

# INTRODUCTION TO NUMERICAL METHODS

Lecture No. 1, June 3, 1954

B. Gilchrist

The intention of the planners of this workshop was to keep the lectures as free from mathematics as possible. The subject is, however, one which essentially involves a great deal of mathematics as no doubt the lectures have already demonstrated. The purpose of this lecture is to indicate very briefly some of the more important mathematical aspects, a knowledge of which is almost essential to any understanding of numerical weather prediction.

The fundamental mathematical assumption underlying all phases of numerical weather prediction is that we can replace a continuous derivative by its finite difference equivalent. Thus, considering the curve ABCDE in Fig. 1, we assume that the continuous derivation (slope of the tangent) at C can be replaced by the slope of the chord BD. If EF, CG and DH are perpendicular to the x-axis, the finite difference is said to be centered when  $FG = GH$ . In general we try to use centered differences whenever possible. The accuracy with which the finite difference equivalent will approach the true derivative clearly depends on the shape of the curve and on the location of the points B and D. Assume now that  $FG = GH = \Delta x$ . The centered finite difference approximations to the first and second derivatives at C are therefore  $(y_D - y_B)/2\Delta x$  and  $(y_D + y_B - 2y_C)/\Delta x$  respectively. The errors introduced by using these finite differences instead of the continuous derivatives are known as "truncation" errors.

Consider now a function  $f$  of both  $x$  and  $y$  for which we require the centered finite difference approximation to the horizontal Laplacian  $\nabla^2 f$  at a point C. Using the notation given in Fig. 2 it follows directly from the one dimensional case that this approximation, indicated by  $\nabla^2 f$ , is

$$\nabla^2 f = \{f_{x+1,y} + f_{x-1,y} + f_{x,y+1} + f_{x,y-1} - 4f_{x,y}\} / (\Delta s)^2$$

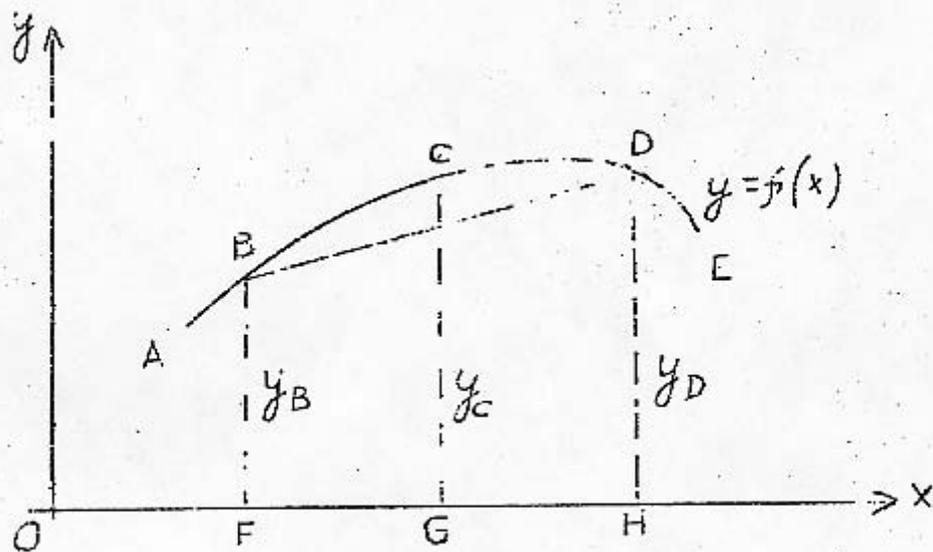


FIGURE 1

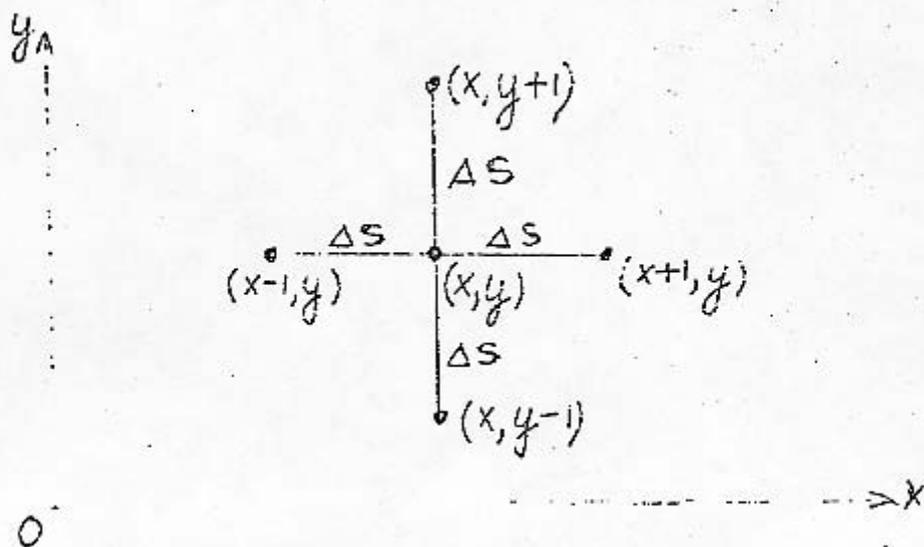


FIGURE 2.

$$R = \sin\left(\frac{2\pi x}{L}\right) \cdot \sin\left(\frac{2\pi y}{L}\right)$$
  
ERROR IN EVALUATING  $\nabla^2 R$  BY  
FINITE DIFFERENCES

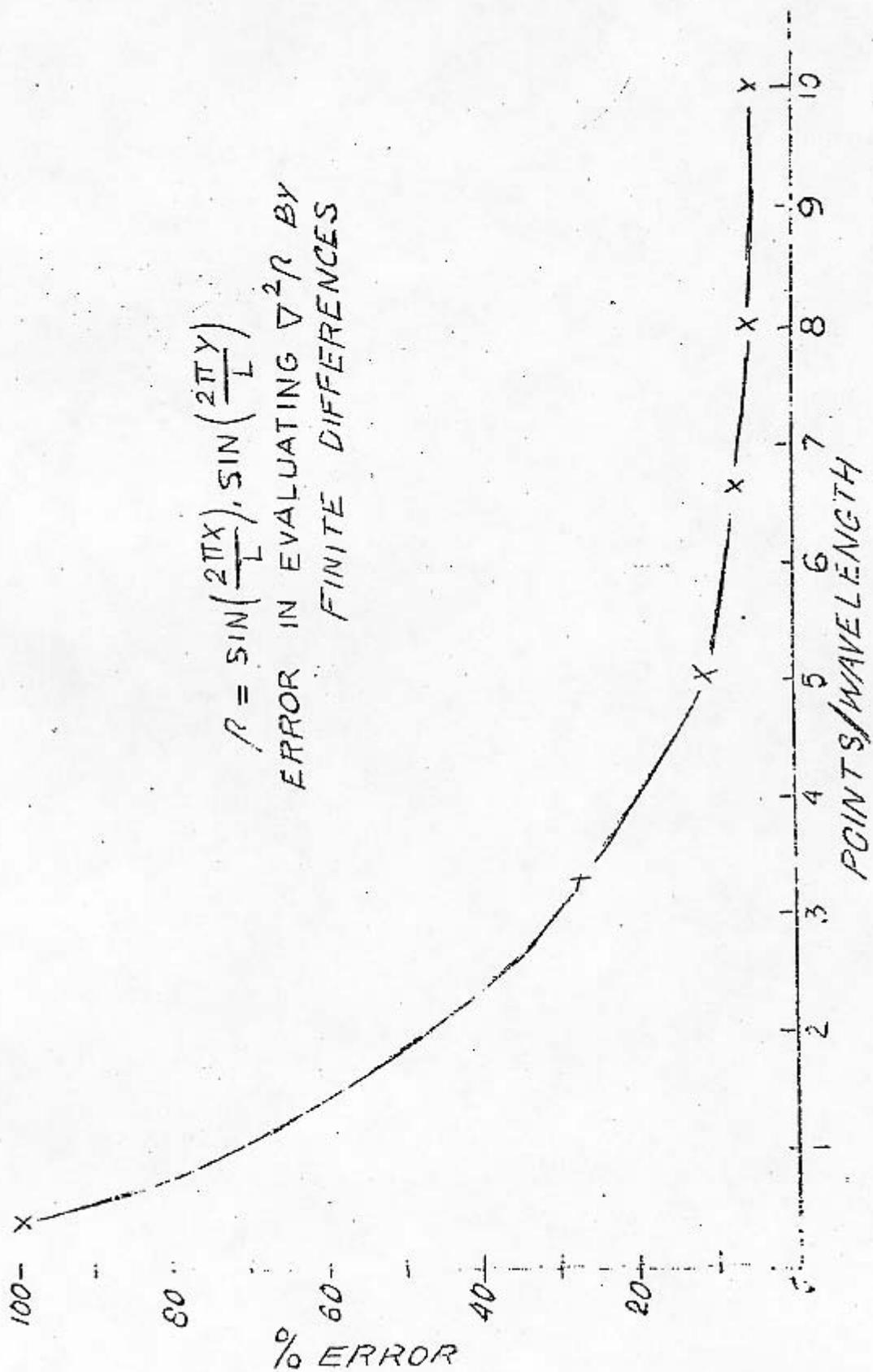


FIGURE. 3

Since the horizontal Laplacian of the pressure field is proportional to the geostrophic relative vorticity, a quantity used considerably in numerical weather prediction, it is of interest to estimate the magnitude of the error introduced by using the finite difference Laplacian in place of the continuous one. If we assume  $R$  to have the form

$$R = \sin \frac{2\pi x}{L} \sin \frac{2\pi y}{L}$$

we can evaluate directly both  $\nabla^2 R$  and  $\nabla^2 R$ . Figure 3 shows the graph of the percentage error in  $\nabla^2 R$  as compared to  $\nabla^2 R$  for various numbers of grid points per wave length of  $R$ . It will be seen that to keep the error below about 10% we have to have at least five or six grid points per wave length.

The grid interval usually met with in numerical weather prediction is 300 km and therefore we can only expect to get an error of less than 10% in the estimation of the geostrophic relative vorticity for wave lengths longer than about 1600 km. This result should be remembered when looking at numerical forecasts as made at the present time. The grid size alone, independent of physical assumptions, limits the amount of small scale detail which can be accurately forecast. Similar errors can also be expected when we replace continuous time derivatives by finite differences.

When we introduce time as well as spatial finite differences in our computations we meet another problem, the so called "computational stability problem". Space and time truncation errors of the type already considered can always be made as small as required by reducing the respective intervals over which the differences are taken. However, irrespective of how small these intervals are, a further condition must be satisfied if the solution of the finite differences problem is not to diverge exponentially from that of the continuous problem as the time increases. In our present forecasting models the most sensitive equation in this respect is the one used to advect the potential vorticity. This is an equation of the form

$$\frac{\partial}{\partial t} Q = -\underline{v} \cdot \nabla_H Q$$

where  $Q$  is the potential vorticity,  $\underline{v}$  the horizontal wind and  $\nabla_H$  the horizontal del-operator. Making the geostrophic assumption this equation becomes

$$\frac{\partial}{\partial t} Q = \frac{1}{f} J(Q, \phi)$$

where  $f$  is the coriolis parameter,  $\phi$  the geopotential and

$$J(\alpha, \beta) = \frac{\partial \alpha}{\partial x} \frac{\partial \beta}{\partial y} - \frac{\partial \alpha}{\partial y} \frac{\partial \beta}{\partial x}$$

Using centered finite differences  $J(Q, \phi)$  becomes

$$\frac{1}{4(\Delta s)^2} \left\{ (Q_{i+1,j} - Q_{i-1,j})(\phi_{i,j+1} - \phi_{i,j-1}) - (Q_{i,j+1} - Q_{i,j-1})(\phi_{i+1,j} - \phi_{i-1,j}) \right\}$$

and  $\partial Q / \partial t$  becomes  $\frac{1}{2\Delta t} (Q_{t+\Delta t} - Q_{t-\Delta t})$  we thus obtain an equation in which errors arising from both the space and time finite difference approximations will be present. It can be shown that these errors will not grow exponentially with time if the following condition is satisfied

$$\frac{\Delta s}{\Delta t} \geq \sqrt{2} |V|_{MAX}$$

where  $|V|_{MAX}$  is the maximum particle speed in the region under consideration. This condition states that the time interval,  $\Delta t$ , must be chosen sufficiently small so that a particle cannot travel for more than  $1/\sqrt{2}$  times a grid interval during the time interval  $\Delta t$ .

If after a time  $\Delta t$  we adjust the  $\phi$  and  $Q$  fields so as to be in agreement we can of course proceed for a further time  $\Delta t$ . This is in fact typical of the way we make our present numerical predictions.

The question of adjusting the  $\phi$ 's to agree with the  $Q$ 's leads us to our next problem. With the more complicated models this can involve the solution of very difficult differential equations but the principles of the methods used can be demonstrated by considering the type of equation which we have to solve in the barotropic model. This is a Poisson equation, i.e.,

$$\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} = \rho(x, y)$$

where  $\phi$  is the unknown and  $\rho(x, y)$  is a function of the vorticity and is known at each grid point. On the boundaries of the rectangular forecast region we assume that we know both  $\phi$  and  $\rho$ .

In finite difference form, using the notation of Fig. 2, the Poisson equation becomes

$$\phi_{i+1,j} + \phi_{i-1,j} + \phi_{i,j+1} + \phi_{i,j-1} - 4\phi_{i,j} = \rho_{i,j}^*$$

If  $i$  and  $j$  vary between 0 and  $I$  and 0 and  $J$  respectively this equation will hold for  $1 \leq i \leq (I-1)$ ,  $1 \leq j \leq (J-1)$ . Thus, since we know the values of  $\phi_{i,j}$  on the boundaries we have a set of  $(I-2)(J-2)$  simultaneous non-homogeneous linear algebraic equations for  $(I-2)(J-2)$  unknowns. If  $I$  and  $J$  are very small these equations can be solved very simply by the elimination method. Such a method, however, becomes prohibitively long for the large values of  $I$  and  $J$  met with in the numerical prediction problem. Various methods similar to Southwell's "relaxation" process have therefore been developed to reduce the amount of work involved in obtaining the solution of this set of equations. These methods will now be described.

The finite difference Poisson equation can be rewritten in the form

[1]

$$R_{ij} \equiv \phi_{i+1,j} + \phi_{i-1,j} + \phi_{i,j+1} + \phi_{i,j-1} - 4\phi_{i,j} - \rho_{ij}^* = 0$$
$$1 \leq i \leq I-1, 1 \leq j \leq J-1$$

and the problem can be regarded as that of finding a set of  $\phi_{ij}$  such that, for a given set of  $\rho_{ij}^*$ , the "residuals",  $R_{ij}$  are zero. Since in actual practice we only require the  $\phi_{ij}$  to a certain accuracy it is sufficient to require  $|R_{ij}|$  to be smaller than  $\delta$  where  $\delta$  depends on the accuracy to which we require the  $\phi_{ij}$ .

In order to start the relaxation process we first guess the  $\phi_{ij}$  and then compute the set of residuals  $R_{ij}$ . We then correct the  $\phi_{ij}$  so as to reduce the  $|R_{ij}|$ . From equation (1) we see that by increasing  $\phi_{ij}$  by one unit we can reduce  $R_{ij}$  by 4 units. At the same time, however,  $R_{i+1,j}$ ,  $R_{i-1,j}$ ,  $R_{i,j+1}$  and  $R_{i,j-1}$  are each increased by one unit if they correspond to internal points. Since we know the correct  $\phi_{ij}$  on the boundaries the residuals there will always remain zero. Thus for points  $(i,j)$  adjacent to a boundary the average residual the 5 points concerned can be reduced by altering the  $\phi_{ij}$ . We can thus see that there is at least a possibility that this correction process, if applied a sufficient number of times to a sufficient number of points, will eventually reduce the  $|R_{ij}|$  to the required size. The process can, in fact, in normal circumstances, be shown to converge to the correct solution.

In the original Southwell relaxation process the first  $\phi_{i,j}^{(1)}$  together with the corresponding  $R_{ij}^{(1)}$  are written down for each grid point. The grid is then inspected to find the largest  $|R_{ij}|$  and the corresponding  $\phi_{ij}$  changed so as to reduce this  $|R_{ij}|$  to zero. The resultant changes in the surrounding R's are then calculated and written down. The grid is then again inspected for the largest  $|R_{ij}|$  and the process repeated. This is continued until the  $|R_{ij}|$  are so small that changes in them do not significantly alter the  $\phi_{ij}$ .

Southwell's "over-relaxation" process is similar to the one just described except that now instead of reducing  $R_{ij}$  to 0 it is changed to  $-\alpha R_{ij}$  where  $\alpha$  lies between 0 and 1, the exact value used usually being chosen by judicious inspection at each point. With experience over-relaxation can considerably increase the rate of convergence of the process.

Southwell's two methods are very useful for hand computations but are not directly suitable for machine computations because of their requirement that the grid be continually searched for the largest  $|R_{ij}|$ , a process which is simple for the human eye but which involves the machine looking at each quantity. The over-relaxation process also involves the very subjective choice of  $\alpha$ . More systematic methods have therefore been developed for use with computing machines.

The simplest method adaptable to machine use is due to L. F. Richardson. In this method, and those following, the grid is scanned systematically and the value of  $\phi$  at each point changed according to a definite rule. We say that one iteration has been performed when each point in the grid has been scanned once. Suppose that after the  $\nu$ -th iteration we have a set of  $\phi_{ij}^\nu$  and a corresponding set of residuals  $R_{ij}^\nu$ . In general  $R_{ij}^\nu$  will not be zero. The Richardson process is to correct  $\phi_{ij}^\nu$  by adding one-fourth the residual to it, i.e.,

$$\phi_{ij}^{\nu+1} = \phi_{ij}^\nu + \frac{1}{4} R_{ij}^\nu \quad (2)$$

This has the effect of reducing the residual at the point  $i, j$  to zero, provided one leaves undisturbed all the other  $\phi_{ij}^\nu$ . Since, however, all the  $\phi_{ij}^\nu$ 's are simultaneously corrected by the formula (2) the resulting residuals  $R_{ij}^{\nu+1}$  will not necessarily be zero. It can, however, be shown rigorously that the  $\phi_{ij}^\nu$ 's will converge to the correct solution with increasing  $\nu$ . In actual practice the process is stopped when  $|\phi_{ij}^{\nu+1} - \phi_{ij}^\nu|$  is sufficiently small.

An improvement on the Richardson method is due to Liebmann. In his method the  $\phi_{ij}^\nu$  are not all altered simultaneously according to formula (2). Instead, as each new  $\phi_{i,j}^{\nu+1}$  is calculated it replaces  $\phi_{ij}^\nu$  in the computation of  $R_{ij}^\nu$  for succeeding points. Thus, if we scan the grid from left to right starting with the bottom row, we will use two values of  $\phi_{ij}^\nu$  and two of  $\phi_{ij}^{\nu+1}$  in the computation of  $R_{ij}^\nu$ , then usually written  $R_{ij}^{\nu, \nu+1}$ , since corrections will already have been at the points  $(i-1, j)$  and  $(i, j-1)$ , i.e.,

$$R_{ij}^{\nu, \nu+1} = \phi_{i+1,j}^\nu + \phi_{i-1,j}^{\nu+1} + \phi_{i,j+1}^\nu + \phi_{i,j-1}^{\nu+1} - 4\phi_{ij}^\nu - \rho_{ij}^*$$

The correction formula is then exactly similar to that of Richardson, viz.

$$\phi_{ij}^{\nu+1} = \phi_{ij}^\nu + \frac{1}{4} R_{ij}^{\nu, \nu+1} \quad (3)$$

Since in Southwell's method it was found that "over-relaxation" was advantageous, Frankel applied the same principle to the Liebmann process, the only difference being that the amount of over-relaxation was kept constant over the whole grid. This method is called the "extrapolated Liebmann method" and is characterized by the correction equation

$$\phi_{ij}^{\nu+1} = \phi_{ij}^\nu + \frac{1+\alpha}{4} R_{ij}^{\nu, \nu+1}$$

where  $\alpha$  lies between 0 and 1, the exact value being determined by the grid size and the type of errors expected in the initial guess  $\phi_{ij}^0$ . For our present prediction models this last method has been found to give the fastest convergence and therefore to require the least number of iterations.

A further source of error which is important both in hand as well as machine computations is known as "round-off" error. This arises from the fact that after we perform such numerical operations as multiplication and division we only write down or otherwise record the answer to a finite number of digits. For instance, if we have room on our paper or in our machine to write or to store ten decimal digits per number and we multiply two such numbers together storing the result to ten significant digits we will introduce a possible error of  $\pm 5$  in the eleventh decimal place. If we now add together the results of two such operations we will introduce a maximum possible error of  $\pm 1$  in the tenth decimal place. We can thus see the serious possibility of the growth of such errors if the computation involves numerous operations.

In the example given above the maximum possible error has been used for purpose of emphasis. On the average, of course, the errors will not grow anywhere near as fast. In fact, if we have  $m$  operations each with a maximum possible error  $\epsilon$  the probable error at the completion of the computation will be  $m^{1/2}\epsilon$  provided the round-off errors are random.

It is important that round-off errors should not be confused with errors due to inaccuracy of the initial data. Round-off errors can always be minimized by storing a sufficient number of digits, whereas no computation can ever give an answer more accurate than the initial data. For example, consider a computation in which the initial data is the height of a given pressure surface accurate to the nearest ten feet and from which we require a forecast of the height values twelve hours later. Assuming that our forecasting system is entirely correct, we can never expect to get an answer more accurate than to within ten feet. We can, however, always remove the possibility of round-off error producing an even larger error by keeping a sufficient number of digits in the computation, i.e., if the initial data is 100 feet we might store it during the computation as 100.00000 feet and then the round-off error  $\epsilon$  at each stage would only be  $\pm 5.10^{-6}$  feet as compared with an observational error of 10 feet.