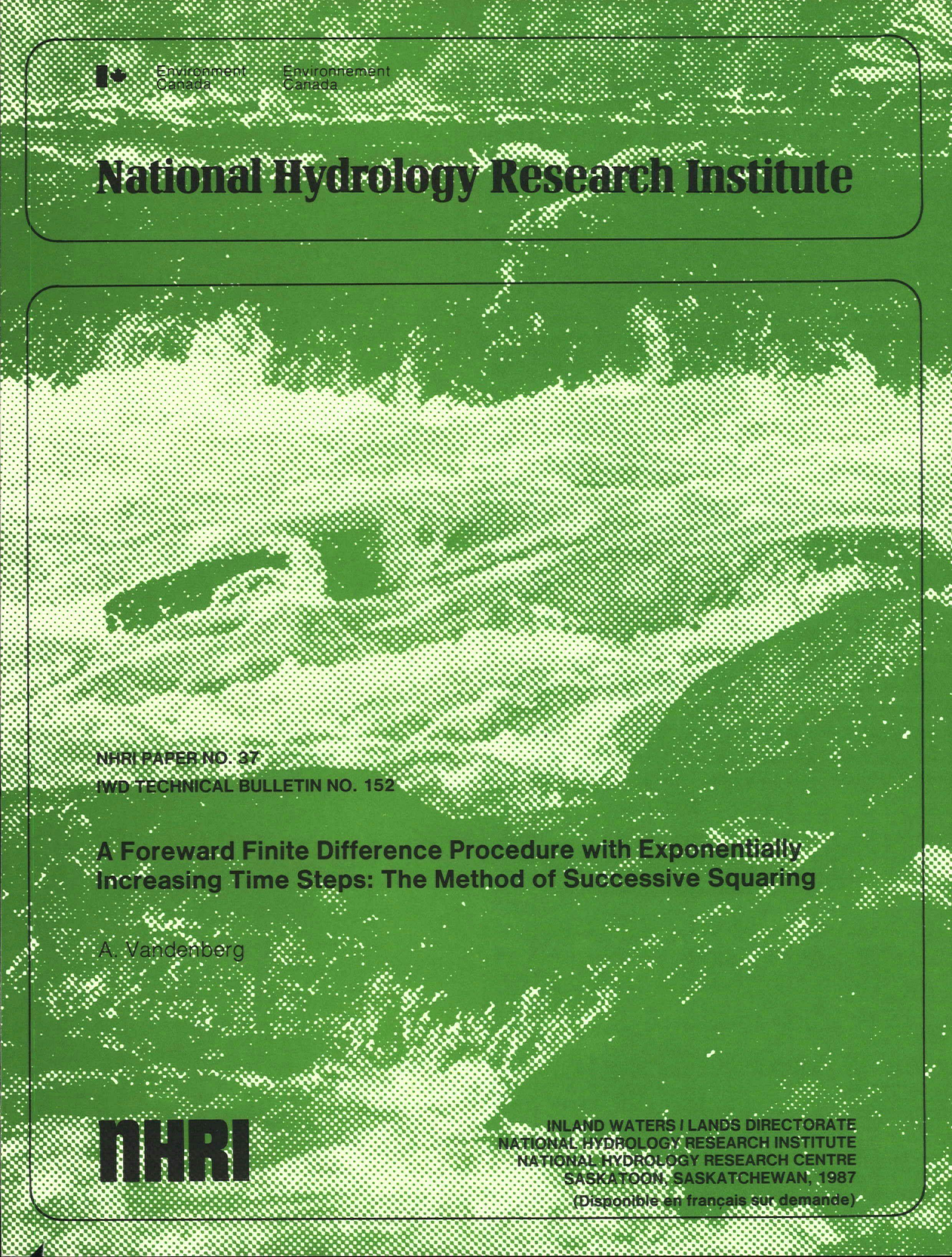
Environment Canada Environnement Canada

National Hydrology Research Institute



NHRI PAPER NO. 37
IWD TECHNICAL BULLETIN NO. 152

A Foreward Finite Difference Procedure with Exponentially Increasing Time Steps: The Method of Successive Squaring

A. Vandenberg

NHRI

INLAND WATERS / LANDS DIRECTORATE
NATIONAL HYDROLOGY RESEARCH INSTITUTE
NATIONAL HYDROLOGY RESEARCH CENTRE
SASKATOON, SASKATCHEWAN, 1987
(Disponible en français sur demande)

Environment Canada Environnement Canada

National Hydrology Research Institute

NHRI PAPER NO. 37
IWD TECHNICAL BULLETIN NO. 152

A Foreward Finite Difference Procedure with Exponentially Increasing Time Steps: The Method of Successive Squaring

A. Vandenberg

NHRI

INLAND WATERS / LANDS DIRECTORATE
NATIONAL HYDROLOGY RESEARCH INSTITUTE
NATIONAL HYDROLOGY RESEARCH CENTRE
SASKATOON, SASKATCHEWAN, 1987
(Disponible en français sur demande)

Published by authority of
the Minister of the Environment

Cat. No. En 36-503/152E

ISBN 0-662-15489-4

Contents

Page

Illustrations

Abstract

The forward finite difference formulation of linear partial differential equations leads to the matrix equation

$$u^{n+1} = Au^{n} + c$$

in which the matrix A and the vector c are constant whenever the differential equation and boundary conditions have no time-dependent coefficients. By a slight artifice, this equation can be changed to

$$u^{n+1} = A^{*} u^{n}$$

after which, on the m^{th} successive squaring of A*, the value of u is calculated at a time that is 2^{m} times the original time step.

Résumé

La formulation à différences finies en avant d'équations aux dérivées partielles conduit à l'équation matricielle

$$u^{n+1} = Au^{n} + c$$

dans laquelle la matrice A et le vecteur c sont des constantes, tandis que l'équation différentielle et les conditions limites ne possèdent pas de coefficients dépendant du temps. Par un artifice simple, cette équation peut s'écrire

$$u^{n+1} = A^{*} u^{n}$$

après quoi, à la $m^{\text{ième}}$ mise au carré successive de A*, la valeur de u est calculée au moment égal à 2^{m} fois le pas temporel initial.

A Foreward Finite Difference Procedure with Exponentially Increasing Time Steps: The Method of Successive Squaring

A. Vandenberg

INTRODUCTION

Of all the procedures for the numerical solution of partial differential equations, the forward finite difference method is the simplest, since values of the independent variable at the space nodes after a time step Δt are calculated explicitly from values before the time step, avoiding the need for simultaneous solution at all the nodes, which is characteristic of implicit procedures. The forward method, however, has been abandoned in practice, since the time step must always be kept under a maximum critical size for the method to remain stable. In this paper, however, it will be shown that for certain types of boundary value problems involving linear differential equations, the step size of the forward finite difference formulation can be increased each iteration by a factor of two.

THE METHOD

To develop the method, the parabolic equation in two-space dimensions will be used, although the method may be applied equally well to other types of linear equations:

$$\partial u/dt = \alpha(x,y)\partial^2 u/\partial x^2 + \beta(x,y)\partial^2 u/\partial y^2 + \gamma(x,y) \qquad (1)$$

In conjunction with Equation 1, a set of boundary conditions is prescribed:

$$u = U(x,y) \text{ on } \Gamma_1 \tag{2a}$$

$$du/ds = \delta(x,y) \text{ on } \Gamma_2 \tag{2b}$$

$$c_1 u + c_2 du/ds = \varepsilon(x,y) \text{ on } \Gamma_3 \tag{2c}$$

where du/ds = the derivative in the direction perpendicular to the boundary Γ_2 or Γ_3, c_1 and c_2 are constants, and the α, β, γ, δ, ε, and U must be independent of time.

In the forward finite difference method, Equation 1 is replaced by its finite difference approximation

$$u^{n+1}_{i,j} = u^n_{i,j} + l_{i,j}\ (u^n_{i-1,j} - 2u^n_{i,j} + u^n_{i+1,j}) + k_{i,j}\ (u^n_{i,j-1} - 2\ u^n_{i,j} + u^n_{i,j+1}) + \gamma'_{i,j} \tag{3}$$

where

$$l_{i,j} = \alpha_{i,j}\ \Delta t/\Delta x^2$$
$$k_{i,j} = \beta_{i,j}\ \Delta t/\Delta x^2$$
$$\gamma'_{i,j} = \gamma_{i,j}\ \Delta t$$

the superscript n indicates the value of u at time t = nΔt, and the subscripts i and j designate the value of the variable at the intersection, in the x-y plane, of the ith vertical grid line and the jth horizontal grid line of the finite difference grid (Fig. 1).

Equation 3 is then applied to each of the intersections (nodes) in turn, except for nodes on the boundary element Γ_1, which have a constant value. Boundary conditions of type (2b) and type (2c) are

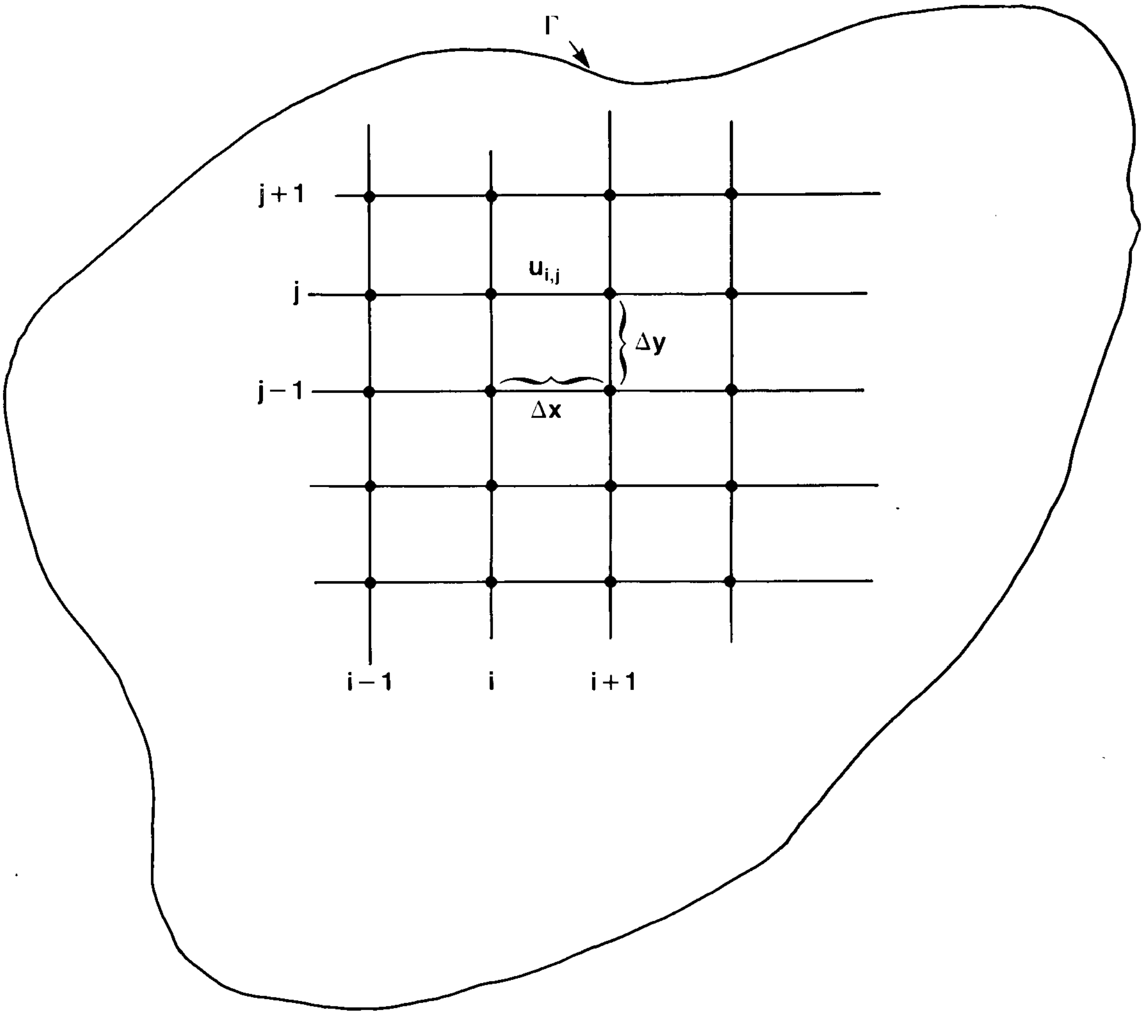


Figure 1. Notation used in the finite difference approximation.

replaced by finite difference approximations from which the value of u at the boundary node is expressed in terms of values at nearby internal nodes.

In matrix notation, then, one iteration, which carries the solution forward in time from $t = n\Delta t$ to $t = (n+1)\Delta t$, is expressed by the equation

$$u^{n+1} = Au^{n} + c \tag{4}$$

where u^n and u^{n+1} are the vectors of nodal values of u at $t = n\Delta t$ and $t = (n+1)\Delta t$, respectively, at all the m nodes where u is not constant; A is the m x m matrix resulting from applying Equation 3 to all of the interior nodes in turn, replacing nonconstant boundary nodes where they occur by appropriate expressions derived from (2b) and (2c); and c is a vector of constant terms deriving from the term $\gamma'_{i,j}$ in Equation 3 and from constant boundary values on Γ_1.

Thus, starting with n = 0, we calculate from u°, the vector of initial conditions:

$$\begin{aligned}
u^1 &= u(\Delta t) = Au^\circ + c \\
u^2 &= u(2\Delta t) = Au^1 + c = A^2u^\circ + Ac + c \\
u^3 &= u(3\Delta t) = A^3u^\circ + A^2c + Ac + c \\
u^n &= u(n\Delta t) = A^nu^\circ + c\sum_{i=0}^{n-1} A^i
\end{aligned} \tag{5}$$

The summation in (5), however, represents a calculating effort that one can very well do without, and it would be much preferable to have, instead of Equation 5, an equation of the form

$$u^n = A^nu^\circ \tag{6}$$

or, in general,

$$u^{k+n} = A^nu^k \tag{7}$$

for then

$$\begin{aligned}
u^2 &= u(2\Delta t) = A^2u^\circ \\
u^4 &= u(4\Delta t) = A^4u^\circ
\end{aligned}$$

etc., and successive squaring of A would result in the calculation of u at $t = \Delta t$, $2\Delta t$, $4\Delta t$, or in general at $t = 2^n \Delta t$ on the nth iteration.

An equation of type (7) can be obtained from Equation 4 by introducing an additional, dummy variable, $\bar{u}$ as the $(m+1)^{th}$ element of u and assigning to $\bar{u}$ the constant value

$$\bar{u} = 1$$

Equation 4 can then be written as

$$u^{n+1} = A^* \, u^n \tag{8}$$

where

$$A^* = \left[\begin{array}{cccc|c} a_{11} & a_{12} & ---- & a_{1m} & c_1 \\ a_{21} & a_{22} & ------ & a_{2m} & c_2 \\ \vdots & \vdots & & \vdots & \vdots \\ a_{m1} & a_{m2} & --- & a_{m}a_{m} & c_{m} \\ \hline 0 & 0 & ------ & 0 & 1 \end{array}\right] = \left[\begin{array}{cc} A & c \\ \hline 0 & 1 \end{array}\right]$$

Thus the matrix now contains the vector of constants c as its last column, which is now regarded as coefficients of the dummy variable $\bar{u}$ in the matrix Equation 8. The last row of the matrix expresses the fact that

$$\bar{u}^{n+1} = \bar{u}^{n}$$

that is, $\bar{u}$ is a constant.

Thus, all the requirements for solution of Equation 3 by a method of successive squaring are met. The method will be illustrated by two examples.

EXAMPLES

In the first example, the solution of Equation 1 is sought in the domain shown in Figure 2; the function u is constant everywhere along the boundary with values as indicated in Figure 2; the initial values at the three internal nodes are indicated in brackets in Figure 2. There are no source/sink terms: γ is zero everywhere. Obviously the solution will tend to a steady state with u_1, u_2, and u_3 equal to 120, 140, and 160, respectively.

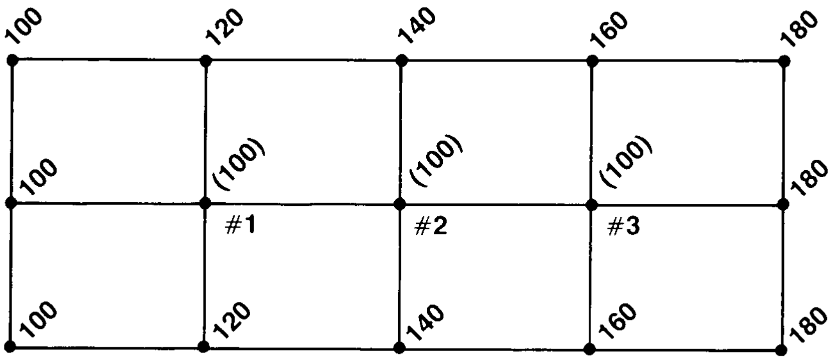


Figure 2. Configuration, initial values, and boundary values of the first example.

Setting $l_{i,j}$ and $k_{i,j}$ equal to 0.2 at all nodes, Equation 3 reduces to the set of three equations:

$$u_1^{n+1} = 0.2u_1^n + 0.2u_2^n \qquad + 0.2\ (100 + 120 + 120)$$

$$u_2^{n+1} = 0.2u_1^n + 0.2u_2^n + 0.2u_3^n + 0.2\ (140 + 140)$$

$$u_3^{n+1} = \qquad 0.2u_2^n + 0.2u_3^n \quad + 0.2\ (180 + 160 + 160)$$

which, on introduction of $\bar{u}$ is expanded to the set of four equations:

$$\begin{aligned}
u_1^{n+1} &= 0.2u_1^n + 0.2u_2^n \qquad\qquad + 68\ \bar{u} \\
u_2^{n+1} &= 0.2u_1^n + 0.2u_2^n + 0.2u_3^n + 56\ \bar{u} \\
u_3^{n+1} &= \qquad\qquad 0.2u_2^n + 0.2u_3^n + 100\ \bar{u} \\
\bar{u} &= \qquad\qquad\qquad\qquad\qquad \bar{u}
\end{aligned}$$

which also can be written in matrix form:

$$u^{n+1} = A{*}u^n$$

where

$$A = \begin{bmatrix} .2 & .2 & 0 & 68 \\ .2 & .2 & .2 & 56 \\ 0 & .2 & .2 & 100 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Successive squaring of A* results in

$$A{*}^2 = \begin{bmatrix} .08 & .08 & .04 & 92.8 \\ .08 & .12 & .08 & 100.8 \\ .04 & .08 & .08 & 131.2 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad A{*}^4 = \begin{bmatrix} .014 & .019 & .013 & 113.54 \\ .019 & .027 & .019 & 130.82 \\ .013 & .019 & .014 & 153.47 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A{*}^8 = \begin{bmatrix} .00074 & .00104 & .00074 & 119.65 \\ .00104 & .00148 & .00104 & 139.50 \\ .00074 & .00104 & .00074 & 159.65 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad A{*}^{16} = \begin{bmatrix} - & - & - & 120.0 \\ - & - & - & 140.0 \\ - & - & - & 160.0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Thus, the powers of A* converge to all zeros except for the last column, which eventually contains the steady state solution of the problem. This final form is typical of all problems that attain a steady state for t → ∞. It can be noted that further squaring of A^{*16} has virtually no effect, that is, for large n

$$A^{*2n} \sim A^{*n}$$

Nor has the initial condition any influence on the steady state solution, which is completely determined by the boundary conditions.

A further note of interest is the existence of the inverse matrices of A and A*, A^{-1} and A^{*-1}, respectively. From Equation 4 we have

$$u^n = A^{-1} (u^{n+1} - c). \tag{9}$$

and from Equation 8

$$u^n = A^{*-1} u^{n+1}. \tag{10}$$

Either equation can be used to step backwards through time, and there is no particular reason why stepping backwards beyond t = 0 would not be permissible. Equation 10 permits us to do so by successive squaring at an increasing rate. It should also be noted, however, that higher powers of A*, as exemplified by A^{*16} of the example, become increasingly ill-conditioned. Thus the closer the solution is to the steady state, the less accurate becomes the retrieval of previous states.

Continuing the evaluation of the sample solution, we calculate

$$A^{*^{-1}} = \begin{bmatrix} 0 & 5 & -5 & 220 \\ 5 & -5 & 5 & -560 \\ -5 & 5 & 0 & 60 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

and we thus have

$$u^{-1} = u\ (t = -\Delta t) = A^{*^{-1}}\ u^{\circ}$$
$$u^{-2} = u\ (t = -2\Delta t) = A^{*^{-1}}\ u^{-1}$$
$$u^{-3} = u\ (t = -3\Delta t) = A^{*^{-1}}\ u^{-2}$$

which gives

$$u^{-1} = \quad 220, \quad -60, \quad +60$$
$$u^{-2} = \quad -380, \quad +1140, \quad -1340$$
$$u^{-3} = +12620, \quad -14860, \quad +7660$$

Thus, although mathematically correct, the "solutions" at negative t soon become meaningless, physically, in this instance.

In the second example, the result is shown when the system does not converge to a steady state, as for example in the configuration of Figure 3, in which the boundary is of the "no-flux across the boundary" type, that is

$$du/ds = 0 \qquad \text{on } \Gamma$$

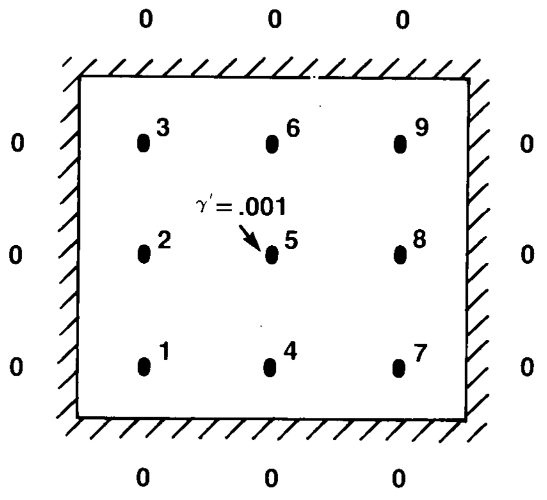


Figure 3. Configuration of the second example.

Thus, for example, both virtual nodes, outside the boundary and opposite node No. 1 in the interior of the region, have the same value at all times as node No. 1. Furthermore a constant source, $\gamma' = .001$, operates at node No. 5 in the centre. Again setting $l_{i,j}$ and $k_{i,j}$ at 0.2, the equivalent of Equation 3 becomes

$$u_{i,j}^{n+1} = 0.2\ (u_{i-1,j}^{n} + u_{i+i,j}^{n} + u_{i,j-1}^{n} + u_{i,j+1}^{n} + u_{i,j}^{n}) + \gamma' \qquad (11)$$

Applying Equation 11 to each of the nine internal nodes of Figure 3, the matrix A* is obtained:

$$A^* = \begin{bmatrix} .6 & .2 & 0 & .2 & 0 & 0 & 0 & 0 & 0 & 0 \\ .2 & .4 & .2 & 0 & .2 & 0 & 0 & 0 & 0 & 0 \\ 0 & .2 & .6 & 0 & 0 & .2 & 0 & 0 & 0 & 0 \\ .2 & 0 & 0 & .4 & .2 & 0 & .2 & 0 & 0 & 0 \\ 0 & .2 & 0 & .2 & .2 & .2 & 0 & .2 & 0 & 0.001 \\ 0 & 0 & .2 & 0 & .2 & .4 & 0 & 0 & .2 & 0 \\ 0 & 0 & 0 & .2 & 0 & 0 & .6 & .2 & 0 & 0 \\ 0 & 0 & 0 & 0 & .2 & 0 & .2 & .4 & .2 & 0 \\ 0 & 0 & 0 & 0 & 0 & .2 & 0 & .2 & .6 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

After a few successive squarings, the matrix will look like

$$A^{*32} = \begin{bmatrix} .111 & .111 & .111 & .111 & .111 & .111 & .111 & .111 & .111 & .00328 \\ .111 & .111 & .111 & .111 & .111 & .111 & .111 & .111 & .111 & .00356 \\ .111 & .111 & .111 & .111 & .111 & .111 & .111 & .111 & .111 & .00328 \\ .111 & .111 & .111 & .111 & .111 & .111 & .111 & .111 & .111 & .00356 \\ .111 & .111 & .111 & .111 & .111 & .111 & .111 & .111 & .111 & .00467 \\ .111 & .111 & .111 & .111 & .111 & .111 & .111 & .111 & .111 & .00356 \\ .111 & .111 & .111 & .111 & .111 & .111 & .111 & .111 & .111 & .00328 \\ .111 & .111 & .111 & .111 & .111 & .111 & .111 & .111 & .111 & .00356 \\ .111 & .111 & .111 & .111 & .111 & .111 & .111 & .111 & .111 & .00328 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

A^{*32} represents a quasi-steady state in which the shape of the surface representing u does not change, but its level rises uniformly at a constant rate. The complete solution at large t can be considered to consist of two parts: (1) an individual part, characteristic for each node, which is steadily increasing since it is the result of the continuously operating source at node No. 5 (this part is contained in the last column and therefore independent of the initial condition); and (2) a part that is equal for all nodes and is the average of the initial values, in this case 1/9 (.111) times the sum of all the nodal values at $t = 0$; it thus represents "what was in the bucket" at the beginning and is still there, but evenly distributed over all the nodes.

EFFICIENCY IN COMPARISON WITH THE STANDARD FORWARD METHOD

Although the exponentially increasing time steps of the method of successive squaring seem, at first sight, to provide a great improvement in calculating efficiency, the method suffers from the disadvantage of a larger number of operations per time step. In the standard forward difference method, the number of additions and the number of multiplications is approximately 3 m, 5 m, or 7 m per time step, for respectively 1-dimensional, 2-dimensional, and 3-dimensional problems, where m is the number of nodes. Thus to carry the solution forward to $t = 2^n \Delta t$ requires $pm2^n$ multiplications, where $p = 3, 5,$ or 7, and approximately the same number of additions.

In the method of successive squaring, the (m+1)m matrix elements (not counting the last row, which does not change) must be calculated at (m+1) additions and the same number of multiplications per element per squaring. Also, to calculate the vector u at each step, an additional m(m+1) multiplications and the same number of additions are required.

Thus, in the method of successive squaring for n iterations, that is, for $t = 2^n \Delta t$, a total of $n[(m+1)^2 m + m(m+1)] = nm(m+1)(m+2)$ additions and the same number of multiplications are needed. And only if

$$(m+1)(m+2) < p\ 2^n/n$$

will the method of successive squaring be more efficient than the standard forward method. For a 2-dimensional problem with $n = 10$, that is, last calculated values at $t = 1024\ \Delta t$, the number of nodes must be less than 21, and with 40 nodes, the solutions must be carried forward at least to $t = 8192\ \Delta t = 2^{13} \Delta t$ for successive squaring to be the more efficient.

APPLICATION TO IMPLICIT FORMULATIONS

Implicit finite difference methods can be expressed in matrix form as

$$Au^{n+1} = Bu^n + c \qquad (12)$$

Again, if the vector c and the matrices A and B are invariant, Equation 12 can be premultiplied by A^{-1}, the inverse of A:

$$u^{n+1} = A^{-1} Bu^{n} + A^{-1}c$$

or

$$u^{n+1} = Du^{n} + d \tag{13}$$

where

$$D = A^{-1}B$$

and

$$d = A^{-1}c$$

Equation 13 can then be extended to

$$u^{n+1} = D^{*} u^{n}$$

where

$$D^{*} = \left| \begin{array}{c|c} D & d \\ \hline 0 & 1 \end{array} \right|$$

Thus the method of successive squaring can be applied to implicit formulations as well.

IN CONCLUSION

Although the method of successive squaring has been described here for the solution of a particular partial differential equation and its forward finite difference approximation, the method is applicable to all types of equations that in finite approximation can be expressed by the matrix equation

$$Au^{n+1} = Bu^{n} + c$$

provided the matrices A and B, and the vector c are time invariants, and the inverse A^{-1} of A exists. These requirements clearly limit the method to linear differential equations.

The method becomes efficient only after a certain number of iterations is exceeded, which number increases with the number of nodes.

As an aside, it has been shown that it is possible to step backwards through time to arrive from an observed state at some point in time to a previous state that gave rise to it. A closer inspection of this possibility may, however, reveal that this type of "historical research" will soon abort because of round-off error.

Canada