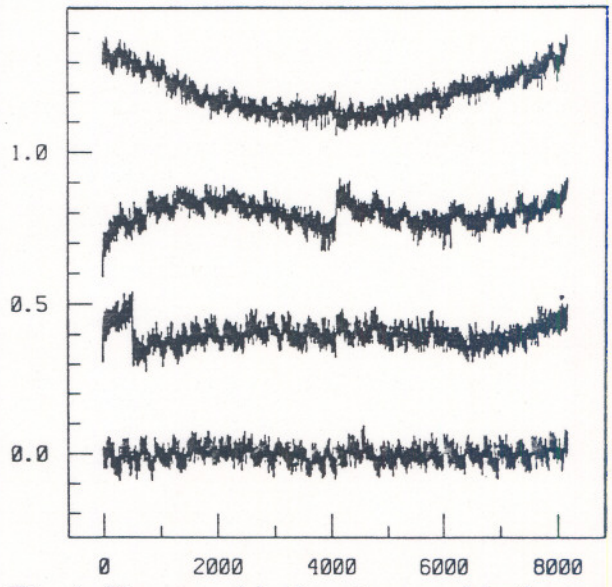

Fig. 4. Four empirical vectors used in the model for the A/D converter example. The vertical scale is LSB's and the horizontal scale is test codes.

3. Example

In [1], an example was given of tests made on

a set of 13-bit A/D converters, using testing strategies based on a model having the form given in Eq. 1. The model used for these converters consists of 18 column vectors of 2^{13} elements each. The first 14 vectors are *a priori* and include the Rademacher functions (fig. 2) corresponding to an offset term and the 13 bit errors. The remaining four vectors were obtained empirically from test data taken on a representative set of 50 devices. Fig. 4 shows these vectors after they have been orthogonalized with respect to the preceding 14 as described in the "Orthogonalizing" section. It can be seen that the first three vectors include substantial structure as well as some measurement noise. In this case the measurement noise had an rms value of 0.021 least significant bits (LSB's). The plot in fig. 5 is of the normalized rms values, $m^{-\frac{1}{2}}r_{j,j}/\sigma$, of the 15th through the 22nd candidate empirical vectors. (One vector beyond that designated by the cutoff criterion of Eq. 6 was included in this model but the effect of this single addition is negligible.)

Fig. 6 presents the test results obtained when the model was used to predict, from a small number (64) of measurements, the largest integral nonlinearity (INL) for each of 77 additional devices. The figure shows a histogram of predicted values vs. measured INL.

4. Conclusions

It can be shown that there is always a linear model of the form given in Eq. 1 that will be complete for a given device and candidate test space; however, there is no guarantee that the model will be efficient. To be efficient, the number of parameters, n, should be small with respect to the number of candidate test points, m. Sometimes, lack of efficiency is simply due to the fact that there are a large number of independent parameters underlying the manufacturing process - the modeler usually

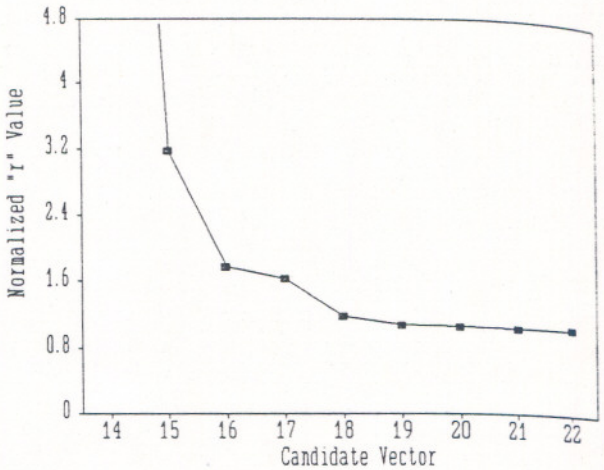

Fig. 5. Plot of normalized "r" values, $m^{-\frac{1}{2}}r_{j,j}/\sigma$, for candidate vectors 14-22. The 14th vector is the 14th Rademacher function; the remaining vectors are empirical.

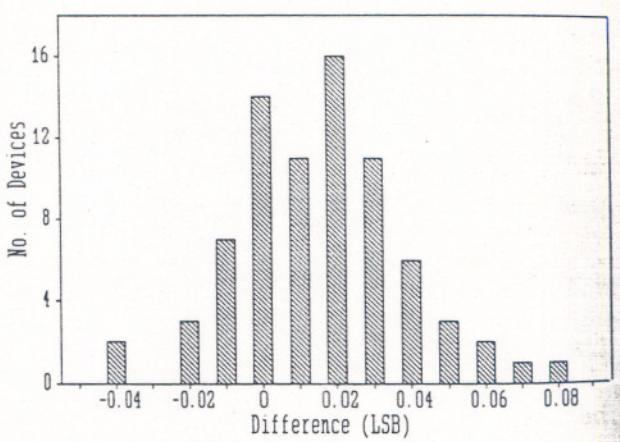

Fig. 6. Test results for the A/D converter example, showing a histogram of differences between the measured and predicted maximum INL for 77 devices tested. The predicted INL values were based on the model, using measurements taken at 64 (out of 8192) test codes.

nonlinear dependence on at least some of the parameters, so that a higher dimensional space is required for an adequate description. In such cases, empirical modeling will find the linear

References

1. T. M. Souders and G. N. Stenbakken, "A Comprehensive Approach for Modeling and Testing Analog and Mixed-Signal Devices," Proceedings, 1990 International Test Conference, IEEE Computer Society Press, Sept. 1990.

2. J. Vlach and K. Signhal, Computer Methods for Circuit Analysis And Design, Van Nostrand Reinhold Co., New York, NY, 1983.

3. H. Dai and T. M. Souders, "Time-Domain Testing Strategies and Fault Diagnosis for Analog Systems," IEEE Trans. Instrum. Meas., Vol. 39, No. 1, February, 1990.

4. G. N. Stenbakken, T. M. Souders and G. W. Stewart, "Ambiguity Groups and Testability," IEEE Trans. Instrum. Meas., Vol. 38, Oct. 1989.

5. T. M. Souders and D. R. Flach, "An NBS Calibration Service for A/D and D/A Converters," Proceedings 1981 International Test Conference, IEEE Computer Society Press, Oct. 1981.

6.* J. J. Dongarra, et. al., LINPACK User's Guide, Society for Industrial and Applied Mathematics, Philadelphia, 1979.

7.* W. D. Gropp, D. E. Foulser, and S. Chang, "CLAM™ User's Guide," Scientific Computing Associates, Inc., New Haven, CT 06510, 1989.

8.* C. Moler, "MATLAB Users' Guide," University of New Mexico Tech. Report CS81-1, 1982.

* *Certain commercial software products are referenced for the convenience of the reader. In no case does such identification imply recommendation or endorsement by the National Institute of Standards and Technology, nor does it imply that the products identified are necessarily the best available for the purpose.*