Available online at www.sciencedirect.com
SCIENCE DIRECT®
Journal of Computational Physics 200 (2004) 1–7
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

Short Note

Computation of the material indicator function near the contact line (in Tryggvason's method)

M. Khenner

Department of Mathematics, State University of New York at Buffalo, Buffalo, NY 14260, USA

Received 17 November 2003; received in revised form 29 March 2004; accepted 29 March 2004
Available online 30 April 2004

Abstract

This note describes a simple and robust procedure for the accurate computation of values of the indicator function along and near the solid material boundary that is in contact with the evolving interface, as is common in crystal growth on a substrate (solidification and vapor/liquid-phase epitaxy).

1. Introduction

The front-tracking method developed by Unverdi and Tryggvason [1] is widely used for modeling multiphase fluid flows, boiling and atomization phenomena, and solidification (see the review paper [2] and references therein for modifications and improvements of the original method, as well as for the discussion of applications). "A stationary rectangular grid is used for the fluid flow, but the interface is tracked by a separate grid of lower dimension. However, unlike front-tracking methods, where each phase is treated separately, here all the phases are treated together by solving a single set of governing equations for the whole flow field" [2]. For the problems that admit reformulation in terms of a single set of equations, the method is very robust and accurate even in 3D. Recent state-of-the-art implementations can handle breaking and merging interfaces.

To distinguish between phases, the method makes use of the material indicator function $\rho(\mathbf{x})$, that takes on prescribed values separated by unity in the bulk of phases 1 and 2; for instance, 0 and 1, respectively. The construction procedure for $\rho(\mathbf{x})$ is described in [1]. We repeat this description here for the sake of completeness. For example, one has to solve the Poisson equation for $\rho(\mathbf{x})$. The RHS of this equation is $\nabla \cdot \mathbf{G}$, where

$$\mathbf{G}(\mathbf{x}) = \Sigma_\ell \mathbf{D}(\mathbf{x} - \mathbf{x}^{(\ell)}) n^{(\ell)} \Delta s^{(\ell)}. \tag{1}$$

E-mail address: mkhenner@nsm.buffalo.edu (M. Khenner).

0021-9991/$ - see front matter
doi:10.1016/j.jcp.2004.03.012

In (1), **D** is a "distribution function" that determines what fraction of some quantity defined on the interface should be assigned to each point in the rectangular grid should one attempt to "distribute" the interfacial quantity to the rectangular grid (the procedure inverse to interpolation at the interface). Also, $\mathbf{n}^{(\ell)}$ is the unit normal vector to an interface element of area $\Delta s^{(\ell)}$ (length in 2D) whose centroid is at $\mathbf{x}^{(l)}$. By construction, $\rho(\mathbf{x})$ changes smoothly between bulk values in the interfacial region (2–3 grid points at each side of the interface), and thus $\rho(\mathbf{x})$ is the smoothed Heaviside function, see [1,2] for more details. The indicator function is constructed from the *known* position of the interface, and it is not used to locate the interface; an interface markers keep track of interface location and shape. $\rho(\mathbf{x})$ is used only to smooth the otherwise discontinuous material properties (such as the fluid density and viscosity, for example) across the interface. The smoothed material properties, in turn, allow for the stable computation of the interface evolution. For the solution of the Poisson equation, the boundary conditions for $\rho(\mathbf{x})$ must be prescribed on the boundaries of the computational domain (e.g., rectangle in 2D or cube in 3D). The indicator function is always computed anew every time step of marching the interface through the computational domain.

The method of [1] is often applied to compute the evolution of closed interfaces not in contact with material boundaries. We are aware of only one application of the method to open interfaces: recently, Zhang et al. [3] studied the drop spreading on the planar substrate. However, the details of the computation near the contact point are not presented in that work. In [4], we numerically studied the 2D electro-epitaxial crystal growth on a patterned masked substrate by using this method. The crystal–melt interface in this problem is an open curve that touches the planar mask by one of its ends (Figs. 1 and 2). As the crystal grows, this contact point moves unidirectionally along the mask. The contact angle θ that the crystal–melt interlace makes with the mask is determined by the condition of local equilibrium at the junction of the crystal–mask, crystal–melt and mask–melt interfaces (the contact angle is measured from within the crystal). That condition is famous Laplace–Young condition (see [5], for example). As a rule, surface energy is constant along each interface; it then follows that θ is a constant parameter. The next section describes difficulties we faced while trying to ensure accurate, stable computation, and the procedure that finally allowed for such computation.

Similar approach was used in [6] for the study of grain-boundary grooving by level set method [7]. The motivation for this note is to present the details and performance of the approach since it also proved useful

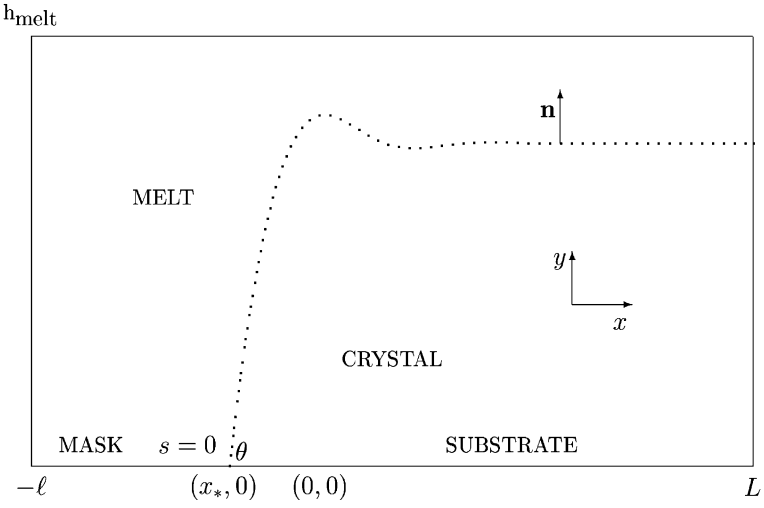

Fig. 1. A sketch of the mathematical situation. The crystal–melt interface (curve in two dimensions) is sketched such that the crystal overgrowth onto the mask is shown. The interface is parametrized by arc length, s. The direction of DC current through the open substrate, masked region of the substrate, crystal and melt is upward. $(x_*, 0)$ is the contact point.

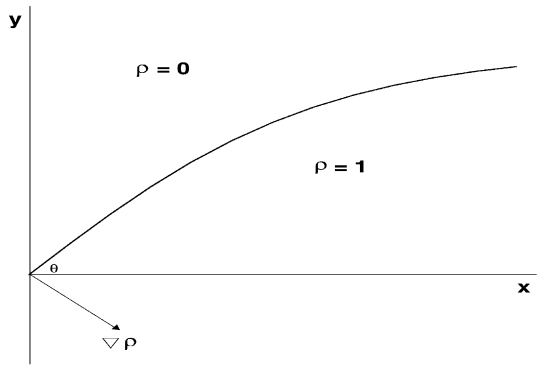

Fig. 2. Schematic view of the interface near tri-junction, at $\theta < 90°$.

in the context of another method. The usefulness of diffuse interface methods for the numerical solution of problems with moving contact lines, and their ability to reproduce the contact line physics is demonstrated in [8]. The contact line spreading when two liquids and a substrate are involved (as in [3,8], for example) is a more difficult problem than the one under study in this note and in [4].

2. Procedure

Every time step, the boundary condition for $\rho(\mathbf{x})$ must be accurately prescribed at those points in 2D grid that lie on the line segment $y = 0$, $x \in [-\ell, L] \equiv I$ (the lower boundary of the computational box). Let phase 1 is the melt, phase 2 is the crystal. The simplest method one could think about is just to set $\rho = 0$ at all grid points to the left of the contact point $(x_*, 0)$, $x_* \in I$ (in the melt), and $\rho = 1$ at all grid points to the right of $(x_*, 0)$ (in the solid). Not surprisingly, this boundary condition leads to the instability of the contact point motion and of the crystal–melt interface in the vicinity of the contact point (Fig. 5). The obvious reason for the instability is the prescribed abrupt change in ρ across the contact point and the resulting abrupt jump in the material parameters (in our case, electrical conductivities of the melt and the crystal, and also electrical mobilities and diffusivities of the solute in both phases). Besides, the available information about the interface slope was not accounted for when ρ was set constant at both sides of the contact point. The following simple method was established that provides stable long-time computation.

Let the grid points along $y = 0$ boundary of the computational box are indexed by i, $i = 1, \ldots, N$, and let $j = 1, \ldots, N$ indexes grid points in y-direction (with $j = 1$ corresponding to $y = 0$). Let at a given time level l the first grid point to the left of x_* is $i = k$. At $l = 0$, we set $\rho = 0$ at the grid points $1, \ldots, k-3$ and $\rho = 1$ at the grid points $k+4, \ldots, N$, but $\rho = 0.125, 0.25, 0.375$ at the grid points $k-2, k-1, k$, respectively, and $\rho = 0.625, 0.75, 0.875$ at the grid points $k+1, k+2, k+3$, respectively. Thus, we artificially smear out the boundary condition for the indicator function at $l = 0$. [1] Denote this approximation $\rho_{i,0}^{(l)(0)}$ Given this boundary condition, the boundary conditions on the rest three boundaries of the computational box and the initial interface, the guess to the indicator function at $t = 0$ is then computed everywhere on 2D grid. Denote this guess as $\rho_{i,j}^{(l)(\mathrm{ini})}$. We next show how to correct $\rho_{i,j}^{(l)(\mathrm{ini})}$.

[1] Of course, smearing can be done more accurately; we found that crude smearing as described is sufficient, and we need to employ it only once in the beginning of the computation (see below).

From the geometry (Fig. 2), the x- and y-components of $\nabla\rho$ at the contact point are

$$\frac{\partial\rho}{\partial x} = |\nabla\rho|\sin\theta, \tag{2}$$

$$\frac{\partial\rho}{\partial y} = -|\nabla\rho|\cos\theta \tag{3}$$

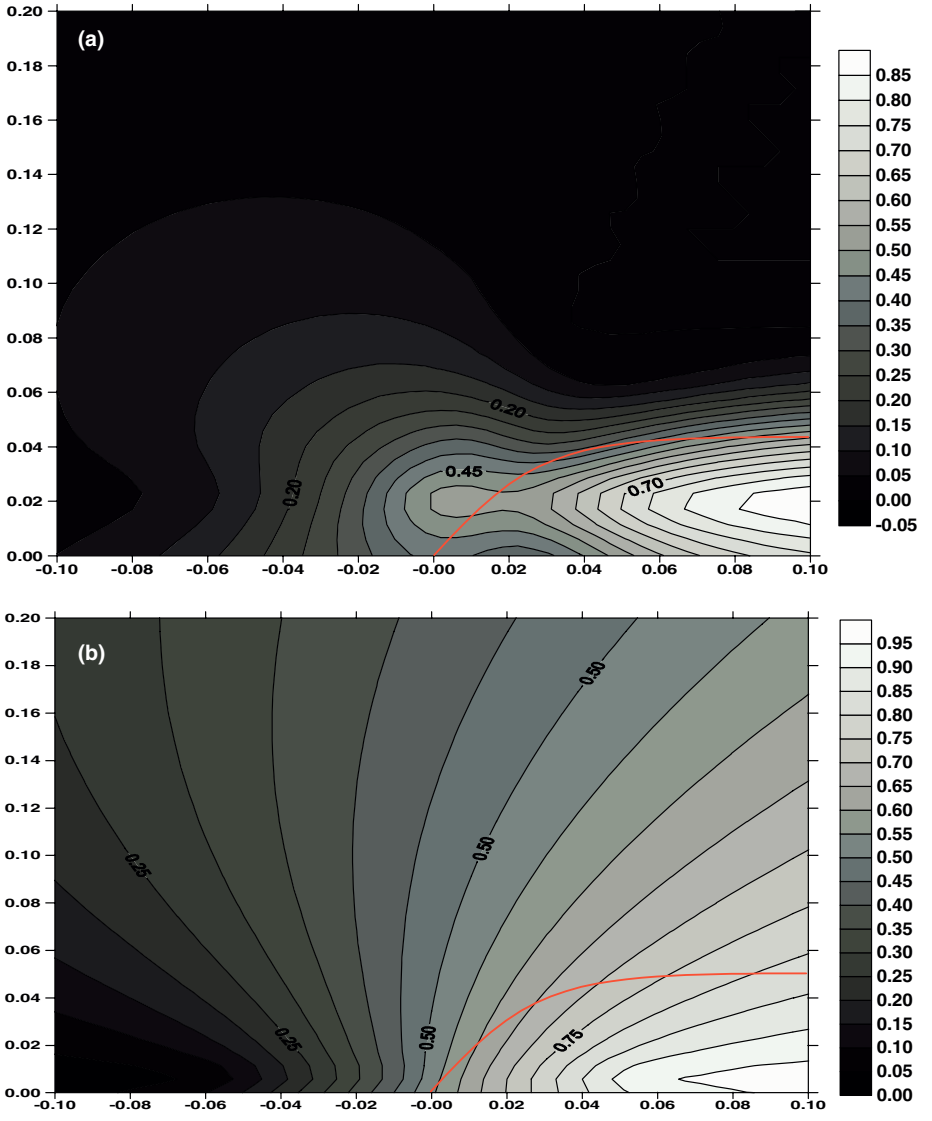

Fig. 3. Contour plots of $\rho(\mathbf{x})$ at $t = 0$, near the tri-junction. Interface is shown in red. The computational domain is $[-0.4 : 1] \times [0 : 4]$. The interfacial grid consists of 250 segments. In (a), the 2D grid is 192×192. In (b), the 2D grid is 192×576. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

if the interface makes angle $\theta < 90^\circ$ with the x-axis. From (2) and (3) we obtain

$$\frac{\partial \rho}{\partial x} = -\frac{\partial \rho}{\partial y} \tan\theta. \tag{4}$$

Recall that the interface is the 0.5 level curve of $\rho(\mathbf{x})$ and require Eq. (4) to hold for other level curves of $\rho(\mathbf{x})$. Thus we require for other level curves of $\rho(\mathbf{x})$ to intersect $y = 0$ boundary also at angle θ. We can then discretize Eq. (4) with central finite difference in x and one-sided finite difference in y:

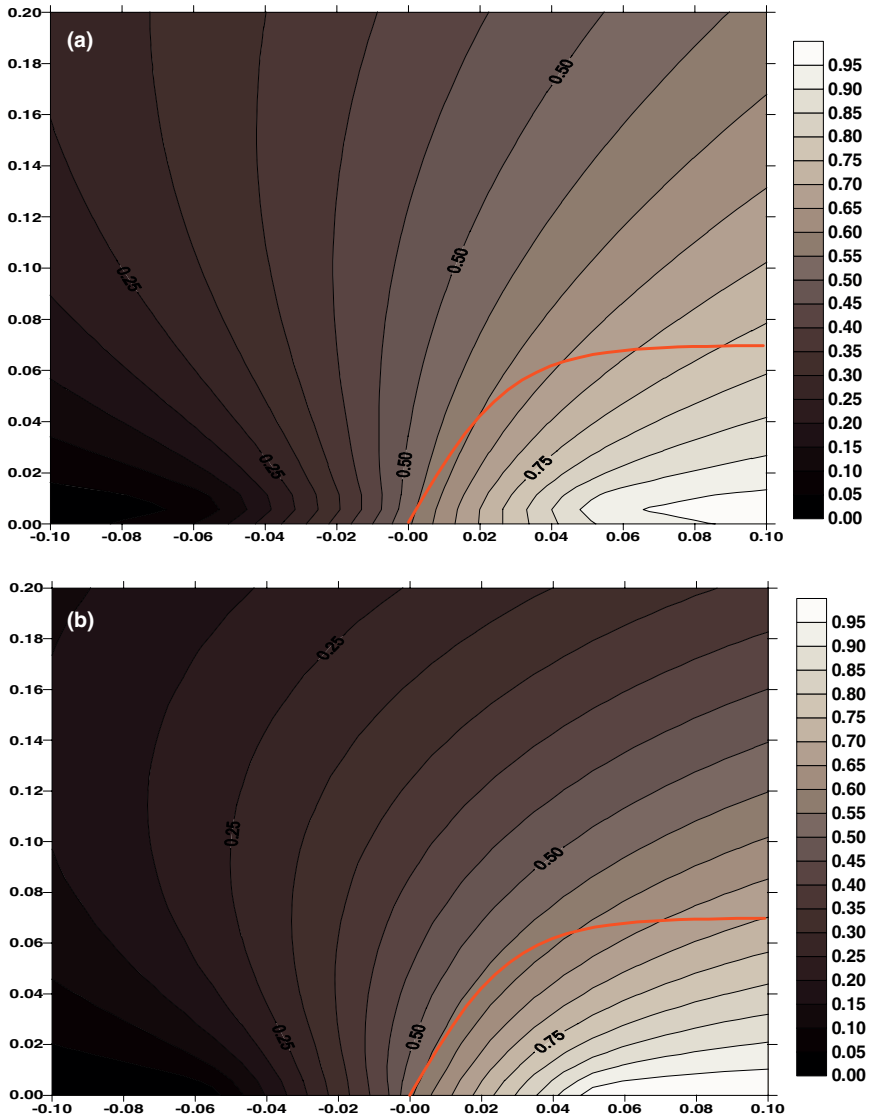

Fig. 4. Contour plots of $\rho(\mathbf{x})$ at $t = 0$, near the tri-junction. The computational domain is $[-0.4 : 1] \times [0 : 4]$ in (a), but $[-0.4 : 1] \times [0 : 0.4]$ in (b). The interfacial grid consists of 500 segments. The 2D grid is 192×576.

$$-\frac{1}{2\Delta x}\rho_{i-1,0}-\frac{3\tan\theta}{2\Delta y}\rho_{i,0}+\frac{1}{2\Delta x}\rho_{i+1,0}=\frac{-4\rho_{i,1}+\rho_{i,2}}{2\Delta y}\tan\theta,\quad i=2,\ldots,N-1. \tag{5}$$

Eq. (5) is a tridiagonal system of algebraic linear equations for unknowns $\rho_{i,0}$. This system can be easily and accurately solved numerically provided right-hand sides of the equations are known. That is, corrected values $\rho_{i,0}^{(l)}$ of the indicator function along x-axis are found from

$$-\frac{1}{2\Delta x}\rho_{i-1,0}^{(l)}-\frac{3\tan\theta}{2\Delta y}\rho_{i,0}^{(l)}+\frac{1}{2\Delta x}\rho_{i+1,0}^{(l)}=\frac{-4\rho_{i,1}^{(l)(ini)}+\rho_{i,2}^{(l)(ini)}}{2\Delta y}\tan\theta,\quad i=2,\ldots,N-1. \tag{6}$$

Eq. (6) can be applied iteratively 2–3 times if more accuracy is needed. The last step is to assign the just computed values of $\rho_{i,0}^{(l)}$ to $\rho_{i,0}^{(l+1)(0)}$ and proceed to the next time level, where the procedure is repeated for the updated location of the interface. Since in our application (as well as in the majority of other) the interface moves less than a distance between two adjacent grid points between the times t and $t+\Delta t$, the usage of $\rho_{i,0}^{(l)}$ as initial approximation on the next time level is justified, and the repetition of the crude smearing is avoided.

At the left boundary of the computational box and at the top boundary we prescribe $\rho=0$; at the right boundary the interface is flat, and $\partial\rho/\partial x=0$ due to symmetry. Also note that simple changes in formulas (2) and (3) are necessary in the case $\theta>90°$.

Figs. 3 and 4 show the contour plots of the indicator function computed with the described procedure (Eq. (6) is applied once per time step). In Fig. 3(a), the resolution of the rectangular grid is insufficient to produce reasonable $\rho(\mathbf{x})$ near the tri-junction. In Fig. 3(b), the contour curves adjacent to the interface are smooth and intersect the x-axis at the required contact angle (60°). Fig. 4(a) shows that denser interfacial grids do not improve the computation near the material boundary. Finally, Fig. 4(b) shows that decreasing the vertical size of the domain ten-fold while keeping constant the number of rectangular grid points along the vertical (and hence, effectively increasing vertical resolution ten-fold) allows to achieve smoothness in the far field to the right, but at the same time contour curves to the left of the interface fail to intersect the x-axis at 60°. Hence, for the computation to be successful the domain size and the rectangular grid must be carefully chosen.

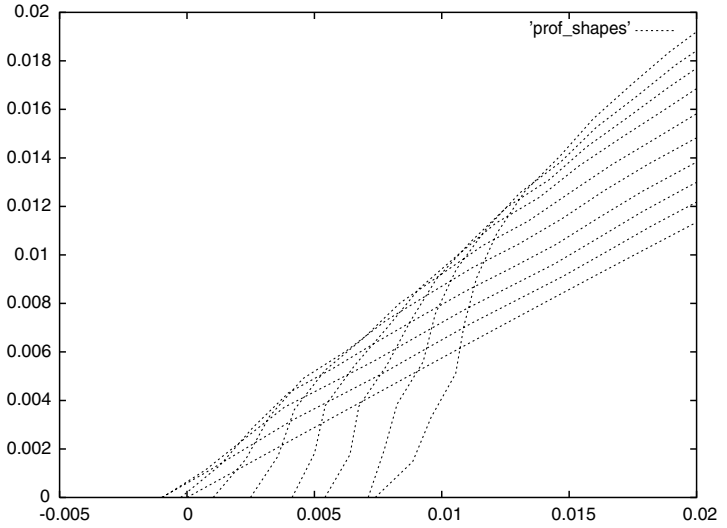

Fig. 5. Interface profiles near tri-junction ($0\leqslant t\leqslant 0.1$). On the x-axis, value of indicator function is permanently set to 1 and 0 to the right and to the left of the contact point, respectively. Not only the contact angle in not conserved in this computation, but after slight initial overgrowth onto the mask the crystal retracts on the substrate and the interface quickly becomes unstable.

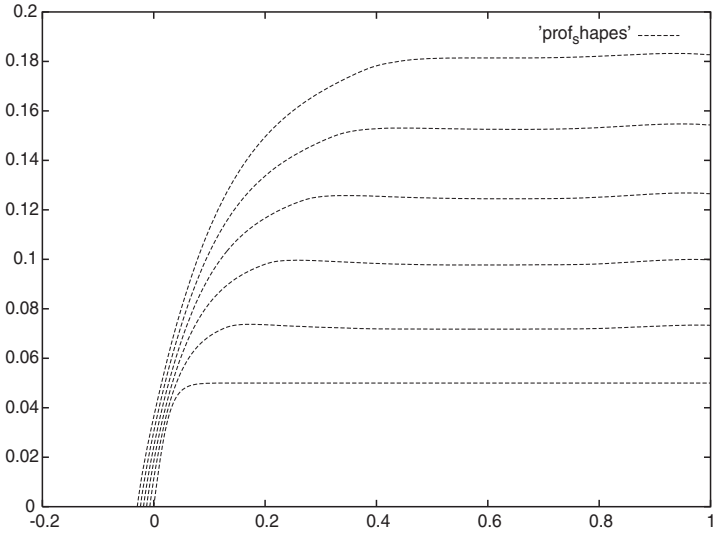

Fig. 6. Longtime, isotropic interface evolution with full procedure in place ($0 < t < 10^3$). The electric current is turned off in this computation.

The procedure works for any fixed value of θ when the above requirement is met; besides, θ may change while the interface evolves as required by underlying physics.

Figs. 5 and 6 show the interface shapes computed without and with the suggested procedure, respectively.

For the solution of Poisson equation we use the HWSCRT code (cyclic reduction method) from FISHPACK software library [9].

The procedure can be easily extended to 3D.

Acknowledgements

Author thanks Prof. M.J. Miksis for helpful discussion.

References

[1] S.O. Unverdi, G. Tryggvason, J. Comput. Phys. 100 (1992) 25.
[2] G. Tryggvason, B. Bunner, A. Esmaeeli, D. Juric, N. Al-Rawahi, W. Tauber, J. Han, S. Nas, Y.-J. Jan, J. Comput. Phys. 169 (2) (2001) 708.
[3] J. Zhang, M.J. Miksis, S.G. Bankoff, Motion of a viscous drop with a moving contact line, (2002) submitted.
[4] M. Khenner, R.J. Braun, Numerical simulation of liquid phase electro-epitaxial selective area growth, in preparation.
[5] J.N. Israelachvili, Intermolecular and Surface Forces, second ed., Academic Press, London, San Diego 1991, c1992.
[6] M. Khenner, A. Averbuch, M. Israeli, M. Nathan, E. Glickman, Comp. Mater. Sci. 20 (2001) 235.
[7] S. Osher, J.A. Sethian, J. Comput. Phys. 79 (1988) 12.
[8] D. Jacqmin, J. Fluid Mech. 402 (2000) 57.
[9] P. Swarztrauber, R. Sweet, Efficient Fortran subprograms for the solution of elliptic equations, NCAR TN/IA-109, July 1975.