Electronic Journal: Southwest Journal of Pure and Applied Mathematics
Internet: http://rattler.cameron.edu/swjpam.html
ISSN 1083-0464
Vol. No. 01, 1997, pp. 8–12.
Submitted: January 14, 1997. Published: May 29, 1997.

A HEURISTIC ALGORITHM FOR CONSTRUCTING OPTIMAL LOOKUP TABLES IN CIRCUIT DESIGN

Bing Shen and Ferenc Szidarovszky

Abstract. A heuristic algorithm is developed to find the optimum locations of interpolation points in constructing lookup tables in electronic circuit design. Cubic spline interpolators are used, and three versions of the algorithm are introduced. Computational results are reported to illustrate the effectiveness of the methodology.

1991 A.M.S. (MOS) subject classification Codes. 65D07,65D17.

Key Words and Phrases. Cubic splines, optimum location of nodes, heuristic algorithm.

1. INTRODUCTION.

Advances in integrated circuit technology have resulted in increased clock frequencies and higher speeds of operations. With an increased demand on smaller chip size and higher density in high performance systems, the analytical modeling became very difficult and complex. Even analytical modeling of passive elements such as resistors, capacitors, inductors is quite complicated and requires so much computer time in simulation that it is often replaced by modeling based on stored tables of relevant values. The tables can be filled using the data computed using analytical models or the data obtained via measurements. In the first case the advantage of a lookup table is that complex analytical models are used to build the tables for subsequent multiple, inexpensive use in the design. In the second case the tables can be constructed for the elements for which we do not have satisfactory analytical models. Examples of modeling high speed interconnections using lookup tables that can be classified as belonging to the first case of problems are described in You et al. (1996), Holt et al. (1992). In the analysis mode, given a set of dimensions the table is able to provide the electrical parameters as the output. In the synthesis mode, the table provides the dimensions and topology of interconnects for a given set of electrical parameters. The electrical parameters and line geometry are then determined by using specialized interpolation schemes based on tabulated data to get the desired values in a fraction of second and with limited memory size. This is achieved by using special, problem oriented interpolation algorithms that optimize response time and table size. Examples illustrating the modeling of the second case problems, where analytical approach is inadequate, are in Root et al. (1993), Detjens (1987). Lookup tables therefore have a significant practical importance in cases when the quantities in question cannot be computed easily. In looking for certain values it is very seldom the case that the required quantity is among the entries of the table. Since lookup tables represent the values of single or multi variable functions, any function value being between the entries requires the use of certain type of interpolation techniques. In the numerical analysis literature the following methods are most frequently applied for function approximation:

1. Least squares methods
2. Interpolation polynomials
3. Spline functions

Systems and Industrial Engineering Department, University of Arizona, Tucson, Arizona 85721, USA
E-mail address: szidar@sie.arizona.edu

Typeset by $\mathcal{A}_\mathcal{M}\mathcal{S}$-TeX

For our purposes spline functions are the most approximation techniques. Least squares methods require the assumption of a certain function type, and do not give back the original function exactly (Ross. 1987). In engineering applications no analytic function forms are usually given, and the assumption of any function type might result in high approximation errors. Interpolation polynomials are also questionable. In the case of a large number of points, the degree of (single or multi variable) interpolating polynomials becomes large, which leads to numerical instability and accumulation of roundoff errors. It is also well known that the increase in the number of the interpolating points (that is the number of table entries) does not necessarily result in smaller approximation error. As the famous Runge example, see for example Yakowitz and Szidarovszky (1989), shows the sequence of interpolation polynomials even might become divergent if the number of interpolation points converges to infinity. For our purpose spline approximations are the methods of choice, since (1) they interpolate; (2) no prior information on function types is needed; (3) at each segment they are low degree polynomials; (4) increasing number of interpolation points implies better approximation; (5) they are simple to compute. Spline functions are used by several authors in constructing lookup tables. For example, Root et al. (1993) have applied spline function interpolation in building tables for nonlinear model functions. Since there are several spline function types (with different approximation properties) know in the literature it is important to find the optimal spline function type that provides the best overall approximation for a given class of models based on tabulated values. A spline function type is considered to be optimal for a class of models if for any given error tolerance it requires the smallest number of tabulated values. That is, the optimal spline assures minimal computation cost in both the construction of the lookup table and its usage, i.e. in interpolating for the model values.

Therefore finding the best interpolation scheme requires the determination of the optimal locations of the nodes in order to minimize the overall error. In this paper a heuristic algorithm will be proposed, and some computational results will be reported. This problem has been first investigated by Chebyshev for interpolation polynomials, when he has introduced a spechial set of points and proved that uniform error distribution is achieved if and only if these points are selected as nodes (Yakowitz and Szidarovszky, 1989). Unfortunately no similar result are known for spline approximations. The optimal location problem can be mathematically formulated as a complicated nonlinear, nonconvex programming problem. Szidarovszky (1983) has developed a methodology and software package to find the best measurement locations for estimating the quantity and quality of mineral resources. A different than spline approximation scheme known as Kriging was then used, but that problem (arising in the mining industry) has an obvious analogy to the construction of lookup tables in electrical engineering.

2. THE ALGORITHMS.

Let f be given continuous function on a given finite interval $[a, b]$, and assume that the number $n+1$ of interpolation points is fixed. The algorithm starts with the equidistant distribution of the nodes, that is, $x_o = a$ and for $i = 1, 2, \dots, n, x_i = a + (i-1)\frac{b-a}{n}$. The goodness of fit may be characterized by the maximum error. If p denotes the interpolator based on the given nodes, then this measure is defined as

$$E = \max_{x \in [a,b]} |f(x) - p(x)|$$

The algorithms are based on a simple procedure moving one of the points at a time optimally, by assuming that all others are fixed. Let x_k be one of the points. Since x_o is assumed to be the left endpoint of the interval and x_n is always the right endpoint, necessarily $k = 1, 2 \dots, n-1$. The value of x_k is a point between x_{k-1} and x_{k+1}, and its optimal location can be determined by a discrete search. Selecting an $N > 1$, we can divide interval $[x_{k-1}, x_{k+1}]$ into N equal parts, and systematically checking each node for possible location of x_k, the best choice is obtained by comparison. Notice that this idea does not depend on the particular interpolating scheme.

The first algorithm is based on a cyclic application of the above simple procedure. Point x_1 is first located optimally, then the best location of x_2 is found, etc.. After x_{n-1} is relocated, then the best selection of x_1 is determined again, and so on. The procedre terminates, if in a whole cycle none of the points is moved to a new location.

The second algorithm is a refinement of the above procedure. After it terminates, we find the optimal locations of the pairs in order $(x_1, x_2), \dots, (x_1, x_{n-1}),(x_2, x_3), \dots, (x_2, x_{n-1}), \dots, (x_{n-2}, x_{n-1})$. If at least one of the points is relocated, then this cycle is repeated again. The procedure terminates if no point changes location in an entire cycle. The third algorithm is a further refinement of the previous procedure. After it terminates the locations of all possible triples of nodes are determined in the order (x_1, x_2, x_3),(x_1, x_2, x_4), $\ldots$, (x_1, x_2, x_{n-1}), $\ldots$, $(x_{n-3}, x_{n-2}, x_{n-1})$. This procedure also terminates if in a whole cycle none of the points change location.

The above procedure can be applied for any given interpolation scheme as well as with any alternative measure of fit. For example, instead of minimizing the maximum error, we can minimize relative error

$$E* = \max_{x \in [a,b]} \frac{|f(x) - p(x)|}{max\{|f(x)|; |p(x)|\}}$$

or we can minimize the difference of the maximum and minimum errors

$$E** = \max_{x \in [a,b]} |f(x) - p(x)| - \min_{x \in [a,b]} |f(x) - p(x)|$$

3. COMPUTATIONAL EXPERIMENTS.

In our computer study a special interpolating cubic spline was selected, which is a sight modification of the natural cubic spline in the following way: Instead of requiring that $p''(a) = p''(b) = 0$, we assumed that these second derivatives are equal to the second derivative approximations at endpoints based on three first (or last) nodes. That is, we used the conditions

$$p''(a) = \frac{2f(x_0)}{(x_0 - x_1)(x_0 - x_2)} + \frac{2f(x_1)}{(x_1 - x_0)(x_1 - x_2)} + \frac{2f(x_2)}{(x_2 - x_0)(x_2 - x_1)}$$

and

$$p''(b) = \frac{2f(x_n - 2)}{(x_{n-2} - x_{n-1})(x_{n-2} - x_n)} + \frac{2f(x_{n-1})}{(x_{n-1} - x_{n-2})(x_{n-1} - x_n)} + \frac{2f(x_n)}{(x_n - x_{n-1})(x_n - x_{n-2})}$$

Three functions, x^2, e^x, and $sin(\pi x/10)$ were considered and examined in interval $[0, 10]$. The number of nodes was also varied, the values $5, 10$, and 20 were selected for n. the above three algorithms were applied for the three functions, the tree values of n, and measures $E, E*, E**$. Hence we have overall $3 \times 3 \times 3 = 27$ results, which are presented in Tables 1, 2, and 3. The first number of each entry gives the corresponding measure of goodness of fit in the case of the initial node selection (that is, if the nodes are evenly distributed, and the second, third, and forth values correspond to algorithms 1, 2, and 3, respectively.

Table 1 shows the improvements most clearly. For smooth functions such as x^2 and $sin(\pi x/10)$, no large improvement can be observed, however for the exponential function it is significant. Since algorithms 2 and 3 are the further improvements of algorithm 1, the numbers in each entry of the table are decreasing. However this monotonicity does not hold in the other two cases, since the objective functions differ from the maximum error that is given in Tables 2 and 3. For the exponential case, between algorithms 1 and 2, and between algorithms 2 and 3, the improvements are much smaller than the improvement obtained by applying algorithm 1. This fact shows that it is important to use any one of the improving algorithms, and it does not make a significant difference which one is selected to be used.

4. CONCLUSIONS.

The problem of finding the optimal locations of nodes for any interpolation scheme is a complicated, nonconvex optimization problem. Heuristic algorithms are presented in this paper to find its approximate solution. The nonconvexity of the problem makes the algorithm development difficult, and there is no guarantee that the methods do not terminate at only local optima.

The numerical study illustrates, that the application of the algorithms being introduced in the paper reduces the measures of goodness of fit significantly compared to initial node selections, which shows the practical importance of the methodology.

The continuation of the algorithms with the simultaneous improvement of four nodes, then that of five, etc. nodes could result in further reduction of the overall errors, however in the case of large value of n, the application requires a large amount of computer time. Theoretically, if all the $(n - 1)$ interior points are simultaneously varied, then global optimum is reached, however computationally it is infeasible.

Table 1. Maximum Errors in Minimizing Maximum Error.

function	$n = 5$	$n = 10$	$n = 20$
x^2	0.134737	0.034302	0.008575
	0.134737	0.032265	0.008041
	0.116474	0.032265	0.008041
	0.116474	0.030876	0.007812
e^x	2048.475586	422.125000	72.188477
	189.112106	74.360107	30.498047
	189.112106	74.360107	30.498047
	168.601562	74.233749	30.498047
$sin(\pi x/10)$	0.014221	0.001932	0.000246
	0.011504	0.001320	0.000246
	0.011320	0.001320	0.000091
	0.010975	0.001168	0.000080

Value 1 = maximum error with evenly selected points
Value 2 = maximum error using algorithm 1
Value 3 = maximum error using algotithm 2
Value 4 = maximum error using algotithm 3

Table 2. Maximum Errors in Minimizing Maximum Relative Error.

function	$n = 5$	$n = 10$	$n = 20$
x^2	0.134737	0.034302	0.008575
	0.002655	0.033699	0.007423
	0.002655	0.033699	0.007423
	0.002655	0.029655	0.006660
e^x	2084.475586	422.125000	72.188477
	1427.731445	287.652344	55.441406
	1427.708984	287.652344	55.441406
	1464.326172	287.654297	55.480469
$sin(\pi x/10)$	0.014221	0.001932	0.000246
	0.023309	0.004641	0.000061
	0.019391	0.004641	0.000061
	0.019391	0.002897	0.000061

Value 1 = maximum error with evenly selected points
Value 2 = maximum error using algorithm 1
Value 3 = maximum error using algorithm 2
Value 4 = maximum error using algorithm 3

Table 3. Maximum Errors in Minimizing the difference Between Maximum and Minimum Errors.

function	$n = 5$	$n = 10$	$n = 20$
x^2	0.134737	0.034302	0.008575
	0.134737	0.026093	0.008041
	0.116983	0.026093	0.008041
	0.113804	0.026093	0.007812
e^x	2084.475586	422.125000	72.188477
	121.843750	74.350586	30.498047
	121.843750	74.360342	30.498047
	116.367188	74.202148	30.498047
$sin(\pi x/10)$	0.014221	0.001932	0.000246
	0.011504	0.001517	0.000246
	0.011320	0.001517	0.000091
	0.010975	0.001342	0.000080

Value 1 = maximum error using evenly selected points

Value 2 = maximum error using algorithm 1
Value 3 = maximum error using algorithm 2
Value 4 = maximum error using algorithm 3

References

Detjens, E., *Multi-Dimensional Table Model Representation for Circuit Stimulation* (1987), Philips J. Res., 42:510-532.

Holt, J.H., O.A. Palusinski, K.F. Menezes, N. Patel, and M. Smith, *A Design Database for Stripline Interconnections*, Proceedings of the International Electronics Manufacturing Technology Symposium (1992), 90-93.

Root, D.E., M. Pirola, S. Fan, and W.J. Anklam, *Measurement Based Large-Signal Diode Modeling System for Circuit and Device Design*, IEEE Trans. On Microwave Theory and Techniques (1993), 41(21):2211-2217.

Ross, Sh. M., *Introduction to Probability and statistics for Engineers and Scientists* (1987), New York: J. Wiley..

Szidarovszky, F. 1983, *Optimal Network in Geostatistics and Underground Hydrology*, Appl. Math. Modeling, 7:25-32.

Winston, W.L., *Introduction to Mathematical Programming* (1995), Belmont, California: Duxbury Press.

Yakowitz, S. and F. Szidarovszky, *Introduction to Numerical Computations* (1989), New York: Macmillan (2nd edition).

You, Y.C., O.A. Palusinski, B.Lee, and F. Szidarovszky, Computer Support for Design of VLSI and Packaging Interconnections (1996), Submitted to IEEE Transactions on CPMT.