FOURTH AUSTRALASIAN CONFERENCE

on

HYDRAULICS AND FLUID MECHANICS

at

Monash University, Melbourne, Australia

1971 November 29 to December 3

# NUMERICAL INTEGRATION OF THE EQUATION OF GRADUALLY VARIED AND SPATIALLY VARIED FLOW

by

C.J. APELT

SUMMARY

Three methods for numerical integration of the differential equation for gradually varied and spatially varied flow are compared and a detailed comparison is made between them. The methods described are the Standard Fourth Order Runge-Kutta Process, the Kutta-Merson process and the Trapezoidal method. All three methods are shown to be capable of integrating upstream or downstream regardless of whether the flow is subcritical or supercritical. The two Runge-Kutta processes are more accurate than the Trapezoidal method and they also have better stability characteristics. The results of comparative computations show that both Runge-Kutta processes require less computational effort than does the Trapezoidal method in computing flow profiles to the same accuracy. The Kutta-Merson process has a further advantage in that it includes a direct means for automatic control of accuracy of solution and of length of interval for integration.

University of Queensland.

## INTRODUCTION

The governing equation for steady, gradually varied flow in open channels is a non-linear first order differential equation which can be integrated by analytical methods only under very restricted conditions. The importance of the phenomena of gradually varied flow in both natural and man-made systems has led to the development of a number of approaches to approximate numerical solution of the equation. Those methods of computation described in treatises on open channel hydraulics such as Chow (1) and Henderson (2) have been developed primarily for desk calculation. However, integration of the gradually varied flow equation is a task particularly suited to the automatic computer and it is important to develop an efficient, reliable method of integration for use with the computer.

A form of the gradually varied flow equation which includes the effects of changes in channel cross-section with distance and of lateral inflow or outflow (spatially varied flow) is,

$$\frac{dy}{dx} = \frac{S_o - S_e + \frac{\beta Q^2 y}{gA^3}\frac{\partial B}{\partial x} - \frac{\delta \beta Q}{gA^2}\frac{dQ}{dx}}{1 - \frac{\beta Q^2 B}{gA^3}} \qquad (1)$$

in which y is the depth of flow, x is the distance along the channel, $S_o$ is the slope of the channel bed, $S_e$ is the energy gradient, Q is the discharge, A is the cross-sectional area of flow, B is the top width of the cross-section of flow, g is the gravitational constant, $\beta$ is the momentum coefficient which accounts for variations in velocity across the cross-section and $\delta$ takes the values, 2 for lateral inflow normal to the direction of stream flow and 1 for lateral outflow, respectively. Different values of $\delta$ are appropriate to more complex conditions of lateral inflow. The coefficient of $\frac{\partial B}{\partial x}$ used in Eq. 1 is that for a cross-section in the shape of a rectangle, triangle or trapezium; for other cross-sectional shapes the coefficient is halved (3). If the discharge and the cross-sectional shape do not vary with distance, Eq.(1) simplifies to

$$\frac{dy}{dx} = \frac{S_o - S_e}{1 - \frac{\beta Q^2 B}{gA^3}} \qquad (2),$$

which is the form commonly referred to as the gradually varied flow equation.

## NUMERICAL SOLUTION

The equations (1) and (2) and all other versions of the gradually varied flow equation can be written in the form,

$$\frac{dy}{dx} = \phi(x,y) \qquad (3),$$

in which $\phi(x,y)$ is defined by the right-hand side of the equation. For a particular solution to be calculated, one condition needs to be specified and this will normally be the depth of flow corresponding to a specified value of x. A class of methods which is particularly suitable for direct solution of Eq. (3) when it is non-linear is that which is described as the Runge-Kutta method. There are many variants of the Runge-Kutta method but they are all derived from the Taylor's series method, which simply evaluates $y(x\pm\Delta x)$ from the values of $y(x)$ and the derivatives of $y(x)$ as follows:-

$$y(x\pm\Delta x) = y(x) \pm \Delta x\, y'(x) + \frac{\Delta x^2}{2!} y''(x) \pm \frac{\Delta x^3}{3!} y'''(x) + \ldots. \qquad (4),$$

where primes denote differentiation with respect to x. The summation on the right of Eq. (4) is convergent provided the interval, $\Delta x$, is within the radius of convergence of the series and, in such circumstances, the summation can be evaluated to a specified accuracy by inclusion of an appropriate number of terms. The second and higher derivatives can be obtained by successive differentiation of the differential equation (3) and, if $y(x)$ is known, then all derivatives of $y(x)$ can be evaluated and $y(x\pm\Delta x)$ can be computed from Eq. (4). A disadvantage of the Taylor's series method is the difficulty which is often encountered in the formation of the higher derivatives of y and it is particularly acute if one attempts to use the method for integration of the gradually varied flow equation. Runge-Kutta methods avoid this difficulty by replacing Taylor's series with an expression, involving only $y(x)$ and the first derivative of y evaluated at a number of values of x over the interval of integration, which is equivalent to the Taylor's series up to a certain order of term.

The writer has applied two Runge-Kutta processes to the integration of the gradually varied flow equation by automatic computer and has obtained excellent results with each. The methods used, the Standard Fourth Order Runge-Kutta Process and the Kutta-Merson process, are both fourth-order processes, i.e. they match the Taylor's series method exactly up to and including the term in $(\Delta x)^4$.

If the notation is adopted that $x_{i+1} \equiv x_i + \Delta x$ and $y_i \equiv y(x_i)$, the two Runge-Kutta processes as applied to Eq. (3), can be expressed as follows (Refs (4),(5)):-

*Standard Fourth Order Runge-Kutta Process (SRK)*

$$y_{i+1} = y_i + (k_1 + 2k_2 + 2k_3 + k_4)/6 \qquad (5)$$

in which

$$k_1 = \Delta x\ \phi(x_i, y_i),$$
$$k_2 = \Delta x\ \phi(x_i + 0.5\Delta x,\ y_i + 0.5\ k_1),$$
$$k_3 = \Delta x\ \phi(x_i + 0.5\Delta x,\ y_i + 0.5\ k_2),$$
$$k_4 = \Delta x\ \phi(x_i + \Delta x,\ y_i + k_3).$$

The truncation error in (5) is of order $(\Delta x)^5$.

*Kutta-Merson Process (KM)*

$$y_{i+1} = y_i + (k_1 + 4k_4 + k_5)/2 \qquad (6)$$

in which

$$k_1 = \Delta x\ \phi(x_i,\ y_i)/3,$$
$$k_2 = \Delta x\ \phi(x_i + \Delta x/3,\ y_i + k_1)/3,$$
$$k_3 = \Delta x\ \phi(x_i + \Delta x/3,\ y_i + 0.5\ k_1 + 0.5\ k_2)/3,$$
$$k_4 = \Delta x\ \phi(x_i + \Delta x/2,\ y_i + 0.375\ k_1 + 1.125\ k_3)/3,$$
$$k_5 = \Delta x\ \phi(x_i + \Delta x,\ y_i + 1.5\ k_1 - 4.5\ k_3 + 6\ k_4)/3.$$

The truncation error in (6) is also of order $(\Delta x)^5$ and an estimate of its value is given by

$$\varepsilon = 0.2\ k_1 - 0.9\ k_3 + 0.8\ k_4 - 0.1\ k_5 \qquad (7).$$

In the use of each algorithm, the $k_1$, $k_2$, etc, are computed in order and their use in the appropriate integration formula, (5) or (6), gives the value of $y_{i+1}$ directly. Since the integration requires the knowledge only of the value of $y_i$ at the beginning of the interval, the value of $\Delta x$ can be changed at will from one interval to the next to meet requirements of accuracy, etc. The SRK process involves the evaluation of $\phi(x,y)$ four times for each interval of integration and the KM process requires five such evaluations. However, the latter process provides a direct estimate of the truncation error from Eq. (7) and this is used to determine whether the interval, $\Delta x$, is satisfactory for the accuracy of integration sought or whether it should be decreased or increased. Interval and accuracy control in the KM process is achieved automatically as follows:- if $\varepsilon$ is greater than a preassigned limit on truncation error the interval is halved and computation is begun again from $x_i$; if $\varepsilon$ is less than 1/32 of the preassigned limit the interval is doubled. There is no direct method available for automatic interval and accuracy control with the SRK process but the writer has had satisfactory results using for interval control a numerical estimate of the truncation over two steps of length $\Delta x$. This estimate is obtained by comparing the value of $y_{i+2}$ given by the Runge-Kutta process with that produced by Simpson Rule quadrature as follows:-

$$y_{i+1} = y_i + \Delta x\{\phi(x_i,y_i) + 4\phi(x_{i+1},y_{i+1}) + \phi(x_{i+2},y_{i+2})\}/3 \qquad (8).$$

## COMPUTATIONAL EXPERIENCE

Both Runge-Kutta processes described above have been found to be very successful in all cases to which they have been applied. A fairly extensive set of computations has been carried out for integration of Eq. 2 and some details of these are provided below. In summary, it has been established that, for both methods, integration can proceed either upstream or downstream and produce excellent results regardless of whether the flow is subcritical or supercritical. For a fixed interval of integration the SRK process achieved a specified accuracy of integration with slightly less computational expense that did the KM process in those cases where the comparison was made. However, if automatic control of accuracy and interval length is employed, the KM process has been found to require significantly less computational effort than does the SRK process, and for this reason the KM is the better process to use in a general purpose computer programme for integration of the varied flow equation. Experience with integration of the more general Eq. (1) is less extensive but in those cases studied to date the performance of the two methods has been similar to that described above.

## COMPARISON WITH OTHER WORK

At a stage when this work was well advanced several workers published details of research into procedures for computer integration of the varied flow equation (refs (6),(7),(8)). Both studies referred to made use of the same method of integration, described as the trapezoidal method of integration (6) and as the depth-step method (8), which can be expressed as,

$$y_{i+1} = y_i + 0.5\ \Delta x\{\phi(x_i,y_i) + \phi(x_{i+1},y_{i+1})\} \qquad (9).$$

This implies an iterative calculation and the r-th cycle of iteration is given by,

$$y_{i+1}^{(r)} = y_i + 0.5\ \Delta x\{\phi(x_i,y_i) + \phi(x_{i+1},y_{i+1}^{(r-1)})\} \qquad (10).$$

For the first cycle, $y_{i+1}^{(r-1)}$ is set equal to $y_i$ and the iteration continues until successive values of $y_{i+1}$ agree to some specified tolerance. An alternative procedure to that given by Eq. (10) is to solve the Eq. (9) by the method described as "interval halving" or "bisection" (See e.g. Hamming (9)), and this procedure possesses advantages over that given in Eq. (10) in certain conditions (8).

The integration procedure defined by Eq. (9) is essentially the same as that used in the Standard Step method for calculating flow profiles* (Chow (1)) and is closely related to a second order Runge-Kutta process. In fact the result of the second cycle of iteration in Eq.(10) is exactly that given by a second order Runge-Kutta process. The truncation error in Eq.(9) is of order $(\Delta x)^3$.

It has been established (6),(8) that the trapezoidal method of integration (TRAP) possesses the same flexibility as that already described for the SRK and KM processes in that integration can proceed upstream or downstream regardless of the type of flow profile being computed. It is, therefore, of interest to compare these three methods in order to determine whether there are other grounds for preferring any one of them. The details of the comparison are given in following sections.

## STABILITY ANALYSIS

In the process of numerical integration of Eq.(1), if the initial values at the beginning of a step are known exactly and if the integration formula is computed exactly, the error in the calculated solution at the end of the step is given by the truncation error. Since the computations are done with finite arithmetic there will be an additional error, the rounding error, due to rounding off. During the process of integration over an extended number of steps the numerical solution at each point will contain the effects of truncation and rounding errors from all previous steps. A tendency for such errors to grow rapidly in magnitude in such a manner as to swamp the true nature of the solution being sought gives rise to "instability" (4). It is not possible with stability theory in its current state of development to carry out a complete stability analysis for the numerical integration of a non-linear equation such as Eq.(1) and the best that can be done is to consider small ranges of x in which Eq.(1) is replaced by the linear approximation,

$$\frac{dy}{dx} = \phi(x_n,y_n) + (x-x_n)\frac{\partial\phi}{\partial x}(x_n,y_n) + (y-y_n)\frac{\partial\phi}{\partial y}(x_n,y_n) \qquad (11).$$

The inhomogeneous terms in Eq.(11) are unimportant from the point of view of stability and the short term stability of Eq.(1) will be the same as that of

$$\frac{dy}{dx} = L\,y, \quad \text{where } L \equiv \frac{\partial\phi}{\partial y} \qquad (12).$$

In the analysis it is assumed that the range of x under consideration if small enough so that L can be taken as approximately constant.

When applied to Eq.(12), the three integration processes under discussion are equivalent to the following difference equations:-

$$\text{SRK, Eq.(5)};\quad y_{i+1} = y_i(1 + Z + Z^2/2 + Z^3/6 + Z^4/24) \qquad (5a),$$

---

* The Standard Step Method as normally employed actually integrates the equation for the elevation of the total head line.

$$\text{KM, Eq.(6)};\quad y_{i+1} = y_i(1 + Z + Z^2/2 + Z^3/6 + Z^4/24 + Z^5/144) \qquad (6a),$$

$$\text{TRAP, Eq.(9)};\quad y_{i+1} = y_i(1 + Z/2)/(1 - Z/2) \qquad (9a),$$

where $Z \equiv L\,\Delta x$.

The complementary function in the solution of the difference equations is, in each case, $c\lambda^i$, where $\lambda$ is the coefficient of $y_i$ in the appropriate difference equation. The complementary function in the solution of the error equation in each case is identical to that of the difference equation. It follows that, if $|\lambda| > 1$, the magnitude of the error will grow without bound. If at the same time the true solution is decreasing in magnitude, instability results and the computed solution will depart more and more from the true solution in such a way as to render the computed solution useless. The variation of $\lambda$ with Z for the three systems is shown in Figure 1.

For $Z < 0$, which implies that the true solution decreases in the direction of integration, the following restrictions can be deduced from Figure 1:-

SRK: For stability, $|Z| < 2.8$ approx.

KM : For stability, $|Z| < 3.6$ approx. In addition the numerical solution will contain oscillatory components unless $|Z| < 2.4$ approx.

TRAP: There is no limit for stability but the numerical solution will contain oscillatory components unless $|Z| < 2.0$.

For $Z > 0$, $\lambda > 1$ in all cases and the error will grow without limit. However, a positive Z implies that the true solution also increases in the direction of integration and the important question is how fast the error grows relative to the true solution. The local behaviour of the complementary function in the true solution will be similar to that of exp(Lx), i.e. of $\{\exp(Z)\}^i$. Hence the important consideration is the magnitude of $\lambda/\exp(Z)$. The variation of $\lambda/\exp(Z)$ with Z is given in Figure 2 which shows that, as regards relative error growth, the KM process is a little better than the SRK process and both are significantly better than the TRAP method for all practical values of Z. The values of Z at which the relative error would increase tenfold after 100 equal integration steps are as follows:

KM : Z = +2.0 and -1.6

SRK : Z = +1.6 and -1.05

TRAP : Z = +0.65 and -0.65.

The results of the stability analysis summarised above show that the KM and SRK processes possess better stability characteristics than does the TRAP method for all practical values of Z. It needs to be emphasised that these are the results of a linearised analysis.

## COMPARISON OF COMPUTATIONAL EFFICIENCY

In all three methods of integration of Eq.(1) the computational effort is almost entirely made up of the repeated evaluations of $\phi(x,y)$ for different values of the arguments. The KM process involves five evaluations of $\phi(x,y)$ per step of integration; the SRK process involves four evaluations per step of integration; the TRAP method requires one such evaluation for each iteration after the first. Since the truncation error is of order $(\Delta x)^5$ for both KM and SRK processes as compared to order $(\Delta x)^3$ for the TRAP method and since the error growth characteristics are better for the KM and SRK processes than for the TRAP method (as shown in the stability analysis) it follows that the KM and SRK processes can employ a much larger interval of integration than can the TRAP method for the same accuracy of solution. This conclusion was tested by numerical experiment in which a number of flow profiles was computed with each method for a series of values of interval length. The number of evaluations, N, of $\phi(x,y)$ for each computation was recorded and this is taken as the measure of computational effort. The results of the experiment are summarised in Figures 3,4 and 5. The flow profiles computed to produce the data of Figures 3,4 and 5 are the same as some of those computed by McBean and Perkins (8) so that direct comparison can be made with their work. The accuracy of the numerical solutions was evaluated by comparing the computed profile with the Bresse solution, calculated to the same precision, all being done on the same computer. (The DEC PDP 10 system at the University of Queensland). In Figures 3,4 and 5 the error plotted is the magnitude of the difference between the computed profile and the Bresse solution.

For the M1 profile, Figure 3, the KM process used at fixed interval length is more efficient than the TRAP method but only marginally so. However, when automatic interval control is incorporated into the KM process, it is significantly more efficient than the TRAP Method,

requiring between one third and one half the computational effort of the TRAP method to compute a profile to the same accuracy. There is one exception to these general observations in that the TRAP method gave surprisingly good results for the largest interval of computation used for integration in the upstream direction. This exceptional result may be due to the small values of $\lambda$ for the TRAP method in the vicinity of Z = -2 (Figure 1), but it is difficult to see how one could make practical use of this. The results of Figure 3 also show that there are only small differences in the computational efficiency of each method for integration upstream and downstream.

For the M2 profile, Figure 4, the KM process at fixed interval length is about three times more efficient in computational effort than the TRAP method for integration in the upstream direction and up to eight times more efficient for integration in the downstream direction. With automatic interval control the efficiency of the KM process is increased further for integration in the upstream direction but is not much affected for integration downstream. For the profile computed for Figure 4, each method required significantly more computation to achieve a given accuracy when the integration was carried downstream than for integration in the upstream direction.

For the M3 profile, Figure 5, the KM process at fixed interval length is about twice as efficient as the TRAP method for both directions of integration. Use of automatic interval cotrol increases the efficiency of the KM process in producing the more accurate solutions but reduces its efficiency somewhat when less accurate solutions are computed. For integration in the upstream direction,each method required approximately twice as much computation as for downstream integration at the same accuracy. The distance computed downstream is less than that upstream.

For all profiles computed it was found that the shapes of the curves of error growth with distance were essentially similar for each method of integration and, consequently, comparison of the computational efficiency of the methods at different stations on the profiles from those used for producing the results in Figures 3,4 and 5 leads to essentially similar results.

The detailed results for the SRK process are not included; in summary, the SRK process was found to be approximately ten percent more efficient than the KM process at fixed interval length but generally less efficient than the KM process with automatic interval control.

## CONCLUSION

Three methods of integrating the varied flow equation have been described and each is capable of direct computation of flow profiles in the upstream or downstream directions regardless of whether the flow is subcritical or supercritical. The chief findings of a detailed comparative study of the three methods are summarised as follows:-

(1) The KM and SRK processes possess better stability characteristics than does the TRAP method.

(2) The KM and SRK processes have truncation errors of order $(\Delta x)^5$ whereas the TRAP method has a truncation error of order $(\Delta x)^3$. Consequently the KM and SRK processes are significantly more accurate than the TRAP method at the same interval length.

(3) When the integration is carried out to the same accuracy the KM and SRK processes require less computational effort than does the TRAP method.

(4) The KM process includes a direct means for automatic control of accuracy of solution and of interval length. Neither of the other methods possesses this feature.

## REFERENCES


1. Chow, V.T., *Open Channel Hydraulics*, McGraw Hill, N.Y., 1959.
2. Henderson, F.M., *Open Channel Flow*, Macmillan, N.Y., 1966.
3. Fox, J.A. and Goodwill, I.M., "Spatially varied flows in open channels", *Proc. I.C.E.*, 1970, 46, pp 311-325.
4. Fox, L. (Ed.), *Numerical Solution of Ordinary and Partial Differential Equations*, Pergamon, Oxford, 1962.
5. Lance, G.N., *Numerical Methods for High Speed Computers*, Iliffe, London, 1960.
6. Prasad, R., "Numerical Method of Computing Flow Profiles", *Proc. A.S.C.E., Jour.Hyd.Div.*, 1970, 96, HY1, pp 75-86.
7. Apelt, C.J., Discussion of Prasad, R., "Numerical Method of Computing Flow Profiles", *Proc.A.S.C.E.,Jour.Hyd.Div.*, 1970, 96, HY10, p 2134.
8. McBean, E.A. and Perkins, F.E., *Error Criteria in Water Surface Profile Computations*, Hydrodynamics Lab. Report No. 124, M.I.T., 1970.
9. Hamming, R.W., *Numerical Methods for Scientists and Engineers*, McGraw Hill, N.Y., 1962.

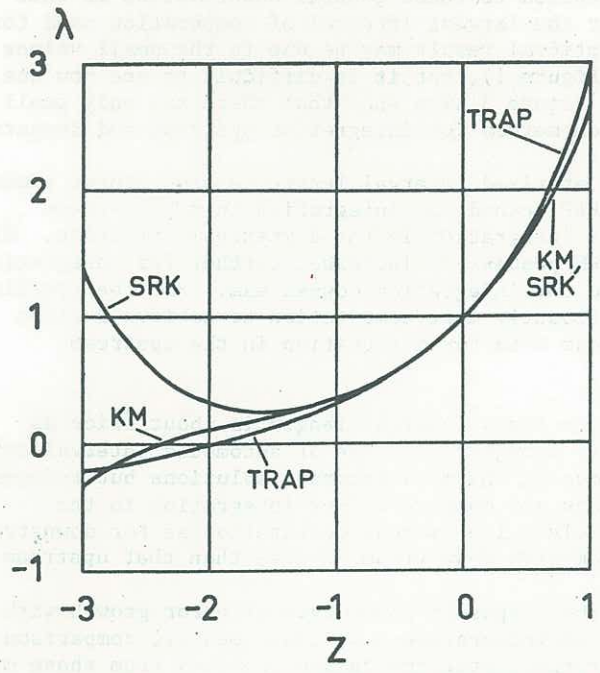



Fig.1 AMPLIFICATION FACTOR

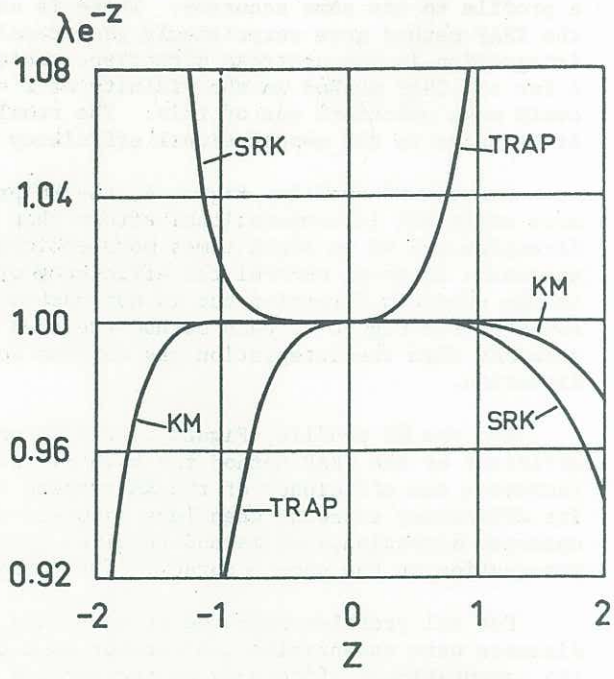
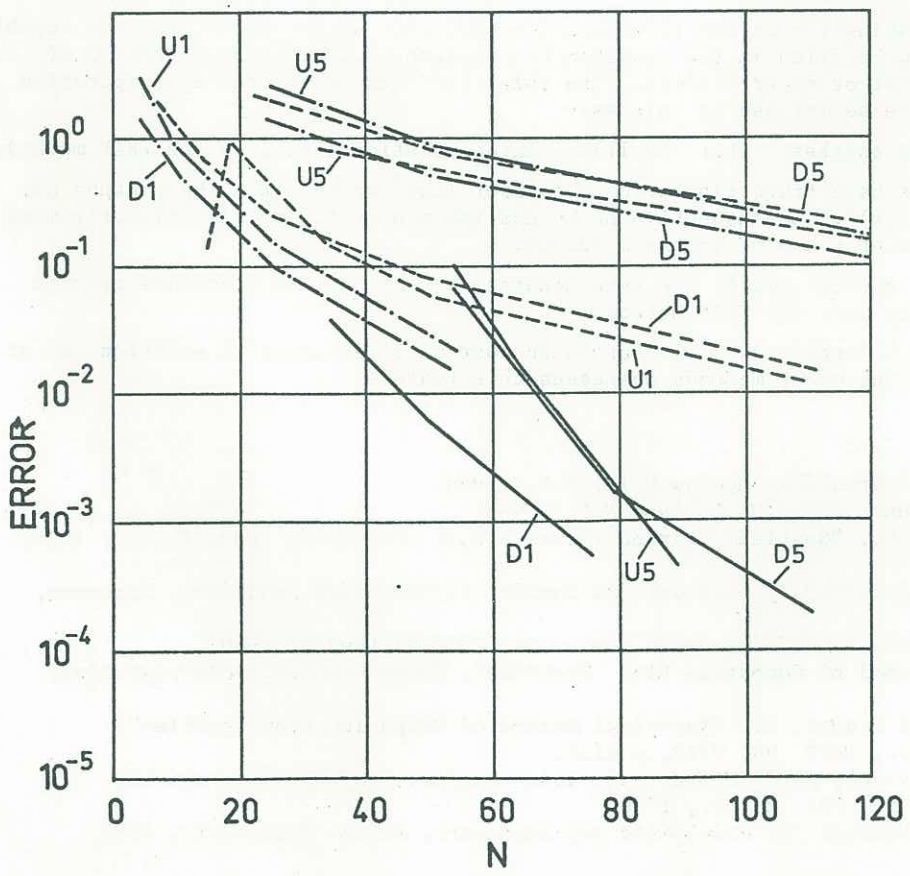



Fig.2 RELATIVE ERROR




| | |
|---|---|
| $S_0$ | 0.005 |
| CHEZY .C | 75 |
| Q | 100 CUSECS/ft |
| $y_N$ | 7.08439 ft |
| $y_c$ | 6.77407 ft |

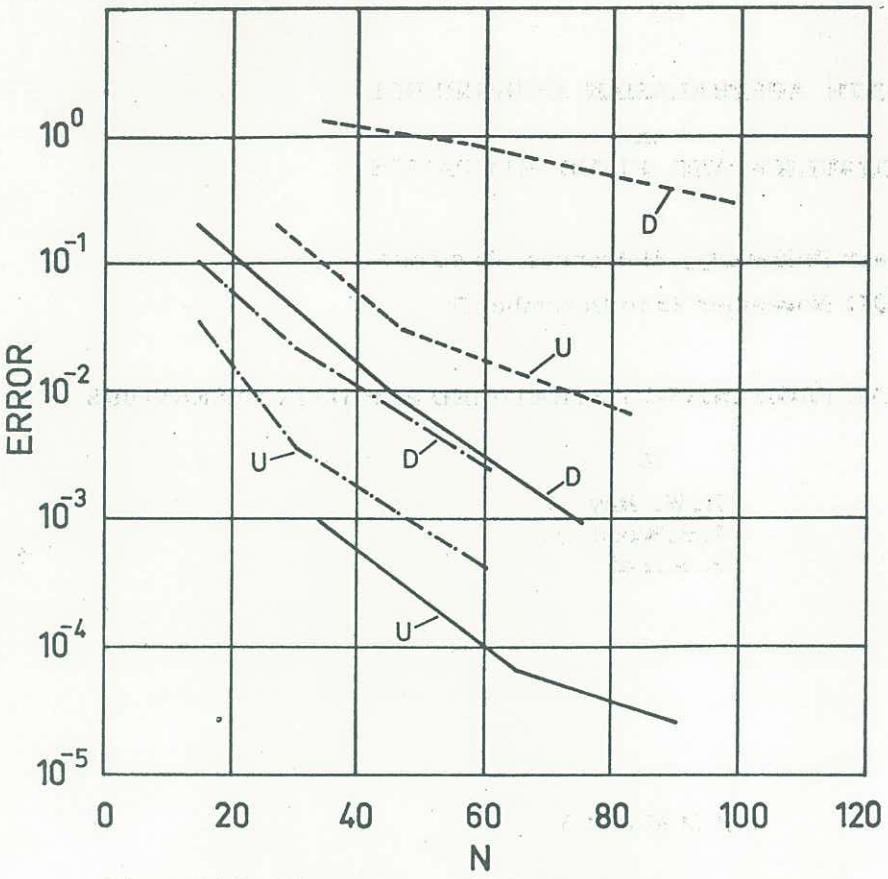
BRESSE SOLUTION

| x(ft) | y(ft) |
|---|---|
| 0 | 8.00000 |
| 10000 | 57.46863 |
| 20000 | 107.46379 |
| 30000 | 157.46276 |
| 40000 | 207.46238 |
| 50000 | 257.46220 |

LEGEND

KM (AUTO.)
KM (Δx FIXED)
TRAP
D1, x = 0 → 10000 ft
D5, x = 0 → 50000 ft
U1, x = 10000 → 0 ft
U5, x = 50000 → 0 ft

Fig.3 M1 PROFILE-COMPUTATIONAL EFFORT




| | |
|---|---|
| $S_0$ | 0.005 |
| CHEZY C | 35 |
| Q | 100 CUSECS/ft |
| $y_N$ | 11.77510 ft |
| $y_c$ | 6.77407 ft |

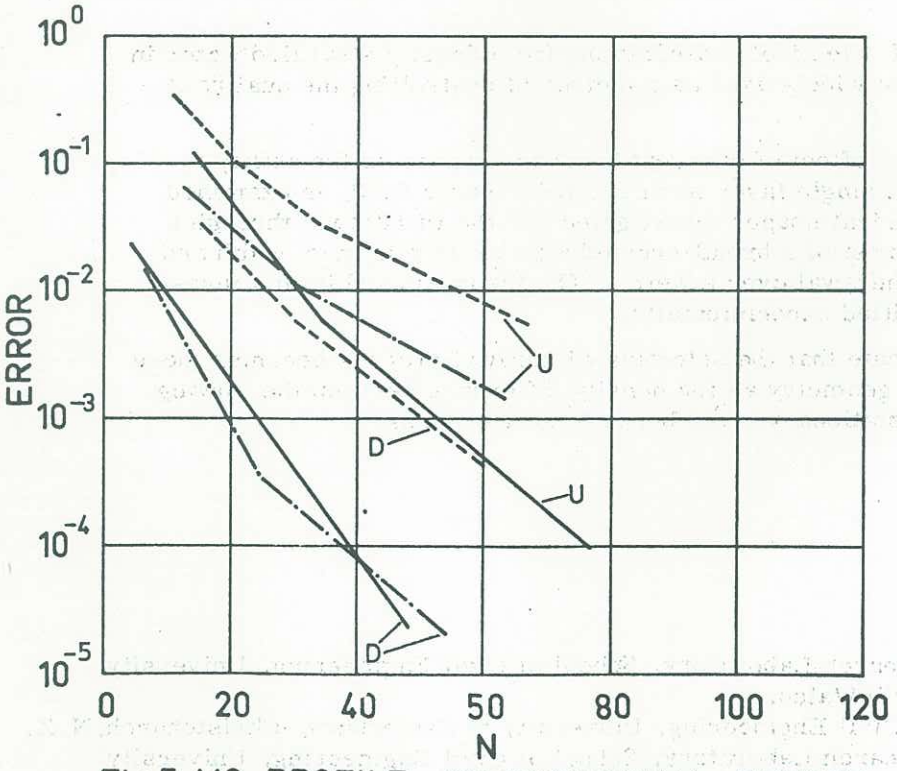
BRESSE SOLUTION

| x(ft) | y(ft) |
|---|---|
| 0 | 11.00000 |
| 200 | 10.61811 |
| 400 | 9.92757 |
| 500 | 9.28957 |
| 550 | 8.76577 |
| 600 | 7.72019 |

LEGEND

KM (AUTO.)
KM (Δx FIXED)
TRAP
D, x = 0 → 600 ft
U, x = 600 → 0 ft

Fig.4 M2 PROFILE - COMPUTATIONAL EFFORT




PROFILE PARAMETERS AS IN FIG 3

BRESSE SOLUTION

| x(ft) | y(ft) |
|---|---|
| -50 | 2.71112 |
| 0 | 3.00000 |
| 200 | 4.17000 |
| 400 | 5.39684 |
| 500 | 6.08540 |
| 550 | 6.53078 |

LEGEND

KM (AUTO.)
KM (Δx FIXED)
TRAP
D, x = 0 → 500 ft
U, x = 550 → -50 ft

Fig.5 M3 PROFILE - COMPUTATIONAL EFFORT