Journal of Pharmacokinetics and Biopharmaceutics, Vol. 6, No. 1, 1978

SCIENTIFIC COMMENTARY

A Comparison of Numerical Integrating Algorithms by Trapezoidal, Lagrange, and Spline Approximation

K. C. Yeh[1] and K. C. Kwan[1]

Received June 9, 1977—Final August 23, 1977

In the trapezoidal method, linear interpolation between data points tends to overestimate or underestimate the area, depending on the concavity of the curve. In some instances, area estimates can be obtained by linear interpolation of logarithmically transformed data. Two alternative algorithms based on known interpolating functions have been implemented for area calculations. In the Lagrange method, the linear interpolations are replaced by cubic polynomial interpolations. In the spline method, the cubic functions are further modified so that the fitted curves are completely smooth. This report describes their computing procedures with numerical examples.

KEY WORDS: numerical integrating algorithms; trapezoidal approximation; Lagrange method; spline method.

INTRODUCTION

It is customary in biopharmaceutics to use a trapezoidal method to calculate areas under the concentration–time curve. The popularity of this method may be attributed to its simplicity both in concept and in execution (1,2). However, in cases where changes in curvature between data points are excessive or there are long intervals between data points, large algorithm errors are known to occur.

To circumvent the curvature problem, two alternative algorithms based on interpolating polynomials have been devised and implemented for area calculations in these laboratories. These polynomials are known as spline functions (3,4) and Lagrange interpolating functions (5). The purpose of this report is to describe computational procedures of these two methods, to compare their solutions along with those obtained by the

[1] Merck Sharp & Dohme Research Laboratories, West Point, Pennsylvania 19486.

linear and log trapezoidal methods, and to discuss the relative merit of the methods.

NUMERICAL METHODS

The purpose of a numerical method is to obtain practical solutions which otherwise would have been difficult or impossible to achieve. Because of two contributing factors, the solutions are seldom error free. First, experimental errors in the data are inevitable. These are called input errors. Second, additional errors are incurred when data are processed to produce numerical solutions. These are called algorithm errors. There are two types of algorithm errors (6). The *truncation* error is the difference between the true functional value and that calculated by numerical approximation. The *round off* error results from the fact that only a finite number of digits can actually be retained after each computational step, and any excess digits are lost.

In biopharmaceutics, experimental data such as plasma concentrations are usually recorded at discrete time points. The purpose of using an approximating function in the present case is to connect all data points so that reliable values of areas can be calculated by integration. Although the selection of a particular procedure is somewhat subjective, two basic factors are usually considered: speed and accuracy. When calculations are to be performed manually, easy and simple methods are clearly preferable. However, with the advent of high-speed electronic computers, accuracy becomes the major consideration since computational steps can be programmed and executed swiftly. Thus a method that increases the accuracy of solutions by minimizing algorithm errors should be attempted whenever the procedure is compatible with the limitations of available facilities.

Because of their convenient mathematical properties, polynomials are the most widely used among various curve-fitting approaches. The four procedures described below represent the application of polynomials to area calculations.

Linear Trapezoidal Method

The linear trapezoidal method is the best known numerical integrating method. The functional value y between two adjacent points (t_{i-1}, y_{i-1}) and (t_i, y_i) is approximated by a straight line:

$$y = a + bt \tag{1}$$

where

$$a = (t_i y_{i-1} - y_i t_{i-1})/h_i$$

$$b = (y_i - y_{i-1})/h_i$$

and

$$h_i = t_i - t_{i-1}$$

Integrating equation 1 from t_{i-1} to t_i gives the incremental area in that interval:

$$[\mathrm{AUC}]_{t_{i-1}}^{t_i} = \int_{t_{i-1}}^{t_i} y\,dt = \tfrac{1}{2}h_i(y_i + y_{i-1}) \tag{2}$$

To obtain the cumulative area over the interval $[t_1, t_n]$, where n is the number of data points, the above procedure is repeated for each i, where $i = 2, 3, \ldots, n$. The cumulative area then becomes

$$[\mathrm{AUC}]_{t_1}^{t_n} = [\mathrm{AUC}]_{t_1}^{t_2} + [\mathrm{AUC}]_{t_2}^{t_3} + \cdots + [\mathrm{AUC}]_{t_{n-1}}^{t_n} \tag{3}$$

It is apparent that linear interpolation between data points will tend to underestimate the area when data form a convex curve and to overestimate when the curve is concave. Further, the greater the h_i, the greater would be the error. The magnitude of error would also depend on the oscillatory nature of the curve, or the lack thereof, between data points.

Log Trapezoidal Method

A direct modification of the linear trapezoidal method is the so-called log trapezoidal method. In this modified version, the y values are assumed to vary linearly within each sampling interval on a semilogarithmic scale:

$$\ln(y) = \ln(y_{i-1}) + (t - t_{i-1}) \cdot \ln(y_i/y_{i-1})/h_i \tag{4}$$

On integration, one obtains

$$[\mathrm{AUC}]_{t_{i-1}}^{t_i} = h_i(y_i - y_{i-1})/\ln(y_i/y_{i-1}) \tag{5}$$

In pharmacokinetics, equation 5 is most appropriate when applied to data which appear to decline exponentially. Under such condition, the error produced is independent of h_i. However, the method may produce large errors when used in an ascending curve, near a peak, or in a steeply descending polyexponential curve. Furthermore, the method cannot be used if either concentration is zero or if the two values are equal. When they occur, equation 2 can be used as an alternative approach. Despite these limitations, the log trapezoidal method can be used advantageously in specific situations or in combination with a second method to yield optimal solutions.

Lagrange Method

In the Lagrange method, the linear function of equation 1 is replaced by a cubic polynomial:

$$y = a_i + b_i t + c_i t^2 + d_i t^3 \qquad (6)$$

To interpolate between $t_{i-1} \le t \le t_i$, the equation is fitted to the nearest four data points (t_{i-2}, y_{i-2}), (t_{i-1}, y_{i-1}), (t_i, y_i), and (t_{i+1}, y_{i+1}). The function is thus forced to pass through all four points. The shape of the fitted curve in the middle interval may not always be linear, but may be parabolic, or sigmoidal with one inflection point. The four coefficients, a_i, b_i, c_i, and d_i, can be obtained by using the Lagrange multiplier formula (5), or by solving the following system of equations:

$$\begin{bmatrix} 1 & t_{i-2} & t_{i-2}^2 & t_{i-2}^3 \\ 1 & t_{i-1} & t_{i-1}^2 & t_{i-1}^3 \\ 1 & t_i & t_i^2 & t_i^3 \\ 1 & t_{i+1} & t_{i+1}^2 & t_{i+1}^3 \end{bmatrix} \begin{bmatrix} a_i \\ b_i \\ c_i \\ d_i \end{bmatrix} = \begin{bmatrix} y_{i-2} \\ y_{i-1} \\ y_i \\ y_{i+1} \end{bmatrix} \qquad (7)$$

Once the coefficients are obtained, the incremental area on the interval $[t_{i-1}, t_i]$ is calculated by integrating equation 6:

$$[\mathrm{AUC}]_{t_{i-1}}^{t_i} = a_i h_i + \tfrac{1}{2} b_i (t_i^2 - t_{i-1}^2) + \tfrac{1}{3} c_i (t_i^3 - t_{i-1}^3) + \tfrac{1}{4} d_i (t_i^4 - t_{i-1}^4) \qquad (8)$$

As an example, Fig. 1 shows the cubic polynomial

$$y = 1 + 7.5t - 5.5t^2 + t^3 \qquad (9)$$

passing through (0,1), (1,4), (2,2), and (3,1). The area over the interval [1,2] by equation 8 is 3.17, whereas that by the trapezoidal method is 3.0

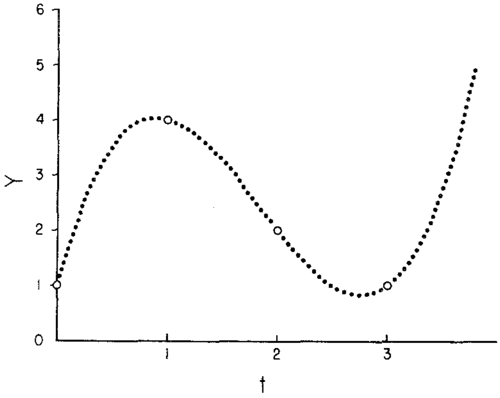

Fig. 1. A cubic polynomial fitted to four data points.

In order to use equation 6, the data must be theoretically smooth over the interval $[t_{i-2}, t_{i+1}]$. Under this condition, the cubic polynomial will usually give a better approximation than a single straight line in the middle interval $[t_{i-1}, t_i]$ because the two contiguous points also contribute to defining the functional behavior over this subinterval. Functional approximations over the intervals $[t_{i-2}, t_{i-1}]$ and $[t_i, t_{i+1}]$ tend to be less reliable in that shaping constraints are one-sided.

Equation 7 can be applied serially for each i, where $i = 3, 4, \ldots, n-1$, but not for the two end intervals, $[t_1, t_2]$ and $[t_{n-1}, t_n]$. To fit these two intervals, the nearest three points are used and fitted with a parabola:

$$y = a_i + b_i t + c_i t^2 \tag{10}$$

The three coefficients are calculated by solving a system of three simultaneous linear equations, analogous to equation 7, and the corresponding areas are obtained by integrating equation 10.

The cumulative area over the entire interval $[t_1, t_n]$ is computed by summation, using equation 3.

Areas calculated by the integration of parabolic equations (equation 10) are subject to greater error than those calculated by the integration of cubic equations (equation 6). In part, this is because approximations are usually better near the middle segment than at the two ends. Since equation 10 is quadratic, it may yield a minimum or a maximum that "should" not have been there. Unwanted oscillations may also occur with cubic equations. Therefore, it is desirable to monitor the interpolated curve over the entire interval. The monitoring can be accomplished by sampling a set of interpolated values within each interval $[t_{i-1}, t_i]$. In so doing, the suitability of the fitted curve can be evaluated intuitively in relation to the prevailing understanding or assumptions concerning the underlying kinetic mechanism.

Thus in the Lagrange method the n experimental points are linked by n-1 smooth curves, with each data point forming the knot of the chain. The fitted curve is piecewise smooth; i.e., it is differentiable within each interval $[t_{i-1}, t_i]$, but not at the data points. In other words, each knot forms a singular point and has no definitive first derivative because the two tangent lines of adjacent cubic functions at t_i may not coincide. This is similar to the trapezoidal method wherein the fitted curve is piecewise linear. However, the curve connected by serial cubic polynomials will look more curvilinear and natural than the polygonal curve formed by the linear trapezoidal method.

The use of serial low-degree (cubic) polynomials, each of which is fitted to a local region, is preferable to the use of a single high-degree polynomial. While a polynomial of degree n-1 or less can be expected to pass through n points, excessive oscillations on the fitted curve would be inevitable. On the other hand, the linear interpolations of either of the trapezoidal methods are overly simplistic, even though the problem of spurious oscillations is completely avoided. In a practical sense, the Lagrange method can be considered a compromise between the two extremes.

Spline Method

General spline functions are defined as piecewise polynomials of degree k. The pieces are connected at each of the several knots such that the fitted curve and its first k-1 derivatives are continuously differentiable over the interval $[t_1, t_n]$ (3).

In this method, the knots are taken to be the data points themselves and k is defined to be 3. Thus the procedure of interpolation by cubic splines is similar to the Lagrange method except for the additional constraint of differentiability at each data point.

Description of spline functions can be found in many sources (3,4,7). The derivation presented below follows closely that of Dunfield and Read (8).

Consider equation 6, which is a cubic function. Differentiating it twice gives the following linear expression:

$$\ddot{y} = 2c_i + 6d_i t \tag{11}$$

Equation 11 shows that $\ddot{y}$ is linear over each interval $[t_{i-1}, t_i]$. Rewriting it in the following form

$$\ddot{y} = \frac{\ddot{y}_{i-1}}{h_i}(t_i - t) + \frac{\ddot{y}_i}{h_i}(t - t_{i-1}) \tag{12}$$

the equation can be integrated to yield

$$\dot{y} = \frac{-\ddot{y}_{i-1}}{2h_i}(t_i - t)^2 + \frac{\ddot{y}_i}{2h_i}(t - t_{i-1})^2 + s_1 \tag{13}$$

$$y = \frac{\ddot{y}_{i-1}}{6h_i}(t_i - t)^3 + \frac{\ddot{y}_i}{6h_i}(t - t_{i-1})^3 + s_1 t + s_2 \tag{14}$$

where s_1 and s_2 are integration constants. Evaluating equation 14 at t_{i-1} and t_i, respectively, gives

$$y_{i-1} = \frac{\ddot{y}_{i-1}}{6h_i}(h_i)^3 + s_1 t_{i-1} + s_2 \tag{15}$$

$$y_i = \frac{\ddot{y}_i}{6h_i}(h_i)^3 + s_1 t_i + s_2 \tag{16}$$

from which the two constants can be solved:

$$s_1 = \frac{1}{h_i}(y_i - y_{i-1}) - \frac{h_i}{6}(\ddot{y}_i - \ddot{y}_{i-1}) \tag{17}$$

$$s_2 = \frac{1}{h_i}(t_i y_{i-1} - y_i t_{i-1}) - \frac{h_i}{6}(t_i \ddot{y}_{i-1} - \ddot{y}_i t_{i-1}) \tag{18}$$

All quantities in equation 14 are known except $\ddot{y}_{i-1}$ and $\ddot{y}_i$, the second derivatives at each data point. These unknown values are determined as follows.

Evaluating equation 13 at t_{i-1} from the right interval $[t_{i-1}, t_i]$ and from the left interval $[t_{i-2}, t_{i-1}]$ gives the following two equations, respectively:

$$\dot{y}_{i-1} = \frac{-\ddot{y}_{i-1} h_i}{2} + \frac{(y_i - y_{i-1})}{h_i} - h_i \frac{(\ddot{y}_i - \ddot{y}_{i-1})}{6} \tag{19}$$

$$\dot{y}_{i-1} = \frac{\ddot{y}_{i-1} h_{i-1}}{2} + \frac{(y_{i-1} - y_{i-2})}{h_{i-1}} - h_{i-1} \frac{(\ddot{y}_{i-1} - \ddot{y}_{i-2})}{6} \tag{20}$$

where $h_{i-1} = t_{i-1} - t_{i-2}$.

Combining Equations 19 and 20 gives a single expression after rearrangement:

$$\frac{h_{i-1}}{6}\ddot{y}_{i-2} + \frac{1}{3}(h_i + h_{i-1})\ddot{y}_{i-1} + \frac{h_i}{6}\ddot{y}_i = \frac{(y_i - y_{i-1})}{h_i} - \frac{(y_{i-1} - y_{1-2})}{h_{i-1}} \tag{21}$$

Thus n-2 equations can be generated from equation 21 where $i = 3, 4, \ldots, n$. Two extra equations are required to solve for n unknowns. They can be obtained by specifying two additional conditions. In the present case, these are $\dddot{y}_2 = \dddot{y}_3$, and $\dddot{y}_{n-1} = \dddot{y}_n$. These third derivatives are obtained by differentiating equation 11 at $i = 2, 3, n$-1, and n:

$$\dddot{y}_2 = (\ddot{y}_2 - \ddot{y}_i)/h_2 \tag{22}$$

$$\dddot{y}_3 = (\ddot{y}_3 - \ddot{y}_2)/h_3 \tag{23}$$

$$\dddot{y}_{n-1} = (\ddot{y}_{n-1} - \ddot{y}_{n-2})/h_{n-1} \tag{24}$$

$$\dddot{y}_n = (\ddot{y}_n - \ddot{y}_{n-1})/h_n \tag{25}$$

Combining equations 22 and 23, and equations 24 and 25, respectively, gives

$$\frac{1}{h_2}\ddot{y}_1 - \left(\frac{1}{h_2} + \frac{1}{h_3}\right)\ddot{y}_2 + \frac{1}{h_3}\ddot{y}_3 = 0 \tag{26}$$

$$\frac{1}{h_{n-1}}\ddot{y}_{n-2} - \left(\frac{1}{h_{n-1}} + \frac{1}{h_n}\right)\ddot{y}_{n-1} + \frac{1}{h_n}\ddot{y}_n = 0 \tag{27}$$

With the addition of the last two equations, the n unknown $\ddot{y}_i$ values are determined by solving equations 21, 26 and 27 simultaneously. The incremental area over each interval $[t_{i-1}, t_i]$ is calculated by integrating equation 14:

$$[\mathrm{AUC}]_{t_{i-1}}^{t_i} = \frac{h_i^3}{24}(\ddot{y}_i + \ddot{y}_{i-1}) + h_i[\tfrac{1}{2}s_1(t_i + t_{i-1}) + s_2] \qquad (28)$$

As before, the cumulative area over the interval $[t_1, t_n]$ is obtained by summation, using equation 3.

The functional behaviors of the fitted curve in each interval $[t_{i-1}, t_i]$ are determined not only by the nearest two or four data points but also by all others. Evidently, data in the immediate neighborhood are the most influential.

Because of these additional constraints, the fitted curve and its first derivatives are completely smooth. This is in contrast to the other methods wherein the fitted curve is only piecewise smooth. Since the spline function is composed of serial cubic polynomials, the actual values of the interpolated curve should also be sampled to monitor the functional behavior of the fitted curve. This is especially critical when data are scattered or sampling intervals are large. Under these conditions, the cubic equations may produce extraneous inflection points and cause serious under- or overestimations of areas.

In practice, equations 26 and 27 are applicable to general types of kinetic data. In specific situations where additional information is available, there may be other functions (e.g., exponential) that are more representative of the data. When used appropriately, such modifications should increase the accuracy of the solutions.

SIMULATIONS AND RESULTS

A series of computations were performed using each of the above procedures. These exercises were designed to examine the validity of the algorithms as well as to compare the reliability of the solutions under simulated conditions. The following examples may serve to provide some perspective on the relative merits of the four methods in specific circumstances.

Example 1. The data were assumed to be linear over the interval $[t_1, t_n]$. The results, shown in Table I, indicate that all three methods gave exact and identical solutions. In this example, the linear trapezoidal method should yield the correct answer since assumption of linearity between points is identical to the nature of data. Moreover, the obser-

Table I. Estimation of Areas Under Three Straight Lines

		Simulated y_i		
i	t	Case 1	Case 2	Case 3
1	0	0	8.0	4.0
2	0.2	0.2	7.8	4.0
3	1.0	1.0	7.0	4.0
4	3.0	3.0	5.0	4.0
5	4.0	4.0	4.0	4.0
6	5.5	5.5	2.5	4.0
7	8.0	8.0	0	4.0
$[\mathrm{AUC}]_0^8$ by				
Linear trapezoidal method		32.0	32.0	32.0
Lagrange method		32.0	32.0	32.0
Spline method		32.0	32.0	32.0

vation that both the Lagrange and the spline methods also produced correct solutions indicates that all polynomials had degenerated to linear functions and that undue oscillations had not been produced.

Example 2. Data displayed in Table II represent sampled values on a monoexponentially decaying curve:

$$y = 16 \exp(-0.4t) \tag{29}$$

On a semilogarithmic plot, equation 29 forms a straight line. The theoretical area values were obtained by using the log trapezoidal method (equation 5).

Table II. Incremental Areas and the Corresponding Algorithm Errors for Example 2

		Area on subinterval $[t_{i-1}, t_i]^{a,b}$			
t_i	Y_i	Theoretical	Linear trapezoidal	Lagrange	Spline
0	16.0000	—	—	—	—
0.2	14.7699	3.0753	3.0770 (0.05)	3.0754 (0.00)	3.0754 (0.00)
0.5	13.0997	4.1754	4.1804 (0.12)	4.1754 (−0.00)	4.1754 (−0.00)
1.0	10.7251	5.9364	5.9562 (0.33)	5.9363 (−0.00)	5.9364 (0.00)
2.0	7.1893	8.8396	8.9572 (1.33)	8.8374 (−0.02)	8.8392 (−0.01)
3.0	4.8191	5.9254	6.0042 (1.33)	5.9231 (−0.04)	5.9250 (−0.01)
4.0	3.2303	3.9719	4.0247 (1.33)	3.9695 (−0.06)	3.9725 (0.02)
6.0	1.4515	4.4471	4.6818 (5.28)	4.4256 (−0.48)	4.4340 (−0.30)
9.0	0.4372	2.5358	2.8330 (11.72)	2.3368 (−7.85)	2.6346 (3.90)

[a] Values shown are rounded off for display.

[b] Values in parentheses are percent algorithm errors.

It is evident from Table II that the linear trapezoidal method consistently produced large, positive errors, while the Lagrange method yielded relatively small negative ones. In contrast, output errors produced by the spline method were generally smaller and were more randomly distributed. However, both the Lagrange and the spline methods produced proportionately large error for the last increment because the sampling interval is large and there are no later data to tie the cubic function. In this example, the log trapezoidal method is obviously the best since its solutions are identical to the theoretical values.

Example 3. Simulated plasma concentrations were generated based on an oral two-compartment open model with elimination occurring from the central compartment (Scheme I):

$$\begin{tikzcd}
G \arrow[r, "k_a"] & B \arrow[r, "k_{10}"] \arrow[d, shift left, harpoon, "h_{21}"] & E \\
& T \arrow[u, shift left, harpoon, "k_{12}"] &
\end{tikzcd}$$

Scheme I

Differential equations corresponding to the model are as follows:

$$\dot{G} = -k_a G \tag{30}$$

$$\dot{C}_p = \frac{1}{V} k_a G - k_{12} C_p + \frac{1}{V} k_{21} T - k_{10} C_p \tag{31}$$

$$\dot{T} = k_{12} V C_p - k_{21} T \tag{32}$$

$$\dot{E} = k_{10} V C_p \tag{33}$$

where G, B, T, and E are drug amounts in the absorption compartment, the central compartment (including blood), the peripheral compartment, and the elimination compartment (sum of biotransformation and excretion), respectively; C_p is the plasma concentrations; V is the apparent volume of distribution of the central compartment; and k_{12}, k_{21}, and k_{10} are first-order rate constants for the designated processes. To simulate the possibility that absorption efficiency may be related to transient location of absorption sites, the absorption parameter k_a was assumed to be time dependent. After several experimentations, the following empirical equation was employed:

$$k_a = A_1 \exp\left[-(A_2 t - A_3)^2\right] + A_4 \sqrt{t} \tag{34}$$

where A_1, A_2, A_3, and A_4 are constants.

Table III. Time-Dependent Variables Generated by the Runge-Kutta Procedure[a,b]

i	t_i (hr)	G (mg)	k_a (hr^{-1})	C_p (μg/ml)
1	0	75.0	0.3092	0
2	0.25	64.6909	0.8288	1.1616
3	0.5	48.8621	1.4496	2.7103
4	1.0	15.2885	3.1992	5.1988
5	1.5	2.1823	4.3524	4.9576
6	2.0	0.2677	3.7487	3.8694
7	3.0	0.0308	0.9961	2.3532
8	4.0	0.0153	0.6140	1.4732
9	6.0	0.0039	0.7425	0.5847
10	9.0	0.0003	0.9093	0.1464
11	12.0	0.0	1.0473	0.0366
12	16.0	0.0	1.2093	0.0058

[a] Parameter used: dose = 75 mg; $V = 8$ liters, $k_{12} = 0.4\ \text{hr}^{-1}$, $k_{21} = 1.8\ \text{hr}^{-1}$, $k_{10} = 0.6\ \text{hr}^{-1}$, $A_1 = 4.0\ \text{hr}^{-1}$, $A_2 = 1.0\ \text{hr}^{-1}$, $A_3 = 1.6$, $A_4 = 0.3\ \text{hr}^{-3/2}$.

[b] The G.E. time-sharing numerical integration program RKPBI/KKPB2 (10) was employed. Step size was set to vary with sampling interval and equaled $0.0625h_i$.

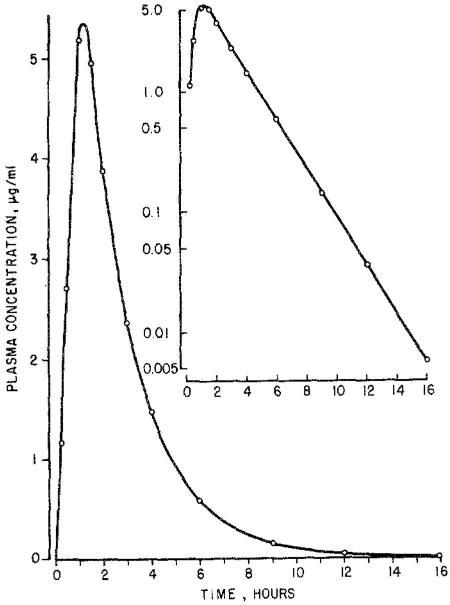

Fig. 2. Time course of simulated plasma levels. Circles represent sampled values.

In the present context, such an assumption also defies a closed-form solution to the rate equations. Therefore, solutions of differential equations were obtained by the fourth-order Runge–Kutta procedure (9). Actual values of model parameters and the generated time-dependent quantities are shown in Table III. The time course of the plasma concentrations is illustrated in Fig. 2.

Using the 12 values of C_p "sampled" at times indicated in the table, areas under the curve were calculated by each of the four methods. Table IV lists the incremental and the total areas. In order to eliminate the large end error discussed in the previous example, plasma levels during the last interval were assumed to decline monoexponentially, and equation 5 was used to calculate the last increment in all computations. This was adapted as a standard procedure since, in actual practice, sampling of plasma concentrations usually terminates in the log linear phase. Thus, in the spline method, the following equation was used in place of equation 27, which fulfills the continuity and smoothness of requirements at the $(n\text{-}1)$th point:

$$\frac{h_{n-1}}{6}\ddot{y}_{n-2}+\frac{h_{n-1}}{3}\ddot{y}_{n-1}=\frac{y_{n-1}}{h_n}\ln\left(y_n/y_{n-1}\right)-\frac{(y_{n-1}-y_{n-2})}{h_{n-1}} \qquad (35)$$

Comparisons of the algorithm errors are given in Fig. 3. These values were obtained by comparing the numerically calculated incremental areas with the corresponding theoretical areas. It is apparent that the largest errors were produced by the linear trapezoidal method. In particular, there were large negative deviations around the peak and large positive deviations elsewhere. While significant error reductions were observed with both the Lagrange and the spline methods, the smallest absolute errors were produced by the latter. It should be noted that the log trapezoidal method produced the best area estimates after the fourth hour. Figure 2 shows that the curve practically declines log-linearly after 4 hr. Since the log trapezoidal method can best approximate the nature of the data, it should be superior to any of the other three empirical methods. One could use a combination procedure of applying the spline method for $t \leq 4$ hr and the log trapezoidal method for $t > 4$ hr. The last column of Table IV shows the excellent result by such a combination.

Example 4. Twelve sets of data were generated by introducing simulated random experimental errors, corresponding to a coefficient of variation of 10%, into the sampled C_p values shown in Table III. These values were then rounded off to retain only two decimal places, as shown in Table

Table IV. Incremental Areas and the Corresponding Algorithm Errors for Example 3

t_i	Theoretical	Area on subinterval $[t_i, t_{i-1}]^{a,b}$				
		Linear trapezoidal	Log trapezoidal	Lagrange	Spline	Spline and log trapezoidal
0.25	0.1340	0.1452 (8.38)	0.1452 (8.38)	0.1371 (2.35)	0.1327 (−0.96)	0.1327 (−0.96)
0.50	0.4794	0.4840 (0.96)	0.4570 (−4.70)	0.4805 (0.24)	0.4804 (0.20)	0.4804 (0.20)
1.0	2.0792	1.9773 (−4.90)	1.9102 (−8.13)	2.0431 (−1.74)	2.0631 (−0.77)	2.0631 (−0.77)
1.5	2.6259	2.5391 (−3.31)	2.5386 (−3.13)	2.6136 (−0.47)	2.6293 (0.13)	2.6293 (0.13)
2.0	2.2045	2.2068 (0.10)	2.1955 (−0.41)	2.2254 (0.95)	2.2071 (0.12)	2.2071 (0.12)
3.0	3.0361	3.1113 (2.48)	3.0487 (0.42)	3.0461 (0.33)	3.0314 (−0.16)	3.0314 (−0.16)
4.0	1.8771	1.9132 (1.92)	1.8790 (0.10)	1.8710 (−0.33)	1.8781 (0.06)	1.8781 (0.06)
6.0	1.9223	2.0579 (7.05)	1.9230 (0.04)	1.9023 (−1.04)	1.9179 (−0.23)	1.9230 (0.04)
9.0	0.9495	1.0967 (15.49)	0.9496 (0.01)	0.9097 (−4.19)	0.9450 (0.48)	0.9496 (0.01)
12.0	0.2377	0.2745 (15.50)	0.2376 (0.04)	0.2210 (−7.02)	0.2378 (0.08)	0.2376 (0.04)
16.0	0.0668	0.0668 (0.0)	0.0668 (0.0)	0.0668 (0.0)	0.0668 (0.0)	0.0668 (0.0)
Total	15.6125	15.8725 (1.7)	15.3512 (−1.7)	15.5165 (−0.6)	15.5896 (−0.1)	15.5991 (−0.1)

[a] Values in parentheses are percent errors.
[b] Last increment calculated from equation 5.

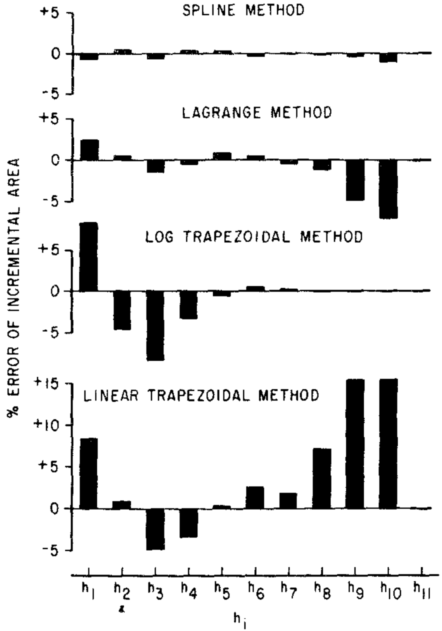

Fig. 3. Algorithm errors produced by the four numerical methods. Errors for the last increment (h_{11}) were obtained by using equation 5.

Table V. Twelve Sets of Simulated Plasma Concentrations (μg/ml) Containing Random Errors

Test	Sampling time (hr)											
set	0	0.25	0.5	1	1.5	2	3	4	6	9	12	16
1	0	1.09	3.19	5.34	5.33	3.96	2.32	1.47	0.51	0.15	0.04	0.01
2	0	1.17	3.20	5.13	4.38	3.19	2.31	1.18	0.57	0.17	0.04	0.01
3	0	1.07	2.87	5.49	5.44	4.06	1.97	1.31	0.44	0.15	0.04	0.01
4	0	1.08	2.27	6.08	5.35	4.56	2.53	1.26	0.57	0.12	0.04	0.01
5	0	1.20	2.94	5.00	4.15	3.93	2.77	1.79	0.59	0.16	0.04	0.01
6	0	1.02	2.45	5.17	4.41	3.70	2.38	1.36	0.62	0.15	0.03	0.01
7	0	1.17	2.08	4.63	4.27	3.83	2.63	1.40	0.52	0.15	0.04	0.01
8	0	1.30	2.99	4.58	5.02	3.39	2.18	1.63	0.63	0.16	0.03	0.01
9	0	1.21	3.05	5.67	5.74	4.06	2.01	1.29	0.54	0.16	0.04	0.01
10	0	1.30	2.82	5.13	4.77	3.87	2.20	1.54	0.63	0.16	0.04	0.01
11	0	1.31	2.82	5.78	5.31	4.46	2.69	1.65	0.55	0.15	0.04	0.01
12	0	1.27	2.91	5.13	4.98	4.51	2.52	1.52	0.54	0.13	0.04	0.01

Table VI. Cumulative Areas Calculated by the Three Numerical Methods

Test set	Cumulative area (μg–hr/ml)			Percent output error		
	Linear trapezoidal	Lagrange	Spline	Linear trapezoidal	Lagrange	Spline
1	16.170	15.806	15.829	3.57	1.24	1.38
2	14.802	14.443	14.553	−5.19	−7.49	−6.79
3	15.485	15.162	15.196	−0.82	−2.88	−2.67
4	16.608	16.187	16.417	6.38	3.68	5.15
5	16.482	16.152	16.243	5.57	3.46	4.04
6	15.277	14.933	15.073	−2.15	−4.35	−3.46
7	15.022	14.631	14.788	−3.79	−6.29	−5.28
8	15.587	15.259	15.187	−0.17	−2.27	−2.72
9	16.118	15.774	15.807	3.24	1.03	1.25
10	15.947	15.657	15.704	2.14	0.28	0.59
11	17.412	17.013	17.125	11.52	8.97	9.69
12	16.533	16.196	16.291	5.90	3.74	4.34
Mean ± SD	15.954 ± 0.754	15.601 ± 0.745	15.684 ± 0.755	2.18	−0.07	0.46

V. Cumulative areas and the corresponding output errors were calculated using the three empirical procedures identical to those of Example 3.[2]

Two important observations are found in the results given in Table VI. First, almost without exception, areas calculated by the spline method were higher than those of the Lagrange method and lower than those of the trapezoidal method. Second, the output errors of each data set were similar in magnitude and in sign regardless of the algorithm, and the average cumulative areas were comparable to those calculated from error-free C_p values (Table IV).

With regard to the first observation, an inspection of the results shows that, for a given set of data, output errors by each of the three methods are either all positive or all negative. Among those with negative errors, invariably the smallest deviations were produced by the trapezoidal method. Since the loci of points are predominantly concave upward, positive biases associated with the trapezoid method (+1.7%) tend to compensate best the negative effect of input errors. The reason that Lagrange interpolations consistently produced the smallest deviations among those with positive errors may be in part related to the negative bias associated with the Lagrange method (−0.6%) in the present example.

With regard to the second observation, simulated errors in each data set were not completely balanced. As a result, a given set would be composed of either more positively deviated values and fewer negatively deviated values or *vice versa*. These errors would inevitably contribute to the output errors, regardless of the algorithm used. It is also obvious that the exact allocation of the random errors would also affect the actual output values. However, with increasing number of data set, the mean areas and output errors should convèrge, in a stochastic manner, to corresponding limiting values that are characteristic of the algorithm. The magnitude of standard deviation, on the other hand, should reflect the scattering of data only. Comparing Tables IV and VI, these appear to be the case,

Monitoring of the interpolated curves revealed no unusual oscillations in any of the 12 data sets. An examination of such behavior is given in the next example.

Example 5. Table VII is based on the data tabulated in Table III with three modifications. First, the data point at 9 hr is deleted as might occur when a sample is missed. Second, the data are rounded off to two decimal places only. Third, the data point at 4 hr is given a positive deviation (1.59 instead of 1.47 μg/ml). Columns 3–5 include the linear trapezoidal, log

[2] Two Fortran subroutines implementing the Lagrange and spline methods are available on request.

Table VII. Summary of Example 5[a]

t (hr)	C_p (μg/ml)	Linear trapezoidal	Log trapezoidal	Spline	Combined spline and log trapezoidal
0.25	1.16 (−0.1)	0.1450 (8.2)	0.1450 (8.2)	0.1324 (−1.2)	0.1324 (−1.2)
0.5	2.71 (−0.0)	0.4825 (0.6)	0.4557 (−4.9)	0.4801 (0.1)	0.4801 (0.1)
1	5.20 (0.0)	1.9750 (−5.0)	1.9072 (−8.3)	2.0634 (−0.8)	2.0634 (−0.8)
1.5	4.96 (0.1)	2.5400 (−3.3)	2.5395 (−3.3)	2.6302 (0.2)	2.6302 (0.2)
2	3.87 (0.0)	2.2075 (0.1)	2.1962 (−0.4)	2.2083 (0.2)	2.2083 (0.2)
3	2.35 (−0.1)	3.1100 (2.4)	3.0471 (0.4)	3.0227 (−0.4)	3.0227 (−0.4)
4	1.59 (7.9)	1.9700 (4.9)	1.9453 (3.6)	1.9339 (3.0)	1.9339 (3.0)
6	0.58 (−0.8)	2.1700 (12.9)	2.0031 (4.2)	2.0930 (8.9)	2.0031 (4.2)
9	N.S.[b]	—	—	—	—
12	0.04 (9.3)	1.8600 (56.7)	1.2116 (2.1)	0.7812 (−34.2)	1.2116 (2.1)
16	0.01 (72.4)	0.0866 (29.6)	0.0866 (29.6)	0.0866 (29.6)	0.0866 (29.6)
Total		16.5466 (6.0)	15.5373 (−0.5)	15.4317 (−1.2)	15.7723 (1.0)

[a] Values in parentheses are percent errors.
[b] No sample.

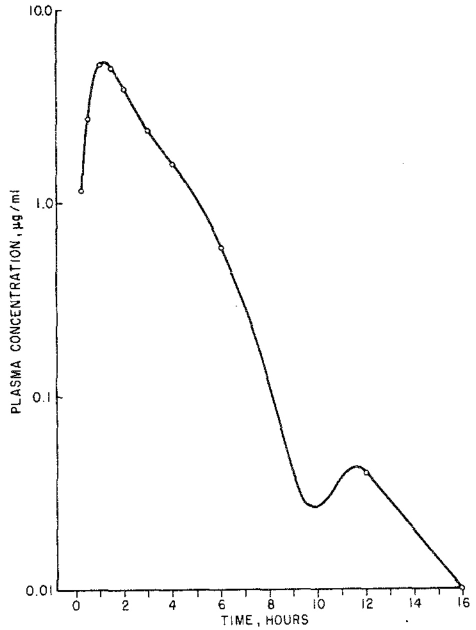

Fig. 4. Example showing the production of spurious oscillation between 6 and 12 hr.

trapezoidal, and the spline areas, respectively. Note the large negative error of the spline area for the interval between 6 and 12 hr and the relatively small error with the log trapezoidal area. Column 6 is a combined method wherein the values at $t \leq 4$ hr are calculated using the spline procedure (column 5), while subsequent ones are calculated by the log trapezoidal method. The results are excellent as seen when one examines the percent error of each incremental area and the cumulative error listed at the bottom of the column. Thus, for best results, data should be examined on a semilogarithmic scale so as to detect the onset of terminal log-linear phase. The spline method may be used for the early and middle segments of the curve, while the log trapezoidal method may be used during monoexponential decline.

The cause of large negative deviation by the spline method is illustrated in Fig. 4 for this example. It is apparent that the large minimum produced at 10 hr is extraneous. As stated earlier, such occasional oscillation can occur when cubic polynomials are used. Possible ways to avoid this problem through splines are being investigated.

DISCUSSION

Four numerical integration methods have been described for area calculations. These methods are empirical, do not require equally spaced data points, and with the exception of the log trapezoidal method make no assumptions regarding the underlying mechanism of the data. Of the four, it is apparent that either the Lagrange or the spline method is probably less biased than either of the trapezoidal methods. This may be attributed to the greater flexibility of cubic polynomials in approximating various local behaviors often seen in actual data. However, such flexibility can occasionally produce unwanted oscillations, as exemplified in Fig. 4. The major advantage of the spline function over the Lagrange polynomials is the complete smoothness of the fitted curve. If data are functionally smooth and error free, it is apparent that the spline function will be the best approximation to the system.

Theoretically, the existence of experimental errors, no matter how small, will render the data discontinuous. Since errors are experimentally inevitable, the superiority afforded by the spline function may become less certain, and this uncertainty may increase with increasing noise in the data. For any given data set, any of the four methods may be superior to the others. However, because of its small inherent algorithm errors, areas calculated by the spline method should on the average be less distorted and more suitable for further data analysis (11), particularly when combined with the log trapezoidal method.

The Legrange method, on the other hand, has the advantage of being computationally less complex, yet comparable in accuracy. With the spline method, parameters required for area calculations are obtainable only by solving a set of n simultaneous equations. Depending on the magnitude of n, such operations may require the aid of large computers. With the Lagrange method, the number of equations is reduced to four or three and can usually be solved with less sophisticated facilities such as programmable electronic calculators (12).

In using either the Lagrange or the spline method, it is required that the data be functionally smooth over the interval $[t_1, t_n]$. In cases where data are only piecewise smooth, the procedure must be applied to individual segments that meet this requirement. For example, at the termination of a constant intravenous infusion, there will be an abrupt change in plasma concentrations, forming two distinctly smooth segments: one during infusion and the other after infusion.

Since both the spline and the Lagrange methods may occasionally produce spurious and unrealistic oscillations, they should not be used blindly. In practice, the tendency of such occurrences increases with increasing noise levels in the data. Inserting hypothetical values will in general have a stabilizing influence, but will introduce bias.

While the two trapezoidal methods are less accurate, they may be the logical choice in some cases because of their simplicity. They are also particularly suitable when area estimates are the end product and are to be compared among data which have similar shapes and sampling schemes. The log trapezoidal method is especially useful for data in exponential decline.

ACKNOWLEDGMENTS

The authors are grateful to Dr. R. L. Davis for helpful discussions on Example 4, and to Dr. S. Riegelman for many constructive comments on the manuscript.

REFERENCES

1. B. Carnahan, H. A. Luther, and J. O. Wilkes. *Applied Numerical Methods*, Wiley, New York, 1969, pp. 71–72.
2. D. D. McCracken and W. S. Dorn. *Numerical Methods and FORTRAN Programming*, Wiley, New York, 1964, pp. 161–163.
3. J. H. Ahlberg, E. N. Nilson, and J. L. Walsh. *The Theory of Splines and Their Applications*, Academic Press, New York, 1967.
4. S. Wold. Spline functions in data analysis. *Technometrics* **16**:1–11 (1974).
5. B. Carnahan, H. A. Luther, and J. O. Wilkes. *Applied Numerical Methods*, Wiley, New York, pp. 27–34.

6. F. Scheid. *Schaum's Outline of Theory and Problems of Numerical Analysis*, McGraw-Hill, New York, 1968, p. 80.
7. T. N. E. Greville. Spline functions, interpolation, and numerical quadrature. In A. Ralston and H. W. Wilf (eds.), *Mathematical Methods for Digital Computers*, Vol. II, Wiley, New York, 1967, pp. 156–168.
8. L. G. Dunfield and J. F. Read. Determination of reaction rates by the use of cubic spline interpolation. *J. Phys. Chem.* **57**:2178–2183 (1972).
9. B. Carnahan, H. A. Luther, and J. O. Wilkes. *Applied Numerical Methods*, Wiley, New York, p. 361.
10. *User's Guide*: *Numerical Analysis Routines*, General Electric, Bethesda, Md., 1971, p. 358.
11. K. C. Yeh, Kinetic parameter estimation by numerical algorithms and multiple linear regression: theoretical. *J. Pharm. Sci.* **66**:1688–1691 (1977).
12. HP-67 Math Pac. Hewlett-Packard, Cupertino, Calif., 1976.