Interpolative Modeling of GaAs FET S-parameter Data Bases for Use in Monte Carlo Simulations

L. Campbell and J. Purviance
NASA SERC for VLSI System Design
University of Idaho, Moscow, Idaho 83843

Abstract - **A statistical interpolation technique is presented for modeling GaAs FET S-parameter measurements for use in statistical analysis and design of circuits. This is accomplished by interpolating among the measurements in a GaAs FET S-parameter data base in a statistically valid manner.**

1 Introduction

Statistical analysis and design of high frequency GaAs circuits requires accurate statistical models of the variation of the GaAs FETs' performance. In this paper we develop a method for modeling a GaAs FET S-parameter data base that is concise, efficient, accurate, and which can generate a simulated data base which is statistically indistinguishable from a measured data base. Two sets of data samples will be said to be indistinguishable if their statistical properties do not differ. This goal is met by introducing and developing the statistical interpolation model which was first presented in this context in Campbell's dissertation [4, 5]. The term "statistical interpolation model" was used there to refer to the density estimation techniques used in this work. These density estimation techniques are based on kernel density estimation and data clustering. "Interpolative model" will be used from here on as a shortened form for statistical interpolation model. The interpolative model is developed here for the purpose of modeling probability density functions (PDFs) for use in statistical modeling of GaAs FETs. A probability density function is defined in Definition 1. The weighted sum of two or more PDFs is also a PDF.

Definition 1 *Probability Density Function.*

$$f(x) \ni: \int_{-\infty}^{+\infty} f(x)dx = 1, f(x) \geq 0 \; \forall \; x$$

2 Modeling Assumptions

As we have already stated above, our objective is to create a sample data base that has the same statistics as a measured data base. To put this in more precise terms, we must find the statistical distribution (PDF) of the population from which the measured samples were taken. Such a PDF cannot simply be directly calculated. There are an infinite number of possible densities from which a data set may have been sampled. For example, it is possible that the PDF is a set of peaks centered at each of the measured data points, a simple uniform distribution, or the PDF might be a series of peaks and valleys similar to the Mandelbrot set [2]. In order to model the data PDF, we must make educated assumptions about its nature.

The assumptions made to construct the model require knowledge about the kind of data expected. In order to model the PDF of a data set we need to ask the following questions:

- What are we modeling and what are the known properties of its PDF?
- What models have others used and could they be improved upon?
- For what are we going to use the model?
- What modeling assumptions should be made based on the answers to the previous questions and what will be the effect of these assumptions?

In this work we are modeling the statistical properties of a set of manufactured transistors. While other kinds of devices could be modeled, our particular application is to GaAs integrated circuits. The knowledge about the statistical properties of the GaAs FET parameters is limited. This is because the GaAs FET manufacturing processes in general are new and because each set of data to be modeled will come from new fabrication lines. It has generally been accepted that the univariate marginal distributions and the joint probability density functions of the parameters are continuous [2]. Their multivariate distributions can be expected to have short tails and have single or multiple modes clustered in a local region [2, 15]. We list these properties below:

- Continuous univariate marginal distributions.
- Continuous joint probability density functions.
- Short tailed multivariate distributions.
- Single or multiple modes clustered in a local region.

Others have used unimodal, univariate and multivariate trimmed and nontrimmed Gaussian distributions as parameter models. This is based on the assumption that the individual parameters are statistically independent. As shown in [2], this assumption is very simple and highly unlikely. Others have also modeled the parameter densities by their marginal distributions and covariance matrices [15]. As shown in [10], the marginal distribution and covariance matrix method is not adequate for an accurate model. The main reason is that the others' techniques do not model the higher order statistical structure of the data. That is to say they do not properly model the local modes and valleys of the joint probability distribution. Since the parameters will be used as the input to a simulator, error from modeling the parameters will affect the accuracy of the simulation [10]. In light of the above stated nature of the data to be modeled, the general direction taken in *this work assumes that the data PDF is a finite mixture of multivariate Gaussian distributions.*

A finite mixture $p(x)$ is a sum of a set of subdistributions $K_i(x)$ where the subdistributions may take any form [16]. Gaussian subdistributions are chosen because they have desirable statistical properties [13]. In addition, the data will be assumed to be time invariant over the time period of its use [2]. It will be assumed that new data will be measured if at any point the underlying process changes. A finite mixture distribution is defined as follows:

Definition 2 *Finite Mixture Distribution*

$$p(x) \doteq \sum_{i=1}^{n} \pi_i K_i(x)$$

In theory, a finite mixture distribution can model closely any distribution since for example as n goes to infinity the distribution has an increasing number of subdistributions. This ultimately becomes an infinite set of points if the Gaussians have zero variance. An infinite collection of points will accurately model any distribution. In practice, n will be a small number, 10 for example. For small n, these modeling assumptions do not model all the variation to be found in the PDFs of all possible data. It is however a substantially more robust model than the previously used techniques. Detailed analysis of the accuracy of the model was presented in Campbell [4]. How this model is constructed from the measured data is the subject of the next sections.

3 Variable Kernel-Based Method

The first technique we will use for statistical interpolative modeling is based on variable kernel density estimation [3]. The variable kernel approach to density modeling is the best suited of the standard nonparametric density estimation techniques for the kinds of data we wish to model. This will be discussed in some detail in the first subsection. Also in the first subsection, we will describe the basic variable kernel density estimation and say why it is useful. Then we will describe what its limitations are, and how we extended it to better suit our particular kind of data.

3.1 Variable Kernel Density Estimation

The variable kernel density estimation method was invented by Brieman *et al.* and presented in the paper[3]. It combines the advantages of the kernel density estimation technique, and the nearest neighbor technique. That is, the data dependency of the kernel estimate, and the local density dependency of the nearest neighbor estimate. In kernel density estimation, the position of data samples is used as the basis for defining the shape of a density estimate. In variable kernel density estimation, the spacing of the data samples is used in addition to their position for defining the shape of the density estimate.

Kernel density estimation is based on the idea that each point of a data set contributes an equal amount of information about the density from which it is sampled. If the density is locally Gaussian, an estimate of the real density may be constructed by putting a Gaussian distribution around each data point. The sum of these Gaussian distributions forms an estimate of the real density. The choice of Gaussian kernel distributions is reasonable since it matches the assumption made previously that the data PDF is a finite mixture of Gaussian distributions.

If we take a PDF as in Figure 1a and sample it as in Figure 1b, then a fixed kernel estimate would take a kernel like the PDF on the right-hand side in Figure 1c and put it around every data point. The normalized sum of these kernels in Figure 1d forms the estimate of the original PDF in Figure 1a. In the variable kernel method, the kernels vary in shape according to the local density of data points in a neighborhood of the data point around which the kernel is put. A data point with a high local density would have a PDF like the one on the right-hand side in Figure 1c, and a data point with a low local density would have a PDF like the one on the left-hand side in Figure 1c. That is to say that the variance of the Gaussian kernels is higher for regions of low density and the variance is lower for regions of high density. The normalized sum of these variable kernels forms the estimate of the original PDF as in Figure 1e.

The variable kernel method combines the kernel estimator and the nearest neighbor method to produce an estimator that is smooth and varies according to the local density of the data [11]. Kernel density estimation can also be thought of as a moving average which averages the points within the kernel window. The nearest neighbor method is based on the idea of smoothing data according to the local density of the data. The shape of the kernel changes according to the width of the window needed to contain a fixed number k of points. The width of the box is found by finding the kth nearest neighbor. If we order the data points near a given data point x, then the kth neighbor at distance $d_k(x)$ is the kth nearest neighbor. Model parameters are chosen so that the PDF of the model smooths or interpolates the data, in order to match the statistics of the data PDF.

3.2 Extension of Variable Kernel Density Estimation

Variable kernel density estimation is limited by the fact that the kernels are not correlated with the local region. The Gaussian distributions used as the kernel PDFs may not accurately reflect local trends in the data. For example, if all the data are in a line then, in order to reflect the local trend, the kernel PDFs should be too. In the variable kernel method however, the kernel distributions will have excess probability off the local trend. This is illustrated in Figure 2 where the data samples are in the middle, the variable kernel method is at the bottom, and the desired result is on top. In order to correct for this deficiency, we developed the concept of a localized nearest neighbor.

The Kernels need to be oriented in the same manner as the orientation of the local trend. The localized nearest neighbor matches the local trend by restricting the choice of nearest neighbor to a "local region" around the anchor point of the kernel. *Each dimension* of the kernel is normalized to reflect the direction of and distance to the localized nearest neighbor in the local region. The shape of the local region is a rectangular box whose sides are proportional to the standard deviation in that dimension for all the data. The size of the local region is a model parameter added to those already required for variable kernel density estimation. These model parameters are optimized to fit the data.

The nearest neighbor found within this local region is then used to orient the kernel PDF so as to reflect the local trend. Consistent with our assumption about the data PDF, the kernel PDF is a multidimensional standard Gaussian distribution with uncorrelated components and zero mean. In order to modify the kernel PDFs to reflect the local trend, each dimension of the kernel PDF is multiplied by the distance in each dimension to the local nearest neighbor. This requires modification to the distance calculation in the variable kernel estimate formula. These modifications are shown below where m is the number of dimensions of the data and each dimension of the kernel is made proportional to the localized

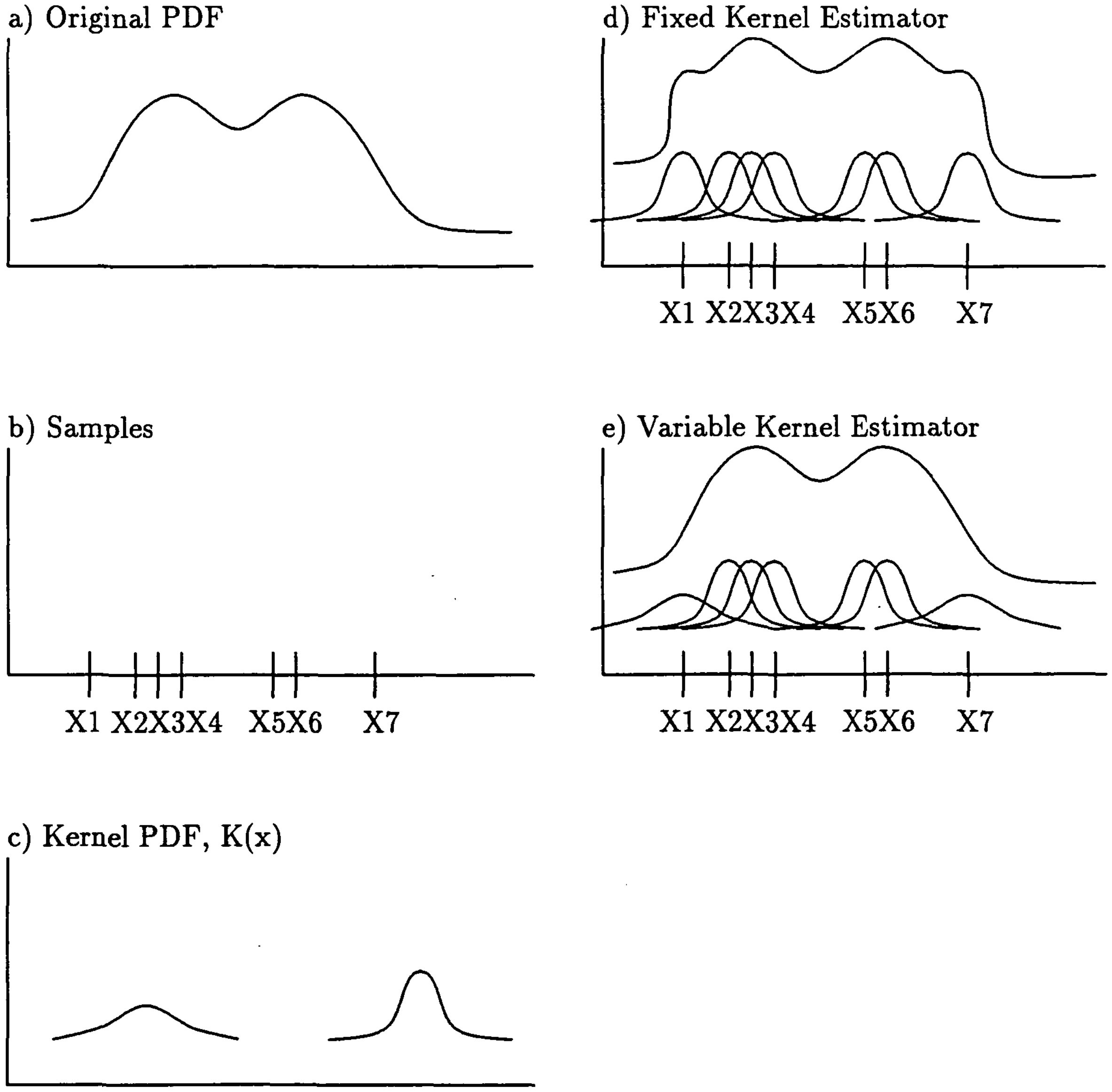


Figure 1: Kernel Variation

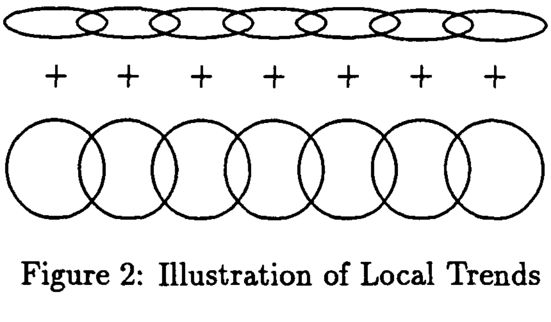


Figure 2: Illustration of Local Trends

nearest neighbor $d_{i,j,k}(x)$.

Definition 3 *Extension of the Variable Kernel Method.*

$$\hat{f}(x) = \frac{1}{n}\sum_{i=1}^{n}\prod_{j=1}^{m}\frac{1}{hd_{i,j,k}(x)}K\left(\frac{x-X_i}{d_{i,j,k}(x)}\right)$$

The sum of all the kernel PDFs forms the model PDF used to generate the simulated data. Figure 3 illustrates the idea, where kernels are shown as ovals around the data points. The generation of points using the extended variable kernel method works as follows. The localized nearest neighbor is found for all the points in the data base of samples from the original PDF. A data point P_i is chosen at random from the data base of samples from the original PDF. The spread around each data point P_i, is determined by the model parameters a and K_i, where K_i is a function of the kth nearest neighbor in the local region determined by q. The choice of a, q, and k is done by an optimization process that is discussed in [4]. The generation of points is done by the following equation where data points are vectors in the data space:

$$\hat{P}_j = P_i + \dot{a}\Delta P_j diag(K_i(k,q)) \tag{1}$$

Where:

$\hat{P}_j$	is a data point vector generated from this model;
P_i	is a measured data point vector chosen at random from the measured data;
a	is a constant model parameter;
ΔP_j	is a vector chosen at random from the kernel PDF;
$K_i(k,q)$	is a scaling vector containing the distance from the chosen P_i to the kth nearest neighbor in each of the data's dimensions [6].

4 Cluster-based methods

The previous technique works well for large data sets in low dimensional spaces [13]. The analysis in [13] suggests that it will not work as well for larger numbers of dimensions. This was investigated in Campbell [4]. This section presents an alternative way of reconstructing the large dimensional PDFs of an S-parameter data base. It has the added benefit that it requires less memory for storage and thus may be used for data reduction. The assumption made in order to model S-parameters was that the PDF was a finite mixture of Gaussian distributions (Section 2 and repeated below). This new process works by finding the groups of data that make up the individual Gaussian distributions $K_i(x)$ of the finite mixture.

Definition 4 *Finite Mixture Distribution*

$$p(x) \doteq \sum_{i=1}^{n}\pi_i K_i(x)$$

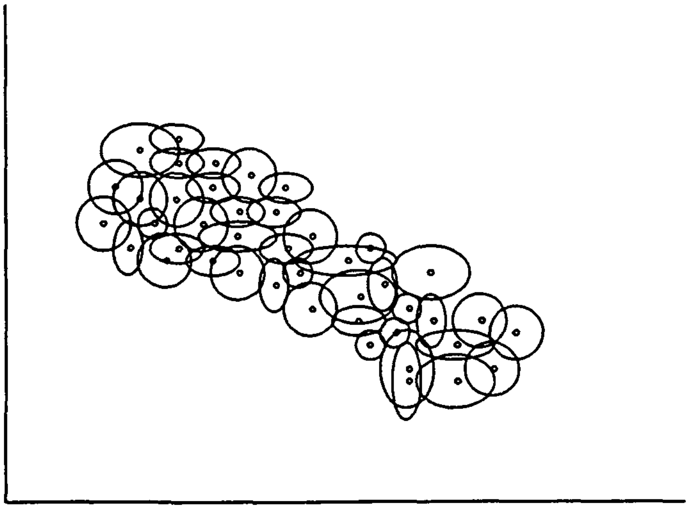


Figure 3: Illustration of Kernels

This presents the problem of deciding from which subdistribution a given data point was sampled. We solve this problem by grouping the data into clusters which represent the subdistributions. A cluster is a collection or set of data points. Each data point is defined as a set of coordinates in a multidimensional Euclidean space [7]. The Euclidean space defines the possible operating properties that may be held by the device the data represents. Data points are assigned to clusters according to which cluster they are closest to.

Below we will briefly describe the methods used for clustering. A more detailed examination is given in Campbell [4] in which we examined the methods of forming groups of clusters from a given set of data and explain our choice of clustering technique. We will examine the various methods for measuring the distance between two clusters in order for the clustering methods to determine the best possible grouping of data points to form clusters. Then we will use an example to describe how the finite mixture distributions are constructed from the clusters.

4.1 Clustering

All methods for clustering data decide which cluster a data point belongs to by the distance between data points. Clustering may be thought of as the process of joining two smaller clusters to form larger clusters, the simplest example being that of forming a cluster from two data points (each of which may be thought of as a cluster of one). How points are chosen to be members of different clusters is what distinguishes the different cluster distance measures.

To find the most compact clusters, the best cluster merging method is complete linkage. In complete linkage, the distance between two clusters is measured by the longest distance between any two points in the two clusters. The two clusters in the data set which have the shortest complete linkage distance are joined to make a larger cluster. Complete linkage tends to find very tight clusters [8]. Cluster distance measuring techniques are the basis for the cluster forming methods which are discussed next.

Cluster analysis is a technique for finding grouping patterns in data [1, 14, 8]. There are a number of clustering techniques, but they are mostly variations on the following techniques. There are two hierarchical clustering methods which produce a hierarchical tree of clusters. The most commonly used techniques are listed below.

- Agglomerative hierarchical clustering
- Divisive hierarchical clustering
- Nonhierarchical K-means clustering

The agglomerative clustering method is the most desirable method since it is efficient and merges outlyers only at the top of the clustering hierarchy. Agglomerative techniques work by starting with all the data points as separate clusters, finding the clusters that are closest together, and merging them one at a time. Ultimately, there is only one cluster. The user of the program must decide by his own criteria how many clusters are desired. Thus the chosen method for finding a Gaussian cluster from a finite mixture is agglomerative clustering using complete linkage.

4.2 Cluster-Based Density Estimation

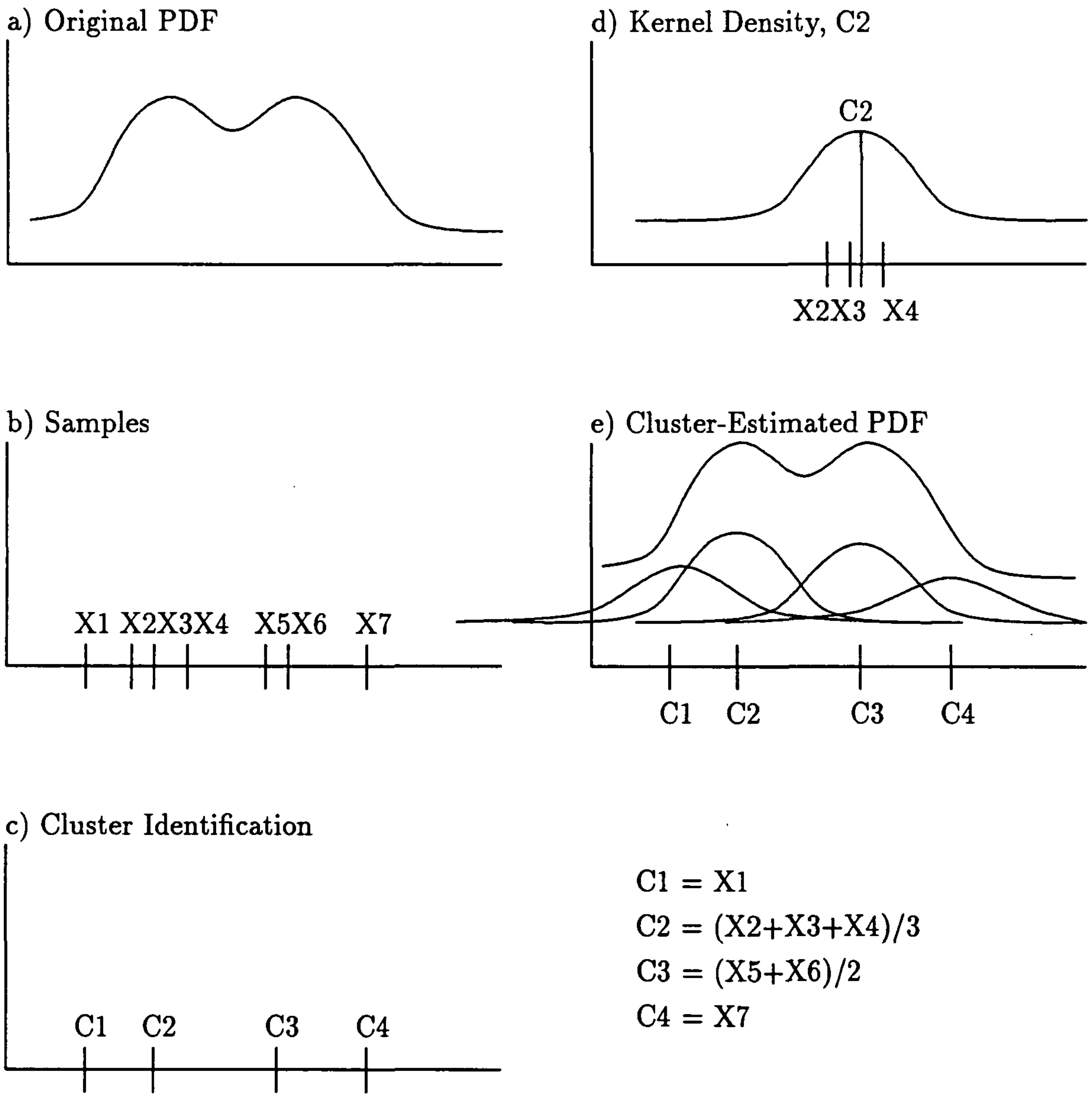
Next we will discuss how finite mixtures are reconstructed using the clustering-based methods. We will do this first using a one-dimensional example which will illustrate the basic method. Then we will show how new simulated data points can be generated from a finite mixture. We will also discuss possible variations to the approach including a method for efficiently storing the finite mixture.

For the example, if seven data points are chosen from a PDF (Figure 4a) at random as illustrated in Figure 4b. These points are labeled $X_1, X_2, ..., X_7$, and are defined by their position values. The next step, as shown in Figure 4c, is to identify the clusters. For this example the data will be grouped into 4 clusters. For the one-dimensional case, a cluster is the average of the values of data points in the cluster.

Definition 5 *1-d Cluster*

$$c_j = \frac{1}{n_j} \Sigma_{i=1}^{n_j} X_i^j$$

For the one-dimensional case, the Gaussian kernel PDF (f_k below) for a cluster is centered at the average of the cluster, and the variance of the Gaussian kernel PDF is proportional to the variance of the data points in the cluster. This is illustrated in Figure 4d. The basic technique is to cluster the data, then model each cluster by a kernel density. Because the clusters can contain different numbers of data points, the kernels will be stored with numbers (π_j below) indicating the proportion of the total number of points each kernel's cluster contained. When points are generated from the kernels, each kernel is allotted a generated point with a probability that is proportional to the number of points in the kernels divided by the total number of points in the original data set. The density estimate for the data is shown in Figure 4e and in the equation below.



C1 = X1

C2 = (X2+X3+X4)/3

C3 = (X5+X6)/2

C4 = X7

Figure 4: Cluster-Based Density Estimation

Definition 6 *1-d Example of Cluster-Based Density Estimate*

$$\hat{F}(x) = \Sigma_{j=1}^{n} \pi_j f_k(x - c_j)$$

In the multidimensional case, there are a number of additional considerations. The coordinates of a cluster kernel are found by the geometric average of the data points in the cluster. The cluster kernel PDFs may then be correlated to the data by using the square-root method [12]. The square-root method uses a square root of the correlation matrix of the cluster data points to correlate vectors generated from the kernel PDFs. The required matrix square root is computed using the Cholesky decomposition [17].

One of the problems in the simulation of circuits is *representing* the distribution of parameters for the devices of a system. The Truth Model [10] proposes to use measurements of actual devices as the data to model their parameter distribution. This has the problem of requiring a considerable amount of storage for the data. In order to reduce the required amount of data stored for each kernel, the kernels may be uncorrelated Gaussian distributions with the variance in each dimension proportional to the cluster data points. This is a considerable storage savings since for correlated kernels the entire correlation matrix must be stored. The resulting model requires far less storage than the Truth model [4].

5 Summary

In this paper, we examined the existing density modeling techniques available, and we gave the details of the density-estimation techniques developed in this work for modeling the data. We considered the assumptions that can be made about the nature of the data to be modeled, and it was assumed that the data PDF is a finite mixture of multivariate Gaussian distributions. In addition, the data were assumed to be time invariant [2] over the time period of its use. We introduced two density estimation techniques to model this data:

1. Extended Variable Kernel Density Estimation.

2. Cluster-based method.

There is a relation between the two density estimation techniques presented here. The extended variable kernel density estimation technique is simply the cluster-based method with a cluster size of one. Together they constitute the statistical interpolative GaAs FET models.

6 Acknowledgements

This research was supported in part by NASA under Space Engineering Research Grant NAGW-1406 and by the NSF under Research Initiation Grant MIP-9109618.

References

[1] M. Anderberg, *Cluster Analysis for Applications*, Academic Press 1973.

[2] P. Becker, and F. Jensen, *Design of Systems and Circuits for Maximum Reliability and Yield*, McGraw Hill 1977.

[3] L. Breiman, W. Meisel, and E. Purcell, "Variable Kernel Estimates of Multivariate Densities," Technometrics, vol 19, no. 2, May 1977, 135-144.

[4] L. Campbell, *Interpolative Modeling for GaAs Simulation*, Ph.D. Dissertation, University of Idaho 1992.

[5] L. Campbell, J. Purviance, and C. Potratz, "Statistical Interpolation of FET Data Base Measurements," Proc. of the 1991 IEEE MTT-S International Microwave Symposium, Boston MA, June 1991.

[6] J. Friedman, F. Bashett and L. Shustek, "An Algorithm for Finding Nearest Neighbors," IEEE Trans. on Computers, Oct 1975, 1000-1006.

[7] K. Hoffman, *Analysis in Euclidean Space*, Prentice Hall 1975.

[8] L. Kaufman and P. Rousseeuw, *Finding Groups in Data: An Introduction to Cluster Analysis*, Wiley 1990.

[9] J. Purviance and M. Meehan, "CAD for Statistical Analysis and Design of Microwave Circuits," International Journal of Microwave and Millimeter-Wave Computer-Aided Engineering, vol 1, no. 1, 1991, 59-76.

[10] J. Purviance, M. Meehan, and D. Collins, "Properties of FET Statistical Data Bases," Proc. of IEEE MTT-S International Microwave Symposium, May 1990.

[11] J. Raategver and R. Duin, "On the Variable Kernel Model for Multivariate Nonparametric Density Estimation," COMPSTAT 78 (eds L. Corsten and J. Hermans), Physics Verlag 1978.

[12] E. Scheuer and D. Stoller, "On Generation of Normal Random Vectors," Technometrics, May 1962.

[13] B. Silverman, "Density Estimation for Statistics and Data Analysis," Chapman and Hall 1986.

[14] H. Spath, *Cluster Analysis Algorithms for Data Reduction and Classification of Objects*, Ellis Horwood 1980.

[15] R. Spence and R. Soin, *Tolerance Design of Electronic Circuits*, Addison-Welsey 1988.

[16] D. Titterington, A. Smith, and U. Makov, *Statistical Analysis of Finite Mixture Distributions*, Wiley 1985.

[17] J.H. Wilkinson and C. Reinsch, *Linear Algebra*, Springer-Verlag Berlin, Heidelberg 1971.