Analysis of Nonlinearities in Ground Water Hydrology: A Hybrid Computer Approach

VENKATESWARARAO VEMURI AND JOHN A. DRACUP

University of California, Los Angeles

Abstract. A hybrid computer method to solve nonlinear partial differential equations describing the flow of fluids in underground formations is described. Using finite difference methods, the problems are first programmed by a flow chart for a pure digital computer solution. Because the ability of resistance networks to solve a set of simultaneous equations instantaneously is recognized, major matrix inversion subroutines in the digital flow chart are replaced by analog resistance network hardware. Incorporation of analog hardware drastically reduces the computation time involved in inversion of large matrices. The resistance network constructed on an analog computer patch board allows an unparalleled flexibility not available in a digital computer. (Key words: Computer; ground water; hydrology; nonlinearities; simulation)

INTRODUCTION

In hydrologic systems analysis an engineer often encounters the problem of solving nonlinear partial differential equations with irregular boundary conditions. For example, problems involving the flow of water in both the saturated and unsaturated zones of Earth's geological formations fall in this category. Stream flow routing and reservoir routing problems also involve this type of differential equations.

A systematic look at the differential equations occurring in hydrology, as well as the basic equations in such diverse fields as aerodynamics, solid mechanics, and heat transfer, indicates that all these areas are intimately related to each other by three equations known as:

1. Continuity equation (The law of conservation of mass);
2. Newton's second law (The law of conservation of momentum);
3. First law of thermodynamics (The law of conservation of energy).

When these conservation laws are formulated as general mathematical equations, the results are highly nonlinear second-order partial differential equations, which are often called the Navier-Stokes equations. There is no known general solution, nor is there any approach to solve these equations analytically, except in some specialized cases. The ability to get a reasonable solution in the general case constitutes a special challenge to the physicist and the engineer.

The classical alternative is to resort to numerical techniques to obtain a solution in the time domain with the help of either an analog or a digital computer. Analog methods are complicated, because such problems generally have two or more independent variables, whereas analog computers are restricted to only one independent variable: time. The application of digital computer techniques leads to uneconomically long computer runs if reasonable accuracies are required. The purpose of this paper is to demonstrate that a novel automatic computer approach, termed hybrid computation [*Karplus,* 1964], representing an attempt to combine the speed of an analog computer with the accuracy of a digital computer, appears to be ideal to solve problems involving nonlinear partial differential equations. Even though this method is applicable to a majority of problems in other areas, including weather forecasting and meteorology, magnetohydrodynamics, nuclear reactor systems, etc., the interest in the present paper is confined to the study of problems in ground water hydrology.

MATHEMATICAL BACKGROUND

A mathematical expression governing the flow of water through porous mediums can be determined from the equations of motion, continuity, and the laws of thermodynamics. If the flow is laminar and if the inertia force can be neglected, Darcy's Law follows from Navier-Stokes equations, and one can write

$$\mathbf{V} = K(x, y, z)\mathbf{F} \tag{1}$$

where $\mathbf{V}$ is the velocity of water, $\mathbf{F}$ is the applied force, and $K(x, y, z)$ is a proportionality constant, depending upon the medium as well as on the fluid, given by

$$K(x, y, z) = (Cd^2\rho g/\eta) = [k(x, y, z)\rho g/\eta] \tag{2}$$

and is often called the coefficient of permeability. In (2), the so-called specific permeability $k(x, y, z) = Cd^2$ is strictly a property of the medium alone, wherein the constant C is determined by such factors as porosity, packing, size, shape, and distribution of the grain, whereas d stands for the average grain diameter, ρ for the density, η for the viscosity of the fluid flowing through the medium, and g for acceleration due to gravity. The force $\mathbf{F}$ is equal to the negative gradient of the hydraulic head h

$$\mathbf{F} = -\text{grad } h \tag{3}$$

Because viscous fluid flow is a mechanical process, friction is a dominating factor, and all such flows are accompanied by an irreversible transformation of mechanical to thermal energy. In potential theory, this mechanical energy per unit mass is termed the *potential* of the fluid. Following *Hubbert* [1940], this potential ψ can be written (neglecting kinetic energy term) as

$$\psi = gz + \int_{p_0}^{p} \frac{dp}{\rho} \tag{4}$$

where p is the pressure at the point in question whose elevation above a datum is z, and p_0 is atmospheric pressure. For liquids (4) further reduces to

$$\psi = gz + (p - p_0)/\rho$$

Because the water table is at atmospheric pressure, for all points on the water table (4) reduces to the yet simpler form

$$\psi = gz$$

Defining the hydraulic head h as

$$h = \psi/g = z \tag{5}$$

it is clear that the potential ψ is nothing but the hydraulic head in disguise, and the use of h as a potential function is justified. Substituting (3) and (5) in (1)

$$\mathbf{V} = -K(x, y, z) \text{ grad } h \tag{6}$$

If θ is the water content of the soil on a volumetric basis (that is, θ is the ratio of the volume of the soil water to the total volume of the soil), and if the density of water is assumed to remain constant, the continuity equation can be written as

$$-\text{Div } (\mathbf{V}) = \partial\theta/\partial t \tag{7}$$

Combining (6) and (7)

$$\text{Div } (K(x, y, z) \text{ grad } h) = \partial\theta/\partial t \tag{8}$$

The dependent variable on the right-hand side of (8) is θ, whereas on the left-hand side it is h. To make the equation consistent with respect to one dependent variable a term $S = S(x, y, z)$, to represent the volume of water that a unit decline in head releases from storage, is introduced. Then θ can be replaced by the product Sh and (8) becomes

$$\text{Div } (K(x, y, z) \text{ grad } h) = S(x, y, z)\, \partial h/\partial t \tag{9}$$

This is the basic equation to be solved in the study of unsteady flow in a three-dimensional field. A variety of problems in ground water hydrology reduce to one of solving (9) with various initial and boundary conditions.

Unsteady flow in saturated aquifers. The recent increase in usage and dependence upon ground water resources throughout the world has brought special significance and attention to the theory of ground water movement. Ground water aquifers are utilized in conjunction with surface water systems for, among other things, municipal, industrial, and agricultural water supplies; disposal of treated industrial and municipal waste waters; storage of artificially recharged water.

In artificial recharge operations, a quantity $Q = Q(x, y, t)$ of water of varying quality and

composition is introduced into the aquifer and mixing occurs between injected and indigenous waters. This kind of mixing between two different bodies of water may occur either by design, as in the case of artificial recharge, or by accident, as in sea-water intrusion. It is highly desirable, in any event, to be able to predict the variations in the elevation $h = h(x, y, t)$ of the water table during these mixing processes. Introducing the quantity $Q(x, y, t)$, which represents such diverse factors as rainfall, recharge, extraction, and leakage as a forcing function, (9) becomes

$$\nabla \cdot (K(x, y, z) \nabla h) = S(x, y, z)\, \partial h/\partial t \pm Q(x, y, t) \qquad (10)$$

A positive sign on Q corresponds to a net upward flow and a negative sign to a net downward flow. Utilizing the relation in (7), this linear three-dimensional equation can be converted into the following nonlinear equation in only two dimensions

$$\frac{\partial}{\partial x}\left[K(x, y, h)\frac{\partial h}{\partial x}\right] + \frac{\partial}{\partial y}\left[K(x, y, h)\frac{\partial h}{\partial y}\right] = S(x, y, h)\frac{\partial h}{\partial t} \pm Q(x, y, t) \qquad (11)$$

Use of this equation is justified in ground water studies because we are interested only in the hydraulic head at the water table, and this is the fundamental partial differential equation to be solved with appropriate initial and boundary conditions.

Studies in unsaturated flow fields. Problems of estimating, quantitatively, the changes in ground water regimes and water balance of soils in the zone of aeration are of vital importance in afforestation of arid zones to combat drought, dry wind, soil erosion, and for storage of soil moisture. Studies of ground water regime must be carried out in liaison with studies of soil moisture regime and water balancing studies on the surface and in the aeration zone. Some of the objectives of these studies are: 1. To obtain a satisfactory model that could reasonably duplicate field measurements in soil drainage experiments; 2. To use this tailored model in extrapolation and also to obtain evidence concerning drainage problems for which field measurement is not possible; 3. To estimate drainage in soils having diffusivities that differ from those of the test soil.

In unsaturated flow fields, it is not meaningful to talk about the hydraulic head h as the dependent variable. It is more desirable to obtain solutions in terms of spatial distribution of pressure or the spatial distribution of water content. This problem is particularly difficult to solve, because the infiltration of water in soils is a simultaneous flow of a mixture of air and water through soil. This phenomenon induces a vertical variation in the degree of saturation. The hydraulic engineer, therefore, is more interested in the vertical profiles of pressure distribution. Considering only the z-component of (8) and substituting for the hydraulic head the sum of the pressure and elevation heads (that is, $h = p + z$), thus neglecting the velocity head, the result is

$$K\,\partial^2 p/\partial z^2 + \partial K/\partial z\,(\partial p/\partial z) + \partial K/\partial z = \partial\theta/\partial t \qquad (12)$$

This equation, as it stands, is difficult to solve analytically, because the coefficient K is a function of both θ and p. Because K is more sensitive to the variations in p, it is convenient to modify (12) by considering both K and θ as a single function of p. This is achieved by writing (12) as

$$K\frac{\partial^2 p}{\partial z^2} + \frac{\partial K}{\partial p}\left(\frac{\partial p}{\partial z}\right)^2 + \frac{\partial K}{\partial p}\frac{\partial p}{\partial z} = \frac{\partial \theta}{\partial p}\cdot\frac{\partial p}{\partial t} \qquad (13)$$

where the z coordinate is measured positively upwards. Dividing throughout by $\partial\theta/\partial p$ results in

$$D\,\partial^2 p/\partial z^2 + B\cdot\partial p/\partial z(\partial p/\partial z + 1) = \partial p/\partial t \qquad (14)$$

where D and B are defined as

$$D = K\,\partial p/\partial\theta \qquad B = \partial K/\partial\theta \qquad (15)$$

Equation 14 is the fundamental partial differential equation governing one dimensional unsaturated flow.

THE HYBRID COMPUTER METHOD

The two cases leading to the nonlinear equations 11 and 14 of the preceding section are but two representative occasions, and equations like (11) or (14) or modified forms thereof occur very often in nonlinear hydrology. The nature of the nonlinearities and the irregularity of the boundaries often render the prob-

lem intractable by analytical means. When analytical methods fail, a partial differential equation must be solved as it stands on a computer using numerical techniques. There are two classes of problems that an engineer often faces: the analysis problem and the design problem. The analysis problem, also called the direct problem, is concerned with the determination of the potential function when the parameters are specified. The other class of problems is concerned with the more difficult task of identifying the parameters with a known potential distribution. This second class of problems is variously known as the inverse problem or the identification problem.

In spite of the high speeds of modern computers, solution of the direct and inverse problems associated with nonlinear partial differential equations is still a formidable task. The application of analog computer techniques [*Karplus,* 1958] to transient field problems has not been practical. Simulation of partial differential equations by electrical means would involve a provision for distributed electrical elements. For this reason, just to mention one, all analog computer techniques involve the discretization of at least one of the many independent variables. In the so-called discrete space continuous time *(DSCT)* methods [*Karplus,* 1966], the usual practice is to discretize the space variable by replacing the space derivatives by suitable finite difference expressions [*Karplus and Vemuri,* 1967] and to leave the time continuous. This analog computer technique yields very quickly a reasonable solution, but a major difficulty arises when it is necessary to simulate nonlinear and time-varying fields. Under these conditions, it is necessary to adjust a multitude of circuit elements in the course of a computer run. This is always difficult and sometimes impossible, so that analog computer methods are essentially limited to linear constant parameter fields.

When transient field problems are solved on a digital computer using discrete space-discrete time *(DSDT)* methods, all independent and dependent variables must be discretized, since a digital computer is capable of solving only algebraic equations. The first step in solving field problems by digital methods is, therefore, to define a finite difference grid. For a one-dimensional diffusion equation, such a grid will have two coordinates: distance x and time t, as shown in Figure 1. The intervals Δx and Δt are selected in such a manner that the truncation error (the error resulting from the finite difference approximation of continuous partial derivatives) falls within a specified tolerance.

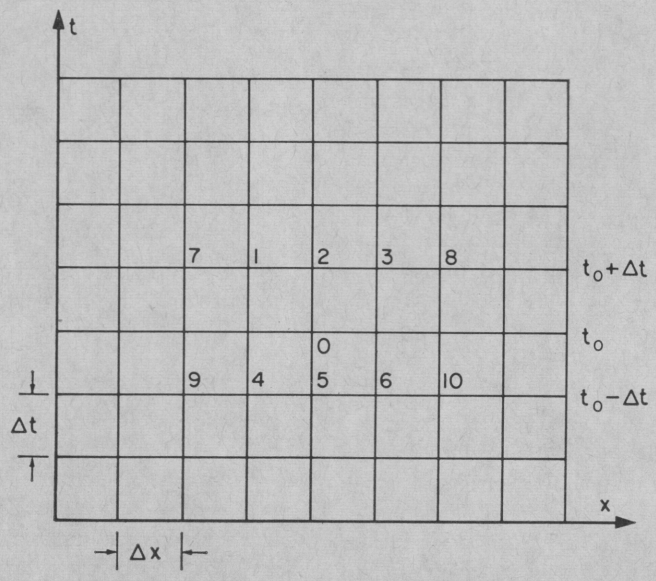

Fig. 1. A finite difference grid in a one-dimensional space-time continuum.

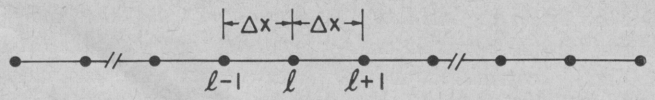

Fig. 2. A typical node point l and its neighbors used to derive the finite difference approximation in (17).

In general the x-coordinate will be bounded at $x = 0$ and $x = X$ (where X is the total linear dimension of the one-dimensional field under study), whereas variable t starts at zero and proceeds to infinity. Every point in this finite difference grid corresponds to a specific point in space at a specific instant of time. It is often convenient to choose a regular grid of square or rectangular shape, but this is by no means the only way. Sometimes irregular, asymmetric [*MacNeal,* 1953] grids may prove to be more useful.

A typical point within this finite difference grid is now selected arbitrarily and denoted by the subscript 0 (Figure 1). The coordinates of this point are assumed to be (x_0, t_0). The field potential occurring at this point at a time Δt later than t_0 is denoted by subscripts 1, 2, 3, 7, and 8, whereas the field potentials at the point in question and at the neighboring points that occurred at a time Δt earlier than t_0 are denoted by the subscripts 4, 5, 6, 9, and 10.

One-Dimensional Diffusion Equation

To solve a one-dimensional diffusion equation, for example

$$\partial^2\varphi/\partial x^2 = (1/\sigma)\cdot\partial\varphi/\partial t \qquad (16)$$

by this method, a set of boundary conditions and one initial condition at each point of the grid are required. This equation may represent the flow of water in a narrow tube filled with sand, in which case the potential function stands for the hydraulic head. This equation may also represent the flow of heat along a rod whose ends are kept at prescribed temperatures, in which case the temperature φ is the scalar potential. The boundary conditions in the former case may be interpreted as the head maintained at both ends of the tube. The initial conditions may represent the initial temperature of the rod or the initial head in the tube at various points at the beginning of the experiment. Typically, then, the field potentials (say, the temperature) at the initial time t_0 are given, and the field potentials at later times $t_0 + n(\Delta t)$, for $n = 1, 2, 3, \ldots$, are sought.

Referring to Figure 2, the finite difference approximation to a second space derivative at a typical node point l can be written as

$$\partial^2\varphi/\partial x^2|_{at\ l} \simeq (\varphi_{l-1} - 2\varphi_l + \varphi_{l+1})/(\Delta x)^2 \qquad (17)$$

In approximating time derivatives by finite difference expressions two possibilities exist, namely, 'backward difference' and 'forward difference' approximations [*Richtmyer,* 1957]. For instance, using (17) and backward time differences, (16) becomes

$$(\varphi_{l-1}{}^{j} - 2\varphi_l{}^{j} + \varphi_{l+1}{}^{j})/(\Delta x)^2 \simeq (\varphi_l{}^{j} - \varphi_l{}^{j-1})/\sigma(\Delta t) \qquad (18)$$

where Δt is the size of the step taken along the timelike axis (see Figure 1) to make the solution march forward in time, and the superscript j indicates the time level at which the approximation is written. In backward differencing, as performed above, $\partial\varphi/\partial t$ is approximated by the first difference between the value of φ measured at the present time level j and the value of φ at the previous time level $j - 1$. Similarly using a forward difference scheme, the finite difference approximation to (16) becomes

$$(\varphi_{l-1}{}^{j} - 2\varphi_l{}^{j} + \varphi_{l+1}{}^{j})/(\Delta x)^2 \simeq (\varphi_l{}^{j+1} - \varphi_l{}^{j})/\sigma(\Delta t) \qquad (19)$$

Knowing the value of the potential φ at the time level j, the value of the unknown potential at the $(j + 1)^{\text{th}}$ level can be computed by a simple algebraic procedure, because in (19) $\varphi_l{}^{j+1}$ is the only unknown variable. Since $\varphi_l{}^{j+1}$ is obtained explicitly, methods employing forward time differences are called explicit methods. Computationally this is a simple process, but by repeated application of (19) round-off errors (that is, errors committed by

the digital computer owing to its inability to carry the computations beyond a certain number of significant digits) committed in the solution process will grow and make the solution meaningless unless the inequality

$$(\Delta x)^2/\sigma(\Delta t) \geq 2 \tag{20}$$

is satisfied [*Richtmyer,* 1957]. The quantity on the left-hand side of (20), however, depends on the nature of the difference approximation used. Some representative difference approximations to (16) along with the required stability conditions are available in the literature [*Karplus and Vemuri,* 1967]. To traverse a given time domain T, condition 20 makes it necessary to make many more steps in time, that is, to make Δt much smaller than in the application of the backward difference equation 18. Moreover, when the field parameter σ is a function of the field potential φ, the appropriate value of Δt generally cannot be chosen in advance. For these reasons, the implicit approach or the method of backward time differences is generally preferred.

Equations employing backward time differences are solved implicitly, a process that is computationally stable. In implicit methods employing (18), only the potential at time level $j-1$ is known, and all other potentials at time level j are unknown. Thus the unknown potentials at a given time step are contained in a system of simultaneous algebraic equations. If the space domain has been replaced by N finite difference grid points it becomes necessary to invert an $N \times N$ matrix at each time step to solve the above system of equations. The presence of nonlinear parameters makes it further necessary to iterate at each time step, that is, to invert the said $N \times N$ matrix several times until a specified convergence condition is satisfied. It is this matrix inversion that occupies about 90% of the digital computer running time and occupies a considerable amount of storage space, making it necessary to use expensive high speed computers with large memories.

The above discussion points out that digital methods suffer either from problems of computational instability or from prolonged running times, and analog methods suffer from the lack of accuracy. The discrete space-discrete time (*DSDT*) hybrid computer represents a hybridization of the analog and digital methods. This method represents an attempt to combine in one system the accuracy, memory, decision-making capability, and programmability of a digital computer with the speed and flexibility of an analog computer.

In the so-called digital computer oriented hybrid methods [*Karplus,* 1965], the problem to be solved is first programmed by a flow chart for a conventional digital computer solution. The flow chart is then examined for any loops involving matrix inversion subroutines. Taking advantage of the well known ability of electrical resistance networks to solve simultaneous equations [*Liebmann,* 1950], these matrix inversion loops are replaced by the corresponding resistance networks. This concept of using analog hardware as subroutines in a digital computer solution would result in an appreciable reduction in the over-all computation time.

To illustrate the hybrid method, equation 18 is considered once again. Allowing the subscript l to run through the values $1, 2, \cdots, N-1$, the following set of equations, representing the finite difference approximations to (16) at various grid points at a given time level (that is, for a specific value of j), is obtained:

$$\begin{aligned}
&\varphi_0^{\,j} - 2\varphi_1^{\,j} + \varphi_2^{\,j} + 0 + \cdots + 0 \\
&\qquad = k(\varphi_1^{\,j} - \varphi_1^{\,j-1}) \\
&0 + \varphi_1^{\,j} - 2\varphi_2^{\,j} + \varphi_2^{\,j} \cdots + 0 \\
&\qquad = k(\varphi_2^{\,j} - \varphi_2^{\,j-1}) \\
&\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\
&\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\
&0 + 0 + \cdots + \varphi_{N-2}^{\,j} - 2\varphi_{N-1}^{\,j} + \varphi_N^{\,j} \\
&\qquad = k(\varphi_{N-1}^{\,j} - \varphi_{N-1}^{\,j-1})
\end{aligned} \tag{21}$$

Here k stands for $(\Delta x)^2/\sigma(\Delta t)$ and $\varphi_0^{\,j}$ and $\varphi_N^{\,j}$ are the given boundary values $\varphi_i^{\,j-1}$; $l = 1, 2, \ldots, (N-1)$ are the values of φ obtained from the calculations at the previous time level and are known. Rearranging these equations with all the unknowns on the left-hand side and known values on the right-hand side, the following equations are obtained:

$$\begin{aligned}
&-(2+k)\varphi_1^{\,j} + \varphi_2^{\,j} + 0 + \cdots + 0 \\
&\qquad = -\varphi_0^{\,j} - k\varphi_1^{\,j-1} \\
&\varphi_1^{\,j} - (2+k)\varphi_2^{\,j} + \varphi_3^{\,j} + \cdots + 0
\end{aligned}$$

$$= -k\varphi_2^{\,i-1}$$
$$\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \quad (22)$$
$$\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots$$
$$0 + \cdots 0 + \varphi_{N-2}^{\,i} - (2+k)\varphi_{N-1}^{\,i}$$
$$= -\varphi_N^{\,i} - k\varphi_{N-1}^{\,i-1}$$

In matrix notation this can be written as

$$\begin{bmatrix} -(2+k) & 1 & 0 & \cdots & 0 \\ 1 & -(2+k) & 1 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & 0 & \cdots & 1 \quad -(2+k) \end{bmatrix} \begin{bmatrix} \varphi_1 \\ \cdot \\ \cdot \\ \cdot \\ \varphi_{N-1} \end{bmatrix}^{i}$$

$$= \begin{bmatrix} -\varphi_0 \\ 0 \\ \cdot \\ 0 \\ -\varphi_N \end{bmatrix}^{i} + \begin{bmatrix} -k\varphi_1 \\ -k\varphi_2 \\ \vdots \\ -k\varphi_{N-1} \end{bmatrix}^{i-1} \quad (23)$$

or

$$A\varphi = \beta + \mathbf{c} \quad (24)$$

where φ, β, and $\mathbf{c}$ are vectors, and A is a matrix. The matrix A in (24) is called the analog matrix, and the vectors β and $\mathbf{c}$ are, respectively, the boundary vector and the analog input vector.

The procedure here takes a major turn from the conventional digital methods. Rewriting a typical equation from (22)

$$\varphi_{l-1}^{\,i} - (2+k)\varphi_l^{\,i} + \varphi_{l+1}^{\,i} = -k\varphi_l^{\,i-1} \quad (25)$$

and setting $l = 2$, $(2 + k) = b$ and $-k\varphi_l = c(\varphi_l)$ for convenience results in

$$\varphi_1^{\,i} - b\varphi_2^{\,i} + \varphi_3^{\,i} = c^{i-1}(\varphi_2) \quad (26)$$

Equation 26 has a close resemblance to Kirchoff's current law equation of electrical circuit theory. Recognizing this formal similarity, one can represent (26) by a passive resistance network as shown in Figure 3. Such a resistance network representing a single equation of the system is termed a node module. It is evident that as many modules are required as there are equations in the system of finite difference equations. To solve a set of equations of the type (25) it is necessary to interconnect several such modules as shown in Figure 4. In Figure 4 each box represents one node module, and the inputs c_1, c_2, $\cdots$, c_{N-1} correspond to $-k\varphi_l$ for $l = 1$, 2, $\cdots$, $N - 1$, computed at the previous time level, so that at any stage of the computation the vector $\mathbf{c}$, whose components are c_1, c_2, $\cdots$, c_{N-1}, is known. Because the values c_1, c_2, $\cdots$, c_{N-1} are analog voltages, the vector $\mathbf{c}$ is called the analog input vector. The boundary conditions are applied as voltages to the modules at the boundaries. Consequently, the vector whose components are the boundary values, that is, the vector β, is called the boundary vector. When the boundary and input voltages are applied to the network of Figure 4, the output voltages represented by φ_1, φ_2, $\cdots$, φ_{N-1} yield the solution of the set of equations at the indicated time level.

The iterative network method. Referring to (26), it is seen that the coefficients b and c are functions of k, which is in turn a function of the field parameter σ. In highly nonlinear cases σ will be, in general, a function of the coordinates, the potential φ, and perhaps of the derivatives of φ. It is conceivable, therefore, that the values of b and c, in general, change from time level to time level as the computation progresses. This means that certain resistor values in the passive analog computer (Figure 3) must change, in general, from time level to time level. To perform these changes automatically would require sophisticated hardware. To obviate this difficulty, an iterative method [*Russell*, 1965] is adopted to solve the set of simultaneous equations.

To facilitate the implementation of the iterative method, equation 26 is rearranged as follows:

$$\varphi_1^{\,i} - 3\varphi_2^{\,i} + \varphi_3^{\,i} = c(\varphi_2^{\,i-1}) + (b-3)\varphi_2^{\,i}$$
$$= g(\varphi_2) \quad (27)$$

where

$$g(\varphi_2) = c(\varphi_2^{\,i-1}) + (b-3)\varphi_2^{\,i}$$

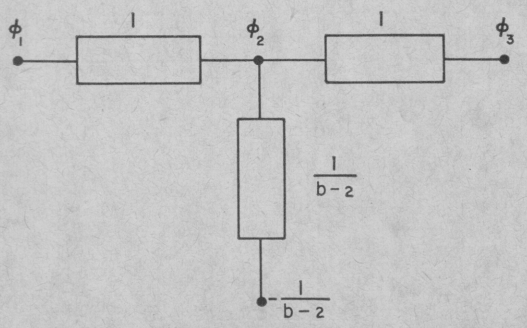

Fig. 3. Node module for a passive analog circuit.

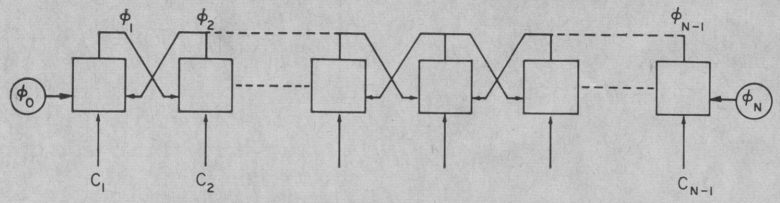

Fig. 4. Schematic diagram showing interconnection of node modules.

Equation 27 is obtained from (26) by subtracting $3\varphi_2^j$ from both sides of the equation. Here $g(\varphi_2)$ is a function of φ_2 evaluated at the previous time level and also at the present time level. Because φ_2^j is the function to be evaluated by solving (27), it is not possible to determine $g(\varphi_2)$ explicitly. Equation 27 has to be solved, therefore, implicitly by starting with a nominal value of $g(\varphi_2)$ and iterating until convergence is achieved. The node module for solving this equation is shown in Figure 5. As in the previous case a set of simultaneous equations can be solved, by the iterative method, by interconnecting several modules, as in Figure 4. The node module shown in Figure 5 differs from that in Figure 3 in one important respect; in Figure 5 all resistors are equal to unity, and they do not change their values in the course of a computer run. It is the nonlinear function g that changes from iteration to iteration.

To automate the iterative process completely the analog resistance network can be connected in a closed loop with a digital computer. The digital computer is used to evaluate the nonlinear function $g(\varphi_2)$. The boundary conditions are applied to all the boundary nodes and the initial conditions to the rest of the nodes, and the iterative computations are carried out until the potential φ_2 in two successive iterative cycles fails to show any appreciable change. The digital computer serves to test this convergence [*Karplus and Kanus*, 1965] and also to store the solution, namely, the potentials at time $(t_0 + \Delta t)$ at each node in order that these data may be employed as initial conditions at the subsequent time step. To translate the analog voltages appearing simultaneously at the output terminals of the node modules into a serial digital form, a multiplexer (scanner) is employed to sample in turn the potential at each node. These d.c. voltages are converted into a digital code by an analog-digital converter and applied to the memory of the digital computer, where they are stored. Subsequently these data, or modified forms thereof, are read out of the digital computer, reconverted to analog form, and applied to the input terminals of the node modules by means of a second scanner called a distributor. This arrangement is shown in Figure 6.

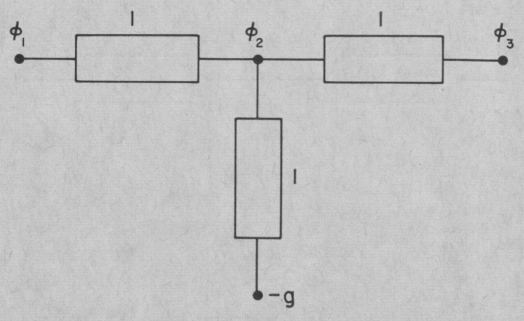

Fig. 5. Node module of a passive iterative analog circuit.

The entire iterative resistance network described so far acts only as a subroutine in a digital computer loop. This analog subroutine performs the most time-consuming digital operation, namely, solving a set of simultaneous equations. Thus the hybrid system takes advantage of the fact that a rectangular network of electrical resistors relaxes automatically and almost instantaneously to the solution of the circuit equations as expressed by Kirchoff's node law. The total computation time for each iterative subcycle is therefore determined by the length of the time required to translate the serial digital information into parallel analog form and vice versa.

Ground water flow. The hybrid computer method of solving (11) occurring in the study of ground water flow is considered. The method of solving (14), or for that matter any other partial differential equation, is similar. The first step in solving (11) is to select a set of nodes within the basin under study. In two-

dimensional problems with irregular boundaries of this nature it is seldom practical to choose a regularly spaced array of node points, since this introduces errors at the boundaries, which are almost always irregular. Also, it either causes an excessive number of nodes to be employed or introduces excessive truncation error at some portion of the field. Furthermore, the historical data on water levels and the replenishment-extraction data are seldom available at regularly spaced intervals. Therefore, the finite difference network is sometimes chosen to be an asymmetric one. The result of discretizing the space domain of (11) using an asymmetric grid is [*MacNeal*, 1953]

$$\sum_l (h_l - h_B) Y_{lB} = A_B S_B \, \partial h_B/\partial t + A_B Q_B \qquad (28)$$

where

$$Y_{lB} = (J_{lB} K_{lB})/L_{lB}$$

and

A_B = area associated with the Bth node;
Y_{lB} = conductance of the path between nodes l and B;
S_B = value of S associated with the polygonal zone centered at B;
Q_B = volumetric flow rate per unit area at node B;
K_{lB} = value of K at the midpoint between the nodes l and B;
L_{lB} = distance between nodes l and B;
J_{lB} = length of the perpendicular bisector associated with the nodes l and B.

A typical node point, its neighbors, and the associated polygonal zone are shown in Figure 7. The left-hand side of (28) is the summation of subsurface flows between a given area and its surrounding areas. The rate of change of storage is given by the first term on the right-hand side, and the second term represents the surface flow rate from the ground surface into or out of the zone of saturation of the given unit area. If the polygon borders the basin boundaries, any subsurface flow crossing the boundary is usually included into $A_B Q_B$.

Discretizing the time derivative in (28) by backward (implicit) differences, one gets

$$\sum_l (h_l^{\,i} - h_B^{\,i}) Y_{lB} = [A_B S_B/(\Delta t)](h_B^{\,i} - h_B^{\,i-1}) + A_B Q_B \qquad (29)$$

Here the superscript j denotes, as usual, the time index, and the subscript l denotes the set of nodes adjacent to the node under question, that is, node B.

Solution of (29) on a hybrid computer is straighforward if the values of the parameters Y_{lB} and S_B are known beforehand. This is not

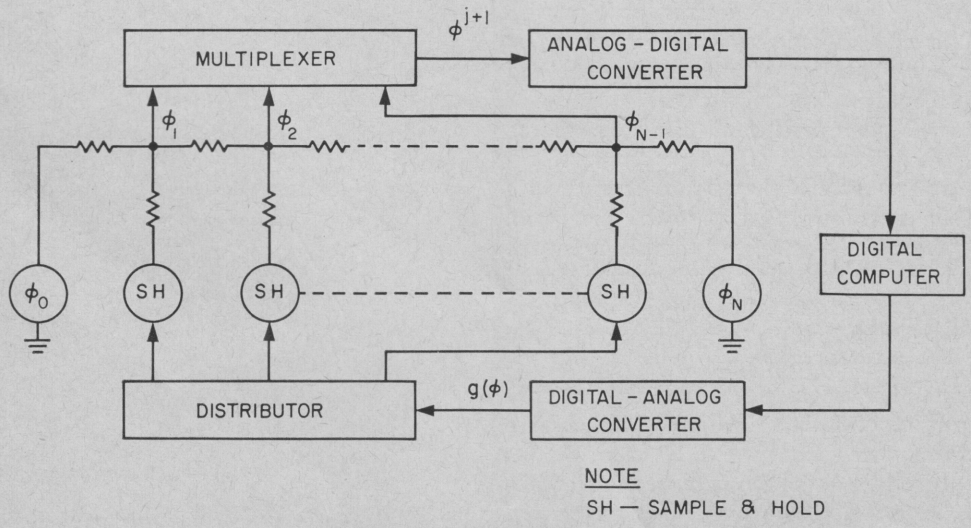

Fig. 6. Schematic of a *DSDT* computer.

the case in general; usually the elevation of the water table above a certain reference and the accretion to the water table are the known quantities. But a good knowledge of the basin parameters is necessary for predicting the basin behavior for all future times and for successful ground water management. The computational procedure is therefore divided into two phases; in phase 1 the values of the basin parameters, namely K and S, are identified, and in phase 2 the values of K and S determined in phase 1 are used to predict the elevation of future ground water levels.

Phase 1 (identification phase). This phase may be regarded as the crux of the whole problem, and it is in this phase that the versatility of the hybrid computer is uniquely demonstrated. An analog resistance network to solve (29) is constructed as shown in Figure 8, wherein only the configuration of the resistors is shown. The values of these resistors are functions of the unknown parameters Y_{IB} and S_B, which in turn depend, perhaps nonlinearly, on the value h to be determined by solving (29). The voltages at each point of this network correspond to the elevation of water table at the corresponding point in the basin. During phase 1 the parameters K and S are repeatedly adjusted (by adjusting the resistors representing Y_{IB} and S_B) until the computed water levels (that is, the voltages at each node of the resistance network) match the known historical water level records. The simplest way to do this adjustment is by trial and error, perhaps with the help of a cathode ray tube display, or by using a combination of steepest descent and random optimization techniques.

At each time level j all the resistors are adjusted, independent of each other, until the measured response of the network at each node matches the observed response of the basin. The values of the set of resistors that give the best match at each time level are noted and the whole procedure repeated for the next time level, namely $j + 1$. By this method any local nonlinearity in T and S is quickly determined. The values of T and S at each node as functions of h are stored in the digital computer for future use in phase 2.

Phase 2 (computational phase). After the identification of the nonlinear parameters the mathematical model of the basin is subjected to various operating conditions corresponding to future pumping, artificial recharge, expected rainfall, or any other pertinent situation to predict the future water level trends. During this phase it may be more convenient to use the iterative network setup than the special network of Figure 8 used in the identification phase.

To demonstrate the capabilities and advantages of the hybrid computer method, a 19-node model of the San Fernando Valley ground water basin is at present under investigation at the Hybrid Computer Laboratory of UCLA.

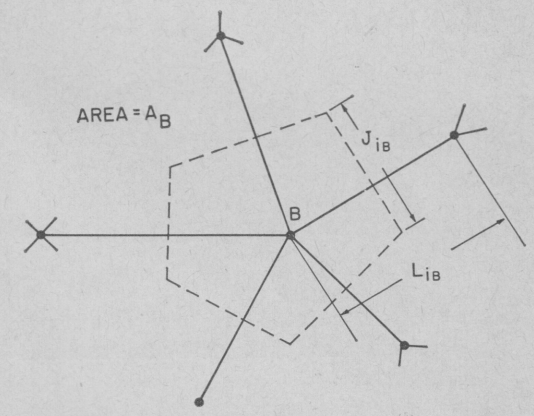

Fig. 7. A typical node in an asymmetric grid.

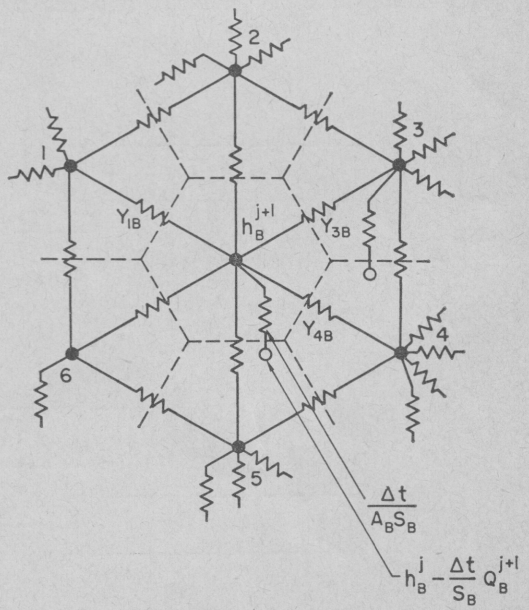

Fig. 8. Analog resistance network for an iterative process using an asymmetric grid.

Advantages and disadvantages of the hybrid method. One of the major advantages of the hybrid computer method is its flexibility; another is the speed gained by the use of analog hardware for matrix inversion.

If an electronic analog computer is used during phase 1, the replenishment-extraction flow data must be simulated using diode function generators with one function generator at each node. This either makes phase 1 an expensive and time-consuming operation, or it restricts the number of nodes to be used in the problem. The price paid by the engineer in either case is the accuracy of the solution. In the hybrid computer method, the replenishment-extraction flow data (as obtained from rainfall-pumping records) are approximated by piecewise-linear curves, and their turning points are stored in the digital memory. With a reasonable number of line segments in the function generation can be made quite accurate without the use of expensive analog function generation hardware. Hence the hybrid computer is more economical and accurate than the analog computer.

In digital computer methods [*Tyson and Weber,* 1964], the matrix inversion operation not only takes a major portion of the computer running time but also heavily taxes the digital computer storage space when a large number of nodes (more than 100, for example), are used. The resistance network on one hand saves a considerable part of the computer time and on the other hand makes the storage requirement more or less independent of the size of the matrix to be inverted. Thus a hybrid computer facility using a small and relatively inexpensive digital computer becomes competitive with a large digital computer facility. This is one case where increasing the size of the problem decreases the cost per unit accuracy, thus making it very attractive from an engineering point of view.

Some of the other advantages of a hybrid computer are procedural. The present tendency of operating large digital computer facilities on a closed shop basis is making it very difficult for the engineer to assist the computer with his engineering judgment by interfering with the solution process. The inexpensive hybrid computer allows the engineer to play around with various design alternatives and to see immediately the way he is influencing the solution. This is an advantage that cannot be expressed in dollars and cents.

On the debit side are the errors introduced owing to quantization and analog-digital conversion [*Vidal and Karplus,* 1965; *Karplus,* 1966], and methods are being developed either to minimize or to compensate for these errors.

Acknowledgments. The authors wish to thank W. J. Karplus of UCLA, who first suggested this problem and who provided constant encouragement throughout the preparation of this paper. Some of the material presented in this paper appears as a part of a doctoral dissertation of Venkateswararao Vemuri at UCLA under the supervision of W. J. Karplus.

The work upon which this paper is based was supported in part by funds provided by the U. S. Department of the Interior, Office of Water Resources Research, as authorized under the Water Resources Research Act of 1964, by a National Science Foundation Grant No. GK 1758, and by the University of California Water Resources Center.

REFERENCES

Hubbert, M. K., The theory of ground-water motion, *J. Geol., 48* (8), 1, 785–944, Nov.-Dec. 1940.

Karplus, W. J., *Analog Simulation—Solution of Field Problems,* McGraw-Hill Book Company, Inc., New York, 1958.

Karplus, W. J., A new active-passive network simulator for transient field problems, *Proc. IRE, 49,* 268–275, 1961.

Karplus, W. J., Hybrid computer technique for treating nonlinear partial differential equations, *IEEE Trans. Electronic Computers, 13,* 597–605, 1964.

Karplus, W. J., Analog subroutines for digital computer programs, *Simulation, 4,* 145–150, 1965.

Karplus, W. J., Error analysis of hybrid computer systems, *Simulation, 6,* 120–136, 1966.

Karplus, W. J., and K. Kanus, Convergence considerations in the hybrid computer treatment of partial differential equations, *AFIPS Spring Joint Computer Conf.,* 1965.

Karplus, W. J., and V. Vemuri, Numerical solution of partial differential equations, (in *Digital Computer User's Handbook,* ed. Korn and Klerer), McGraw-Hill Book Company, Inc., New York, 1967.

Liebmann, G., Solution of partial differential equations with a resistance network analog, *Brit. J. Appl. Phys., 1,* 92–103, 1950.

MacNeal, R. H., An asymmetric finite difference network, *Quart. J. Appl. Math., 11,* 295–310, October 1953.

Richtmyer, R. D., *Difference Methods for Initial Value Problems,* Interscience Publishers, New York, 1957.

Russell, R., Unpublished Report, Department of Engineering, University of California, Los Angeles, 1965.

Tyson, H. N., and E. M. Weber, Groundwater management for the nation's future: Computer simulation of groundwater basins, *Am. Soc. Civil Engrs., Hydraul. Div., 90,* 59–78, 1964.

Vidal, J., and W. J. Karplus, Characterization and evaluation of hybrid systems, *Proc. IFAC Symposium on System Engineering,* Tokyo, Japan, August 1965.

(Manuscript received March 8, 1967;
revised May 18, 1967.)