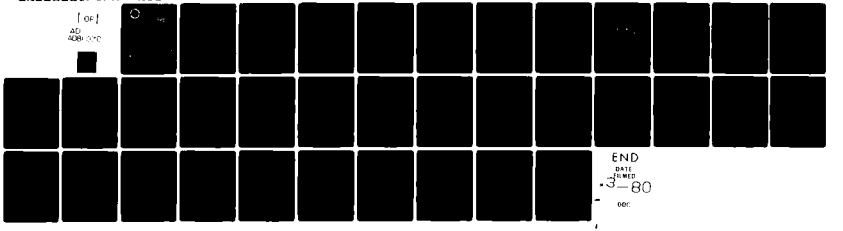
AD-A081 070 STANFORD UNIV CALIF SYSTEMS OPTIMIZATION LAB F/G 12/1
A NESTED PARTITIONING PROCEDURE FOR NUMERICAL MULTIPLE INTEGRAT--ETC(U)
NOV 79 J H FRIEDMAN, M H WRIGHT DAAG29-79-C-0110
UNCLASSIFIED SOL-79-23 ARO-16470.5-M NL



END
DATE FILMED
3-80
DDC

16470.5-M

(1)

Systems
Optimization
Laboratory

LEVEL II

Department of Operations Research
Stanford University
Stanford, CA 94305

SYSTEMS OPTIMIZATION LABORATORY
DEPARTMENT OF OPERATIONS RESEARCH
Stanford University
Stanford, California
94305

A NESTED PARTITIONING PROCEDURE FOR NUMERICAL MULTIPLE INTEGRATION

by

Jerome H. Friedman* and Margaret H. Wright

TECHNICAL REPORT SOL-79-23
November 1979

DAAG29-79-C-0110, EY-76-C-03-0515

164774.5-M

*Stanford Linear Accelerator Center, Stanford, California, 94305.
The work of Jerome H. Friedman was partially supported by the Department of Energy Contract EY-76-C-03-0515.

Research and reproduction of this report were partially supported by the Department of Energy Contract DE-AS03-76-SF00326, PA Number DE-AT-03-76ER72018; National Science Foundation Grant MCS76-20019 A01; and the U.S. Army Research Office Contract DAAG29-79-C-0110.

A NESTED PARTITIONING PROCEDURE FOR NUMERICAL MULTIPLE INTEGRATION

Jerome H. Friedman
Stanford Linear Accelerator Center
Stanford, California
and
European Organization for Nuclear Research, CERN
Geneva, Switzerland

and

Margaret H. Wright
Department of Operations Research
Stanford University
Stanford, California

ABSTRACT

An algorithm is presented for adaptively partitioning a multidimensional coordinate space based on optimization of a scalar function of the coordinates. The goal is to construct a set of hyperrectangular regions, such that the variation of function values within each region is small. These regions are then used as the basis for a stratified sampling estimate of the definite integral of the function.

1. INTRODUCTION

Although the numerical evaluation of multiple integrals has received considerable attention, it remains a difficult problem. Most general-purpose techniques today apply to integrands that are relatively "well behaved" in that the integrand can be reasonably well approximated by a low order polynomial within the region of integration. One technique for trying to deal with integrands that are not so "well behaved" is to partition the region of integration "adaptively" into subregions, chosen so that the integrand is well-behaved within each subregion [Halton and Zeidman, 1971; Lautrup, 1971, Genz, 1972; Kahaner and Wells, 1979; Sasaki, 1978; LePage 1978]. Standard techniques can then be applied to evaluate the integral in each subregion, and the integral over the entire region is taken as the sum of the integrals over the subregions.

Such partitioning can sometimes be carried out manually for integrands of simple functional form with few variables, but this is usually not possible for complicated or large-dimensional integrals. Several automatic partitioning strategies have recently been proposed, based on estimated properties of the integrand. Some of these strategies rely on factored approximations [Lautrup, 1971; LePage, 1978; Sasaki, 1978] while the others are general multidimensional adaptive procedures [Halton and Zeidman, 1971; Genz, 1972; Kahaner and Wells, 1979].

All of these adaptive techniques (as well as the one presented here) are iterative, and are based on top-down successive refinement. At each iteration, a particular region is considered (initially, the entire integration region). A sampling of the integrand within the region is

used to estimate various properties of the integrand, which then guide a strategy for division into several subregions. This process continues until the partitioning has achieved some specified reduction in the estimated error of the approximate integral.

The effectiveness of a partitioning strategy depends, in large part, upon the degree to which the properties of the integrand deduced during the sampling, accurately reflect the characteristics of the function. If the function is badly behaved, its predicted and actual behavior may not even resemble one another, which may lead to ineffective or counterproductive partitions. This possibility is especially likely in high dimensions where even samplings of large cardinality are very sparse (for example, in ten dimensions a sampling of 60,000 points is equivalent to about three points per coordinate). On the other hand, a complex partitioning strategy that requires a very large number of integrand evaluations may be inefficient because the additional evaluations might be better expended simply to increase the number of points used to compute the final integral estimate. Thus, a good partitioning strategy must be based on a computationally feasible way of assessing the integrand's behavior.

The main distinctions of the new partitioning strategy from previously proposed methods are:

(1) The behavior of the integrand within a region is estimated via multiparameter optimization rather than sampling;

(2) All subregions are defined by simple bounds on the coordinates.

2. OVERVIEW OF THE ADAPTIVE REFINEMENT PROCEDURE

Consider a hyperrectangular region R, defined by simple bounds on each coordinate:

$$R = \{x \mid x_i^L \le x_i \le x_i^U\} \tag{1}$$

where x is the vector $(x_1, x_2, .., x_n)^T$.

The essence of an adaptive strategy for partitioning R can be specified by three attributes:

(1) A measure s(R) that indicates the "badness" of the integrand's behavior within R;

(2) A method for subdividing the region after s(R) has been determined;

(3) A procedure for processing the new subregions and for terminating the partitioning.

The quantity used in the new algorithm to characterize the integrand is the difference of extreme values within R, weighted by the volume of R. Let

$$v(R) = \max_{x \in R} f(x) - \min_{x \in R} f(x), \tag{2}$$

where f(x) is a scalar-valued function (presently, the integrand function). The spread s(R) is then defined by:

$$s(R) = v(R) \cdot vol(R). \tag{3}$$

The spread measure s(R) bounds the uncertainty of a quadrature or Monte Carlo estimate of the integral over R, and is taken to indicate the contribution of R to the uncertainty of a global estimate of the integral.

The choice of the measure (3) depends in two crucial ways on the simply-bounded form (1) of R. First, the volume of such a region is

easy to compute:

$$\text{vol}(R) = \prod_{i=1}^{n} (x_i^U - x_i^L).$$

This would not be true if more complicated regions were allowed. The other term (2) may seem, at first glance, to be computationally intractable since two optimization problems must be solved to calculate it. However, methods for optimization with simple bounds on the variables are well developed and, thus, the sub-problems associated with (2) can be solved quite efficiently if f is a reasonable function. Section 3 gives some details of the optimization procedure.

Given that (3) is the spread measure, the second element of the partitioning algorithm involves dividing the single region (1) into disjoint simply-bounded subregions. The strategy for subdivision is based on the assumption that the same quadrature rule will be applied in each subregion at the conclusion of the partitioning so that the aimed-for final result is a list of regions with "similar" spread measures. Section 4 presents the method by which a given region is refined to achieve this goal.

Finally, after R has been partitioned, the daughter subregions are merged into the list of all regions. If the global stopping criteria are satisfied, the partitioning terminates. Otherwise, the list of regions is scanned for the one with the largest spread measure, which is then considered for refinement at the next iteration. Details of this aspect of the algorithm are given in Section 5.

Figure 1 illustrates the partitioning achieved by applying this recursive partitioning procedure to the function

$$f(x_1,x_2) = \exp\{-15[x_1^2 + (x_2 - 0.5)^2]\}$$

$$+ \exp\{-15[(x_1 + 0.433)^2 + (x_2 + 0.25)^2]\}$$

$$+ \exp\{-15[(x_1 - 0.433)^2 + (x_2 + 0.25)^2]\}$$

with $-1 \le x_1 \le 1$ and $-1 \le x_2 \le 1$.

Figure 1a shows an isometric representation of the surface defined by $y = f(x_1,x_2)$, Figure 1b displays some isopleths of the function on the plane, and Figure 1c shows the partitioning of the plane achieved by applying the above procedure recursively, in this case creating eleven subregions. The numbers indicate the order in which the corresponding cuts were made.

3. OPTIMIZATION WITH SIMPLY-BOUNDED VARIABLES

To compute the spread measure (3) at each step of the partitioning procedure, it is necessary to solve two bounds-constrained optimization problems of the form:

M1: $\min f(x)$

subject to $x^L \le x \le x^U$

M2: $\max f(x)$

subject to $x^L \le x \le x^U$,

where the scalar-valued function f drives the partitioning; and the vectors x^L and x^U contain, respectively, the lower and upper bounds that define the desired region.

The problem M2 can be treated as a minimization problem involving $(-f(x))$, and therefore all subsequent discussion will concern minimiza-

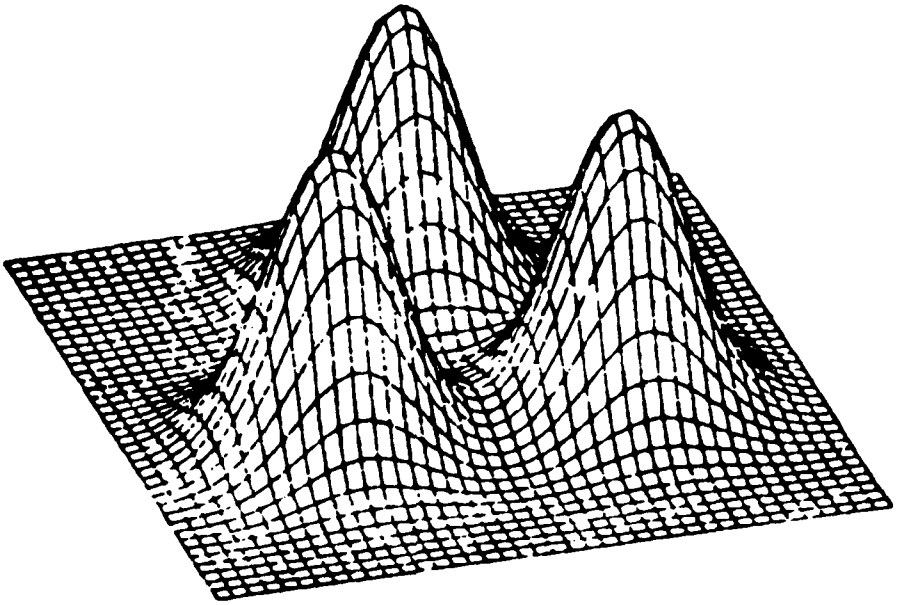


FIGURE 1a

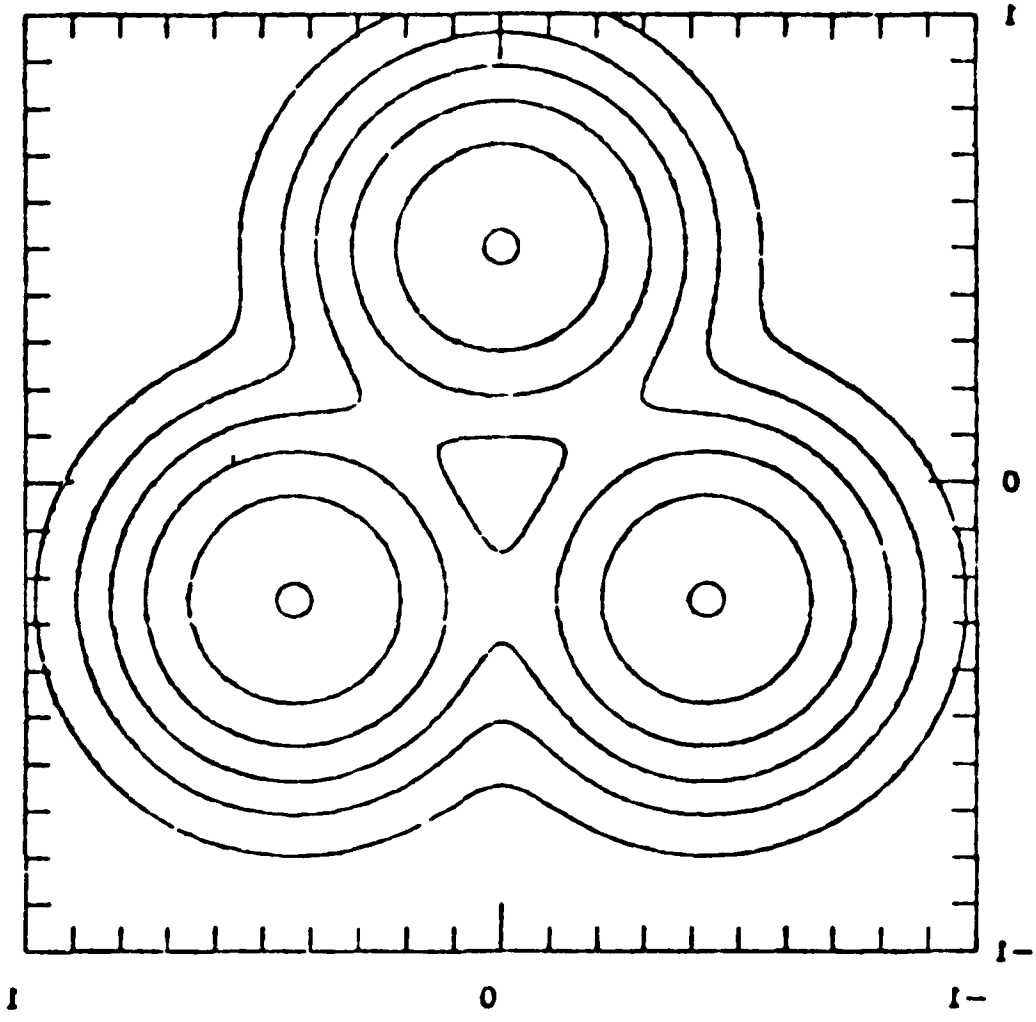


FIGURE 1b

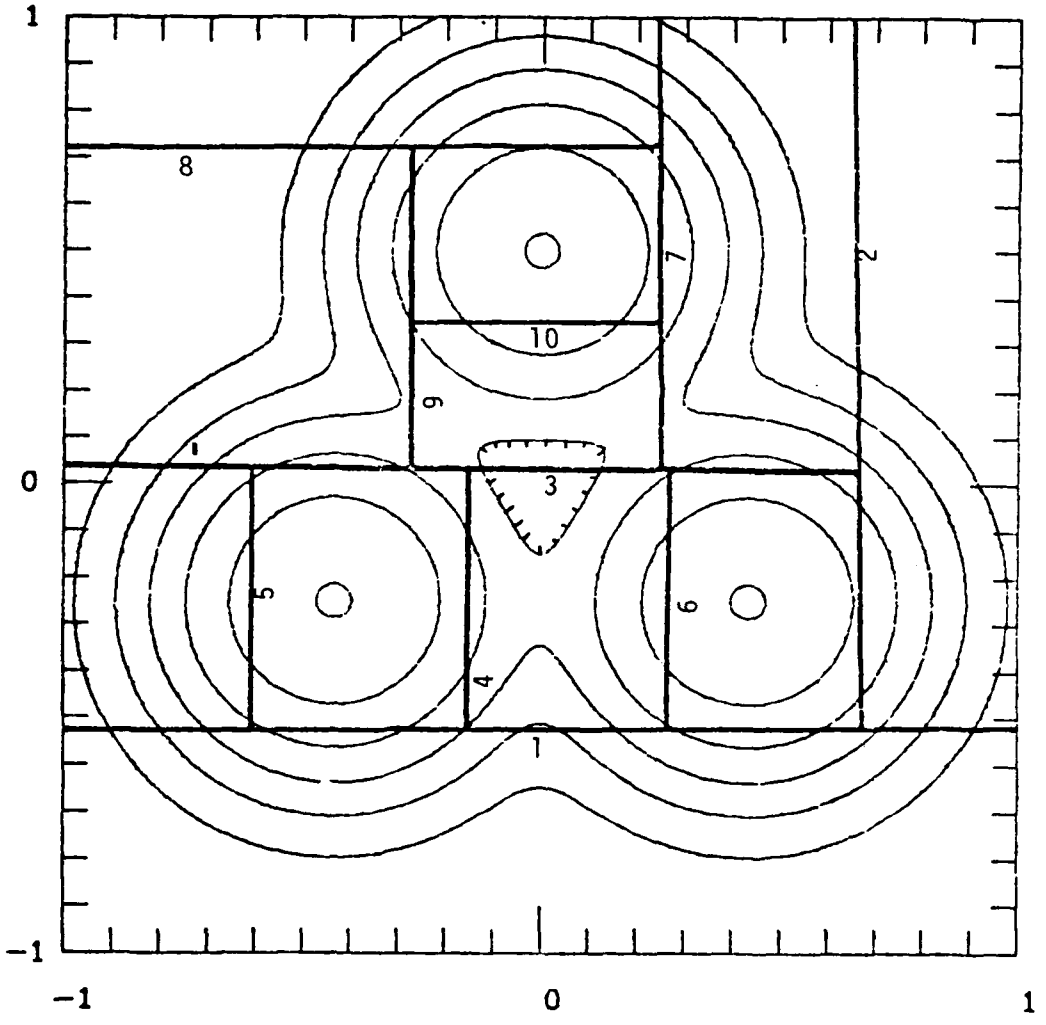


FIGURE 1c

tion only.

In a typical quadrature problem, f(x) will be twice continuously differentiable, or at least will be non-smooth only at isolated points. The algorithm selected to solve problem M1 should consequently be able to perform well on a smooth function. However, in most instances, the derivatives of f will not be available so that the method of choice should require function values only. Based on these considerations, the optimization method used in the partitioning algorithm is a bounds-constrained quasi-Newton method with finite-difference approximations to first derivatives.

Quasi-Netwon methods for unconstrained optimizations have a remarkable history, beginning with Davidon [1959], and Fletcher and Powell [1963]; a recent summary of their motivation and properties is given by Dennis and Moré [1977]. Quasi-Newton methods have been extremely successful on a wide variety of problems; if properly implemented, they are quite robust and usually display superlinear convergence. The idea of a quasi-Newton method is to build up second-order information about the function to be minimized, by incorporating the observed changes in the gradient into a matrix that approximates the underlying matrix of second partial derivatives (Hessian matrix), so that the method should eventually behave like Newton's method.

A typical iteration of an unconstrained quasi-Newton method begins with the current iterate, x; the gradient vector of f, g; and an approximation to the Hessian, the matrix B.

(i) If the norm of the gradient is sufficiently small, the procedure terminates. Otherwise, proceed to step (ii).

(ii) Solve the linear system

$$Bp = -g$$

for the search direction, p. In practice, numerical stability is insured by using a Cholesky factorization of the matrix B, so that p is always a direction of descent for f. This essential feature is due to Gill and Murray [1972].

(iii) Find a steplength $\alpha > 0$ that yields a sufficient decrease in f, so that

$$f(x + \alpha p) < f(x).$$

The steplength algorithm used in the current procedure is the safeguarded quadratic interpolation procedure implemented by Gill and Murray [1974a].

(iv) Evaluate the gradient at $x + \alpha p$, and produce an updated Hessian approximation by modifying the Cholesky factorization of B with the BFGS quasi-Newton update [see Gill and Murray, 1974b]. Return to step (i) with $x + \alpha p$ as the next iterate.

In the present algorithm, it is assumed that the analytic gradient of f is not available, so that the calculation of the vector g is carried out using finite differences.

When the variables are constrained to be between simple bounds, the above algorithm can be modified in a straightforward manner. At each iteration, it is determined whether a given variable is "free" to vary or is to be held "fixed" at one of its bounds. After this decision, the unconstrained algorithm is applied with the following changes:

(1) The gradient, direction of search, and approximate Hessian represent the free variables only;

(2) The steplength in step (iii) may need to be restricted to prevent a free variable from violating a bound during an iteration. In this case, the variable subsequently becomes fixed on that bound.

(3) The updates to B involve only the free variables.

(4) The test for convergence in (i) is based on the norm of the gradient with respect to the free variables. When this quantity is sufficiently small, it is necessary to check whether freeing any variable currently held fixed on its bound will lead to a reduction in f. This determination is made by checking the sign of the gradient with respect to all fixed variables - e.g., if the ith variable is fixed on a lower bound and the ith component of the gradient is negative, the ith variable can be released from its bound.

The many additional details of the algorithm are given in full in the software documentation [Friedman and Wright, 1979], including user-controlled tolerances that define, for example, "sufficiently small" in step (i).

Before beginning to solve problems M1 and M2, the function f is evaluated at a random sample of points (say, 50) in R, and the point with the smallest (largest) function value is used as the initial point for the minimization (maximization). Although for some regions the extrema computed at previous stages could be used as part of the initial sample, this information is not retained in order to improve robustness. Espe-

cially in high dimensions, convergence to a spurious saddle point might preclude any further search for the true extremum since the convergence criteria would be satisfied at the initial point.

An additional feature of the algorithm that is designed to improve robustness is a "local search", to be used if the gradient is small at the initial point. The idea is again to avoid a spurious indication of convergence at a saddle point. The details of the local search are rather complicated and only the general idea will be sketched. First, a point perturbed from the initial point is generated by moving a small, feasible amount along each coordinate until the function value changes sufficiently. Next, a feasible descent direction is constructed at whichever point is lower and an exact line search is carried out along that direction. Then, a second feasible descent direction, orthogonal to the first, is generated at the lowest point found and a second exact line search is performed. If this process fails to yield a "sufficiently lower" function value, the initial point is accepted; otherwise, the quasi-Newton procedure begins at the new point.

This local search cannot be guaranteed to move away from a saddle point since the function is evaluated only at a finite number of points along two directions. In practice, the local search has been quite successful on all the examples tested.

4. REFINEMENT INTO SUB-REGIONS

In this section, we shall be concerned with a particular region, R, defined by (1). Let x^{max} and x^{min} denote the points in R where f achieves its maximum and minimum, respectively, with f^{max} and f^{min} the corresponding function values. For simplicity, we assume that there are no other local extrema in R; the implemented procedure contains provisions to handle situations where this assumption is not satisfied.

A partitioning strategy that allowed completely general regions might divide R into two disjoint parts with equal spread measures, as follows. Suppose that for a given value $\tilde{f}$, $f^{min} \leq \tilde{f} \leq f^{max}$, one could determine an isoplethic surface $\{x | f(x) = \tilde{f}\}$ that separates R into two parts, R^{max} (which contains x^{max}) and R^{min} (which contains x^{min}). Under the uniqueness assumption, the differences of extreme function values within R^{max} and R^{min}, respectively, would be $(f^{max} - \tilde{f})$ and $(\tilde{f} - f^{min})$. The desired choice for $\tilde{f}$ would make the spread measures associated with R^{max} and R^{min} equal, i.e.,

$$(f^{max} - \tilde{f}) \text{ vol } (R^{max}) = (\tilde{f} - f^{min}) \text{ vol } (R^{min}). \qquad (4)$$

The strategy just described is, of course, impractical since the determination of the isopleths would, in general, be an extremely complex numerical procedure. Since the resulting subregions R^{max} and R^{min} would no longer be defined in general by simple bounds, the next optimization problem would also be much more complicated to solve. Furthermore, it would be much more difficult to compute the volume of such a general region.

The strategy adopted in the present algorithm divides the region R into a collection of simply-bounded subregions by constructing an axis-oriented hyperrectangular approximation to either R^{max} or R^{min}. Either f^{max} or f^{min} is selected as the "major" extremum (f^M), i.e., that for which the corresponding function value is farthest from the mean function value $\bar{f}$ in R ($\bar{f}$ is the average of function values at the initial random sample of points).

If $\frac{f^{max} + f^{min}}{2} \geq \bar{f}$, then $f^M = f^{max}$, $f^m = f^{min}$; otherwise, $f^M = f^{min}$, $f^m = f^{max}$ (with the corresponding choices for x^M, x^m).

We then seek to define a region R^M containing x^M by two sets of "cuts" (δ^+, δ^-) along the positive and negative coordinate directions from x^M:

$$R^M = \{x \mid x_i^M - \delta_i^- \leq x_i \leq x_i^M + \delta_i^+\},$$

$$\delta_i^-, \delta_i^+ \geq 0, \quad i = 1, \ldots, n$$

with δ^+, δ^- chosen such that

$$f(x^M + \delta_i^+ e_i) = \tilde{f}$$

$$f(x^M - \delta_i^- e_i) = \tilde{f}, \quad i = 1, \ldots, n \tag{5}$$

where e_i is the ith column of the identity matrix.

The equation to be satisfied by $\tilde{f}$ is a re-arrangement of (4)

$$\gamma f^M + (1-\gamma) f^m = \tilde{f}, \tag{6}$$

where $\gamma = vol(R^M)/vol(R)$.

Because R^M is defined by simple bounds,

$$\text{vol}(R^M) = \prod_{i=1}^{n} (\delta_i^+ + \delta_i^-).$$

Thus, the vectors δ^+, δ^- are the solution of the 2n nonlinear equations:

$$f(x^M + \delta_i^+ e_i) = \frac{\prod_{i=1}^{n} (\delta_i^+ + \delta_i^-)}{\text{vol}(R)} f^M + [1 - \frac{\prod_{i=1}^{n} (\delta_i^+ + \delta_i^-)}{\text{vol}(R)}] f^m \qquad (7)$$

$$f(x^M - \delta_i^- e_i) = \frac{\prod_{i=1}^{n} (\delta_i^+ + \delta_i^-)}{\text{vol}(R)} f^M + [1 - \frac{\prod_{i=1}^{n} (\delta_i^+ + \delta_i^-)}{\text{vol}(R)}] f^m,$$

$i=1, \ldots, n.$

Several considerations affect the choice of solution method for the nonlinear system (7). It is undesirable to expend too much effort in solving (7) since the solution need not be computed with very high accuracy. This means that a Newton-type method based on standard finite differences is unacceptable because of the 2n function evaluations required at every iteration to compute the Jacobian. A reasonable alternative is to use a secant-type method, where the elements of the Jacobian are approximated by differencing the function values from the previous iteration. The switch to a secant method is worthwhile because such methods display local superlinear convergence, and are typically as effective as a Newton-type method in moving from a poor initial estimate of the solution to a reasonably good one (which is all that is required in this case) [Moré, 1977].

Even a secant method for solving (7) could be considered objectionable because it requires the solution of a 2n by 2n linear system at each

iteration. However, the special form of (7) allows it to be transformed to an equivalent, but simpler, nonlinear system.

The right-hand side of (7) is a vector whose components are all equal, and hence the vectors δ^+, δ^- also satisfy the 2n nonlinear equations

$$\begin{aligned} f(x^M + \delta_1^+ e_1) - f(x^M + \delta_2^+ e_2) &= 0 \\ f(x^M + \delta_2^+ e_2) - f(x^M + \delta_3^+ e_3) &= 0 \\ &\vdots \\ f(x^M + \delta_1^+ e_1) - \tilde{f} &= 0 \end{aligned} \tag{8}$$

The attractive feature of the system (8) is that since each equation (except for the last) involves only two adjacent unknowns, the Jacobian displays the following special structure:

$$\begin{matrix} x & x & 0 & . & . & . & . & 0 \\ 0 & x & x & 0 & . & . & . & 0 \\ 0 & 0 & x & x & 0 & . & . & 0 \\ . & . & & & & & & . \\ . & . & & & & & & . \\ . & . & & & & & & . \\ 0 & 0 & 0 & . & . & . & . & x \\ x & x & x & . & . & . & x & x \end{matrix} \tag{9}$$

If no interchanges are necessary, the matrix (9) will be reduced to upper triangular form very easily, by simply subtracting multiples of

each successive row from the last. This means that solving the linear system at each iteration is extremely fast.

Numerous safeguards are included in the secant procedure - in particular, each variable δ_i^+, δ_i^- is constrained to remain within the range where the solution must lie, and the norm of the vector that is the left-hand side of (8) is required to decrease at every iteration.

For simplicity of exposition, certain complications were not included in the preceding discussion of the partitioning strategy - in particular, the fact that not all possible directions are considered as candidates for cuts. Since the bookkeeping overhead for any cut is the same, it is prudent to disregard cuts that appear to be ineffective. Certain directions are eliminated for two reasons. First, x^M may be very close to an upper or lower bound, so that the cut would be insignificant. Therefore, no cut is made along the ith positive direction if

$$x_i^U - x_i^M \leq \beta(x_i^U - x_i^L),$$

nor along the ith negative direction if

$$x_i^M - x_i^L \leq \beta(x_i^U - x_i^L),$$

where $0 < \beta < 1$ (currently, $\beta = .05$).

Furthermore, the solution values for $\delta_i^{\pm}$ are constrained to satisfy

$$\begin{aligned} 0 < \delta_i^+ &\leq \alpha(x_i^U - x_i^M) \\ 0 < \delta_i^- &\leq \alpha(x_i^M - x_i^L), \end{aligned} \tag{10}$$

where $0 < \alpha < 1$ (currently, $\alpha = 1/2$). Before beginning the iteration procedure to solve for the cuts, f is evaluated at the points where $\delta_i^{\pm}$, i=1, ..., n, are at the upper bounds given by (10); the values of γ

and $\tilde{f}$ from (6) are then computed at this initial configuration. Assume that $f^M = f^{max}$ (a similar analysis holds when $f^M = f^{min}$), and consider a possible positive cut along the ith coordinate. By assumption, f is monotonic along the ith coordinate (moving away from the extremum) so that the value of f corresponding to the maximum δ_i^+ will be smaller than $\tilde{f}$ at any other δ_i^+ in the acceptable range. In addition, the initial $\tilde{f}$ will be the maximum possible value. Thus, if f at the initial δ_i^+ exceeds $\tilde{f}$, there can be no solution to (7) in the desired range, and so no attempt will be made to find δ_i^+ (the ith upper bound remains unaltered).

The above procedure is carried out for all possible directions. It should be noted that if any direction is eliminated, the values of γ and $\tilde{f}$ in (6) must be recomputed.

5. MULTIPLE INTEGRATION

The result of applying the nested refinement procedure described in the previous three sections is a set of hyperrectangular subregions of the domain of integration which are mutually exclusive and collectively cover the integration region. The variation of integrand values within the regions has been designed to be substantially less than between the regions. Since the regions are mutually exclusive, the results can be summed to form the global integral estimate.

A variety of methods exist for evaluating the definite integral within each subregion. (For an excellent and rather complete survey, see Stroud, 1971). These methods can be crudely characterized by the regularity they require of the integrand (and/or its derivatives to various orders) and their accuracy per integrand evaluation. At one extreme are the Monte Carlo methods [Halton, 1970] which usually require very little

of the integrand (and thus are quite robust) but which converge rather slowly. Monte Carlo methods also yield a simple uncertainty estimate. At the other extreme are the high degree quadrature formulae [Stroud, 1971] which can be applied only to very regular integrands, but which can yield high accuracy for very few integrand evaluations.

Limited experience has indicated that the more robust methods perform best in conjunction with this partitioning method. Of these, the greatest success has, so far, been obtained with the quasi uniform Monte Carlo methods of Korobov [1963]. Reasonable success has also been achieved with simple pseudo random Monte Carlo methods [Halton, 1970].

6. EXAMPLES

In this section, we attempt to illustrate some of the properties of this nested refinement procedure for multiple integration by applying it to several examples presented by others to illustrate their integration procedures.

The integration method used in each subregion for the examples below is (with one exception) the quasi uniform Monte Carlo method of Korobov [1963], as described in Stroud [1971]. The rate of convergence of this method with increasing N depends upon the smoothness of the integrand, but it is never slower than 1/N. This method (like most quadrature methods) does not provide a simple estimate of the uncertainty associated with the integral evaluation in each subregion. It has been found empirically that the quantity

$$\sigma_i = \frac{1}{2} S_i/N$$

provides a reliable (and usually quite conservative) estimate of the uncertainty associated with this method. Here S_i is the spread of the ith region, N is the number of sample points, and the factor 1/2 is introduced because of the convention of reporting uncertainty as a symmetric (plus or minus) half value about the estimate. Since the integral estimates in each subregion are independent, the total uncertainty σ is taken as the square root of the sum of squares of the individual region uncertainties

$$\sigma = \frac{1}{2N}\left[\sum_{i=1}^{M} S_i^2\right]^{1/2} . \tag{11}$$

Here M is the number of subregions.

This uncertainty estimate (11) can be used as a basis for terminating the partitioning. At a given stage of partitioning, let there be M subregions and $N_p(M)$ integrand evaluations. If one wishes to estimate the integral with (prespecified) uncertainty σ_0, then from (11)

$$N_I(M) = \frac{1}{2\sigma_0}\left[\sum_{i=1}^{M} S_i^2\right]^{1/2} \tag{12}$$

integrand evaluations will be required to estimate the integral in <u>each</u> subregion. Therefore, the total number of integrand evaluations needed to achieve accuracy σ_0 if the partitioning is stopped after M regions is

$$N_T(M) = N_p(M) + N_I(M)\cdot M . \tag{13}$$

As the partitioning proceeds (increasing M) $N_p(M)$ increases (approximately linearly) while $N_I(M)$ decreases due to the reduction in spread. This reduction tends to be very rapid initially, tapering off to slow

reduction for large M. Therefore, $N_T(M)$ tends to decrease for small (increasing) M, reaching a mimimum, and then starts to slowly increase. An optimal strategy is to terminate partitioning at the point at which $N_T(M)$ achieves its minimum value. Since it is not possible to know (in advance) this optimum value, we terminate the partitioning at the first point for which $N_T(M)$ (13) fails to decrease for several (five) successive iterations. Equation (12) is then used to determine the number of evaluation points $N_I(M)$ to perform the final integration in each subregion.

Table 1 shows results of applying this procedure to a series of integrals presented by LePage [1978]. The table presents the answer obtained with estimated uncertainty, the total number of evaluations of the integrand (partitioning plus integral evalation) N_T, the number used for the partitioning stage alone, N_p, and the number of subregions resulting from the partitioning. For comparison, the results presented by LePage for both his method and a Guass-Legendre product rule are also presented.

LePage employs a factored approximation of the form

$$\hat{f}(\vec{x}) = \prod_{i=1}^{p} f_i(x_i) \tag{14}$$

which is used as a basis for pseudo random Monte Carlo importance sampling within the region of integration. This procedure should be especially suitable for the integrals presented in Table 1a since the factored approximation of (14) is exact. As seen in Table 1a, it considerably outperforms the one presented here in high dimensionality. However, the

$$I_p = \left(\frac{10}{\sqrt{\pi}}\right)^p \int_0^1 d^p x \exp\left[-100 \sum_{i=1}^{p} (x_i - 1/2)^2\right]$$

$$= 1.0$$

Integral	This Method	LePage [1978]	Gauss-Legendre
I_4	0.999 ± 0.007	0.994 ± 0.007	.892
	$N_T = 7403$	$N_T = 10000$	$N_T = 10000$
	$N_p = 3851$		
	24 regions		
I_9	$1.01 \pm .008$	$1.001 \pm .005$	71.364
	$N_T = 277238$	$N_T = 100000$	$N_T = 2.0 \times 10^6$
	$N_p = 83626$		
	388 regions		0.774
			$N_T = 10^9$

TABLE 1a

best procedure in this case would be to integrate each one-dimensional function separately and then form the combined integral as the product of the one-dimensional integrals. The integrals presented in Table 1b do not conform exactly to the factored approximation of (14), and the comparison for these cases is more favorable to the partitioning method presented here.

Table 2 shows results of applying this procedure to four integrals presented by Sasaki [1979]. He employs a factored approximation of the form

$$\hat{f}(x) = \prod_{i=1}^{p-1} f_i\,(x_i, x_{i+1}). \tag{15}$$

Each function $f_i(x_i, x_{i+1})$ is represented by an adaptive piecewise constant approximation on the plane. For the first two integrands of Table 2, the factored approximation (15) is exact, while for the last two, it is not. The method presented here is seen to perform well in comparison to that of Sasaki for these integrands. However, as Table 2 indicates, these integrands are well approximated by low order polynomials and a simple Gauss-Legendre product rule outperforms both methods in this case.

Table 3 shows results on several of the integrals presented by Halton and Zeidman [1971]. They describe a nested refinement procedure based on successive bisection. The procedure described in this report was motivated, to a substantial degree, by this MCSS (Monte Carlo Sequential Stratification) technique. Inspection of Table 3 indicates that the optimization method described in this report compares favorably with the MCSS procedure.

$$I_p = \frac{1}{2}\left(\frac{10}{\sqrt{\pi}}\right)^p \int_0^1 d^p x \left\{ \exp\left[-100 \sum_{i=1}^{p} \left(x_i - \frac{1}{3}\right)^2\right] + \exp\left[-100 \sum_{i=1}^{p} \left(x_i - \frac{2}{3}\right)^2\right] \right\} = 1.0$$

Integral	This Method	LePage [1978]	Gauss-Legendre
I_2	1.000 ± 0.003	0.999 ± .002	.999
	N_T = 2278	N_T = 300000	N_T = 2304
	N_p = 1339		
	8 regions		
I_4	0.998 ± 0.007	1.003 ± 0.006	.927
	N_T = 10230	N_T = 300000	N_T = 10000
	N_p = 4662		
	29 regions		
I_7	0.994 ± 0.005	0.991 ± 0.007	2.27
	N_T = 190894	$N_T = 2.4 \times 10^6$	N_T = 279936
	N_p = 43449		
	99 regions		
I_9	1.03 ± 0.025	0.96 ± 0.04	240.08
	N_T = 303228	$N_T = 1.5 \times 10^6$	N_T = 262144
	N_p = 151033		0.0065
	305 regions		$N_T = 1.95 \times 10^6$

TABLE 1b

$$I_a = 100 \int_0^1 d^6x \exp\left[-\sum_{i=1}^{p} (0.6 + 0.4i)x_i\right] = 0.719022$$

$$I_b = 0.01 \int_0^1 \frac{d^6x}{\prod_{i=1}^{5} [0.1 + 0.01 + (x_i + x_{i+1})^2]}$$

$$I_c = 0.01 \int_0^1 \frac{d^6x}{\left[0.1 + \sum_{i=1}^{6} 0.1i\ (x_i - 0.1i)^2\right]^3}$$

$$I_d = 100 \int_0^1 \frac{d^6x}{1.0 + \left[\sum_{i=1}^{6} (1.0 + 0.1i)x_i\right]^4}$$

Integral	This Method	Sasaki (1979) Method 1	Sasaki (1979) Method 2	Gauss-Legendre
I_a	0.7193 ± 0.0005 N_T = 40526 N_p = 7010 48 regions	0.7182 ± 0.0005 N_T = 7000	0.7194 ± 0.0003 N_T = 84529	.71902 N_T = 15625 .71902 N_T = 46656
I_b	0.1740 ± 0.0004 N_T = 46656 N_p = 6183 66 regions	0.1718 ± 0.0004 N_T = 70000	0.1721 ± 0.0004 N_T = 84529	.1722 N_T = 15626 .1723 N_T = 46645
I_c	0.5072 ± 0.0004 N_T = 10676 30 regions	0.5005 ± 0.0004 N_T = 70000	0.5017 ± 0.0004 N_T = 84529	.498 N_T = 4096 .502 N_T = 15625
I_d	0.8193 ± 0.0008 N_T = 46663 N_p = 10798 23 regions	0.816 ± 0.00] N_T = 70000	0.8214 ± 0.0008 N_T = 84529	.8208 N_T = 15626 .8208 N_T = 46645

TABLE 2

$$I_a = \int_0^1 d^{10}x \prod_{i=1}^{10} i\, x_i^{i-1} = 1.0$$

$$I_b = \int_0^1 d^5x\, f(x) = 1/54 = 0.01851851$$

$$f(x) = 1.0, \text{ if } \begin{cases} 0 \le x_1 \le 1 \\ 0 \le x_2 \le 1/2 \\ 0 \le x_3 \le 1/3 \\ 0 \le x_4 \le 2/3 \\ 1/3 \le x_5 \le 1/2 \end{cases}$$

$$f(x) = 0, \text{ otherwise}$$

$$I_k = \int_{-5}^{5} d^kx \prod_{i=1}^{k} \left[(2\Pi)^{-1/2} \exp(-1/2\, x_i^2)\right] = 1.0$$

Integral	This Method	Halton & Zeidman (1971)	Gauss-Legendre
I_a	1.03 ± 0.02[1] $N_T = 91096$ $N_p = 50138$ 273 regions	0.944 ± 0.029 $N_T = 205677$	.921 $N_T = 59049$
I_b	$0.018517 \pm 0.68 \times 10^{-5}$ $N_T = 16237$ $N_p = 7837$ 27 regions	$0.018516 \pm 0.11 \times 10^{-4}$ $N_T = 120145$	.01969 $N_T = 32768$
I_5	$.999 \pm 0.003$ $N_T = 15378$ $N_p = 7816$ 38 regions	0.96 ± 0.02 $N_T = 105821$	1.155 $N_T = 16807$
I_{10}	1.017 ± 0.007 $N_T = 360609$ $N_p = 73225$ 184 regions	0.90 ± 0.11 $N_T = 453872$	0.0076 $N_T = 1.05 \times 10^6$

TABLE 3

7. DISCUSSION

The results of the previous section indicate that the partitioning strategy presented here can be useful for multiple integration. Although one might have rejected out of hand (as being hopelessly too expensive) the notion of applying function optimization to these problems, the examples illustrate that, at least for difficult problems in high dimensionality, this is not the case.

The purpose of applying the partitioning is to divide integration region into subregions, such that the behavior of the integrand within each is relatively good when compared to its behavior over the entire integration region. In this context, "bad behavior" is ideally defined as error associated with a particular integration method. Our choice of spread (3) as such a measure is motivated by the relative ease and reliability with which it can be estimated (using function optimization), as compared to other properties of the integrand (such as the variance) which require adequate sampling to be reliably estimated. The partitioning procedure will be most effective in those cases for which the spread measure closely corresponds to with the uncertainty in the integral estimate.

The main objective of the isoplethic division strategy is to make finer divisions (locally) in those directions in which the integrand is most rapidly varying. The strategy will tend to accomplish this even if the resulting hyperrectangle does not closely correspond to a function isopleth. However, the resulting reduction in spread will be greater the closer the approximation is to a function isopleth. Thus, the partitioning strategy will be most effective when the function isopleths

tend to be convex over the bulk of the integration region and somewhat less effective to the extent that this is not the case. (It should be noted that this generally is the case for the examples of the previous section.) As with most numerical integration methods, this method will have difficulty with highly oscillatory integrands.

An important consequence of adopting the spread as an indication of difficulty is that one is less likely to be deceived into thinking that an integral estimate is accurate when it is not. Undersampling can cause both the integral and its associated error estimate to be seriously undervalued. This tendency is more pronounced the more difficult the problem. Owing to the fact that the estimation of the spread does not rely on sampling, it is less vulnerable to this problem.

The memory requirement associated with the method is not severe. For each of the M hyperrectangular regions, one must store the spread measure, S_i (3), and the region boundaries x_i^U, x_i^L. The boundaries can be arranged in a binary tree requiring 3M-2 integers and M-1 real quantities. Thus, in all, storage for 3M-2 integers and 2M-1 real numbers is required. For the examples of the previous sections, the largest number of regions was M=388. Storage for several thousand regions could easily be accommodated on most medium to large scale computers.

There are several avenues of investigation that are not addressed in this report. It has been assumed that the object function used to drive the partitioning was identical to the integrand. This is not a fundamental requirement and different choices may prove to be useful. For example, if several integrals are to be evaluated with similar integrands over the same region of integration, it might be that a partition-

ing based on one of them will be effective for integrating all of them. Many integration formulae have the property that they are exact for linear functions. Thus, within any region, the linear component of the integrand is exactly integrated and one would like a partitioning of the domain of integration, such that the range or spread of derivatives within each subregion is small. An object function of the form

$$f'(x) = \sum_{i=1}^{p} \left|\frac{\partial f(x)}{\partial x_i}\right| \qquad \text{or} \qquad f'(x) = \left\{\sum_{i=1}^{p} \left[\frac{\partial f(x)}{\partial x_i}\right]^2\right\}^{1/2}$$

might be useful for driving the partitioning in these cases.

The possibility of using this partitioning method in conjunction with other adaptive methods might also be considered. Owing to its robustness, this procedure might be applied as the first stage of a combined procedure. In those subregions for which the spread measure is relatively large, one could apply an adaptive procedure based on sampling. The methods of Genz [1972], LePage [1978], and Kahaner and Wells [1979] appear as good candidates for this combined application.

A FORTRAN program [Friedman and Wright, 1979] implementing the nested refinement partitioning procedure described in this report, along with several numerical integration methods, is available from either author upon request.

ACKNOWLEDGMENTS

Helpful discussions with David Carey, Fred James and G. Peter LePage are gratefully acknowledged. We thank Eric Grosse for providing the graphical example.

8. REFERENCES

Davidon, W.C. (1959). Variable metric method for minimization, Report ANL-5990, Argonne National Laboratory.

Dennis, J.E. and Moré, J.J. (1977). Quasi-Newton methods, motivation and Theory, SIAM Review, Vol. 19, pp. 46-89.

Fletcher, R. and Powell, M.J.D. (1963). A rapidly convergent descent method for minimization, Computer Journal, Vol. 6, pp. 163-168.

Friedman, J.H. and Wright, M.H. (1979). User's guide for DIVONNE, Stanford Linear Accelerator Center, Computation Research Group Technical Memo 193.

Genz, A. (1972). An adaptive multidimensional quadrature procedure, Computer Physics Comm., Vol. 4, pp. 11-15.

Gill, P.E. and Murray, W. (1972). Quasi-Newton methods for unconstrained optimization, J. Inst. Math. Appl., vol. 9, pp. 91-108.

Gill, P.E. and Murray, W. (1974a). Safeguarded steplength algorithms for optimization using descent methods, Report NAC 37, National Physical Laboratory, England.

Gill, P.E. and Murray, W. (1974b). "Quasi-Newton methods for linearly constrained optimization," in Numerical Methods for Constrained Optimization (P.E. Gill and W. Murray, eds.), Academic Press, London and New York.

Halton, J.H. (1970). A retrospective and prospective survey of the Monte Carlo method, SIAM Review, Vol. 12, pp. 1-30.

Halton, J.H. and Zeidman, E.A. (1971). The evaluation of multidimensional integrals by the Monte Carlo sequential stratification technique, Technical Report 137, Computer Science Department, University of Wisconsin.

Kahaner, D.K. and Wells, M.B. (1979). An experimental algorithm for N-dimensional adaptive quadrature, ACM Trans. Math. Software, Vol. 5, pp. 86-96.

Korobov, N.M. (1963). Number-Theoretic Methods in Approximate Analysis (in Russian), Goz. Izdat. Fiz.-Mat. Lit., Moscow, MR 28#716.

Lautrup, B.E. (1971). "An adaptive multidimensional integration technique," in Proceedings of the Second Colloquium in Advanced Computing Methods in Theoretical Physics, C.N.R.S., Marseille.

LePage, G.P. (1978). A new algorithm for adaptive multidimensional integration, Journ. of Computational Physics, Vol. 27, pp. 192-203.

Moré, J.J. (1977). "The Levenberg-Marquardt algorithm: implementation and theory," in Numerical Analysis (G.A. Watson, ed.), Springer-Verlag, New York.

Sasaki, T. (1978). Multidimensional Monte Carlo integration based on factorized approximation functions, SIAM J. Num. Anal., Vol. 15, pp. 938-952.

Stroud, A.H. (1971). Approximate Calculation of Multiple Integrals, Prentice-Hall, Englewood Cliffs, New Jersey.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER SOL 79-23	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A Nested Partitioning Procedure for Numerical Multiple Integration		5. TYPE OF REPORT & PERIOD COVERED Technical Report 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Jerome H. Friedman and Margaret H. Wright		8. CONTRACT OR GRANT NUMBER(s) DAAG29-79-C-0110 ✓
9. PERFORMING ORGANIZATION NAME AND ADDRESS Operations Research Department - SOL ✓ Stanford University Stanford, CA 94305		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS U.S. Army Research Office Box CM, Duke Station Durham, NC 27706		12. REPORT DATE November 1979 13. NUMBER OF PAGES 31
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

This document has been approved for public release and sale; its distribution is unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

THE VIEW, OPINIONS, AND/OR FINDINGS CONTAINED IN THIS REPORT ARE THOSE OF THE AUTHOR(S) AND SHOULD NOT BE CONSTRUED AS AN OFFICIAL DEPARTMENT OF THE ARMY POSITION, POLICY, OR DECISION, UNLESS SO DESIGNATED BY OTHER DOCUMENTATION.

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Numerical integration
Quadrature
Partitioning algorithm
Bounds-constrained optimization

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

SEE ATTACHED

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

UNCLASSFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

SOL 79-23 Jerome H. Friedman and Margaret H. Wright

A NESTED PARTITIONING PROCEDURE FOR
NUMERICAL MULTIPLE INTEGRATION

An algorithm is presented for adaptively partitioning a multidimensional coordinate space based on optimization of a scalar function of the coordinates. The goal is to construct a set of hyperrectangular regions, such that the variation of function values within each region is small. These regions are then used as the basis for a stratified sampling estimate of the definite integral of the function.