

An efficient iterative scheme for series solutions to Laplacian free boundary problems

W. W. Read S. R. Belward P. J. Higgins*

(Received 1 October 2001; revised 31 October 2002)

Abstract

Non-linear Laplacian free boundary problems arise in many places in the physical sciences and engineering. Typical applications include locating the water table in ground-water problems, and fully non-linear problems such as flow over topography. Analytic series methods are used to solve these problems by iteratively improving an initial estimate of the free boundary location — at each step, the problem reduces to solving a known boundary problem. As the boundary geometry is not regular, the series coefficients at each iteration are obtained by solving a matrix equation, instead of using an orthogonality relationship. The components of the matrix equations are inner products that result

*School of Mathematical & Physical Sciences, James Cook University, Townsville AUSTRALIA. <mailto:Wayne.Read@jcu.edu.au>

⁰See <http://anziamj.austms.org.au/V44/CTAC2001/Read> for this article, © Austral. Mathematical Soc. 2003. Published 1 April 2003, amended April 8, 2003. ISSN 1446-8735

from minimising the boundary errors in the least squares (or L_2 norm) sense. As the size of the (normal) matrices generated are relatively small, most of the computational effort is spent evaluating these inner products. In this paper, an efficient method is presented to evaluate these integrals that result in an order of magnitude increase in the overall efficiency of the solution process. This increase in efficiency does not come at the cost of accuracy, after suitable modifications are made to the iterative process.

Contents

1	Introduction	C646
2	Brief problem description	C648
2.1	Analytic series solution	C650
3	Eigenfunction expansions	C651
4	Least squares estimation	C653
5	The interpolation approach	C654
5.1	Discrete least squares and interpolation	C656
6	Results and conclusions	C656
	References	C662

1 Introduction

Solutions for Laplacian free boundary problems are required in a large number of practical applications. Free boundary problems are usually very difficult to solve, as the location of one (or more) of the boundaries has to be determined as part of the solution process. Problems of this type include steady seepage through an aquifer and fully nonlinear flow over topography. Typical solution methods for arbitrarily shaped flow domains are iterative numerical schemes such as the Boundary Integral Equation Method (BIEM) [5, 1]. In these methods, an initial estimate of the free boundary location is updated using one of the free boundary conditions as a cost function. However, analytic series methods can also be used to provide accurate, continuous solutions for the flow field. The series coefficients are obtained using least squares methods applied to non-orthogonal basis functions. Solutions thus obtained have all the advantages inherent in analytic solutions, including exact maximum error bounds [3].

Steady seepage problems have been solved using analytic series methods for arbitrarily shaped flow domains [6]. This method has also been used on more difficult seepage problems, where steady solutions can be much harder to obtain [8, 9]. The process consists of iteratively updating an initial estimate of the water table location, in the same fashion as the numerical schemes, until a (preset) error tolerance is met. However, in all of these cases, the free boundary conditions are linear, and the free boundary varies slowly and smoothly along its length. Consequently, only ten to twenty terms are necessary in the truncated series to ensure an accurate solution, and the iterative scheme usually converges in ten to fifty iterations.

Solutions for supercritical flow over topography have also been obtained using analytic series methods [7]. Free boundary problems of this type are much more difficult to solve than seepage problems,

as the free boundary condition is genuinely nonlinear and cannot be linearized using the stream function, as is the case for the steady water table [6]. In addition, subcritical ‘lee wave’ solutions vary strongly along the length of the free surface, and in the limiting cases the wave crests are sharply pointed. For problems of this type, 100 to 150 terms are necessary to ensure an accurate series solution at each iterative step, and the iterative process itself may take hundreds of iterations.

Both the numerical and analytic series schemes require relatively large amounts of computational time to resolve the free boundary for the more difficult problems. Both methods involve generating and solving large systems of equations at each step of the iterative process. Typical $n \times n$ matrix sizes for the BIEM are $n \approx 300$ – 400 , whereas for the analytic series methods $n \approx 100$ – 150 . Although the size of the matrices generated using the analytic series method are significantly smaller than for BIEM, there is a significantly larger amount of computational time required to generate each matrix entry. Each matrix entry consists of an inner product integral which has to be evaluated numerically in nearly all cases. Consequently, the series methods usually take more time at each iteration than BIEM. However, the analytic series method usually requires fewer iterations to converge than BIEM. Note that in this paper an iteration is called a solution of the matrix equations, not an iteration of Newton’s method or some similar optimization scheme.

An approximate interpolation approach [4] can be used on the boundary conditions, thus negating the need for the inner products in the least squares series method. The analytic nature of the solution is still retained, as the truncated series still satisfies Laplace’s equation and some of the boundary conditions exactly. However, the boundary errors are somewhat poorer, and to achieve the same accuracy as the least squares method, the number of terms

in the series has to be doubled. This results in similar sized matrices to BIEM, and also causes a loss of accuracy in the derivatives of the series solution, necessary to obtain velocity profiles in the solution domain.

In this paper, an interpolation method is used to replace the inner product integrals used in the least squares series method. As a consequence, most of the features of the pure least squares series solution are preserved. In particular, the boundary errors are not significantly larger than for the pure least squares method. Consequently, the same number of terms in the truncated series solution is appropriate for both methods. However, the matrices used at each step in the iterative process can be generated in constant time, reducing the computational complexity by at least an order of magnitude. In Section 2, a brief description of the salient features of the series solution method used at each iterative step is provided. The eigenfunction expansion approach used to represent the series coefficients is detailed in Section 3. Then the least squares method used to evaluate the expansion coefficients is given in Section 4, and the interpolation based approach in Section 5. Finally, the boundary errors and computational times for both approaches are compared and discussed in Section 6.

2 Brief problem description

In this section, a brief summary of the mathematical problem formulation that is directly relevant to each iterative step of the series method is presented. The specific formulation used is for incompressible fluid flowing over an obstacle with uniform upstream initial velocity—full details of the problem formulation can be found in [1] or [2]. The problem reduces to solving Laplace's equation for

a stream function $\Psi(x, y)$:

$$\nabla^2 \Psi(x, y) = 0, \quad (1)$$

subject to the boundary conditions:

$$\Psi(x, y) = 0 \quad \text{on} \quad y = B(x), \quad (2)$$

$$\Psi(x, y) = 1 \quad \text{on} \quad y = \eta(x), \quad (3)$$

where $y = B(x)$ and $y = \eta(x)$ are the bottom and top boundaries, respectively. After truncating the infinite flow domain to the interval $x \in [-L, L]$, where L is chosen large enough to capture most of the flow characteristics, the side boundary conditions become

$$\Psi(-L, y) = y, \quad \Psi(L, y) = ay, \quad (4)$$

where a is the reciprocal of the fluid height downstream.

The series method is directly applicable when the side boundary conditions are homogeneous. Thus the original problem is now transformed to a related function $\psi(x, y)$, so that homogeneous boundary conditions exist at $x = \pm L$:

$$\Psi(x, y) = \psi(x, y) + y + \frac{(x + L)}{2L}(a - 1)y. \quad (5)$$

The problem then reduces to solving

$$\nabla^2 \psi(x, y) = 0, \quad (6)$$

subject to the homogeneous side boundary conditions

$$\psi(-L, y) = \psi(L, y) = 0 \quad (7)$$

and

$$\psi[x, B(x)] = -B(x) - \frac{(x + L)}{2L}(a - 1)B(x) = h^b(x), \quad (8)$$

$$\psi[x, \eta(x)] = 1 - \eta(x) - \frac{(x + L)}{2L}(a - 1)\eta(x) = h^t(x). \quad (9)$$

Equations (6) to (9) represent the transformed problem, which is now solved assuming the free surface location has been estimated. Details of the functional approximations of the free surface $\eta(x)$ are given in [2], and the iterative method used to update the free boundary location is given in [7]. The cost function used for these updates is the integrated form of the momentum equation, that is, the Bernoulli equation, which when evaluated on the free surface yields

$$\frac{1}{2}F^2 \left[\left(\frac{\partial \Psi}{\partial x} \right)^2 + \left(\frac{\partial \Psi}{\partial y} \right)^2 \right] + \eta(x) = \frac{1}{2}F^2 + 1 \quad \text{on} \quad y = \eta(x). \quad (10)$$

F is the nondimensional Froude number which characterises whether the flow regime is supercritical, critical or subcritical.

2.1 Analytic series solution

After applying the method of separation of variables to (6), the series solution is

$$\psi(x, y) = \sum_{n=1}^{\infty} \alpha_n u_n(x, y) + \beta_n v_n(x, y),$$

where

$$\begin{aligned} u_n(x, y) &= \sinh\left(\frac{n\pi y}{2L}\right) \sin\left(\frac{n\pi(x+L)}{2L}\right), \\ v_n(x, y) &= \cosh\left(\frac{n\pi y}{2L}\right) \sin\left(\frac{n\pi(x+L)}{2L}\right). \end{aligned}$$

Note that $\psi(x, y)$ satisfies Laplace's equation (6) and the side boundary conditions (7) exactly. The constants α_n and β_n are determined

using the upper and lower boundary conditions (8), (9). Here

$$h^b(x) = \sum_{n=1}^{\infty} \alpha_n u_n^b(x) + \beta_n v_n^b(x), \quad (11)$$

$$h^t(x) = \sum_{n=1}^{\infty} \alpha_n u_n^t(x) + \beta_n v_n^t(x), \quad (12)$$

where $u_n^b(x) = u_n[x, B(x)]$, $u_n^t(x) = u_n[x, \eta(x)]$, etc. At this point in the solution process the classical approach breaks down. When the bottom and top boundary geometries $B(x)$ and $\eta(x)$ are constant, an orthogonality relationship can be used to determine the series coefficients. However, this is clearly not the case here, and a different approach must be used.

3 Eigenfunction expansions

Although the traditional approach (based on eigenfunction expansions) can't be used to evaluate the series coefficients, this method can be extended to non-orthogonal basis functions. Consider the bottom boundary condition (11). Assuming that $u_n^b(x), h^b(x) \in \text{span} \{v_1^b(x), v_2^b(x), \dots\}$, then $u_n^b(x)$ and $h^b(x)$ are expanded in terms of $v_i^b(x)$, $i = 1, 2, \dots$:

$$u_n^b(x) = \sum_{i=1}^{\infty} k_{in}^{ub} v_i^b(x), \quad n = 1, 2, \dots \quad (13)$$

$$h^b(x) = \sum_{n=1}^{\infty} k_n^{hb} v_n^b(x), \quad (14)$$

where k_{in}^{ub} and k_n^{hb} are the expansion coefficients. Substituting into the bottom boundary condition (11),

$$\sum_{n=1}^{\infty} k_n^{hb} v_n^b(x) = \sum_{n=1}^{\infty} \left[\alpha_n \left(\sum_{i=1}^{\infty} k_{in}^{ub} v_i^b(x) \right) + \beta_n v_n^b(x) \right]. \quad (15)$$

After some re-arrangement, this becomes

$$\sum_{n=1}^{\infty} \left(\sum_{i=1}^{\infty} k_{ni}^{ub} \alpha_i - k_n^{hb} + \beta_n \right) v_n^b(x) = 0. \quad (16)$$

Assuming the basis functions $v_n^b(x)$, $i = 1, 2, \dots$ are linearly independent, then

$$\beta_n = k_n^{hb} - \sum_{i=1}^{\infty} k_{ni}^{ub} \alpha_i, \quad n = 1, 2, \dots \quad (17)$$

A similar procedure is used for the top boundary condition (12). First, let

$$u_n^t(x) = \sum_{i=1}^{\infty} k_{in}^{ut} v_i^t(x), \quad n = 1, 2, \dots \quad (18)$$

$$h^t(x) = \sum_{n=1}^{\infty} k_n^{ht} v_n^t(x), \quad (19)$$

where k_{in}^{ut} and k_n^{ht} are the expansion coefficients. After some manipulation, the top boundary condition (12) becomes

$$\sum_{i=1}^{\infty} \left(k_{ni}^{ut} - k_{ni}^{ub} \right) \alpha_i = k_n^{ht} - k_n^{hb}, \quad n = 1, 2, \dots \quad (20)$$

Equations (17) and (20) are two (infinite) sets of linear relationships for α_n and β_n . In practice, the series solution is truncated after a sufficient number of terms have been included. Then

$$\psi(x, y) \approx \psi_N(x, y) = \sum_{n=1}^N a_n u_n(x, y) + b_n v_n(x, y), \quad (21)$$

where ψ_N , a_n and b_n are estimators of ψ , α_n and β_n respectively.

The $N \times N$ matrix equations that need to be solved to obtain $[\mathbf{a}]_i = a_i$, $[\mathbf{b}]_i = b_i$ are

$$\left(K^{ut} - K^{ub}\right) \mathbf{a} = \mathbf{k}^{ht} - \mathbf{k}^{hb}, \quad \mathbf{b} = -\mathbf{k}^{hb} - K^{ub} \mathbf{a}, \quad (22)$$

where K^{ut} , K^{ub} , \mathbf{k}^{ht} , \mathbf{k}^{hb} are matrices of expansion coefficients:

$$[K^{ut}]_{ij} = k_{ij}^{ut}, \quad [K^{ub}]_{ij} = k_{ij}^{ub}, \quad [\mathbf{k}^{ht}]_i = k_i^{ht}, \quad [\mathbf{k}^{hb}]_i = k_i^{hb}, \quad (23)$$

for $i, j = 1, \dots, N$.

4 Least squares estimation

The matrices of eigenfunction expansion coefficients (23) must be evaluated before the series coefficients a_n and b_n are determined using (22). The direct approach is to use least squares to minimize the error in the eigenfunction expansions (13), (14), (18) and (19). These expansions are truncated after N terms, and then the squared residual error ε_n^{ub} of (13) is minimised:

$$\varepsilon_n^{ub} = \int_{-L}^L \left(u_n^b(x) - \sum_{i=1}^N k_{in}^{ub} v_i^b(x) \right)^2 dx. \quad (24)$$

After taking partial derivatives and setting the result to zero, the Normal equations for the expansion coefficients $[K^{ub}]_{ij} = k_{ij}^{ub}$ are:

$$V_v^b K^{ub} = V_u^b, \quad \text{where} \quad [V_v^b]_{ij} = \langle v_i^b, v_j^b \rangle, \quad [V_u^b]_{ij} = \langle u_i^b, v_j^b \rangle, \quad (25)$$

where the inner product

$$\langle p, q \rangle = \int_{-L}^L p(x)q(x) dx. \quad (26)$$

Similar sets of equations are obtained for the other expansion coefficients $[\mathbf{k}^{hb}]_i = k_i^{hb}$, $[K^{ut}]_{ij} = k_{ij}^{ut}$, $[\mathbf{k}^{ht}]_i = k_i^{ht}$:

$$V_v^b \mathbf{k}^{hb} = \mathbf{h}_v^b, \quad \text{where } [V_v^b]_{ij} = \langle v_i^b, v_j^b \rangle, \quad [\mathbf{h}_v^b]_i = \langle h^b, v_i^b \rangle, \quad (27)$$

$$V_v^t K^{ut} = V_u^t, \quad \text{where } [V_v^t]_{ij} = \langle v_i^t, v_j^t \rangle, \quad [V_u^t]_{ij} = \langle u_i^t, v_j^t \rangle, \quad (28)$$

$$V_v^t \mathbf{k}^{ht} = \mathbf{h}_v^t, \quad \text{where } [V_v^t]_{ij} = \langle v_i^t, v_j^t \rangle, \quad [\mathbf{h}_v^t]_i = \langle h^t, v_i^t \rangle. \quad (29)$$

At each step in the iterative process, K^{ut} , \mathbf{k}^{ht} are evaluated. Note that the expansion coefficients K^{ub} and \mathbf{k}^{hb} need only be evaluated once, as the bottom boundary does not change with each iteration. In practice, the inner products are calculated using numerical quadrature routines. All of these integrals are oscillatory, and can be computationally expensive to evaluate. Typically each integral will take of $\mathcal{O}(N)$ time, and thus generating the matrix entries will take $\mathcal{O}(N^3)$ time to evaluate. For large N (> 50), this becomes computationally inefficient.

5 The interpolation approach

An interpolation approach has been used directly on the top and bottom boundary conditions (11), (12) for seepage problems [4]. Using this method, the series coefficients are forced to satisfy the top and bottom boundary conditions at N points $x_i \in (-L, L)$, $i = 1, \dots, N$. That is,

$$h^b(x_i) = \sum_{n=1}^N a_n u_n^b(x_i) + b_n v_n^b(x_i), \quad (30)$$

$$h^t(x_i) = \sum_{n=1}^N a_n u_n^t(x_i) + b_n v_n^t(x_i), \quad (31)$$

for $i = 1, \dots, N$. This leads directly to a set of matrix equations for $[\mathbf{a}]_i = a_i$ and $[\mathbf{b}]_i = b_i$:

$$F^{ub} \mathbf{a} + F^{vb} \mathbf{b} = \mathbf{f}^b, \quad (32)$$

$$F^{ut} \mathbf{a} + F^{vt} \mathbf{b} = \mathbf{f}^t, \quad (33)$$

where

$$[F^{ub}]_{ij} = u_j^b(x_i), \quad [F^{vb}]_{ij} = v_j^b(x_i), \quad [\mathbf{f}^b]_i = h^b(x_i), \quad (34)$$

$$[F^{ut}]_{ij} = u_j^t(x_i), \quad [F^{vt}]_{ij} = v_j^t(x_i), \quad [\mathbf{f}^t]_i = h^t(x_i), \quad (35)$$

The series coefficients \mathbf{a} and \mathbf{b} are obtained from

$$\left(F^{ut} - F^{vt} (F^{vb})^{-1} F^{ub} \right) \mathbf{a} = \mathbf{f}^t - F^{vt} (F^{vb})^{-1} \mathbf{f}^b, \quad (36)$$

$$\mathbf{b} = (F^{vb})^{-1} \mathbf{f}^b - (F^{vb})^{-1} F^{ub} \mathbf{a}. \quad (37)$$

The main advantage of this approach is that the entries in matrices F^{ut} , F^{vt} and \mathbf{f}^t are all generated in constant time, and (compared to the inner products case in the previous section) the matrices themselves are generated in constant time. Unfortunately, this method is not accurate enough, and twice as many terms are used to obtain the same accuracy. Although generating the necessary matrices is still much more efficient than for the least squares method, the matrices become too large to ignore the cost of the matrix inversions needed to solve for the series coefficients. In addition, the derivatives needed for the Bernoulli cost function tend to be much more inaccurate. This situation is not acceptable, as the inaccuracies in the cost function leads to a significant degradation in the performance of the iterative process. In some cases the iterative scheme will not converge at all.

5.1 Discrete least squares and interpolation

An alternative approach is to use discrete least squares at the eigenfunction expansion level. Consider the expansion equation (13) for k_{lj}^{ub} , with the error minimised at $M \geq N$ discrete points x_i , $i = 1, \dots, M$:

$$u_j^b(x_i) = \sum_{l=1}^N v_l^b(x_i) k_{lj}^{ub}, \quad i = 1, \dots, M, \quad j = 1, \dots, N. \quad (38)$$

In matrix form, the over determined system is given by

$$\underbrace{U^b}_{M \times N} = \underbrace{V^b}_{M \times N} \underbrace{\hat{K}^{ub}}_{N \times N}, \quad (39)$$

where, for $i = 1, \dots, M$, $j = 1, \dots, N$:

$$[U^b]_{ij} = u_j^b(x_i), \quad [V^b]_{ij} = v_j^b(x_i), \quad [\hat{K}^{ub}]_{ij} = \hat{k}_{ij}^{ub}. \quad (40)$$

The standard discrete least squares method is now used to minimize the error at the M points. That is, multiply (39) by $(V^b)^T$:

$$(V^b)^T V^b \hat{K}^{ub} = (V^b)^T U^b. \quad (41)$$

This reduces the overdetermined system to a square $N \times N$ system of equations that is readily solved for the expansion coefficients. Note that when $M = N$, this method reduces to interpolation, which also minimizes the least squares error—that is, the error at the N points is zero.

6 Results and conclusions

The accuracy and efficiency of the least squares and interpolation methods are compared by focusing on a particular problem. Ini-

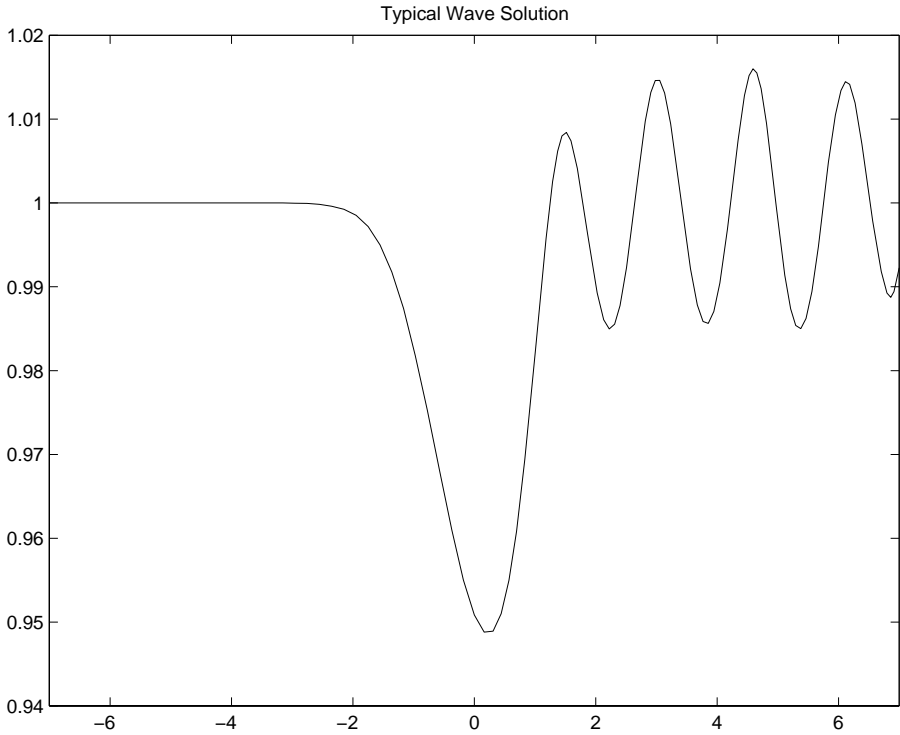


FIGURE 1: A typical free surface profile.

tially, consider a fully developed wave profile for one of the limiting cases—that is, approaching the maximum obstacle height. The wave profile is given in Figure 1. For the interpolation approach, $M = N$ equally spaced points were used. This minimum number of points has been deliberately chosen so that the worst behaviour of the method is apparent.

The accuracy of both methods is given by the root mean square (rms) errors in the top and bottom boundary conditions (2) and (3), as well as the errors in the cost function (10). Consider the stream function $\Psi(x, y)$ approximated by the truncated series $\Psi_N(x, y)$

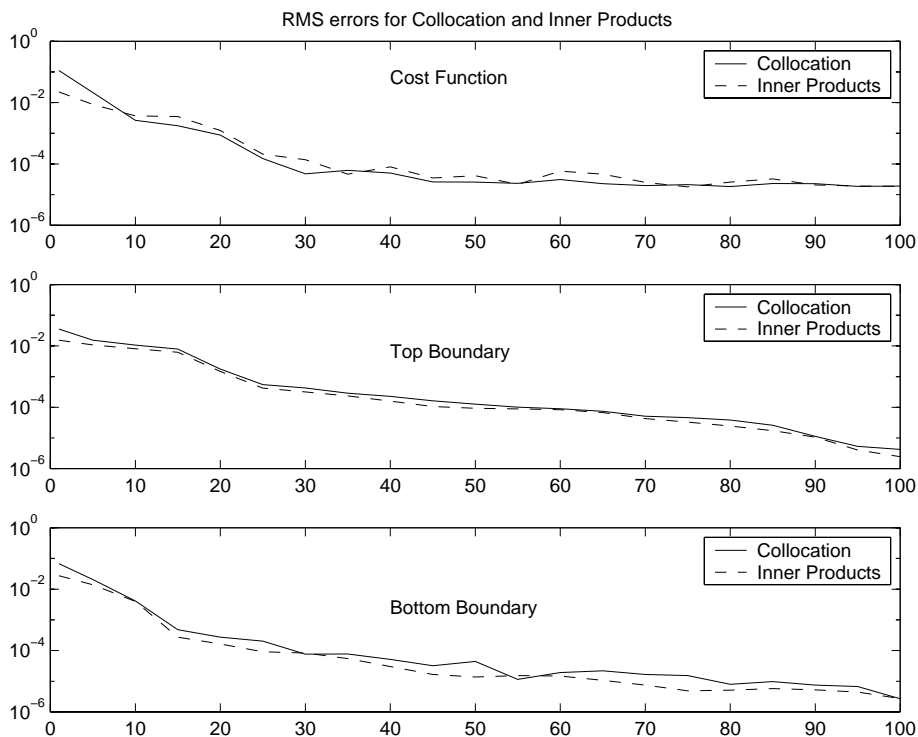


FIGURE 2: rms errors versus N , the number of terms in the series.

along $y = f(x)$, where $f(x)$ is the top or bottom boundary. The rms error

$$\epsilon_f = \left[\frac{1}{2L} \int_{-L}^L (\Psi(x, f(x)) - \Psi_N(x, f(x)))^2 dx \right]^{1/2}. \quad (42)$$

A similar equation for the rms error in the cost function is also readily available.

Figure 2 shows the rms errors of the least squares and interpolation methods for the wave profile given in Figure 1. The number of terms in the series varies from two to one hundred. For each rms

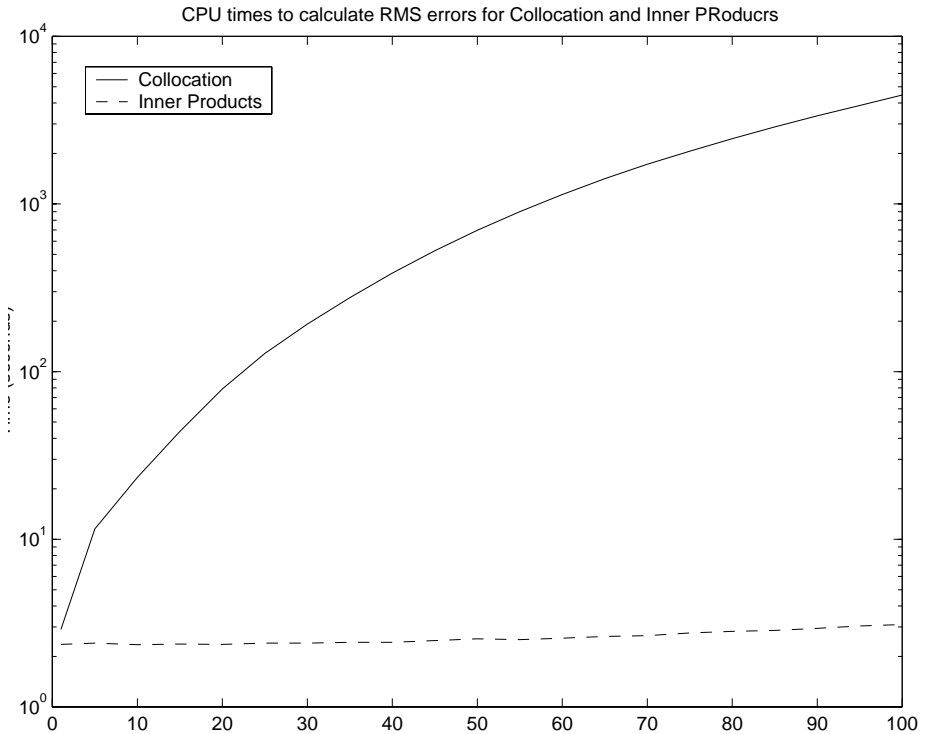


FIGURE 3: CPU times for interpolation and least squares method

error type, the behaviour of the two methods is very similar. The series appears to converge as the number of terms in the series increases. This is not totally unexpected as an increase in the number of interpolation points effectively improves the discrete least squares approximation to the continuous case. The conclusion is that the interpolation method gives sufficiently accurate solutions compared to the continuous least squares method, without having to increase the number of terms in the series.

Next, the actual CPU execution times necessary to calculate the series solution, for different numbers of terms in the series, are com-

pared. Figure 3 shows the CPU times taken to generate a solution for the wave profile in Figure 1, for N varying from two to one hundred (the times given have been obtained using the UNIX time command). Clearly, the interpolation method is at least one order of magnitude more efficient than the least squares method, and is more like two to three orders of magnitude faster. To reinforce this point, consider the times taken for each method, for ten, fifty and one hundred terms in the series solution. The interpolation times are 2.35, 2.55 and 3.10 respectively, whereas the corresponding times for the least squares approach are 23.4, 697.5 and 4463.7.

Finally, the two methods are compared as they iteratively update a free boundary problem. For this analysis, $N = 100$ terms are used in both series solutions. The starting profile of the free surface is constant: $\eta(x) = 1$. Figure 4 shows the cost function and top boundary errors for both methods. See that both methods are converging to the same wave profile. Note that the very low errors at the start of the iterative sequence are due to the constant initial profile—with no variation in the free surface, the series solution satisfies the boundary conditions almost exactly.

In conclusion, an interpolation approach has been presented that improves the efficiency of the analytic series method by at least an order of magnitude, without significantly degrading the accuracy. Note that all the essential features of the analytic series solution are maintained. In particular, the series solution still satisfies Laplace's equation and some of the boundary conditions exactly, and thus exact maximum error bounds are still immediately available. Although arguably the interpolation approach is accurate enough, the iterative scheme can be slightly modified to produce a solution that is as accurate as those obtained using the full least squares implementation. All that needs to be done is to iterate to convergence using the interpolation approach, and then perform an extra itera-

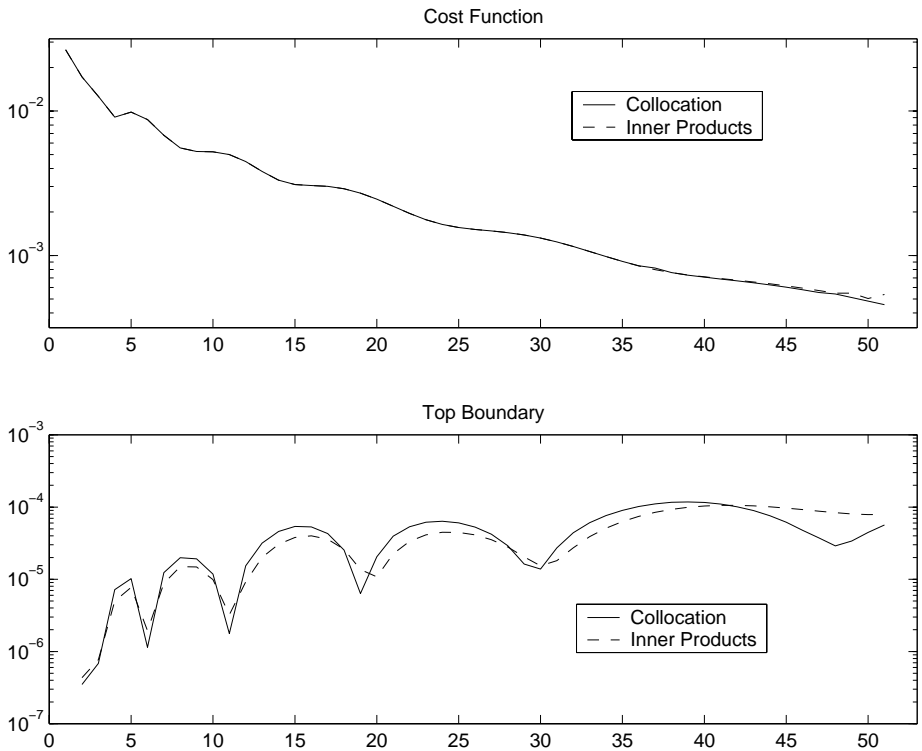


FIGURE 4: rms errors during 50 iterations on a free boundary problem

tion or two using the full least squares method.

References

- [1] S. R. Belward and L. K. Forbes, Fully nonlinear two-layer flow over arbitrary topography, *J. Eng. Math.* **27** 419–432, 1993. C646, C648
- [2] S. R. Belward, W. W. Read and P. J. Higgins, Efficient Series Solutions for Non-Linear Flow Over Topography, *ANZIAM J.*, 44(E), 2003. [Online]
<http://anziamj.austms.org.au/V44/CTAC2001/Belw>.
C648, C650
- [3] A. G. Gill, W. W. Read and G. E. Sneddon, On Series Solutions to Laplacian Free Boundary Problems, *Computational techniques and applications: CTAC95* (eds. R. L. May and A. K. Easton), (World Scientific, Singapore) 343–350, 1996. C646
- [4] A. Gill and W. W. Read, Efficient analytic series solutions for two-dimensional potential flow problems, *International Journal for Numerical Methods in Fluids* **23** (5) 415–430, 1996. C647, C654
- [5] J. A. Liggett and P. L-F. Liu. The Boundary Integral Equation Method for Porous Media Flow. Allen & Unwin, London, 1983. C646
- [6] W. W. Read and R. E. Volker, Series solutions for steady seepage through hillsides with arbitrary flow boundaries, *Water Resources Research*, **29** (8) 2871–2880, 1993. C646, C647

- [7] W. W. Read, S. R. Belward and P. J. Higgins, Iterative schemes for series solutions to Laplacian free boundary problems, *Computational techniques and applications: CTAC95* (eds. R. L. May and A. K. Easton), (World Scientific, Singapore) 669 – 676, 1996. C646, C650
- [8] W. W. Read, Hillside Seepage and the Steady Water Table I: Theory, *Advances in Water Resources* **19** (2) 63–73, 1996. C646
- [9] W. W. Read, Hillside Seepage and the Steady Water Table II: Applications, *Advances in Water Resources* **19** (2) 75–81, 1996. C646