AN ITERATIVE METHOD FOR SOLVING FREE BOUNDARY PROBLEMS

By

E. H. DOHA

Dept. of Mathematics Faculty of Science, Cario University, Giza, Egypt

(Received 28 September, 1979)

1. Introduction

Free boundary problems are boundary–value problems in which part of the boundary, the free boundary, is unknown and must be determined as a part of the solution. When a problem involves free boundaries, the boundary conditions must be sufficient to determine both the solution of the partial differential equation and the position of the free boundary. On each of the free boundaries one extra condition, in addition to those normally required by the differential equation, is needed to determine the solution.

In the case of two–dimensional problems, where function theory can be applied, the solution in certain cases can be found analytically (see for instance, Mason and Farkas (1971), Charmonman (1965, 1966), and Cryer (1970)). For three–dimensional flow problems, for example those with axial symmetry, solutions to familiar models (Young et al. (1955)) are not known, although existence and uniqueness theorems have been established (Garabedian (1964)). Other instances of free boundary problems have been considered in (Doha, 1977).

When solving free boundary problems by the method of finite differences, it is implicitly assumed that the solution is sufficiently often differentiable and that the free boundary is smooth.

The main aim of this paper is the development of a general automatic iterative finite difference technique based on Newton's method. The use of Newton's method in solving free boundary–value problems was proposed as future work by Young et al. (1955) and Sankar (1967). It seems that no work has been done so far, this perhaps is due to the difficulty in dealing with the partial derivatives with respect to the free boundary ordinates.

In the present paper, the general formulation of the problem is described in section 2, while the numerical method is explained in section 3, solution of one test problem is given in section 4, and in the final section discussion of the results with some concluding remarks are explained.

2. General formulation of the problem

We are concerned with free boundary problems of the following class. It is required to find a twice differentiable function $u(x, y)$ which satisfies the second—order self—adjoint elliptic partial differenctial equation

$$(1) \qquad \frac{\partial}{\partial x}\left[P(x, y)\frac{\partial u}{\partial x}\right]+\frac{\partial}{\partial y}\left[P(x, y)\frac{\partial u}{\partial y}\right]=0 \quad x, y \in D.$$

Here $P(x, y)$ is a positive continuously differentiable function of x and y, while D is a bounded domain in the xy—plane with a boundary C which has a continuously turning tangent except at a finite number of corners.

The solution $u(x, y)$ of the free boundary problem must satisfy the following conditions:

$$(2) \qquad \alpha_1(x, y)\, u+\beta_1(x, y)\, u_n = \nu_1(x, y) \quad x, y \in C$$

where $\alpha_1(x, y) \geq 0$, $\beta_1(x, y) \geq 0$, $\alpha_1+\beta_1>0$, x, $y \in C \cdot$ α_1 and β_1 are piecewise continuously differentiable functions of x and y, $u_n = \partial u/\partial n$, n is the outward normal to C.

The final condition which must be satisfied by u is the one which characterized the problem as a free boundary problem. Part of the boundary, c_f, is unknown and must be determined as part of solution. The free boundary is required to satisfy certain geometric constraints and the following additional boundary condition must also be satisfied:

$$(3) \qquad \alpha_2(x, y)\, u+\beta_2(x, y)\, u_n = \nu_2(x, y) \quad x, y \in c_f$$

where α_2, β_2 and ν_2 are of the same form as α_1, β_1, ν_1. It is assumed that the six coefficients, α_1 through ν_2 are continuous on c_f.

It should be noted that any two linearly independent combinations of conditions (2) and (3) could be used, and it will be convenient to consider the conditions in the simple form

$$(4) \qquad u(x, y) = f(x,y),$$

$$(5) \qquad u_n(x, y) = g(x,y), \quad x, y \in c_f.$$

3. Numerical method

An iterative method is used to generate a sequence of approximations $c_f^{(k)}$ to the free boundary and $u^{(k)}$ to the unknown function satisfying equations (1), (2), and (5) on $c_f^{(k)}$ $(k = 0, 1, \ldots)$. $u^{(k)}$ will generally not satisfy condition (4) and the boundary is thus moved to $c_f^{(k+1)}$ to approximate equation (4) better using Newton's method.

The method of computing $u^{(k)}$ is given in section 3.1 and that for $c^{(k)}$ in section 3.2.

3.1 Finite – difference solution for $u^{(k)}$

A non-uniform mesh is used, the spacing in the x-direction is chosen arbitrarily. The spacing in the y-direction is again chosen arbitrarily except that of the free boundary corresponds to grid line and the spacing for that part of the mesh is chosen so that mesh corners lie on the free boundary as shown in Figure 1. This part is effectively represented by the polygon formed by the diagonals of meshes lying on it.

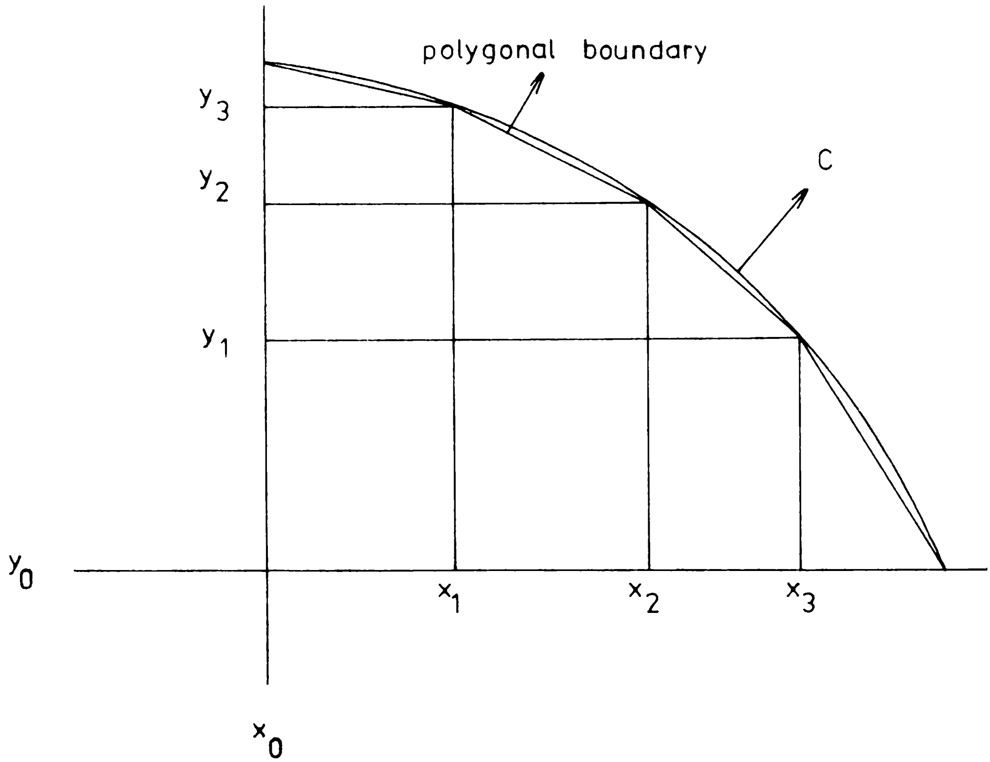

Fig. 1.

We take the vertical mesh lines as $x = x_i$, $i = 1, 2, \ldots, M$, with spacing $p_i = x_{i+1} - x_i$ and horizontal lines as $y = y_j$, $j = 1, 2, \ldots, N$, with $s_j = y_{j+1} - y_j$. Following Varga (1962), we associate with each point (x_i, y_j) interior to the grid an area R_{ij} bounded by the lines

$$x = x_i - \frac{1}{2} p_i, \quad x = x_i + \frac{1}{2} p_{i+1}, \quad y = y_j - \frac{1}{2} s_j \quad \text{and} \quad y = y_j + \frac{1}{2} s_{j+1}.$$

For points on the free boundary R_{ij} is a quadrilateral as shown in Figure 2.

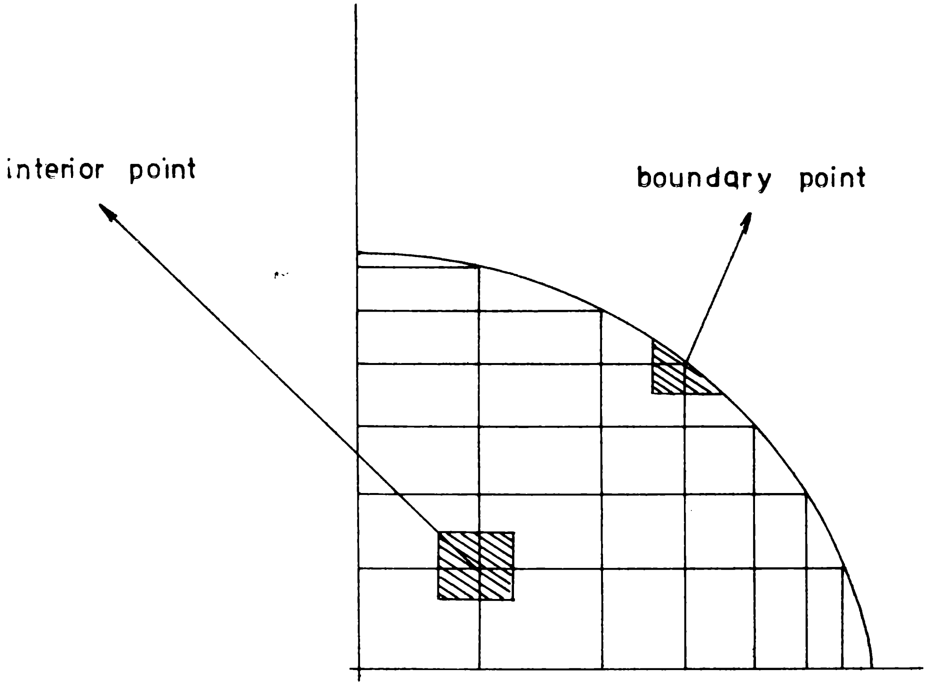

Fig. 2.

For each gridpoint (x_i, y_j) where the function $u(x, y)$ is unknown, Green's theorem is applied to the region R_{ij} to give

$$\iint_{R_{ij}} \left[\frac{\partial}{\partial x}\left\{P(x, y)\frac{\partial u}{\partial x}\right\} + \frac{\partial}{\partial y}\left\{P(x, y)\frac{\partial u}{\partial y}\right\}\right] dx\, dy = \tag{6}$$

$$= \int_{C_{ij}} \left[P(x, y)\frac{\partial u}{\partial x}\, dy - P(x, y)\frac{\partial u}{\partial y}\, dx\right] = 0\,,$$

where C_{ij} is the boundary of R_{ij}.

At internal points the line integral in (6) is approximated by central differences, giving

$$E_{ij}\, u_{ij} - A_{ij}\, u_{i+1,\, j} - B_{ij}\, u_{i-1,\, j} - C_{ij}\, u_{i,\, j+1} - D_{ij}\, u_{i,\, j-1} = 0\,, \tag{7}$$

where

$$A_{ij} = P_{i+\frac{1}{2},\, j}(s_{j-1} + s_j)/p_i\,,$$

$$B_{ij} = P_{i-\frac{1}{2},\, j}(s_{j-1} + s_j)/p_{i-1}\,,$$

$$C_{ij} = P_{i,j+\frac{1}{2}}(p_{i-1}+p_i)/s_j ,$$

$$D_{ij} = P_{i,j-\frac{1}{2}}(p_{i-1}+p_i)/s_{j-1} ,$$

and

$$E_{ij} = A_{ij}+B_{ij}+C_{ij}+D_{ij} ,$$

where

$$P_{i+\frac{1}{2},j} \equiv P\left(x_i+\frac{1}{2}p_i, y_j\right), \quad P_{i-\frac{1}{2},j} \equiv P\left(x_i-\frac{1}{2}p_{i-1}, y_j\right),$$

$$P_{i,j+\frac{1}{2}} \equiv P\left(x_i, y_j+\frac{1}{2}s_j\right) \quad \text{and} \quad P_{i,j-\frac{1}{2}} \equiv P\left(x_i, y_j-\frac{1}{2}s_{j-1}\right).$$

For points along the free boundary, the line integral takes the simpler form

$$\frac{1}{2}(p_{i-1}+p_i)\left[P_{i,j-\frac{1}{2}}(u_{ij}-u_{i,j-1})/s_{j-1}\right]+$$

$$+\frac{1}{2}(s_{j-1}+s_j)\left[P_{i+\frac{1}{2},j}(u_{ij}-u_{i+1,j})/P_i\right]+\int_P^Q P\frac{\partial u}{\partial n}ds = 0 ,$$

the boundary condition (5) gives $\frac{\partial u}{\partial n} = g(x, y)$ on the free boundary, and so the finite difference equations for the points on the boundary may be written as

$$F_{ij}u_{ij}-A_{ij}u_{i+1,j}-D_{ij}u_{i,j-1}+2P_{i,j}g_{i,j}s_{ij} = 0 ,$$

where

$$F_{ij} = E_{ij}-B_{ij}-C_{ij} ,$$

$$s_{ij} \equiv \frac{1}{2}\left(\sqrt{p_{i-1}^2+s_{j-1}^2}+\sqrt{p_i^2+s_j^2}\right).$$

Expressions can be obtained similarly for mesh points (x_i, y_j) on the other boundaries, where u_{ij} is unknown. This method is called the integration method. The great advantage of it is that it is easy to incorporate the condition of the normal derivative, u_n, on the free boundary into the finite difference equations.

For problems where analytic solutions are not known (see, Doha (1977)), the matrix derived from the boundary value problem (1), (2) and (5) has a very simple block tridiagonal form, and so the system is solved by the successive over-relaxation method.

3.2 Moving the free boundary

Let $u(x_i, y_j, Y_j^{(k)})$ denote the value of $u_{ij}^{(k)}$ at any mesh point (x_i, y_j) $(1 \leq i \leq M,\ 1 \leq j \leq N)$ in the domain of solution $D^{(k)}$ where the free boundary ordinate at x_i is given by $Y_j^{(k)}$, then $u(x_i, Y_j^{(k)}, Y_j^{(k)})$ is the value at the boundary point $(x_i, Y_j^{(k)})$. Similarly, we define $u(x_i, y_j, Y_j^{(k+1)})$ and $u(x_i, Y_j^{(k+1)}, Y_j^{(k+1)})$ to give the interior and boundary values of $u_{ij}^{(k+1)}$ in the domain $D^{(k+1)}$, whose free boundary ordinates are given by $Y_j^{(k+1)}$.

Let $P_j \equiv (x_i, Y_j^{(k)})$, $i = 1, 2, \ldots, M$; $j = 1, 2, \ldots, N$ be the grid points on $C_f^{(k)}$ as shown in Figure 3.

The condition (5) is already satisfied on $C_f^{(k)}$, while the second condition on $C_f^{(k)}$, condition (4), is of course, in general not satisfied at P_j. Let $\overline{P}_j \equiv$ $\equiv (x_i, Y_j^{(k+1)})$ be the point on the line $x = x_i$ through the point P_j at which (4) is approximately satisfied. Then,

$$u(x_i, Y_j^{(k)}, Y_j^{(k)}) - f(x_i, Y_j^{(k)}) \neq 0$$

unless $C_f^{(k)}$ is the true free boundary, so we choose $Y_j^{(k+1)}$ to make

$$u(x_i, Y_j^{(k+1)}, Y_j^{(k)}) - f(x_i, Y_j^{(k+1)}) = 0. \tag{8}$$

Following hydrodymanics usage, define a "total" derivative of the function $u(x, y, Y)$ on the free boundary as

$$Du(x, Y, Y)/DY = \partial u/\partial y|_{y=Y} + \partial u/\partial Y|_{y=Y}$$

and application of Newton's method to (8) gives

$$Y_j^{(k+1)} = Y_j^{(k)} - \frac{[u(x_i, Y_j^{(k)}, Y_j^{(k)}) - f(x_i, Y_j^{(k)})]}{\left[\frac{D}{DY_j^{(k)}} u(x_i, Y_j^{(k)}, Y_j^{(k)}) - \frac{d}{dY_j^{(k)}} f(x_i, Y_j^{(k)})\right]}. \tag{9}$$

It is to be noted that the total derivative $Du(x, Y, Y)/DY$ gives the rate of change of u with respect to positional change and boundary change. Because if we take $C_f^{(k-1)}$ to describe the shape and position of the free boundary for $(k-1)$-th iteration, and $C_f^{(k)}$ for the k-th iteration, then two changes happen.

(i) One is due to the positional change of the free boundary which means that the derivative is evaluated at a different point of space, this is given simply by $\partial u(x_i, y_j, Y_j^{(k)})/\partial y_j|_{y_j = Y_j^{(k)}}$.

(ii) The second is due to the change in the domain of solution and is given by

$$\partial u(x_i, y_j, Y_j^{(k)})/\partial Y_j^{(k)}|_{y_j = Y_j^{(k)}} =$$

$$= \lim_{Y_j^{(k-1)} \to Y_j^{(k)}} \frac{u(x_i, Y_j^{(k)}, Y_j^{(k)}) - u(x_i, Y_j^{(k)}, Y_j^{(k-1)})}{Y_j^{(k)} - Y_j^{(k-1)}}.$$

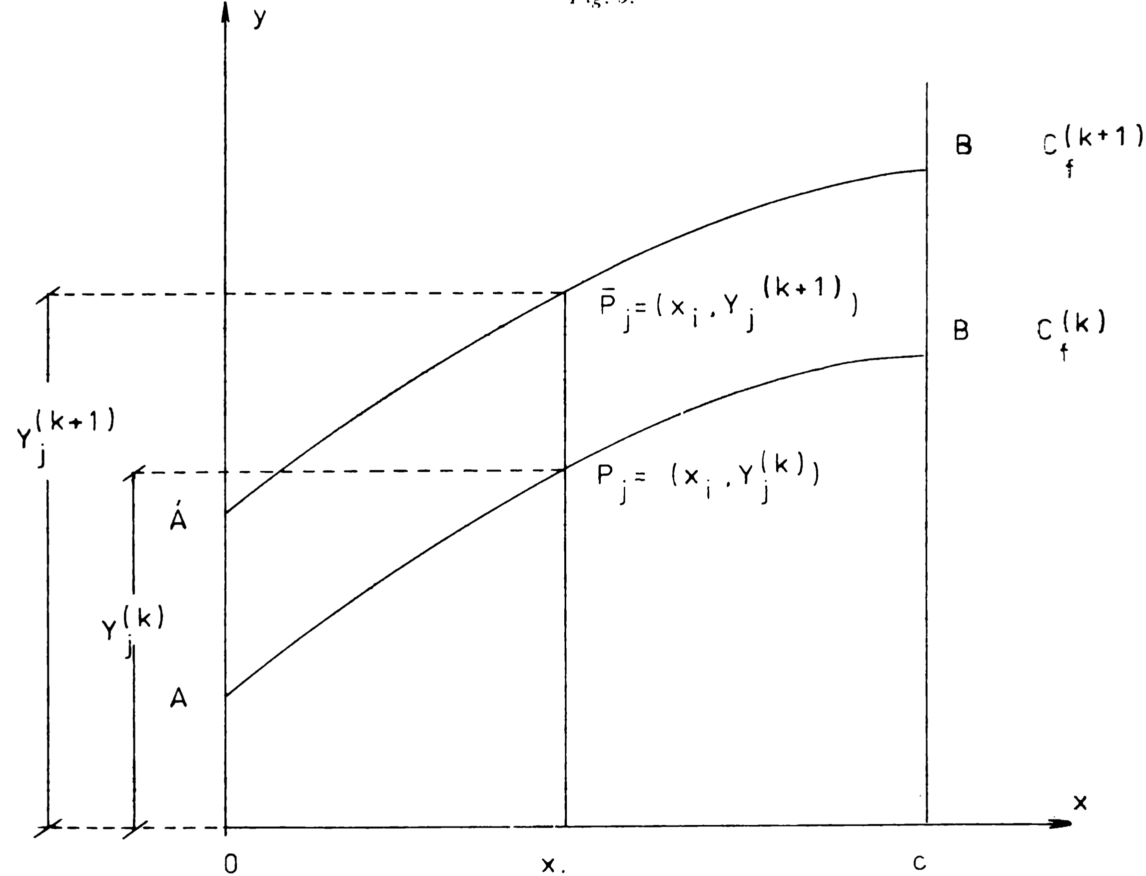

Fig. 3.

$C_j^{(k)}$ is the free boundary AB.
$C_f^{(k+1)}$ is the free boundary A'B'.
$D^{(k)}$ is the domain OABC.
$D^{(k+1)}$ is the domain OA'B'C.

$u(x_i, Y_j^{(k)}, Y_j^{(k)})$ is the value of the distribution at P_j.
$u(x_i, Y_j^{(k+1)}, Y_j^{(k+1)})$ is the value of the distribution at $\bar{p}_j$.
$u(x_i, y_i, Y_j^{(k)})$ is the value of the distribution at any point in $D^{(k)}$.
$u(x_i, y_j, Y_j^{(k+1)})$ is the distribution of u at any point in $D^{(k+1)}$.

Therefore,

$$(10)\qquad Du(x_i, Y_j^{(k)}, Y_j^{(k)})/DY_j^{(k)} =$$

$$= \frac{\partial}{\partial y_j} u(x_i, y_j, Y_j^{(k)})|_{y_j=Y_j^{(k)}} + \frac{\partial}{\partial Y_j^{(k)}} u(x_i, y_j, Y_j^{(k)})|_{y_j=Y_j^{(k)}}.$$

Clearly only a change in the local ordinate of boundary affects the solution (strictly speaking the total derivative should include the effect of movement of the complete boundary).

The first partial derivative in (10) may be calculated from the three-point formula

$$(11)\qquad \frac{\partial}{\partial y_j} u(x_i, y_j, Y_j^{(k)})|_{y_j=Y_j^{(k)}} = l_1' u(x_i, Y_j^{(k)}, Y_j^{(k)}) + l_2' u(x_i, y_{j-1}^{(k)}, Y_j^{(k)}) + + l_3' u(x_i, y_{j-2}^{(k)}, Y_j^{(k)}),$$

where

$$l_1' = (2Y_j^{(k)} - y_{j-1}^{(k)} - y_{j-2}^{(k)})/(Y_j^{(k)} - y_{j-1}^{(k)})(Y_j^{(k)} - y_{j-2}^{(k)}),$$
$$l_2' = (Y_j^{(k)} - y_{j-2}^{(k)})/(y_{j-1}^{(k)} - Y_j^{(k)})(y_{j-1}^{(k)} - y_{j-2}^{(k)}),$$
$$l_3' = (Y_j^{(k)} - y_{j-1}^{(k)})/(y_{j-2}^{(k)} - Y_j^{(k)})(y_{j-2}^{(k)} - y_{j-1}^{(k)}).$$

The second partial derivative may be approximated by

$$(12)\qquad [u(x_i, Y_j^{(k)}, Y_j^{(k)}) - u(x_i, Y_j^{(k)}, Y_j^{(k-1)})]/(Y_j^{(k)} - Y_j^{(k-1)})$$

where $u(x_i, Y_j^{(k)}, Y_j^{(k)})$ is already known, but we have to calculate $u(x_i, Y_j^{(k)}, Y_j^{(k-1)})$, that is, the value of u from the previous iteration evaluated on the new boundary. This can be obtained by extrapolating through the three points $(x_i, Y_j^{(k-1)})$, $(x_i, y_{j-1}^{(k-1)})$ and $(x_i, y_{j-2}^{(k-1)})$.

Alternatively, the second member of (10) can be approximated by

$$(13)\qquad [u(x_i, Y_j^{(k-1)}, Y_j^{(k)}) - u(x_i, Y_j^{(k-1)}, Y_j^{(k-1)})]/(Y_j^{(k)} - Y_j^{(k-1)})$$

and since $u(x_i, Y_j^{(k-1)}, Y_j^{(k-1)})$ is known, so we have to calculate $u(x_i, Y_j^{(k-1)}, Y_j^{(k)})$ which may be obtained by interpolation. Some other ways of approximating the total derivative have been discussed in (Doha, 1977).

We should like to compare two ways of moving the free boundary in a simple context. The alternatives are

(i) Newton's method with $D(x, Y, Y)/DY$ evaluated exactly;

(ii) Newton's method with $D(x, Y, Y)/DY$ approximated by $\partial u(x, y, Y)/\partial y|_{y=Y}$ (i.e. ignoring effect of domain change).

To compare these two methods, calculations on one test problem are described below in section 4.

4. Solution of Laplace's equation inside a rectangle

Let $u(x, y)$ be a function of x and y which satisfies Laplace's equation

$$\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}=0 \tag{14}$$

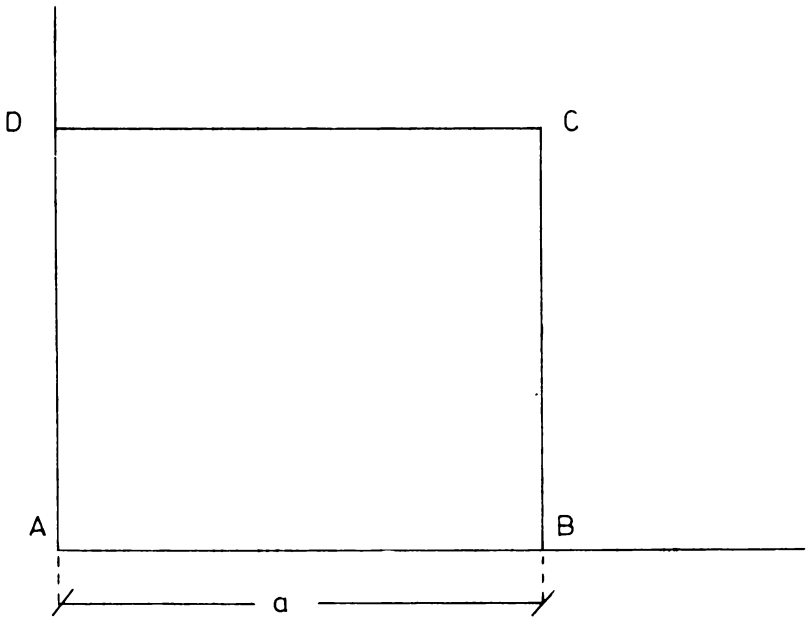

Fig. 4

in a domain D of the form shown in Figure 4. The points A and B are fixed, but the position of the side CD is not known in advance. The boundary conditions are:

$$u(x, y) = 0 \quad x, y \in DABC, \tag{15}$$

and on the free boundary DC:

$$u(x, y) = f(x, y) \quad x, y \in DC, \tag{16}$$

$$\frac{\partial u}{\partial n} = g(x, y) \quad x, y \in DC. \tag{17}$$

We can satisfy all conditions (15) by taking

$$u(x, y) = \sum_{n=1}^{\infty} A_n \sinh\left(\frac{n\pi}{a} y\right) \sin\left(\frac{n\pi}{a} x\right), \tag{18}$$

where A_n, $n = 1, 2, \ldots$ are arbitrary constants.

Let $y = C_k$ be the k-th approximation of the shape and position of the free boundary CD, and imposing the boundary condition (17) on it, we get

$$g(x, C_k) = \sum_{n=1}^{\infty} \frac{n\pi}{a} A_n \cosh\left(\frac{n\pi}{a} C_k\right) \sin\left(\frac{n\pi}{a} x\right)$$

so that,

$$A_n = \frac{2}{n\pi \cosh \frac{n\pi}{a} C_k} \int_0^a g(\xi, C_k) \sin\left(\frac{n\pi}{a} \xi\right) d\xi . \tag{19}$$

Substitution of (19) into (18) gives:

$$u(x, y, C_k) = \frac{2}{\pi} \sum_{n=1}^{\infty} \frac{\sinh \frac{n\pi}{a} y \sin \frac{n\pi}{a} x}{n \cosh \frac{n\pi}{a} C_k} \int_0^a g(\xi, C_k) \sin \frac{n\pi}{a} \xi \, d\xi ,$$

therefore

$$u(x_i, y_j, C_k) = \frac{2}{\pi} \sum_{n=1}^{\infty} \frac{\sinh \frac{n\pi}{a} y_j \sin \frac{n\pi}{a} x_i}{n \cosh \frac{n\pi}{a} C_k} \int_0^a g(\xi, C_k) \sin \frac{n\pi}{a} \xi \, d\xi , \tag{20}$$

hence

$$\left.\frac{\partial u(x_i, y_j, C_k)}{\partial y_j}\right|_{y_j = C_k} = \frac{2}{a} \sum_{n=1}^{\infty} \sin \frac{n\pi}{a} x_i \int_0^a g(\xi, C_k) \sin \frac{n\pi}{a} \xi \, d\xi$$

and

$$\left.\frac{\partial u(x_i, y_j, C_k)}{\partial C_k}\right|_{y_j = C_k} = \frac{2}{\pi} \sum_{n=1}^{\infty} \frac{1}{n} \tanh \frac{n\pi}{a} C_k \sin \frac{n\pi}{a} x_i \int_0^a \frac{\partial g(\xi, C_k)}{\partial C_k} \sin \frac{n\pi}{a} -$$

$$- \frac{2}{a} \sum_{n=1}^{\infty} \tanh^2 \frac{n\pi}{a} C_k \sin \frac{n\pi}{a} x_i \int_0^a g(\xi, C_k) \frac{n\pi}{a} \xi \, d\xi ,$$

then

$$Du(x_i, C_k, C_k)/DC_k = \frac{2}{a} \sum_{n=1}^{\infty} \operatorname{sech}^2 \frac{n\pi}{a} C_k \sin \frac{n\pi}{a} x_i \int_0^a g(\xi, C_k) \sin \frac{n\pi}{a} \xi \, d\xi +$$

$$+ \frac{2}{\pi} \sum_{n=1}^{\infty} \frac{1}{n} \tanh \frac{n\pi}{a} C_k \sin \frac{n\pi}{a} x_i \int_0^a \frac{\partial g(\xi, C_k)}{\partial C_k} \sin \frac{n\pi}{a} \xi \, d\xi . \tag{21}$$

It should be noted that if we put $y_j = C_k$ in (20) and differentiate with respect to C_k (the free boundary ordinates) we would get the same expression given by (21).

Returning to (9), and its application we get

$$C_{k+1} = C_k - \frac{u(x_i, C_k, C_k) - f(x_i, C_k)}{\dfrac{Du(x_i, C_k, C_k)}{DC_k} - \dfrac{df(x_i, C_k)}{dC_k}}, \tag{22}$$

where $u(x_i, C_k, C_k)$ is given by (20) and Du/DC_k by (21), $f(x_i, C_k)$ is given explicitly by (16) and then df/dC_k.

Special case.

If we take $f(x, y) = \sin \frac{\pi}{a} x$ and $g(x, y) = \frac{\pi}{a} \coth \frac{\pi b}{a} \sin \frac{\pi x}{a}$, then it is easily verified that the solution of (14)–(17) is given by

$$u(x, y) = \frac{\sinh \frac{\pi}{a} y \sin \frac{\pi}{a} x}{\sinh \frac{\pi}{a} b}, \tag{23}$$

and the free boundary is given by $y = b$.

In this special case we have

$$\int_0^a g(\xi, C_k) \sin \frac{n\pi}{a} \xi \, d\xi = \frac{\pi}{a} \coth \frac{\pi b}{a} x \begin{cases} \dfrac{a}{2} & n = 1 \\ 0 & \forall\, n \geq 2, \end{cases}$$

hence, equations (20) and (21) give

$$u(x_i, C_k, C_k) = \tanh \frac{\pi}{a} C_k \coth \frac{\pi}{a} b \sin \frac{\pi}{a} x_i ,$$

and

$$\frac{Du(x_i, C_k, C_k)}{DC_k} = \frac{\pi}{a} \operatorname{sech}^2 \frac{\pi}{a} C_k \coth \frac{\pi b}{a} \sin \frac{\pi}{a} x_i$$

and accordingly (22) takes the form

$$C_{k+1} = C_k - \frac{a}{\pi} \frac{\left(\tanh \frac{\pi}{a} C_k - \tanh \frac{\pi}{a} b\right)}{\operatorname{sech}^2 \frac{\pi}{a} C_k} \quad (k = 0, 1, 2, \ldots). \tag{24}$$

If we take $b = 2$, and $a = \pi$ as a numerical example, then (24) takes the form

$$C_{k+1} = C_k - \frac{(\tanh C_k - \tanh 2)}{\operatorname{sech}^2 C_k} \quad (k = 0, 1, 2, \ldots). \tag{25}$$

Take $C_0 = 1.0$ as a first guess, then (25) generates the successive values for C_k which are stopped when the absolute error is less than 10^{-6} (see, Table 1).

Table 1

k	c_k
0	1.0
1	1.482014
2	1.815120
3	1.970916
4	1.999201
5	1.999999
6	2.000000

Returning to (22), if we neglect the contribution to the total derivative due to the change in the domain of solution, then (22) takes the form

$$C_{k+1} = C_k - (\tanh C_k - \tanh 2) \quad (k = 0, 1, 2, \ldots). \tag{26}$$

Again if we take $C_0 = 1.0$ as before, then we get the successive values for C_k given in Table 2.

Another problem has been solved analytically in (Doha, 1977), but details will not be given here.

Table 2

k	c_k
0	1.0
5	1.558432
10	1.741407
20	1.901212
40	1.980436
60	1.996160
80	1.999049
100	1.999746
120	1.999941
140	1.999985
150	1.999995
154	2.000000

5. Discussion of the results

In the previous section, we found that the chief difficulty was how to move the free boundary. However, we have already used two alternative ways (i) and (ii) mentioned in section 3.2.

Tables 1 and 2 show that the number of iterations needed for the solution of the problem to converge is 6 iterations by using (i) and 154 by using (ii). The obvious reason for this increase of the number of iterations is due to ignoring completely the contribution to the total derivative due to the change in the domain of solution.

Various ways of moving a free boundary have been tried by, for example, Garabedian (1956), Cryer (1970), Aitchison (1972), and Fox and Sankar (1973). The last ones applied Regula Falsi, a close relation of Newton's method, but they treated the problem as an N-variable problem in the free boundary ordinates. It was thought that individual adjustment of boundary points might lead to irregular shaped boundary and that some smoothing might be needed, but this has not proved to be the case.

From the above discussion, we conclude the following:

(1) Method (i) is superior if we could calculate the total derivative analytically. Unfortunately, this is impossible for problems which have no analytic solutions.

(2) Method (ii) is easy to apply, and it is a stable one. In general it converges slowly, but we could accelerate its convergence by using Aitkens δ^2-process. The obvious reservation is that the partial derivative due to positional change must have the same sign as the total derivative.

The chief limitation of the method in its present form is that one coordinate of the free boundary must be a monotonic function of the other for the mesh to be defined.

REFERENCES

[1] *Aitchison, J.* (1972): Numerical Treatment of a Singularity in a Free Boundary Problem, Proc. Roy. Soc., Vol. A 330, pp. 573–580.

[2] *Charmonman, S.* (1965): A Solution of the Pattern of Fresh–water Flow in an Unconfined Coastal Aquifer, Jour. Geoph. Res., Vol. 70, pp. 2813–2819.

[3] *Charmonman, S.* (1966): A Numerical Method of Solution of Free Surface Problems, Jour. Geoph. Res., Vol. 71, pp. 3861–3868.

[4] *Cryer, C.W.* (1970): On the Approximate Solution of Free Boundary Problems Using Finite Differences, Jour, Assoc. Comp. Mach., Vol. 17, pp. 397–411.

[5] *Doha, E. H.* (1977): Iterative Finite Difference Method for the Solution of Free Boundary Value Problems, D. Phil. Thesis, University of Lancaster.

[6] *Fox, L.*, and *Sankar. R.* (1973): The Regula – Falsi Method for Free Boundary Problems, Jour. Inst. Maths. Applics., Vol. 12, pp. 49–54.

[7] *Garabedian, P. R.* (1956): Calculation of Axially Symmetric Cavities and Jets, Pac. Jour. Math., Vol. 6, pp. 611–684.

[8] *Sankar, R.* (1967): Numerical Solution of Differential Equations, D. Phil, Thesis, University of Oxford.

[9] *Mason, J. C.*, and *Farkas, I.* (1971): Continuous Methods for Boundary Problems, Tech. Rep. No. 25, Dept. of Computer Science, Univ. Toronto, Canada.

[10] *Varga, R. S.* (1962): Matrix Iterative Analysis, Prentice – Hall.

[11] *Young, D.* et al. (1955): The Computation of an Axially Symmetric Free Boundary Problem on NORC, Pt. I. N. P. G. Report No. 1413.