CWI Syllabi

Centrum voor Wiskunde en Informatica
Centre for Mathematics and Computer Science
P.O. Box 4079, 1009 AB Amsterdam, The Netherlands

The CWI is a research institute of the Stichting Mathematisch Centrum, which was founded on February 11, 1946, as a nonprofit institution aiming at the promotion of mathematics, computer science, and their applications. It is sponsored by the Dutch Government through the Netherlands Organization for the Advancement of Pure Research (Z.W.O.).

CWI Syllabus 5

Colloquium topics in applied numerical analysis

Volume 2

J.G. Verwer (ed.)

Centrum voor Wiskunde en Informatica
Centre for Mathematics and Computer Science

1980 Mathematics Subject Classification: 65XX06
ISBN 90 6196 282 X

PREFACE

The colloquium "Topics in Applied Numerical Analysis" was held at the Department of Numerical Mathematics of the Centre for Mathematics and Computer Science during the academic year 1983/1984. The aim of this colloquium was to draw attention to the widespread use of numerical mathematics in scientific real life problems, as well as to foster co-operation between mathematicians working in an academic environment and representatives from industries and institutes where the numerical solution of real life problems is studied.

These two volumes contain, in complete form, the papers presented by the speakers among the participants. Also it contains a contribution by J.L.O. Vranckx who, at the last minute, was unable to attend the colloquium in person.

The greater part of the papers deal with the numerical solution of a certain mathematical problem from practice. It was very interesting for the participating mathematicians to attend the lectures of the practitioners and to see the wide range of difficult technical problems which arise in, e.g., the engineering sciences.

The success of the colloquium was due principally to the speakers. To all of them I extend my sincere thanks and appreciation.

October 1984

J.G. Verwer (ed.)

CONTENTS VOLUME 1

CONTENTS VOLUME 2

NAMES AND ADDRESSES OF THE AUTHORS

H. Akse
Landbouwhogeschool Wageningen, Sectie Proceskunde,
De Dreijen 12, 6703 BC Wageningen.

G.L.M. Augenbroe
Technische Hogeschool Delft, Afd. Civiele Techniek,
Vakgroep Bouwfysica, Stevinweg 1, 2628 CN Delft.

A.O.H. Axelsson
Katholieke Universiteit Nijmegen, Mathematisch Instituut,
Toernooiveld, 6525 ED Nijmegen.

M. Bakker
Centrum voor Wiskunde en Informatica,
Kruislaan 413, 1098 SJ Amsterdam.

P.M. van den Berg
Technische Hogeschool Delft, Afd. Elektrotechniek,
Postbus 5031, 2600 GA Delft.

J.J. Bisschop
Shell Research B.V.,
Badhuisweg 3, 1031 CM Amsterdam.

J.W. Boerstoel
Nationaal Lucht- en Ruimtevaartlaboratorium,
Postbus 90502, 1006 BM Amsterdam.

B.J. Braams
FOM-instituut voor Plasma-Fysica,
Postbus 1207, 3430 BE Nieuwegein.

R. de Bruin
Rekencentrum der Rijksuniversiteit Groningen,
Postbus 80, 9700 AV Groningen.

J.G.M. de Bruin
Neratoom B.V.,
Postbus 93244, 2509 AE 's-Gravenhage.

A. van Calster
Universiteit van Gent, Laboratorium voor Electronica,
Sint Pietersnieuwstraat 41, 9000 Gent, België.

C. Flokstra
Waterloopkundig Laboratorium "De Voorst",
Voorsterweg 28, 8316 PT Marknesse.

R.H.J. Gmelig Meyling
Universiteit van Amsterdam,
Instituut voor Toepassingen der Wiskunde,
Roetersstraat 15, 1018 WB Amsterdam.

J.P. Hollenberg
Rekencentrum der Rijksuniversiteit Groningen,
Postbus 800, 9700 AV Groningen.

F.J. Jacobs
Koninklijke Shell Exploratie & Productie Laboratorium,
Postbus 60, 2280 AB Rijswijk.

A. Kassies
Nationaal Lucht- en Ruimtevaartlaboratorium.
Postbus 90502, 1006 BM Amsterdam.

A. Kooistra
Instituut TNO voor Wiskunde, Informatieverwerking en Statistiek,
Postbus 297, 2501 DD 's-Gravenhage.

I. Kuijper
Neratoom B.V.,
Postbus 93244, 2509 AE 's-Gravenhage.

C.G. van der Laan
Rekencentrum der Rijksuniversiteit Groningen,
Postbus 800, 9700 AV Groningen.

D. Loret
Universiteit van Gent, Laboratorium voor Electronica,
Sint Pietersnieuwstraat 41, 9000 Gent, België.

G. de Mey
Universiteit van Gent, Laboratorium voor Electronica,
Sint Pietersnieuwstraat 41, 9000 Gent, België.

S.J. Polak
Nederlandse Philips Bedrijven B.V., ISA-ISC-TIS/CARD,
Gebouw SAQ 2, 5600 MD Eindhoven.

N. Praagman
Svasek B.V.,
Heer Bokelweg 146, 3032 AD Rotterdam.

W.H.A. Schilders
Nederlandse Philips Bedrijven B.V., ISA-ISC-TIS/CARD,
Gebouw SAQ 2, 5600 MD Eindhoven.

G.S. Stelling
Dienst Informatie Verwerking, Rijkswaterstaat,
Nijverheidsstraat 1, 2288 BB Rijswijk.

A.G. Tijhuis
Technische Hogeschool Delft, Afdeling der Elektrotechniek,
Postbus 5031, 2600 GA Delft.

G.K. Verboom
Waterloopkundig Laboratorium,
Rotterdamseweg 185, 2600 MH Delft.

J. Vranckx
Agfa-Gevaert, N.V., Mathematisch Centrum 3523,
Septestraat 27, B-2510 Mortsel, België.

A. Wachters
ISA-ISC-TIS/CARD, Nederlandse Bedrijven B.V.,
Gebouw SAQ 2, 5600 MD Eindhoven.

A.J. van der Wees
Nationaal Lucht-en Ruimtevaart Laboratorium,
Postbus 153, 8300 AD Emmeloord

J.B.T.M. Willemse
Dienst Informatieverwerking, Rijkswaterstaat,
Nijverheidsstraat 1, 2288 BB Rijswijk.

W. Zijl
Dienst Grondwaterverkenning TNO,
Postbus 285, 2600 AG Delft.

IMPROVEMENT OF THE ACCURACY OF NUMERICAL SIMULATORS FOR FLOW THROUGH OIL/GAS RESERVOIRS

F.J. JACOBS

Contents

1. INTRODUCTION

Enhanced oil recovery (EOR) processes [1], nowadays being developed to produce the bypassed fifty per cent of proven reserve which is left after natural depletion and/or water injection, have induced KSEPL to reconsider the accuracy of their numerical reservoir simulators.
As in EOR the number of components in the mass balance system is greatly increased in comparison with the classical black oil description, which obscures the interpretation of numerically diffused composition profiles in terms of waves (penetrating water tongues) and shocks (oil/gas banks), and as at the same time the occurrence of shocks depends more critically on the system parameters, simulation of enhanced oil recovery requires also enhanced accuracy.
It is doubtful whether the simple standard discretisation of SHELL's simulators (fixed grid during the entire simulation, one-step in time, 5-point (2D) in space for the pressure equation and '5'-point (2D) in space

(i.e. one-point upstream in each grid direction) for the transport part) is able to satisfy the demands by means of brute force even with the help of vectorisation.

For the model of isothermal multicomponent multiphase flow such as it forms the basis of the simulator MULTISIM, the concepts of two alternatives for raising the accuracy are presented. The first method is an implementation of flux correction, or rather of correcting the flux evaluation point, which has led to FCMULTISIM, a variant of incompressible MULTISIM that contains an improved discretisation for the transport part, but does not involve any further modifications of the standard algorithm. The second method implements the idea of adaptive local grid refinement. So far, it has been worked out only for the special case of incompressible immiscible two-phase flow (oil + water). However, the fundamental problem of constructing an efficient multigrid solver for the pressure equation has been overcome. In addition, preliminary versions of local time stepping in the transport part and of adaptation criteria have been developed.

2. FLOW MODEL

Isothermal multicomponent multiphase flow through porous media.

Concepts

ϕ, porosity, fluid volume in unit volume of rock

m_i $(i=1...I)$, $\Sigma_i\, m_i = 1$, mass fraction of component i in unit mass of fluid

S_j $(j=1...J)$, $\Sigma_j S_j = 1$, saturation, volume fraction of phase j in unit volume of fluid

$r_{i,j}$, $\Sigma_i\, r_{i,j} = 1$, relative mass fraction of component i in unit mass of phase j

ρ_j, density of phase j

$\bar{\rho} = \Sigma_j\, \rho_j S_j$, mean density

μ_j, viscosity of phase j

p_j, pressure of phase j

p, reference pressure

$pc_j = p_j - p$, capillary pressure

$\underline{v}_j$, volume flow rate of phase j (vector)

$\underline{\underline{ka}}$, absolute permeability (tensor)

kr_j, relative permeability of phase j

$\lambda_j = kr_j/\mu_j$, mobility of phase j

$\lambda_T = \Sigma_j\, \lambda_j$, total mobility

$\underline{g}$, gravity

Assumptions

Darcy's law:

$$\underline{v}_j = -\underline{\underline{ka}}\,\lambda_j\,(\text{grad}\,p_j - \rho_j\underline{g}).$$

Fluid-rock properties:

$$kr_j = kr_j(\underline{S})$$
$$pc_j = pc_j(\underline{S}).$$

Rock properties:

$$\phi = \phi(x,p)$$
$$\underline{\underline{ka}} = \underline{\underline{ka}}(x,p)$$

Fluid properties (phase equilibrium):

$$p,\underline{m}(\Sigma_i m_i = 1) \to r_{i,j},\ S_j,\ \rho_j,\ \mu_j \text{ for all } i \text{ and } j.$$

Mass balance equations

$$\partial/\partial t[\phi\bar{\rho}m_i] = -\text{div}[\Sigma_j\ \rho_j r_{i,j}\underline{v}_j] + q_i,\quad i = 1,\ldots,I$$

$$q_i = \Sigma_k\ q_{i,k}\delta(x-x_k),\ \text{source function}$$

$$q_{i,k} = \begin{cases} q_{i,k}(t), & \text{rate constrained well} \\ q_{i,k}(\underline{S},p-p_{well}(t)), & \text{pressure constrained well} \end{cases}$$

No flow boundary conditions

$$(\underline{v}_j,\underline{n}) = 0,\quad j = 1,\ldots,J.$$

Expressed in the I unknowns $p,\hat{\underline{m}}(=\underline{m}-\text{one component})$ the mass balance equations define a system of the following form.

Compressible (general case):

$$\partial/\partial t\ f_i(p,\hat{\underline{m}}) = \text{div}[\Sigma_j\ g_{ij}(p,\hat{\underline{m}})\{\text{grad}\,p + \text{grad}\,pc_j(p,\hat{\underline{m}}) - \rho_j(p,\hat{\underline{m}})\underline{g}\}] + q_i(p,\hat{\underline{m}}). \tag{1}$$

Incompressible (rock and fluid properties independent of p, in particular constant component density ρ_i and $1/\bar{\rho} = \Sigma_i\ m_i/\rho_i$):

(2) $$\partial/\partial t\, f_i(\hat{\underline{m}}) = \mathrm{div}[\Sigma_j\, g_{ij}(\hat{\underline{m}})\{\mathrm{grad}\, p + \mathrm{grad}\, pc_j(\hat{\underline{m}}) - \rho_j(\hat{\underline{m}})\underline{g}\}] + q_i(p,\hat{\underline{m}}).$$

Incompressible, immiscible (the phase equilibrium computation reduces to $r_{i,j} = \delta_{ij}$, $\hat{\underline{S}}$(=$\underline{S}$-one phase) is chosen as unknown, the i-th equation is divided by ρ_i):

(3) $$\partial/\partial t[\phi S_i] = \mathrm{div}[\underline{\underline{ka}}\, \lambda_i(\hat{\underline{S}})\{\mathrm{grad}\, p + \mathrm{grad}\, pc_i(\hat{\underline{S}}) - \rho_i\underline{g}\}] + q_i(p,\hat{\underline{S}})/\rho_i.$$

3. CHARACTERISTIC FEATURES

3a. Pressure equation

To elucidate the rôle of p in the system, we observe that $\{\partial f_i/\partial \hat{m}_j\}$ as a I × (I-1) matrix admits α_i such that $\Sigma_i\, \alpha_i\, \partial f_i/\partial \hat{m}_j = 0$. Therefore, multiplication of the i-th equation with α_i and summation results in the pressure equation:

(4) $$\Sigma_i\alpha_i\partial f_i/\partial t = [\Sigma_i\alpha_i\partial f_i/\partial p]\partial p/\partial t = \Sigma_i\alpha_i\, \mathrm{div}[\Sigma_j g_{ij}\{\mathrm{grad}\, p + \mathrm{grad}\, pc_j - \rho_j\underline{g}\}] + \Sigma_i\alpha_i q_i.$$

The coefficient $\Sigma_i\alpha_i\partial f_i/\partial p = [\Sigma_i\alpha_i m_i]\partial(\phi\bar{\rho})/\partial p$ will be recognized as a kind of compressibility modulus. It vanishes in the incompressible case. Then the α_i are given by the constants $1/\rho_i$ and under the assumption of perfect mixing ($1/\rho_j = \Sigma_i r_{i,j}/\rho_i$) the pressure equation for this case reduces to

(5) $$0 = \mathrm{div}[\underline{\underline{ka}}\Sigma_i\lambda_i(\underline{S}(\hat{\underline{m}}))\{\mathrm{grad}\, p + \mathrm{grad}\, pc_i - \rho_i\underline{g}\}] + \Sigma_i q_i/\rho_i.$$

The same equation follows in the immiscible situation directly from summation of (3).

Eq. (4) expresses that p follows a parabolic 'diffusion' equation (highly non-linear via its coefficients), which degenerates into an elliptic equation (linear in p with the usual assumption that pressure constrained wells are

linear in p) when the compressiblity decreases. From this we conclude that
(i) a discretisation of (1) must proceed implicitly in p,
(ii) unique existence of a p-solution is always guaranteed, because, even in the critical case of incompressibility, the presence of at least one pressure constrained well (a production well) at some x_k, which satisfies the normal condition $\partial q_{i,k}/\partial p < 0$ for all i, will keep the (b.c. + differential) operator for p positive.

3b. Transport part

Assuming that p and grad p have been solved by means of (4) or (5), we may consider the remainder of (1) as a non-linear first order hyperbolic transport system in divergence form for $\hat{\underline{m}}$, except for the presence of a generalised diffusion introduced by pc. The nature of the diffusion becomes very obvious from the incompressible immiscible case. If we introduce the total velocity $\underline{v}_T = \Sigma_j \underline{v}_j$, which follows from the solution of (5), and express grad p in terms of $\underline{v}_T$ and $\underline{S}$, we transform (3) into

$$(6)\qquad \partial/\partial t[\phi S_i] = -\mathrm{div}[\underline{v}_T \lambda_i/\lambda_T] - \mathrm{div}[\underline{\underline{k}} a \Sigma_j \lambda_i \lambda_j (\rho_i - \rho_j)\underline{g}/\lambda_T] + \mathrm{div}[\underline{\underline{k}} a \Sigma_j \lambda_i \lambda_j \mathrm{grad}(pc_i - pc_j)/\lambda_T] + q_i/\rho_i$$

(of which only I - 1 equations apply in combination with (5)). It appears that the diffusion coefficients are only non-zero in the transition zone between injection fluid and reservoir fluid, because the relative permeability for a specific phase is zero where that phase is not present.
From the above remarks we conclude that:
(i) if $pc \neq 0$, a discretisation of (1) must proceed implicitly in $\hat{\underline{m}}$,
(ii) if $pc = 0$, shock phenomena must be expected, which are best followed with an explicit discretisation (if high accuracy is required, the CFL condition must be satisfied anyway and implicit methods present only a computational burden).
We will restrict our further attention to the case $pc = 0$. Of course, the explicit discretisation must be 'upstream' in some sense and, as we are dealing with a non-linear system in multidimensional space, this in turn requires some form of flux splitting that decides which component combinations stream from which directions. In accordance with Darcy's law,

we split the system in its phases propagating along $\underline{v}_j$.

Remark. If all the ground characteristics of the system would coincide (as in 1D always, or in horizontal 2D if $\underline{g}$ is absent and the ground characteristics are all equal to $\underline{v}_T$, see (5) and (6)), one could think of following the system along the unique ground characteristic with the superior Godunow version of the upstream method, which is based on solving local (1D) Riemann problems. Indeed, in 1D this approach differs from the phase splitting above, if gravity effects are important. However, with normally two fields ($\underline{v}_T$ and $\underline{g}$) present, a general multidimensional extension seems impossible.

3c. Conservation

To close the discussion of the flow model, we emphasize the importance of the discretisation scheme being conservative (i.e. following the component masses f_i without losses). Then on a coarse grid at least one essential property of (1), namely total component mass conservation, is reflected in the discrete approximation. This implies that, for the general compressible case, it does not make sense to rewrite (1) in the form of a pressure equation and a remaining independent transport part with the object of constructing separate discretisations for these two parts.

4. MULTISIM

4a. Standard discretisation

The requirements formulated in section 3 are all satisfied by the standard discretisation for $pc \simeq 0$ of SHELL's reservoir simulator MULTISIM, of which a 2D description follows.
It is a one-step method in time, which operates on a fixed block centered grid. Δx, Δy denote the mesh widths; (x_m, y_n) is the center of block (m,n); $(x_{m+1/2}, y_n)$ is the center of the interface between blocks (m,n) and (m+1,n); the grid axes are oriented in the principal directions of $\underline{\underline{k}}a$.
The formulae for the evolution $t^k \to t^{k+1} = t^k + \Delta t$ are:

$$(7) \qquad (f^{k+1}_{i,m,n} - f^{k}_{i,m,n})/\Delta t = (F^{k+\frac{1}{2}}_{i,m+\frac{1}{2},n} - F^{k+\frac{1}{2}}_{i,m-\frac{1}{2},n})/\Delta x + (F^{k+\frac{1}{2}}_{i,m,n+\frac{1}{2}} - F^{k+\frac{1}{2}}_{i,m,n-\frac{1}{2}})/\Delta y + q^{k+\frac{1}{2}}_{i,m,n}$$

(8) $$f_{i,m,n}^{k+1} = f_i(x_{m,n}, p_{m,n}^{k+1}, \underline{\hat{m}}_{m,n}^{k+1})$$

(9) $$q_{i,m,n}^{k+\frac{1}{2}} = q_i(x_{m,n}, p_{m,n}^{k+1}, \underline{\hat{m}}_{m,n}^{k}, t^{k+\frac{1}{2}})$$

(10) $$F_{i,m+\frac{1}{2},n}^{k+\frac{1}{2}} = -\Sigma_j(\rho_j r_{i,j})_{up_j(m+\frac{1}{2},n)}^{k} v_{x,j,m+\frac{1}{2},n}^{k+\frac{1}{2}}$$

(11) $$v_{x,j,m+\frac{1}{2},n}^{k+\frac{1}{2}} = -ka_{xx,m+\frac{1}{2},n}\,\lambda_{j,up_j(m+\frac{1}{2},n)} \times$$
$$[\{(p_{m+1,n}^{k+1} + pc_{j,m+1,n}^{k}) - (p_{m,n}^{k+1} + pc_{j,m,n}^{k})\}/\Delta x - (\rho_{j,m+1,n} + \rho_{j,m,n})g_x/2]$$

(12) $$ka_{xx,m+\frac{1}{2},n} = 2/(1/ka_{xx,m+1,n} + 1/ka_{xx,m,n}), ka_{xx,m,n} = ka_{xx}(x_{m,n}, p_{m,n}^{k})$$

(13) $$up_j(m+\tfrac{1}{2},n) = \begin{cases} (m,n), & \text{if } \operatorname{sign}(v_{x,j,m+\frac{1}{2},n}^{k+\frac{1}{2}}) > 0 \\ (m+1,n), & \text{if } \operatorname{sign}(v_{x,j,m+\frac{1}{2},n}^{k+\frac{1}{2}}) < 0 \end{cases}$$

To solve for $(p^{k+1}, \underline{\hat{m}}^{k+1})$ we use a Newton process based on linearisation of $f_i(p^{k+1}, \underline{\hat{m}}^{k+1})$ and of the pressure constrained wells, if necessary. Evaluation of the coefficients α_i defined in section 3a at the last iteration $(p^{k+1(\ell)}, \underline{\hat{m}}^{k+1(\ell)})$ and summation of all equations after multiplication with α_i results in a linear equation for $p^{k+1(\ell+1)}$ alone (a discretisation of the pressure equation (4)). Its solution is substituted in the right hand member of (7) for the computation of new values $f_i^{k+1(\ell+1)}$, which are inverted to $\underline{\hat{m}}^{k+1(\ell+1)}$ by means of $m_i = f_i/\Sigma_i f_i$. In the case of incompressibility when the α_i are constant, the iteration reduces to one step.

4b. Criticism and possibilities for improvement

i) The weakest point of the method is the computation of $up_j(m+1/2,n)$ in (13) as either (m,n) or (m+1,n). This one-point upstream feature introduces numerical diffusion. An improvement would be given by taking into account the actual distance, which is covered by phase j in the x-direction during time $\Delta t/2$. Evaluation of the flux for the value of $\underline{\hat{m}}^{k}$ at the resulting point must follow from interpolation. Well-known methods in 1D interpolate between x_m, x_{m+1} and at least a second upstream point x_{m-1} or x_{m+2}. Hence the proposal leads to an enlargement of the difference molecule for $\underline{\hat{m}}$.

ii) The transport method in 2D can be interpreted as an approximate backward tracing of the characteristic per phase from $(x_{m,n}, t^{k+1})$ to a point $(\bar{x}, t^k)$. For flow in the positive x and y direction, $\bar{x}$ lies in the triangle $(x_{m,n}, x_{m-1,n}, x_{m,n-1})$, if Δt satisfies the stability condition. The phase values at $\bar{x}$ are determined from interpolation, which implies that there is always an interpolation error if the flow does not coincide with one of the grid directions. In a different formulation: the magnitude of the numerical diffusion 'parallel' to the flow depends on Δt and vanishes for the optimal Δt, but the magnitude of the 'perpendicular' component depends on the grid orientation and vanishes only for flow in the grid directions. The phenomenon is known as the grid orientation effect. An obvious remedy is again the extension of the difference molecule for $\underline{\hat{m}}$, now with the diagonal upstream points at least.

iii) If we want to reduce the grid orientation of the $\underline{\hat{m}}$-molecule with a one-step method in time, while maintaining conservation, we are forced to hexagonal grids [2] or to combinations of rotated 5-point molecules [3] for p at the same time. Extension of the p-molecule means larger matrix inversions.

An alternative way of extending the $\underline{\hat{m}}$-molecule (in both respects i and ii) is by means of an interpolation between values of $\underline{\hat{m}}$ at level t^k and preliminary values at level t^{k+1}. Thus flux evaluation points $\underline{\hat{m}}^{k+1/2}_{j,m+1/2,n}$ can be determined for the replacement of $\underline{\hat{m}}^{k}_{up_j(m+1/2,n)}$ in (10) and (11). This is the idea of flux correction which will be discussed in section 5.

iv) The main obstacle for obtaining high shock resolution is the fixed grid. It can be refined around wells and porosity or absolute permeability transitions (all with fixed positions), but it cannot be adjusted to the propagating fronts. Because of the conservation condition and the coupling between $\underline{\hat{m}}$ and p, the first task in the introduction of adaptable grids is the construction of efficient linear algebra solvers for the pressure equation. A multigrid solution for this problem will be discussed in section 6.

5. FLUX CORRECTION FOR MULTISIM

For a linear 1D equation

(14) $$\partial S/\partial t = -\partial/\partial x[vS], \quad v > 0 \text{ constant},$$

the distance covered by the advancing profile during $\Delta t/2$ is $v\Delta t/2$. Obviously, for (14) $x_m + (1-\sigma)\Delta x/2$ (where $\sigma = v\Delta t/\Delta x$) is the point at which, according to the suggestion of section 4b, the value of S^k should replace the one-point upstream value S_m^k in the evaluation of the flux $F_{m+1/2}^{k+1/2}$ for the conservative scheme

$$(15) \qquad (S_m^{k+1}-S_m^k)/\Delta t = (F_{m+1/2}^{k+1/2}-F_{m-1/2}^{k+1/2})/\Delta x.$$

As S_m^k is a mean block value, the profile S^k in block m is unknown. If we suppose that it can be approximated with a linear profile

$$(16) \qquad S^k(x) = S_m^k + \delta_m^k(x-x_m),$$

we must estimate its slope δ_m^k by comparison of S_m^k with at least its closest neighbours S_{m-1}^k and S_{m+1}^k. Van Leer [4] gives the arguments for taking

$$(17) \qquad \delta_m^k = \alpha(S_{m+1}^k-S_m^k)/\Delta x + (1-\alpha)(S_m^k-S_{m-1}^k)/\Delta x$$

with

$$(18) \qquad \alpha = (S_m^k-S_{m-1}^k)^2/\{(S_{m+1}^k-S_m^k)^2 + (S_m^k-S_{m-1}^k)^2\}.$$

This defines the corrected one-point upstream evaluation point for $F_{m+1/2}^{k+1/2}$ as

$$(19) \qquad S_{m+1/2}^{k+1/2} = S_m^k + \frac{(1-\sigma)}{2}\{\alpha(S_{m+1}^k-S_m^k)+(1-\alpha)(S_m^k-S_{m-1}^k)\}.$$

If $\bar{S}$ denotes the profile at time level t^{k+1} obtained with (15) for one-point upstream evaluation of $F_{m+1/2}^{k+1/2}$, an equivalent expression for $S_{m+1/2}^{k+1/2}$ is

$$(20) \qquad S_{m+1/2}^{k+1/2} = S_m^k + \tfrac{1}{2}\{\alpha(\bar{S}_{m+1}-S_m^k) + (1-\alpha)(\bar{S}_m-S_{m-1}^k)\},$$

which is easily computable for non-linear systems

$$(21) \qquad \partial \underline{S}/\partial t = -\partial/\partial x[v\underline{f}(\underline{S})], \quad v\{\partial \underline{f}/\partial \underline{S}\} > 0.$$

For a system, the computation of α_i with (18) for each component S_i separately seems questionable and superfluous: $\alpha = 1/2$ indicates a wave and $\alpha = 0$ or 1 a shock, whereas these different phenomena cannot occur at the

same time and place for the normal injection type problem (decay of a discontinuity) because of their different propagation speeds. In addition, with different α_i, $\Sigma_i S_i^{k+1/2} = 1$ does not follows from $\Sigma_i S_i^k = 1$ for dependent systems with $\Sigma_i S_i = 1$, which would disqualify $\underline{S}^{k+1/2}$ as an evaluation point. Therefore, we replace (18) for systems with

$$\alpha = \|\underline{S}_m^k - \underline{S}_{m-1}^k\|_2^2 / \{\|\underline{S}_{m+1}^k - \underline{S}_m^k\|_2^2 + \|\underline{S}_m^k - \underline{S}_{m-1}^k\|_2^2\}. \tag{22}$$

Extending the idea to (1), we experimented with (19) and estimates of σ obtained by extrapolation from former time levels, but we have only succeeded in constructing a robust method with (20). Choosing the normal one-point upstream method (7)-(13) of MULTISIM for the computation of the preliminary profile $\underline{\bar{m}}$, we arrive at the following expressions for the corrected upstream evaluation points:

$$\underline{m}_{j,m+1/2(-)}^{k+1/2} = \underline{m}_m^k + \tfrac{1}{2}[\alpha(\underline{\bar{m}}_{m+1} - \underline{m}_m^k) + (1-\alpha)(\underline{\bar{m}}_m - \underline{m}_{m-1}^k)] \tag{23}$$

for contributions to $F_{m+1/2}^{k+1/2}$ by phases j with $v_j > 0$,

$$\underline{m}_{j,m-1/2(+)}^{k+1/2} = \underline{m}_m^k + \tfrac{1}{2}[\alpha(\underline{\bar{m}}_m - \underline{m}_{m+1}^k) + (1-\alpha)(\underline{\bar{m}}_{m-1} - \underline{m}_m^k)]$$

for contributions to $F_{m-1/2}^{k+1/2}$ by phases j with $v_j < 0$.

Extension of (23) to a multi-dimensional algorithm FCMULTISIM is straightforward. For its justification in 2D, we remark that the result is almost identical to the simple second order conservative scheme

$$S^{k+1} = [1 - \sigma_x \Delta_x \{1 + \tfrac{1}{2}(1-\sigma_x)\delta_x - \tfrac{1}{2}\sigma_y \Delta_y\} - \sigma_y \Delta_y \{1 + \tfrac{1}{2}(1-\sigma_y)\delta_y - \tfrac{1}{2}\sigma_x \Delta_x\}] S^k \tag{24}$$

(Δ_x denotes the backward difference operator), which is proposed by van Leer in [5]. We assume that the local slope in block (m,n) of the S^k profile is taken as

$$\delta_{x,m,n}^k = \alpha(S_{m+1,n}^k - S_{m,n}^k)/\Delta x + (1-\alpha)(S_{m,n}^k - S_{m-1,n}^k)/\Delta x, \tag{25}$$

by the analogy of (17). Method (24) adds flux corrections to the one-point upstream method with 2D 9-point difference molecule:

$$S^{k+1} = [1 - \sigma_x \Delta_x \{1 - \tfrac{1}{2}\sigma_y \Delta_y\} - \sigma_y \Delta_y \{1 - \tfrac{1}{2}\sigma_x \Delta_x\}] S^k. \tag{26}$$

In (26) we recognize a dimensional splitting of the transport: $[1-\frac{1}{2}\sigma_y\Delta_y]$ represents 1D transport in the y-direction. If v_y is not constant, $\sigma_y\Delta_y$ must be replaced with $\frac{\Delta t}{\Delta y}[\Delta_y v_y]$. Then, for dependent systems with an additional algebraic relation $\Sigma_i\ S_i=1$, the intermediate evaluation points $[1-\frac{\Delta t}{2\Delta y}\Delta_y v_y]\underline{S}^k$ are not admissible because $\Delta_y v_y \neq 0$. The same objection can be raised against the intermediate points in (24). However, if $\sigma_y\Delta_y$ is changed into $[(1-\alpha)\sigma_y\Delta_y+\alpha\sigma_y\Delta_y E_x^+]$, where E_x^+ denotes the shift operator in the positive x-direction, the modified intermediate evaluation point for the x-direction in (24) can be expressed as

$$(27)\qquad S^{k+1/2}_{m+1/2,n} = S^k_{m,n} + \tfrac{1}{2}\{\alpha(\bar{S}_{m+1,n}-S^k_{m,n})+(1-\alpha)(\bar{S}_{m,n}-S^k_{m-1,n})\}.$$

Again $\bar{S}$ is the result which is obtained with the normal one-point upstream method with 2D 5-point molecule. For (27) $\Sigma_i\ S_i^{k+1/2}=1$ holds. Of course, in (27) we have recovered the FCMULTISIM algorithm. The inclusion of diagonal upstream points in (24) and (26) by means of $\Delta_x\Delta_y$ diminishes the grid orientation effect. The same may be expected of FCMULTISIM because of the resemblance above. As far as tests (including counter-current flow examples) have been performed, FCMULTISIM confirms the expectation that it will reduce shock diffusion and grid orientation in MULTISIM. The efficiency of the improvement is indicated by the observation that FCMULTISIM $(\Delta t,\Delta)$ performs at least as well as MULTISIM $(\Delta t/2,\Delta/2)$, whereas the computing costs, which are determined by the matrix inversions for the pressure equation, have been lowered with at least a factor of 2^n in nD.

6. LOCAL GRID REFINEMENT AND MULTIGRID FOR THE PRESSURE EQUATION

Once a flexible administration has been established for the grids of different levels that compose the locally refined grid on which a matrix equation for an implicit discretisation must be solved, it is only natural to use the same grids in a multigrid solution method. Successful black box multigrid methods derive the matrices on the coarse grids from the matrix on the fine grid with the Galerkin P^TAP construction. To prevent that the addition of finer levels during the refinement process will change the coarse matrices, which hence all must be consistent discretisations, it is necessary that the multigrid Galerkin prolongation/restriction coincides with a finite element Galerkin prolongation/restriction. Experience makes numerical reservoir simulation adhere to the inverse weighting of

$\underline{\underline{k}}$a (12) for obtaining good discretisations. An experience that is shared by Kettler and Meyerink [6] in their construction of multigrid prolongations which are based on flow continuity and differ from (bi)linear interpolation. What seems to matter is that in the problem

(28) $\quad -\mathrm{div}(\underline{\underline{k}}\ \mathrm{grad}\ p) + c.p = f$

primarily the velocity $\underline{v} = -\underline{\underline{k}}\ \mathrm{grad}\ p$ is the smooth variable and not p. Being committed to finite element Galerkin, we cannot obtain the inverse weighting and the flow prolongation from finite element methods for (28) in p. However, we succeed with mixed finite element methods in p and $\underline{v}$, which are based on the splitting

(29) $\quad \mathrm{div}\ \underline{v} + c.p = f$

$$\underline{\underline{k}}^{-1}\underline{v} + \mathrm{grad}\ p = 0.$$

Of course, the resulting matrix is no longer symmetric positive definite. The lowest order Raviart-Thomas element on a rectangular grid is defined as

(30) $\quad p$: piecewise constant

v_x: piecewise constant in y-direction,
piecewise linear and continuous in x-direction

v_y: similar to v_x

For this element Schmidt [7] has designed a FAS multigrid method, for which convergence has been proved in the case of constant compressibility c.

REFERENCES

[1] ---, *Verhoging opbrengst oliewinning*, De ingenieur, Nov. 1983, pp. 36-39.

[2] PRUESS, K. & G.S. BODVARSSON, *A seven-point finite difference method for improved grid orientation performance in pattern steam floods*, SPE 12252 (Seventh SPE Symposion on Reservoir Simulation, San Francisco, Nov. 16-18, 1983).

[3] COATS, K.H. & A.D. MODINE, *A consistent method for calculating transmissibilities in nine-point difference equations*, SPE 12248 (Seventh SPE Symposion).

[4] ALBADA, VAN G.D., B. VAN LEER & W.W. ROBERTS, *A comparative study of computational methods in cosmic gas dynamics*, Astron. Astrophys. 108, 76-84 (1982).

[5] LEER, VAN B., *Multidimensional explicit difference schemes for hyperbolic conservation laws*, Proc. 6th Int. Conf. on Computing Methods in Applied Sciences and Engineering, Versailles, Dec. 1983. (to be published).

[6] KETTLER, R., *Analysis and comparison of relaxation schemes in robust multigrid and preconditioned conjugate gradient methods*, Report 82-17 Department of Math. and Inf. TH Delft.

[7] SCHMIDT, G.H., *Local refinement and multigrid for the convection-diffusion equation*, (to be presented at the Int. Conf. on accuracy estimation and adaptive refinement in finite element computations, Lisbon, June 19-22, 1984).

FAST SOLUTION OF LINEAR EQUATIONS WITH SPARSE SYSTEM MATRICES : AN APPLICATION

A. KOOISTRA

1 INTRODUCTION

The dynamical behaviour of linked mechanical systems being frequently used in crash victim simulation may be given by the set $S\ddot{q} = b$ or ordinary differential equations (ode's). S is called the system matrix and b the power vector. The vector q, which has to be solved, represents the translation and the rotation of the member elements of the system, thus defining the whole movement of the system.

In the course of time S and b will be changing and so will q. On solving the set of ode's, by a numerical process, sets of linear equations of the form $Sx = b$ are to be solved many times. The speed of the solving process depends for a great deal on how quickly $Sx = b$ is solved. Now S is sparse, i.e. S contains many zero elements. The structure of S, i.e. the pattern of zero elements, depends only on the structure of the mechanical system and on its description.

It is obvious that a solution method exploiting this structure will be much faster than the traditional ones. In this paper we will discuss such a method.

Moreover a strategy will be given to find the optimal way of describing the mechanical system; optimal in the sense of speed of solving $Sx = b$.

2 STRUCTURE OF THE MECHANICAL SYSTEM

The structure of the matrix S is completely defined by that of the mechanical system. Therefore we shall have to describe and analyse that system first of all.

The system consists of n elements coupled together by hingelike joints. One joint is exactly linked up with 2 elements. On the contrary one single element can be linked up with an arbitrary number of joints. The elements are stiff, they can only rotate (under more or less resistance) around the joints. Loops are not allowed in the structure, but branches are.

To define the structure we have to number the elements. Joints do not need to be numbered. We shall call the i^{th} element, e(i), and choose the origin in one of the joints.

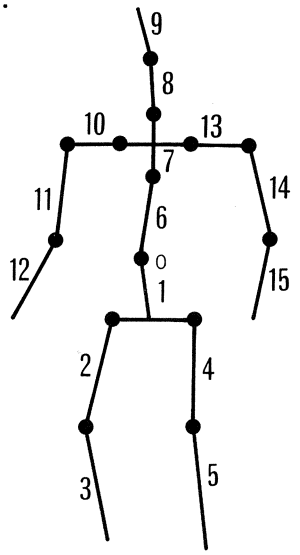


Fig. 1

In fig. 1 we see a linked structure with the numbers of the elements; the origin is marked 0.

We imagine tracks starting at the origin and running to the ends of all branches of the system. All elements must be part of at least one track. Some elements can be part of more than one track however. Tracks are defined by their elements. The elements must be numbered starting at the origin. So e(1) has to be linked up with the origin.

Besides tracks we distinguish branches. They need not start at the origin but may sprout from other branches. Every element has to be part of just one branch.

In the example of fig. 1 we see the branches: 1-2-3, 4-5, 6-7-8-9, 10-11-12, 13-14-15 and the tracks: 1-2-3, 1-4-5, 6-7-10-11-12, 6-7-8-9, 6-7-13-14-15.

There is an elegant way to define the structure through its branches using pointers. This method matches the structure of the matrix nicely. Every element points to its adjacent element closer to the origin. So we can use the same pointer for an element which is situated on different tracks. This is possible because, if we walk towards the origin, the tracks coincide; they converge towards the origin.

We define the whole structure by means of one single integer array called R of n variables. The rank number within the array gives the

number of the element, while the value gives the number of the adjacent element closer to the origin. Elements joining the origin point to 0. Save some jumps, the pointer of an element points to its predecessor. The jumps are caused by the junction of branches.

In our example R is:

Rank number 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 = element

value 0 1 2 1 4 0 6 7 8 7 10 11 7 13 14 points to

We can easily extract the tracks (in reverse) from R:

15-14-13-7-6-0, 12-11-10-7-6-0, 9-8-7-6-0, 5-4-1-0, 3-2-1-0.

We can verify this in fig. 1.

3 CONSTRUCTION OF THE SYSTEM MATRIX

We consider the movement of a mechanical system of n elements, a movement in 3 dimensions. The system matrix S follows from Lagranges equations and is a (n+1)*(n+1) matrix of elements, being 3×3 submatrices. We may consider S as a matrix of submatrices and as a matrix of numbers. In the latter case S is a (3n+3)*(3n+3) matrix. To avoid confusion we will denote the matrix in the former case by S3.

Being m = 3n + 3, S is a m*m matrix, symmetric and positive definite. The greater part of the submatrices is empty, i.e. all their elements are zero. Nonempty means: not previously known, whether all elements are zero or not. The submatrix $S3_{ij}$ is nonempty if element e(j) co-rotates with e(i) when e(i) is rotated in respect of the origin, or in reverse e(i) co-rotates with e(j). This means that e(i) and e(j) are situated on the same track. In the other case submatrix $S3_{ij}$ is empty.

In addition to rotation there will be translation. If the mere translation of the origin causes a co-translation of e(j) then $S3_{0j}$ is nonempty. Because save in case of fracture all elements are connected with the origin, so the whole 0^{th} row is nonempty. If no confusion is possible we will (non) empty submatrices also call (non) zeros.

Our example of fig. 1 gives the following structure of S3, where x's denote nonempty submatrices. Because of symmetry only the upper triangle is given.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	0
	×	×	×	×	×											1
		×	×													2
			×													3
				×	×											4
					×											5
						×	×	×	×	×	×	×	×	×	×	6
							×	×	×	×	×	×	×	×	×	7
								×	×							8
									×							9
										×	×	×				10
											×	×				11
												×				12
													×	×	×	13
														×	×	14
															×	15

Fig. 2

With the array R

element	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
pointer	0	1	2	1	4	0	6	7	8	7	10	11	7	13	14

4 STRUCTURE OF THE MATRIX

As stated before the structure of S3 and so of S is defined as:

- $S3_{ij} = 0$ if there exists no track on which e(i) and e(j) are situated together.
- $S3_{ij} \neq 0$ if there exists at least one track on which e(i) and e(j) are situated both.
- If e(i) and e(j) are on the same track with e(i) closer than e(j) to the origin than i < j.

From this we will deduct 2 useful rules.

1. We choose a (nonzero) element $S3_{ii}$ on the main diagonal, representing element e(i) of the mechanical structure.

   If we pass along its row rightward we will come across nonzeros say $S3_{ij}$. These represent the mechanical elements e(j), situated on the same tracks as e(i) but farther from the origin.

For programming purposes we choose the numbering of the elements in such a way that passing along a row up to a certain point, we only meet nonzeros and beyond that point only zeros. For all rows these points are stored in an array called K, so that element K(i) contains that point for row i.

This means that for every mechanical element e(i) must hold:
All elements situated on tracks together with e(i) but farther from the origin must have numbers greater than i and must form together with i a set of successive natural numbers.
We can achieve this numbering by the following strategy:

Start at the origin.

a If there is no branch joined to the origin of which the elements are still unnumbered, the process has finished.
Otherwise:

b Walk along an unnumbered branch and number its elements. Coming across a junction always take the rightmost branch (seen from the origin).
Reaching the end of a branch, backtrack up to the first junction with a branch to the left or the origin. Having reached the origin repeat the process at a otherwise at b.

2. We choose as before an element $S3_{ii}$ of the main diagonal, representing e(i). If we walk along the i^{th} column towards the top we again meet nonzeros. These represent mechanical elements which we come across walking down the track towards the origin, starting at e(i). We can find the place of these nonzeros by means of the pointer array R. We start at R(i) and just follow the pointers.

In this way we can fill the array K. Therefore we need the numbers of the elements at the ends of the tracks. Let 1 be such a number. If we start following pointers at R(1) we find a series of numbers. We fill the places of K defined by these numbers with the number 1. We do so for all tracks in increasing order, starting with the one with the lowest end number.

5 CHOLESKI DECOMPOSITION

Mainly by intuition the method of Choleski has been chosen to solve

the set of linear equations. This method is rather fast and matches the structure of S very well as we will see.

Using this method we have to decompose S in an upper and a lower triangular matrix which must be their transpose mutually. U is the upper triangular matrix and U^T its transpose. Because S is a symmetric and positive definite the decomposition can be done.

It's even possible to do this in 2 ways: $S = U^T U$ and $S = U U^T$. As we will see later on the latter matches the matrix structure better, so we choose that one though it's not the better known one. In stead of solving the set of equations $S x = b$, we will solve $U U^T x = b$. If we call $U^T x = y$, we can solve $U y = b$ first and then $U^T x = y$. Solving these sets is simply repeated substitution.

To determine the elements of U (and U^T) we use the definition of matrix multiplication, considering that $U^T_{ij} = U_{ji}$.
S, U and U^T being matrices of order m, $S = U U^T$ means:

$$S_{ij} = \sum_{k=j}^{m} U_{ik} U^T_{kj} = \sum_{k=j}^{m} U_{ij} U_{jk} = U_{ij} U_{jj} + \sum_{k=j+1}^{m} U_{ij} U_{jk}$$

For $U_{jk} = 0$ for $k < j$ and $i < j$ because U is an upper triangular matrix.

(5.1) for $i = j$ this gives $S_{ii} = U^2_{ii} + \sum_{k=j+1}^{m} U^2_{ik}$ or $U_{ii} = \sqrt{(S_{ii} - \sum_{k=j+1}^{m} U^2_{ik})}$

(5.2) and for $i \neq j$: $U_{ij} = (S_{ij} - \sum_{k=j+1}^{m} U_{ik} U_{jk})/U_{jj}$

Considering that $\Sigma_{k=j+1}^{m} U_{ik} U_{jk} = 0$ and $\Sigma_{k=j+1}^{m} U^2_{ik} = 0$ when $i \geq m$, we can find a computing sequence for the elements U_{ij}, only using elements of U already computed. We have to start with the m^{th} column. First of all we compute its diagonal element U_{mm} and then its other elements. In this way we may compute all columns down to the first.

We may consider U as a transformation of S.

6 TRANSFORMATION PROPOSITIONS

First of all we notice that for the computation of U_{ij}, besides S_{ij} we only need elements of the i^{th} and the j^{th} row. Of the i^{th} row we use the elements to the right of U_{ij}. Of the j^{th} row we use those to the

right of U_{jj} and U_{jj} itself.
Most of the rows are just filled to a certain point beyond which they are empty.

This yields the following worksaving propositions:

1. If $S_{ij} = 0$ and all elements of the i^{th} row of U, to the right of U_{ij} are zero then $U_{ij} = 0$.
   This immediately follows from the equations 5.2.
2. If $S_{ij} = 0$ and all elements of its row to its right are zero then $U_{ij} = 0$.
   The proof can be given by induction.
   For all elements of the m^{th} column the proposition is clear, because there are no elements to the right of it.
   Under the induction supposition: $U_{ik} = 0 \; \forall \; k \geq j+1$ and $S_{ij} = 0$, we have to prove: $U_{ij} = 0$.
   This follows from proposition 1
3. If $S_{ij} = 0$ and $k > j > i$ then either $S_{ik} = 0$ or $S_{jk} = 0$.
   In terms of the mechanical system this means: If e(i) and e(j) are not on the same track and e(k) is farther from the origin then e(i) and e(j) then: Either e(i) and e(k) are not on the same track or e(j) and e(k) are not.
   This follows from the divergence of the tracks towards their ends.
4. If $S_{ij} = 0$ then $U_{ij} = 0$.
   Again the proof must be given by induction.
   For elements of the m^{th} column the proposition is clear. Under the induction supposition: $U_{ik} = 0$ if $S_{ik} = 0 \; \forall \; k > j+1$ and $S_{ij} = 0$, we have to prove $U_{ij} = 0$.
   This follows from proposition 3 and the equations 5.2. When transforming the structure of the matrix will not be changed. Because those elements which are zero stay zero and those of which we cannot proof they are zero we have to treat as nonzeros.
5. If $S_{ij} \neq 0$ and $S_{jk} \neq 0$ for $i < j < k$ then $S_{ik} \neq 0$.
   Because if $S_{ij} \neq 0$ then $U_{ij} \neq 0$ the proposition means that for the computation of the productsum of $U_{ik}U_{jk}$ we only need those values of k for which $U_{jk} \neq 0$.
   In terms of the mechanical system the proposition is:
   Given: e(i) and e(j) are on the same track and e(j) and e(k) are on the same track, e(i) closer to the origin than e(j) and e(j) closer to the origin than e(k). Then e(i) and e(k) are situated on the same track.

This follows from the convergence of tracks towards the origin.

We notice that proposition 4 need not the being nonzero of a row up to a certain point and zero beyond that point. The only demand is that of 2 elements on the same track, the one who is closer to the origin has the lower number.

7. USE OF THE STRUCTURE OF THE SYSTEM MATRIX

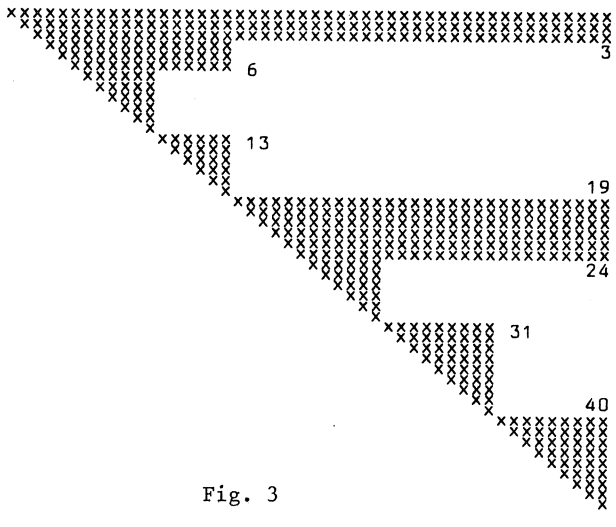


Fig. 3

i	j	R	RK	K	KR
1	0		0	15	48
2			1		48
3			2		48
4	1	0	3	5	18
5			4		18
6			5		18
7	2	1	6	3	12
8			7		12
9			8		12
10	3	2	9	3	12
11			10		12
12			11		12
13	4	1	6	5	18
14			13		18
15			14		18
16	5	4	15	5	18
17			16		18
18			17		18
19	6	0	3	15	48
20			19		48
21			20		48
22	7	6	21	15	48
23			22		48
24			23		48
25	8	7	24	9	30
26			25		30
27			26		30
28	9	8	27	9	30
29			28		30
30			29		30
31	10	7	24	12	39
32			31		39
33			32		39
34	11	10	33	12	39
35			34		39
36			35		39
37	12	11	36	12	39
38			37		39
39			38		39
40	13	7	24	15	48
41			40		48
42			41		48
43	14	13	42	15	48
44			43		48
45			44		48
46	15	14	45	15	48
47			46		48
48			47		48

fig. 4

Besides the array R with pointers to the submatrices of S3 we need a similar array with pointers to the numbers of S, we will call that array RK. We also need an array similar to K with the numbers of the rightmost nonzero's of the rows of S. We can derive RK from R and KR from K considering that:

If RK[i] corresponds with R[j] and KR[i] with K[j], then i = 3j + 1, RK[i] = 3*(R[j]+1) and KR[i] = 3*(K[j]+1). The remainder of the values of RK is part of a continuous series from 0 to m-1. For every element joints to its predecessor, except the jumps. And with these jumps is dealt in R. The unknown values of KR are the same as their known predecessors.

In fig. 3 the system matrix is shown and in fig. 4 the list of i,j,R, KR,K and KR of the example of fig. 1.

A procedure in the programming language Pascal, using the structure of S given by the arrays RK and KR could be:

```
procedure choleski (S:ad2;var U:ad2;RK,KR:ad1; var alarm: boolean);
var i,j,k,upper,lower: integer;
sigma: real;
 {We assume that ad1 = array[1..m] of real;
                  ad2 = array[1..m,1..m] of real;}
begin
   alarm:= false;
   for j:= m downto 1 do {finishing columns}
   begin
      i:= j; {start of pointer following}
      lower:= j+1; upper:= KR[i];
      repeat {finishing elements of column j}
         sigma:= S[i,j];
         for k:= lower to upper do sigma:= sigma + U[i,k]*U[j,k];
         if i ≠ j then U[i,j]:= sigma/U[j,j]     {diagonal element}
         if sigma > 0 then U[j,i]:= sqrt(sigma) else alarm:= true;
         i:= RK[i]; {number of the next nonzero}
      until i = 0;
   end;
end;
```

After the decomposition we have to solve 2 sets of linear equations: $Uy = b$ and $U^T x = y$.

To solve $Uy = b$ we use the formula $y_i = (b_i - \Sigma_{k=i+1}^{m} U_{ik} y_k)/U_{ii}$.

We can use the information of KR to avoid multiplication by zero in the following piece of program:

```
for i := m downto 1 do
begin
  sigma:= b[i];
  for j:= i+1 to KR[i] do sigma:= sigma - U[i,j]*y[i];
    {if i+1 > Kr[i] the do loop is neglected}
  y[i]:= sigma/U[i,i];
end;
```

To solve $U^T x = y$ we use the formula: $x_i = (y_i - \Sigma_{k=1}^{i-1} U_{ik} x_k)/u_{ii}$ because $U_{ik}^T = U_{ki}$ the formula becomes:

$$x_i = (y_i - \sum_{k=1}^{i-1} U_{ki} x_k)/u_{ii}.$$

Here we can use the information of RK in:

```
x[1]:= y[1]/u[1,1];
for i:= 2 to m do
begin
   sigma:= y[i];
   k:= i;
   repeat
      k:= RK[k];
      sigma:= sigma - U[k,i]*x[k];
   until RK[k] = 0;
   x[i]:= sigma/U[i,i];
end;
```

8 MINIMUM NUMBER OF NONZEROS

The choice of the origin determines the number of nonzero elements of the matrix S3. It would be very advantageous, if we could find a strategy to choose the origin in such a way that the number of nonzeros would be minimal.

For this purpose we consider a pivot element e(p), which is connected to the w joints S(i), i = 1..w.
First of all we try to find out which of the joints gives the fewest nonzeros in case of being the origin. We call that joint the most favourable one. In counting the nonzeros we leave the 0^{th} column and row out of account, for these are full of nonzeros.

In case of removal of e(p), the mechanical structure falls apart in a set of w pieces: {D(i)|i=1..w}.

Before the removal D(i) was attached to e(p) by the joint S(i).

Let D(i) contain E(i) elements and give with S(i) as origin N(i) nonzeros. We say that E(i) elements are hanging on S(i).

Let us consider the piece D(i) and choose the joint S(i) as origin, we have N(i) nonzeros. If we put e(p) back we have N(i) + 1 nonzeros. If after that we fasten the piece D(j) then the number of nonzeros increases with N(j) + E(j) up to N(i) + 1 + N(j) + E(j). We have to add this E(j) because it's the number of nonzeros in the p^{th} row to the right of $S3_{pp}$. If we put back al the other pieces, the number of nonzeros will become the total number of nonzeros in case S(i) is the origin. That number we call NN(i).

$$NN(i) = N(i) + 1 + \sum_{j=1}^{i-1} \{E(j) + N(j)\} + \sum_{j=i+1}^{w} E(j) + E(j)\} \text{ or}$$

$$NN(i) = (\sum_{j=1}^{w} N(j) + 1 + \sum_{j=1}^{w} E(j)) - E(i) \text{ so}$$

$$NN(i) + E(i) = \{\sum_{j=1}^{w} N(j) + \sum_{j=1}^{w} E(j)\} + 1 = \text{constant.}$$

So NN(i) is minimal in case E(i) is maximal.

This means: The joint S(w) on which most elements are hanging is the most favourable joint of element e(p). All other joints of e(p) are less favourable. This holds even more for joints linked with the other joints of e(p) via other elements. This is because the number of elements which hangs on those joints is even less.

However, it is possible that one of the joints which is linked with S(w) by means of another element than e(p), say e(q), is more favourable than S(w). Whether this occurs, we can find out by counting the elements hanging on the joints of e(q). If we find again S(w) the most favourable one, then we have found the most favourable joint of the whole structure. Otherwise we have to take e(q) for pivot element and continue the process. Of course it is possible that we find joints which are equally favourable. In that case we can choose arbitrarily.

NUMERICAL MATHEMATICS IN PRACTICAL DATA FITTING

C.G. van der LAAN

A few examples of problems arising from practical situations, where splines are used, are discussed. The problem situations are:
- separation of exponentials
- determination of growth curves
- computer aided repositioning and quantification of facial swelling volume.
In all these cases the algorithmic and numerical aspects are a vital but small part; the totality of: numerical mathematics, statistics, graphics, software engineering in a variety of environments, with emphasis on the proliferating software-tailered general purpose, personal computers and microsystems, is getting more and more important. The latter aspect is examplified by the NAG library because of the graphical supplement, the statistics chapters and the PC-subcollection.

1. SEPARATION OF EXPONENTIALS

A well-known problem is the determination of n and the coefficients (a_k,α_k) from the model

$$\sum_{k=1}^{n} a_k e^{\alpha_k x}$$

for given measurements $(x_k,f_k)_{k=1}^{p}$, p >> n. Thomasson & Clark (1974) surveyed four classes of techniques: graphical, algebraic iterative and transform, and proposed a combination of a graphical method, a transform method and an iterative method. As Lanczos (1957) has pointed out separation of exponentials is an ill-posed problem, therefore we propose a modification of the model into the *direct sum* of two exponentials with in between a linear combination of cubic B-splines, i.e. the function M(x) defined by

exponential: $f(x) = b_0 + a_0 e^{\alpha_0 x}, \quad x < x_1$

B-spline: $S(x) = \sum_{j=1}^{\ell} c_j B_{j,4}(x;t), \quad x_1 \le x \le x_m$ (with knots t)

exponential: $g(x) = b_\infty + a_\infty e^{\alpha_\infty x}, \quad x_m < x$

$B_{j,4}(x;t)$ denotes a cubic B(asic) spline with knot sequence t. The general idea is that the extreme data represent the exponential behaviour of the data on either end rather well, while in between no clear distinction between exponentials can be made: so don't. Besides, the user was not interested in the coefficients (no identification only representation) but in: the area under the curve, the horizontal asymptote and the intersection of the extrapolated curve and the Y-axis. In order to get a smooth approximation continuity conditions up to the second derivative were added at the breakpoints x_1 and x_m, i.e.:

$$f^{(k)}(x_1) = S^{(k)}(x_1), \quad k = 0,1,2$$

$$g^{(k)}(x_m) = S^{(k)}(x_m), \quad k = 0,1,2.$$

From these conditions and the model assumption we have for the coefficients of the exponentials.

$$\alpha_0 = S''(x_1)/S'(x_1)$$

$$a_0 = S'(x_1)e^{-\alpha_0 x_1}/\alpha_0$$

$$b_0 = S(x_1) - a_0 e^{\alpha_0 x_1}$$

and analogously

$$\alpha_\infty = S''(x_m)/S'(x_m)$$

$$a_\infty = S'(x_m)e^{-\alpha_\infty x_m}/\alpha_\infty$$

$$b_\infty = S(x_m) - a_\infty e^{\alpha_\infty x_m}.$$

The solution of the users problem is then given by

- determination of an approximating spline in the least square sense (via NAG or de Boor routines (1978));
- calculation of the exponential coefficients;

- determination of required quantities. (Integral, asymptote and intersection of extrapolated data with Y-axis.)

Discussion. A refinement of the above approach could be to optimize also the locations x_1 and x_m, i.e. to optimize for the extend to which the end data behave like an exponential. This can be stated as the nested minimization problem: minimize x_1 and x_m over the constrained least squares problem

$$\min_{\alpha_0,a_0,b_0,\{c_j\},\alpha_\infty,a_\infty,b_\infty} \|M(x) - y\|_2$$

with the following nonlinear constraints

$$\alpha_0 = S''(x_1)/S'(x_1) \qquad \alpha_\infty = S''(x_m)/S'(x_m)$$

$$a_0 = S'(x_1)e^{-\alpha_0 x_1}/\alpha_0 \qquad a_\infty = S'(x_m)e^{-\alpha_0 x_m}/\alpha_\infty$$

$$b_0 = S(x_1) - a_0 e^{\alpha_0 x_1} \qquad b_\infty = S(x_m) - a_\infty e^{\alpha_\infty x_m}.$$

The practice in the above is that users often come up with an ill-posed problem; the numerical analist is urged to consider a different problem formulation: no separation of exponentials but approximation and extrapolation.

2. DETERMINATION OF GROWTH CURVES

In Gerver et al. the problem of determination of age-dependent reference values is treated. The classical approach is grouping into age classes, calculating mean and standard deviation for each group, plotting the values and smoothing by eye. This process is laboursome and in fact not reproducible. Our approach was to abandon the grouping into age classes and to create a smooth mean-line via a cubic spline model function in the least squares sense as follows.
Given the data (x_i,y_i), i=1,...,n, the problem is to construct a smooth function f such that

$$Q(f) = \sum_{i=1}^{n} [y_i - f(x_i)]^2$$

is small. Various linear spaces F of cubic splines are considered, each space being characterized by two numbers k and m and a sequence of k

so-called variable knots. Computation of f_F^* such that

$$Q(f_F^*) = \min_{f \in F} Q(f)$$

is a matter of elementary least squares. In this respect it is useful if F is spanned by the so-called B (asic)-splines (De Boor (1978)). Given the numbers k, m and the k variable knots, the dimension d of F is given by $d = (k+1)(m+1) + 3$ and the knot-sequence by $t_1,\ldots,t_{d+4}$, where t_{m+5}, $t_{2m+6},\ldots,t_{km+k+4}$ correspond with the so-called "variable" knots and the others are called "intermediate". Any $f \in F$ can now be represented by

$$f(x) = \sum_{j=1}^{d} c_j B_{j,4}(x;t_1,\ldots,t_{d+4})$$

where the $B_{j,4}$'s are the B-splines. The optimal element f_F^* is determined by the corresponding optimal weights $c_1^*,\ldots,c_d^*$ which can be obtained as the solution of the overdetermined set of equations.
Given the number k of variable knots and the number m of intermediate knots (in each of the k+1 intervals defined by the variable knots) and starting from an initial guess of the k variable knots, the optimal space $F_{k,m}^*$ is determined by maximizing $Q(f_{F_{k,m}}^*)$ over all space $F_{k,m}$ satisfying:

(1) $t_1 = t_2 = t_3 = t_4$ = age of youngest child measured

(2) $t_{d+1} = t_{d+2} = t_{d+3} = t_{d+4}$ = age of oldest child

(3) the intermediate knots between two consecutive variable knots are chosen at equal distances.

More precisely, the NAG library is used to calculate the "variable" knots t_{m+5}, $t_{2m+6},\ldots,t_{km+k+y}$ in such a way that the corresponding $Q(f_F^*)$ is minimal. Here f_F^* belongs to the knot-sequence $t_1,\ldots,t_{d+4}$ where the mentioned restrictions are satisfied and in particular the m intermediate knots in the intervals (t_4,t_{m+5}), $(t_{m+5},t_{2m+6}),\ldots,(t_{d-m},t_{d+1})$ are for each interval at equidistant positions:

$$t_5 - t_4 = t_6 - t_5 = \ldots = t_{m+5} - t_{m+4};\ t_{m+6} - t_{m+5} = t_{m+7} - t_{m+6},\ldots$$

The numbers k and m were modified such that the graphical representation was as nice as possible. This technique with two variables k and m is a convenient generalization of the cases: m=0, of variable knots only; and k=0, of fixed equidistant knots only.

The statistical part of the problem is the determination of the standard deviation, σ. It would have been natural from a statistical point of view

to draw a smooth line, e.g. by applying a spline approximation similar to the one used for the determination of the mean, through the following points:

$$(x_i, [r^{-1} S_r(x_i)]^{\frac{1}{2}}), \quad i = \tfrac{1}{2}r + 1, \ldots, n - \tfrac{1}{2}r,$$

with

$$S_r(x_i) = \sum_{j=-r/2}^{r/2} (y_{i+j} - \tilde{y}(x_{i+j}))^2, \quad i = \tfrac{1}{2}r + 1, \ldots, n - \tfrac{1}{2}r,$$

where $\tilde{y}$ denotes the mean. From these quantities the confidence intervals for the σ are given by

$$\sqrt{S_r(x) / (r\chi^2_{r;\alpha/2})} < \sigma(x) < \sqrt{S_r(x) / (r\chi^2_{r;1-\alpha/2})}$$

for selected arguments x.

A nice smooth curve $\tilde{\sigma}(x)$, through the intervals belonging to x_i with $i = i_1, \ldots, i_k$, was obtained by applying the algorithm of Reinsch as available in IMSL. The application of this algorithm requires that again some subjective choices have to be made, e.g. the choice of k. From this $\tilde{\sigma}$ and the mean, the p_{97}, p_{90}, p_{75}, p_{25}, p_{10}, p_3 curves were obtained via $p_k = p_{50}(1 \pm f_k \tilde{\sigma})$, f_k a factor, see figure (2.1).

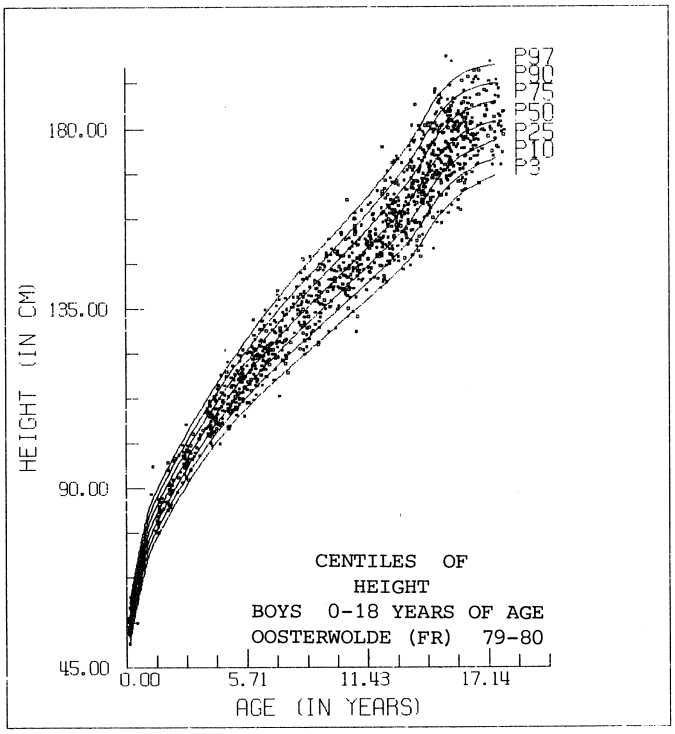


Fig. 2.1.

The practice in the above is that simple problems give sometimes rise to a modern, time saving and more scientific, i.e. reproducible method, where the final choice between alternatives for some parameters remains with the user, in other words: Computer assisted and Intuition controlled Heuristics, to paraphrase Bauer. Honestly speaking variable knots were introduced in order to locate changes in pre-adolescence growth behaviour which we could not find, of course; after all a general and flexible program remained.

3. COMPUTER AIDED REPOSITIONING AND QUANTIFICATION OF FACIAL SWELLING VOLUME

To determine the effectiveness and optimal dosage scheme of corticosteroids in reducing postoperative swelling and other complaints a new method of quantifying the swelling volume is in development, see van Rijn & van der Laan for more details. Accurate three-dimensional repositioning of the patients head, which is necessary for the pre- and postoperative measurements to be comparable, and calculation of the swelling volume is processed by the computer.

Quantification of the swelling.

The problem how to quantify the swelling volume can be split into mathematical repositioning of the patient's head and calculation of the swelling volume.

Repositioning.

From a mathematical point of view the pre- and postoperative data matrices describe roughly the same surface except of course for the swelling, but the relative position of their co-ordinate systems is shifted and rotated because of the repositioning error.

EXAMPLE. Consider a plane given by the equation

$$x + y + z = 1,$$

in the normal x-y-z-Cartesian co-ordinate system. The intersection points of the co-ordinate axes and the plane are described by the triples

$$\begin{bmatrix}1\\0\\0\end{bmatrix}, \begin{bmatrix}0\\1\\0\end{bmatrix}, \begin{bmatrix}0\\0\\1\end{bmatrix}$$

in the given co-ordinate system. The same points can be described by the triples

$$\begin{bmatrix} 0 \\ -\sqrt{2}/2 \\ 0 \end{bmatrix}, \quad \begin{bmatrix} 0 \\ \sqrt{2}/2 \\ 0 \end{bmatrix}, \quad \begin{bmatrix} 0 \\ 0 \\ \sqrt{6}/2 \end{bmatrix}$$

in the shifted and rotated $\hat{x}$-$\hat{y}$-$\hat{z}$-co-ordinate system, where the axes are given by the lines

$$\begin{aligned} \hat{x}\text{-axis:}\quad & 2x - 2z = 1 \;\&\; x = y \\ \hat{y}\text{-axis:}\quad & x + y = 1 \;\&\; z = 0 \\ \hat{z}\text{-axis:}\quad & 2x + z = 1 \;\&\; x = y \end{aligned}$$

in the x-y-z-system.

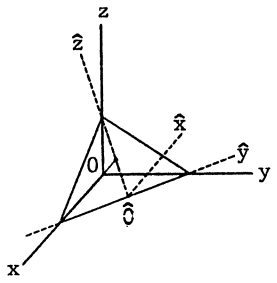


Fig. 3.1. A plane given by the formula x+y+z = 1 in the x-y-z-co-ordinate system and $\hat{x}$=0 in the $\hat{x}$-$\hat{y}$-$\hat{z}$-co-ordinate system.

The $\hat{x}$-$\hat{y}$-$\hat{z}$-co-ordinate system will coincide with the x-y-z-co-ordinate system by a shift of the origin $\hat{0}$ to 0, followed by a rotation around the $\hat{y}$-axis and $\hat{z}$-axis. This process is not unique. In general co-ordinates in the x-y-z-system are related to co-ordinates in the $\hat{x}$-$\hat{y}$-$\hat{z}$-system by the formula

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = R_{\hat{z}}(\phi_{\hat{z}}) R_{\hat{y}}(\phi_{\hat{y}}) R_{\hat{z}}(\phi_{\hat{z}}) \begin{bmatrix} \hat{x} \\ \hat{y} \\ \hat{z} \end{bmatrix} + \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix} \tag{3.1}$$

with

$$R_{\hat{x}}(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix},$$

$$R_{\hat{y}}(\phi) = \begin{bmatrix} \cos\phi & 0 & \sin\phi \\ 0 & 1 & 0 \\ -\sin\phi & 0 & \cos\phi \end{bmatrix},$$

$$R_{\hat{z}}(\phi) = \begin{bmatrix} \cos\phi & \sin\phi & 0 \\ -\sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

where the Δ-vector represents the origin $\hat{0}$ in x-y-z-co-ordinates and $\phi_{\hat{x}}$, $\phi_{\hat{y}}$, $\phi_{\hat{z}}$ represent the rotation angles around the $\hat{x}$-axis, $\hat{y}$-axis and $\hat{z}$-axis, respectively. For the proof see appendix A.
Mathematical repositioning can be formulated as determination of the parameters $p = \{\phi_{\hat{x}}, \phi_{\hat{y}}, \phi_{\hat{z}}, \Delta x, \Delta y, \Delta z\}$, followed by transformation of the postoperative data with these parameters via formula (3.1).
In the sequel we assume that pre-operative data are represented by triples

$$\begin{bmatrix} \alpha \\ z \\ r(\alpha,z) \end{bmatrix}_i, \quad i = 1,2,\ldots,N$$

with N the number of measurements, α the angle, r the complement of the radius and z the cilinder axis. Postoperative data are marked with a ~ (tilde).

Determination of repositioning parameters

Suppose the pre-operative data have been fitted by the formula $r(\alpha,z)$ (See appendix B).
For the determination of the repositioning parameters $\hat{p}$ we have chosen the following least squares criterion

$$\hat{p} = \arg_p \min \Sigma_j \, (r(\alpha_j', z_j') - r_j')^2,$$

where α'_j, z'_j, r'_j are determined via trial repositioning parameters p from the postoperative data $\tilde{\alpha}_j, \tilde{z}_j, \tilde{r}_j$ via formula (3.1) as follows

$$\begin{bmatrix} \alpha' \\ z' \\ r' \end{bmatrix}_j = R_{\hat{x}}(\phi_{\hat{x}}) R_{\hat{y}}(\phi_{\hat{y}}) R_{\hat{z}}(\phi_{\hat{z}}) \begin{bmatrix} \tilde{\alpha} \\ \tilde{z} \\ \tilde{r} \end{bmatrix} + \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix}.$$

In principle, all points outside the swelling could have been taken into account. We considered, however, a subset of the postoperative data, where the independent co-ordinates are contained in a window

$$(\tilde{\alpha}_j, \tilde{z}_j) \in [\tilde{\alpha}_{low}, \tilde{\alpha}_{high}] \times [\tilde{z}_{low}, \tilde{z}_{high}].$$

The window parameters are to be supplied by the user. In our tests the window contained parts of the nose and eyes, in order to create a well-posed problem.

Initially, during the minimization process a well-distributed subset of the window points are considered for efficiency reasons; ultimately all window points are taken into account.
Once the repositioning parameters $\hat{p}$ are found the postoperative data are transformed into repositioned data, i.e.

$$\begin{bmatrix} \tilde{\alpha} \\ \tilde{z} \\ \tilde{r} \end{bmatrix}_j \xrightarrow{\hat{p}} \begin{bmatrix} \hat{\alpha} \\ \hat{z} \\ \hat{r} \end{bmatrix}_j, \quad j = 1,2,\ldots,N$$

via formula (3.1).

Calculation of the swelling volume

After the mathematical repositioning the pre-operative and post-operative data are comparable, and represented by formulas $r(\alpha,x)$ and $\hat{r}(\alpha,z)$, respectively.
The swelling volume bounded by the surfaces r and $\hat{r}$ over the domain

$$[\alpha_{low}, \alpha_{high}] \times [z_{low}, z_{high}]$$

is given by

$$V = \tfrac{1}{2} \int_{\alpha_{low}}^{\alpha_{high}} \int_{z_{low}}^{z_{high}} \{\hat{r}^2(\alpha,z) - r^2(\alpha,z)\}dzd\alpha \tag{3.2}$$

(see appendix C).

The approximation of the face is illustrated in fig. 3.2 and fig. 3.3 which reflect original data and (simulated) data from which the reposition parameters are to be determined.

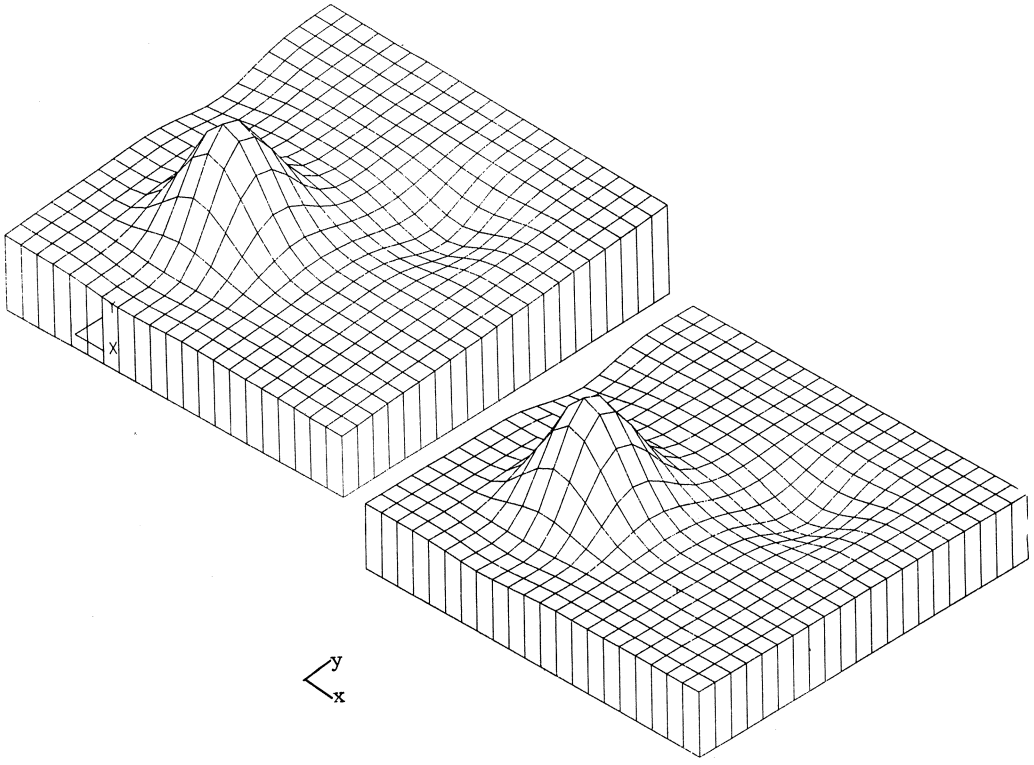


Fig. 3.2 & 3.3. Original data and to be repositioned data.

Conclusions

First, computer aided repositioning is efficient and accurate and gives the possibility of repositioning the patient to obtain comparable measurements, either afterwards or on-line. The duration of a measurement, i.e. scanning of a patient's face, can drasticallt be reduced, because there is no need of accurate repositioning the patient before scanning. The accuracy of repositioning is highly improved by this method.

Second, abstract simulation by computer appeared to be time-saving and required much less effort in testing the computer program than concrete simulation by the plaster head. When comparing the results it became evident that abstract simulation is at least as accurate as concrete simulation.

Future aspects

At this moment we process the scanning results with the university computer, a CYBER 170/760 of Control Data Corporation. In the future we intend to transform dataprocessing to a mini- or microcomputer in order to be independent of access limitations, and datatransfer to, the university computer and to create our own, but coupled, system with scanner, disk units, graphic screen and plotter.
The system described can also be used for other purposes such as quantifying tumor-volume changes in the face, e.g. parotid gland tumors, side-face analysis for osteotomies and orthodontics and quantifying facial oedema in radiotherapy.

Appendix A.(Relation between shifted and rotated and original data).

Co-ordinates in the x-y-z-system and the shifted and rotated $\hat{x}$-$\hat{y}$-$\hat{z}$-system are related by the formula:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = R_{\hat{z}}(\phi_{\hat{z}})R_{\hat{y}}(\phi_{\hat{y}})R_{\hat{x}}(\phi_{\hat{x}})\begin{bmatrix} \hat{x} \\ \hat{y} \\ \hat{z} \end{bmatrix} + \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix}$$

with

$$R_{\hat{x}}(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & \sin\phi \\ 0 & -\sin\phi & \cos\phi \end{bmatrix}, \quad R_{\hat{y}}(\phi) = \begin{bmatrix} \cos\phi & 0 & \sin\phi \\ 0 & 1 & 0 \\ -\sin\phi & 0 & \cos\phi \end{bmatrix}, \quad R_{\hat{z}}(\phi) = \begin{bmatrix} \cos\phi & \sin\phi & 0 \\ -\sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

where the Δ-vector represents the origin of the $\hat{x}$-$\hat{y}$-$\hat{z}$-system in x-y-z-co-ordinates.

PROOF. We can start with rotation around the $\hat{x}$-axis until the $\hat{y}$-axis is in or parallel to the x-y-plane (note: z-axis $\perp$ $\hat{y}$-axis).

Then rotate around the $\hat{y}$-axis until z-axis and $\hat{z}$-axis are parallel and finally rotate around $\hat{z}$-axis until the other axes are parallel. The whole is completed by a shift of $\hat{0}$ to 0. The above process is a repeated 2-dimensional rotation.

The rotation in 2 dimensions can be described by the R-matrices, because co-ordinates in a x-y-system and the rotated x_ϕ-y_ϕ-system are related by:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x_\phi \cos\phi + y_\phi \cos\phi \\ -x_\phi \sin\phi + y_\phi \cos\phi \end{bmatrix} = \begin{bmatrix} \cos\phi & \sin\phi \\ -\sin\phi & \cos\phi \end{bmatrix} \begin{bmatrix} x_\phi \\ y_\phi \end{bmatrix}$$

as can be deduced from the following figure.

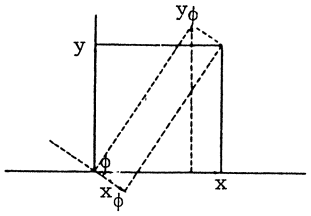


Fig. A. Co-ordinates in mutually rotated co-ordinate systems in a plane.

The above formula can be expanded to 3 co-ordinates by inserting 0 and 1 at the appropriate places, which will yield the R-matrices.

Appendix B. (Fitting 2-dimensional data by product B-splines).

In 1-dimensional fitting the B-spline functions are a versatile tool. A cubic B-spline basic function is positive on the interval (t_i, t_{i+4}), where the points $t_1, t_2, \ldots, t_n$ denote the knot sequence, and zero outside the interval as depicted in figure B.

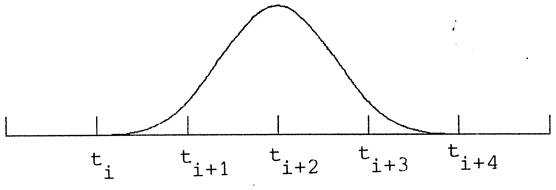


Fig. B. Cubic B-spline basic function on the interval t_1 to t_n.

For 1-dimensional fitting problems the model function

$$\sum_{i=1}^{n-4} c_i B_i(x;t)$$

is used, where the coefficients $\{c_i\}$ are determined from some given data via the least squares criterion. For 2-dimensional data fitting problems product functions analogous to $x_i y_j$ are formed by $B_i(x;u)B_j(y;t)$ with knot sequences

$$u_1, u_2, \ldots, u_n; \quad t_1, t_2, \ldots, t_m.$$

With these product functions the model function

$$\sum_{i=1}^{n-4} \sum_{j=1}^{m-4} c_{ij} B_i(x;u) B_j(y;t),$$

is used. Fitting this model function to data in the least squares sense involves a lot of technical details, such as ordering the data points in such a way that the sparsity in the resulting linear least squares problem is exploited as much as possible and such as the algorithm choice for the solution of the overdetermined system of linear equations. Appropriate choices are made in the NAG-routine E02DAF, and for the details we refer to the documentation of the routine. The evaluation of the model function is done via the NAG-routine E02DBF preceded by E02ZAF for efficiency reasons with respect to grouping argument values.

Appendix C. (Volume of a surface described by product B-splines in cilinder co-ordinates).

In cilinder co-ordinates an infinitesimal volume at point (α,z,r) is given by dz*rdα*dr, as can be seen in the following figure:

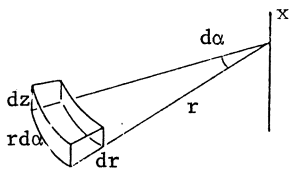


Fig. C. The infinitesimal volume at point (α,z,r).

Integration of this volume yields

$$\int_{\alpha_\ell}^{\alpha_h}\int_{z_\ell}^{z_h}\Big[\int_{r}^{\hat{r}} r\,dr\Big]dz\,d\alpha = \tfrac{1}{2}\int_{\alpha_\ell}^{\alpha_h}\int_{z_\ell}^{z_h}(\hat{r}^2(\alpha,z)-r^2(\alpha,z))\,dz\,d\alpha$$

with r the inner and $\hat{r}$ the outer surface and z_ℓ, z_h the bounds in the z-direction and α_ℓ,α_h the bounds in the angle α.
For the special case that the surfaces are represented by similar product splines,

$$\hat{r}(\alpha,z) = \sum_{i=1}^{n}\sum_{j=1}^{m}\hat{c}_{ij}B_i(\alpha)B_j(z)$$

$$r(\alpha,z) = \sum_{i=1}^{n}\sum_{j=1}^{m}c_{ij}B_i(\alpha)B_j(z)$$

we obtain for the integrand:

$$\hat{r}^2 - r^2 = \{\sum_{i=1}^{n}\sum_{j=1}^{m}c^{-}_{ij}B_i(\alpha)B_j(z)\}\{\sum_{k=1}^{n}\sum_{\ell=1}^{m}c^{+}_{k\ell}B_k(\alpha)B_\ell(z)\}$$

with

$$c^{-}_{ij} = \hat{c}_{ij} - c_{ij},\quad c^{+}_{k\ell} = \hat{c}_{k\ell} + c_{k\ell}.$$

Interchanging summation and integration and factorization of the integrals yields

$$V = \sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{k=1}^{n}\sum_{\ell=1}^{m}c^{-}_{ij}c^{+}_{k\ell}\{\int_{\alpha_\ell}^{\alpha_h}B_i(\alpha)B_k(\alpha)\,d\alpha\}\{\int_{z_\ell}^{z_h}B_j(z)B_\ell(z)\,dz\}.$$

The integrals can be calculated via programs given by the Boor (1975).

The practice in the above is that although from a mathematical software point of view no optimal model function (i.e. the product spline) is used it serves the automation process of integrating local small computer systems and the measurement apparatus. This latter aspect of the proliferation of powerful personal computers or micro computer systems, which must be tailored for a particular application by software, will dominate computer usage in the 80's. Numerical analists and software engineers should be aware of this.

REFERENCES

[1] BOOR, C. de (1975), *On calculating with* B-*splines*. II Integration. in: *Numerische methoden der Approximationstheory*. Band 3. ISNM vol. 30.

[2] BOOR, C. de (1978), *A practical guide to splines*, Springer Verlag.

[3] GERVER, W.J.M., C.G. VAN DER LAAN, W. SCHAAFSMA & N.M. DRAYER (to appear in: International Journal of Bio-Medical Computing), *Smoothing techniques for obtaining age dependent reference values.*

[4] IMSL Houston USA.

[5] LANCZOS, C. (1957), *Applied analysis*, Prentice Hall.

[6] RIJN, L.J. VAN, C.G. VAN DER LAAN, G. BOERING & J.J. TEN BOSCH (in preparation), *Computer aided repositioning and quantification of facial swelling volume: a new three-dimensional method.*

[7] NAG Oxford UK.

[8] THOMASSON, W.M. & J.W. CLARK JR. (1974), *Analysis of exponential decay curves: a three-step scheme for computing exponents*. Mathematical Biosciences 22, 179-195.

MODELLING OF DMOS TRANSISTORS

G. de MEY, D. LORET, A. van CALSTER

The DMOS transistor is a semiconductor component disigned as a switching device, i.e. it can be either in a high conducting state ("on") or a non conducting state ("off"). In the on state, the transistor should be able to conduct a large electric current with a negligible voltage drop. In the off state the current is almost zero, but the trnasistor should be able to withstand a high voltage across its terminals. For practical reasons, this voltage should be as high as possible, which means high electric fields in the semiconductor. However, the maximum electric field may never exceed the ionisation level. The purpose of the present study was to design an optimal transistor geometry in order to get the highest possible terminal voltage. In order to calculate the field distribution, Poisson's equation has been solved using the boundary element method. This numerical technique is extremely usefull if the divice geometry is a variable input parameter.

1. Introduction.

In order to understand the physical behaviour of a D-MOS transistor, the "classical" MOS (metal-oxide-semiconductor) transistor will be outlined first.

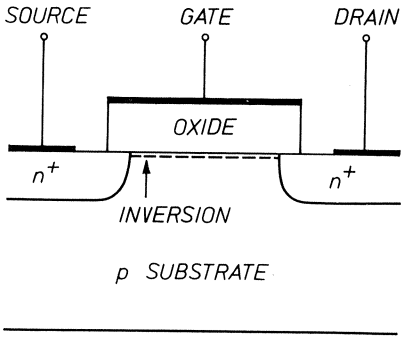


Figure 1.
Structure of a simple MOS transistor.

Fig. 1 shows a typical MOS structure. In a p-type semiconducting substrate, two heavily n-doped (n^+) regions are diffused. A junction between p- and n-type semiconductors gives rise to a diode so that under normal conditions, no current can flow from drain to source because there is always a blocking diode (either the n^+p junction at the source or the drain). By adding a third gate electrode the electron conduction between source and drain can be controlled. In principle this can be done without energy consumption because the gate is insulated from the semiconductor by a non conducting oxide. It is clear that a positive gate voltage will attrack electrons to the oxide-semiconductor interface giving rise to a n-type inversion layer just beneath the gate oxide. There is now a direct conduction path (n^+, n-type inversion layer, n^+) between source and drain and the transistor is in the "on" state. By applying a negative gate voltage electrons will be pulled away from the gate oxide, the inversion layer disappears, the semiconductor is now completely p-type and there is no conduction between source and drain as outlined above. The MOS transistor is now in the "off" state.

We see that conduction between two electrodes can be controlled by a gate electrode which is insulated from the semiconductor. Mentioned so far, the transistor is described as a switch, but it can also be used to amplify electric signals in a more continuous way. For the sake of simplicity we shall limit ourselves to the switching device.

A good electric switch should fullfil several contradictory conditions at the same time :

- in the on state the transistor should require a minimal voltage drop or the on resistance should be as low as possible.
- the transition between the on- and the off-state should be fast.
- in the off state the transistor should withstand high voltages across its terminals without internal ionisation causing short circuiting between source and drain.

The first two conditions can be met by decreasing the gap between source and drain. The last condition can be met by increasing the gap. A high voltage on the drain electrode in the "off" state creates high electric fields which can lead to ionisation.

In order to meet these contradictory requirements, a DMOS (double diffused metal oxide semiconductor) transistor has been introduced (fig. 2). The substrate is now a lightly doped (n^-) layer and there are two regions (n^+ and p^-) beneath the source. These regions are obtained by two sequential diffusions through the same mask, hence the distance AB is small compared to BC (fig. 2).

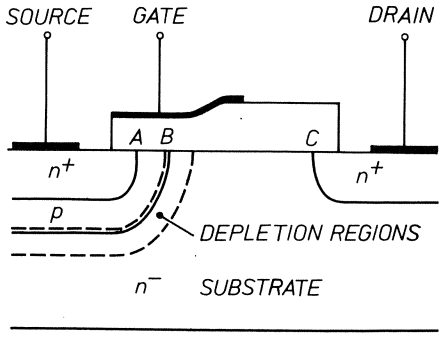


Figure 2.

Structure of double diffused (DMOS) transistor.

In the conducting "on" state, the drain n^+ and the n^- substrate act as a large drain contact. The gate voltage creates a n-type inversion layer between A and B. Due to the small distance between A and B the transistor is very fast and the conduction between A and B requires no voltage drop. In contrast to the classical MOS transistor one has the additional resistance of the low doped n^- region. Nevertheless this resistor can be kept low because the n^- region extends over the whole substrate. In the off state the p^-n^- interface will now be the blocking diode and the depletion region (= absence of charge carriers) is built up in the n^- substrate. Due to the large distance BC, very high voltages can be applied to the drain electrode.

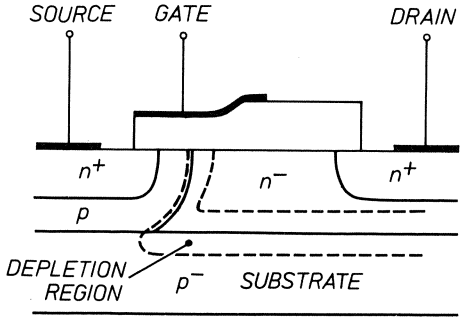


Figure 3.

DMOS transistor on p^- substrate.

Another interesting configuration is shown on fig. 3 using a p^- substrate. The working mechanism is identical to the previous one. For a more precise description of DMOS transistors one is referred to the literature [1][2][3] .

2. Fundamental equations.

The fundamental equations for a semiconductor device are :

$$\frac{\partial n}{\partial t} - \frac{1}{q} \nabla.\overline{J}_n = G_n - R_n \qquad (1)$$

$$\frac{\partial p}{\partial t} + \frac{1}{q} \nabla.\overline{J}_p = G_p - R_p \qquad (2)$$

$$\overline{J}_n = n\, q\, \mu_n \overline{E} + q\, D_n\, \nabla n \qquad (3)$$

$$\overline{J}_p = p\, q\, \mu_p \overline{E} - q\, D_p\, \nabla p \qquad (4)$$

$$-\nabla^2 \phi = \nabla.\overline{E} = \frac{q}{\varepsilon_o \varepsilon_S} (p - n + N_D - N_A) \qquad (5)$$

where :

n : electron concentration
p : hole concentration
$\overline{J}_n$: electron current density
$\overline{J}_p$: hole current density
N_D : donor concentration
N_A : acceptor concentration
G,R : generation and recombination rates
$\mu = qD/kT$: mobility of charge carrier (electron or hole).

Several computer programs have been written to solve the non linear partial differential equations (1)-(5) for a lot of semiconductor devices. For MOS transistors the programs MINIMOS (T.U. Wien) and CADDET (Hitachi) are the best known [4][5].

In several applications, drastic approximations of (1)-(5) can be made. For semiconductor layers with large dimensions and no internal junction, it can be proved that no charge density will be built up. For an n-type layer one can state that [6] :

$$n = N_D \qquad p \ll n \qquad (6)$$

The equation (3) reduces to :

$$\overline{J}_n = \overline{J} = N_D \, q \, \mu_n \, \overline{E} = \sigma E \qquad (7)$$

and the Poisson's equation simply becomes the Laplace' equation. The relation (7) tells us that the layer behaves as a medium with a constant conductivity σ. This approximation will be used in section 4, to investigate the drift resistance of a DMOS transistor.

A second well known approximation is the abrupt depletion method described in many textbooks on semiconductor components. Under zero current condition ($\overline{J}_n \approx 0$, $\overline{J}_p \approx 0$) the equations (3) and (4) lead to :

$$n = n_o \, e^{q\phi/kT} \qquad (8)$$

$$p = p_o \, e^{-q\phi/kT} \qquad (9)$$

Inserting (8) and (9) in the potential equation (5) gives rise to :

$$\nabla^2\phi = -\frac{q}{\varepsilon_o \varepsilon_S} (p_o \, e^{-q\phi/kT} - n_o e^{q\phi/kT} + N_D - N_A) \qquad (10)$$

For a n-type semiconductor $n_o \approx N_D$ and $p_o \approx 0$, hence (10) can be simplified to :

$$\nabla^2\phi = -\frac{qN_D}{\varepsilon_o \varepsilon_S} (e^{q\phi/kT} - 1) \qquad (11)$$

It should be noted that $q/kT = 40$ at room temperature or $\exp(+q\phi/kT) = 4.2 \; 10^{-18}$ for $\phi = -1$ Volt. It is therefore reasonable to replace the right hand member by a step function which equals either 0 or $-qN_D/\varepsilon_o\varepsilon_S$ as soon as ϕ turns out to be negative. The region where the charge density equals $-qN_D/\varepsilon_o\varepsilon_S$ is called the depletion region. The boundary of the depletion region is not always known a priori. The abrupt depletion approximation will be used in section 5 to investigate the off state of a DMOS transistor. Some possible configurations of depletion regions are shown on fig. 2 and fig. 3.

3. The boundary element method.

In the abrupt depletion approximation, the Poisson's equation can be written as :

$$\nabla^2\phi = \frac{-\rho(x,y)}{\varepsilon_o\varepsilon_S} \tag{12}$$

where ρ is a known charge density (fig. 4) and constant in the depletion region and zero elsewhere. Using the equation for the Green's function $G(\bar{r}|\bar{r}') = \frac{1}{2\pi}\ln|\bar{r}-\bar{r}'|$:

$$\nabla^2 G(\bar{r}|\bar{r}') = \delta(\bar{r}-\bar{r}') \tag{13}$$

one gets after applying Green's theorem :

$$\oint_C \phi(\bar{r})\frac{\partial G(\bar{r}|\bar{r}')}{\partial n} - G(\bar{r}|\bar{r}')\frac{\partial\phi(\bar{r})}{\partial n}\,dC$$

$$= \phi(\bar{r}') + \frac{1}{\varepsilon_o\varepsilon_S}\iint_S \rho(\bar{r})G(\bar{r}|\bar{r}')dS \tag{14}$$

If ρ is piecewise constant (= qN_D e.g.) the right hand member of (14) can be calculated analytically for polygonal geometries [7].

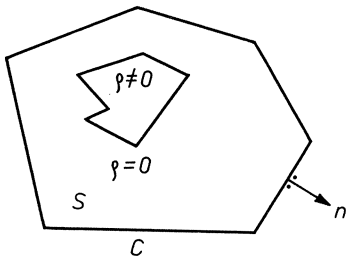


Figure 4.

Polygonal geometry used to outline the boundary element method.

By putting $\bar{r}'$ on the boundary C, (14) turns out to be an integral equation along C. If ϕ is given on a part of the boundary the normal derivative $\partial\phi/\partial n$ will be treated as the unknown function and vice versa on the remaining part of C where $\partial\phi/\partial n$ is given. To solve (14) numerically the boundary C is divided into N elements C_j :

$$\sum_{j=1}^{N}\int_{C_j}\left(\phi\frac{\partial G}{\partial n} - G\frac{\partial\phi}{\partial n}\right)dC = \phi(\bar{r}') + \frac{1}{\varepsilon_o\varepsilon_S}\iint\rho G\,dS \tag{15}$$

If N points $\bar{r}_i'$ are chosen on the elements $\{C_i\}$, (15) reduces to an algebraic set of N equation and N unknowns (either ϕ_j or $(\partial\phi/\partial n)_j$). It should be noted that all the coefficients such as $\int G(\bar{r}|\bar{r}_j)dC$ can be calculated analytically [7].

4. The on-state of the DMOS transistor.

In the "on" state the path AB (fig. 2) becomes a perfect conductor due to the generation of a thin inversion layer.

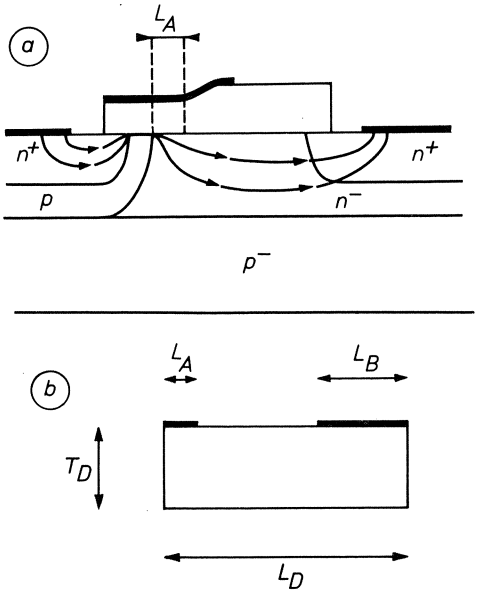


Figure 5.

Conducting paths for the electrons in the on-state.

Fig. 5a shows some typical current lines. It turns out that the current is only limited by the n-region. Hence we just have to calculate the resistance of this region, called the drift resistance. Once observes that (fig. 5a) that the gate electrode has an overlap L_A with the n-region. In order to calculate the drift resistance this overlap is considered as a perfect contact so that we can evaluate the approximate structure of fig. 5b.

Some results are shown on fig. 6, giving the resistance as a function of L_D for different values of the thickness T_D. The discrete points are calculated with the method outlined in section 3 (using $\rho = 0$) and the continuous lines are obtained by conformal mapping techniques. It is well known that the structure of fig. 5b

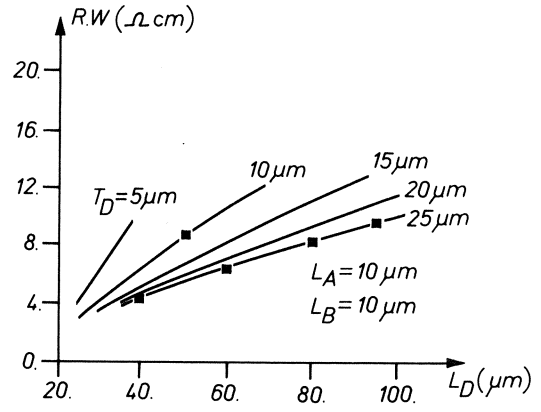


Figure 6.

Drift resistance of a DMOS transistor in the on state as a function of the channel length for various values of the thickness T_D.

can be mapped and that resistance values are invariant under the conformal transformation. Because the overlap L_A is much smaller than the drain contact it is also found that L_A has a major influence on the resistance values. Nevertheless, the calculated resistances were found to be acceptable for industrial applications.

5. The off state of the DMOS transistor.

The purpose is to design a transistor capable to withstand high voltages in the off state without any internal ionisation or avalanche effects. Ionisation occurs when the electric field exceeds a critical value. Mentioning the fact that the potential can be found by integrating the electric field from source to drain, the highest possible voltage will occur with a homogeneous electric field distribution, with the electric field strength just below the ionisation level.

For high voltages (typ. 400 Volt) it turns out that the n^- layer becomes now completely a depletion region as shown on fig. 7a.

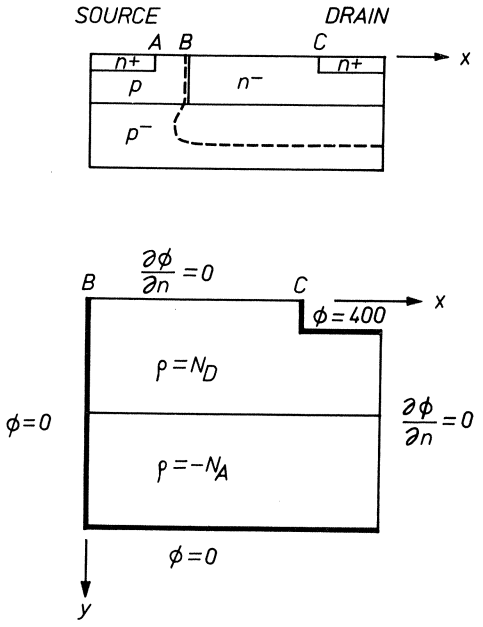


Figure 7.

Simplified geometry to analyse the off state.

This geometry is further simplified to the structure of fig. 7b where the depth is calculated according to the one dimensional solution of the Poisson's equation. The structure of fig. 7b can be calculated by the method outlined in section 3. Because the shortest distance between the ϕ= 0 and ϕ= 400 Volt lines coincide with the x-axis, it is expected the highest electric field will be found on it. Fig. 8 shows the influence of the thickness T_D on the behaviour of the electric field. One observes that T_D = 10 µm is optimal because the field distribution is the most homogeneous and never exceeds 24.10^4 V/cm. Similar graphs have been obtained describing the influence of substrate doping, etc... .

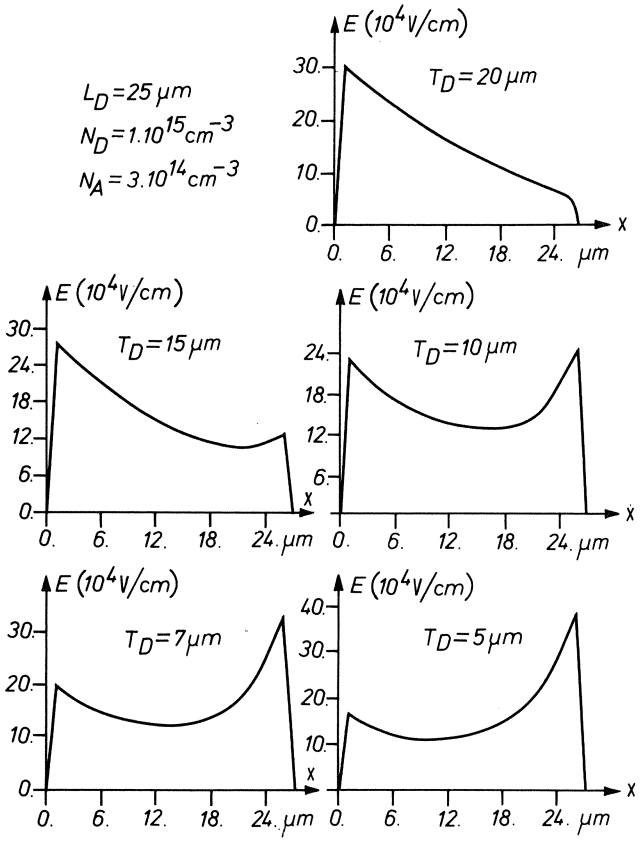


Figure 8.

Electric field strength along the oxide-semiconductor interface for various values of T_D.

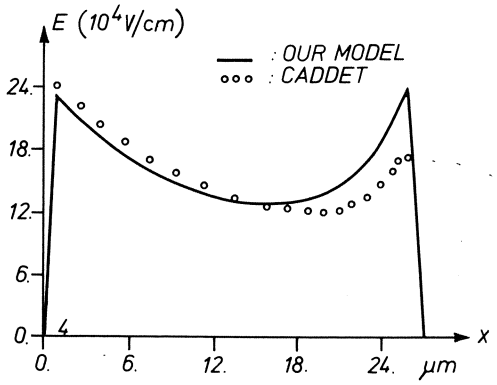


Figure 9.

Comparison with the CADDET results.

The results have also been compared with the program CADDET [5]. This program solves the non linear equations (1) - (5). Only the contribution of the hole current $\overline{J}_p$ is neglected. Fig. 9 shows a good agreement between both results. Note however that the computation time of CADDET was more than one hour whereas the method of section 3 requires only a few minutes. This may not be interpreted as a disapproval of the CADDET program because it can also be used to study transistors for other biasing conditions than the off state. CADDET also provides the (small) drain current, which is completely neglected in our model.

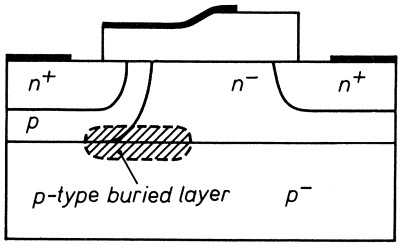


Figure 10.

DMOS transistor with buried layer.

In order to make the field behaviour more rectangular the idea of a buried layer was introduced (fig. 10). This is a bounded p-doped

region with a higher doping than the substrate. Due to the higher charge density when this buried layer becomes depleted it will influence the electric field. Fig. 11 shows a nice result and fig. 12 proves the agreement with CADDET. Nevertheless a three-dimensional plot (fig. 13) proves that the highest electric field occurs in the bulk and no longer along the surface as it was in the absence of a buried layer. Therefore the idea of buried layer has been dropped.

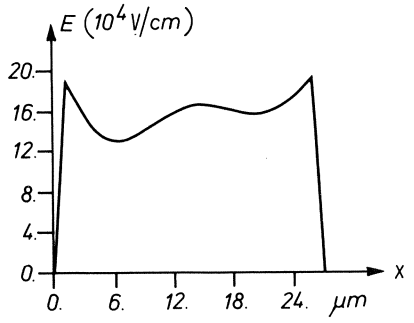


Figure 11.

Electric field strength along the oxide-semiconductor interface for a DMOS with buried layer.

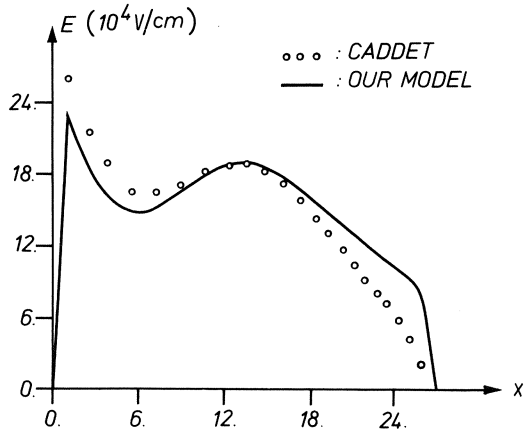


Figure 12.

Comparison with the CADDET results for buried layer.

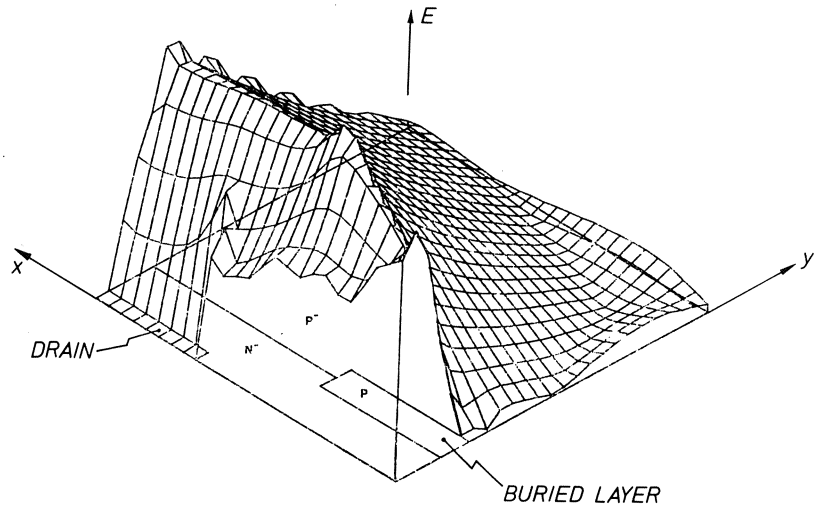


Figure 13.

3-D plot of electric field strength for buried layer DMOS.

6. Conclusion.

It has been demonstrated that a quite simple program based on the boundary element method to solve Poisson's equation has been succesfully used to optimise the parameters of a DMOS transistor. Both the on and off state could be analysed.
The results were verified with the numerical data obtained with the much more sophisticated CADDET program.

7. Acknowledgements

The authors wish to thank E. Palm from the Université Catholique de Louvain for running the CADDET programs. They are also gratefull to J. Danneels, G. Remmerie and L. Vandenbossche of Bell Telephone Antwerp for their continuous interest and practical hints.

References.

1) M.D. Pocha, A.G. Gonzalez and R.W. Dutton :
"Threshold voltage controllability in double diffused MOS transistors"
IEEE Transactions on Electron Devices, 1974, vol. ED-21, p. 778-784.

2) M.D. Pocha and R.W. Dutton :
"A computer aided design model for high voltage double diffused MOS transistors"
IEEE Journal of Solid State Circuits, 1976, vol. SC-11, p.718-726.

3) M.J. Declercq and J.D. Plummer :
"Avalanche breakdown in high voltage DMOS devices"
IEEE Transactions on Electron Devices, 1976, vol. ED-23, p. 1-5.

4) S. Selberherr :
"Zweidimensionale Simulation von MOS Transistoren"
Ph.D. Dissertation, Technische Universität Wien, Vienna, 1981.

5) "CADDET - Computer aided device design in two dimensions"
Internal document on the CADDET program, Hitachi Central Research Laboratory.

6) G. De Mey :
"Potential calculations in Hall Plates"
Advances in Electronics and Electron Physics, 1983, vol. 61, p. 1-62.

7) D. Loret :
"Studie van de hoogspanningsschakeltransistor DMOS"
Afstudeerwerk, Rijksuniversiteit Gent, Laboratorium voor Elektronika en Meettechniek, 1983.

BOX SCHEMES FOR THE SEMICONDUCTOR CONTINUITY EQUATION

S. POLAK, W. SCHILDERS, A. WACHTERS

Abstract.

Box schemes are often used for the semiconductor equations. The usual box schemes suffer from two disadvantages. The boxes must satisfy cumbersome geometrical requirements and in the case of the singularly perturbed semiconductor equations they do not sufficiently reduce to a stable integrator for the reduced (characteristics) problem. In this paper we briefly discuss the disadvantages of the classical scheme and we present a class of box schemes that do not have any geometrical problems and give the possibility to adapt better to the singularly perturbed character of the semiconductor problem.

1. INTRODUCTION

Box schemes are difference schemes obtained by applying a Green's theorem transforming a surface integral into a loop integral. This scheme is described in section 2 and a new scheme of this type is proposed in section 4.
In the mathematical literature we have found very little on such schemes. In this paper we are not filling that gap. We are just proposing a new scheme together with a heuristic reasoning about the merits of this scheme.
In the engineering literature many instances of the application of box schemes can be found. Especially in device modelling this is the most used discretisation technique.
In section 2 we discuss the standard box scheme used in device modelling. We also describe some disadvantages of this scheme.
In section 3 the singularly perturbed nature of the continuity equation is investigated, in section 4 the new class is described and in section 5 it is applied to the continuity equation.

2. THE CLASSICAL BOX SCHEME

Let us consider the equation

(2.1) $\Delta u = f$ in Ω
and $u = u$ on $\delta\Omega = \Gamma$

Suppose a mesh given in Ω, with a meshpoint x_0 and several meshlines coinciding in x_0 as shown in fig. 1

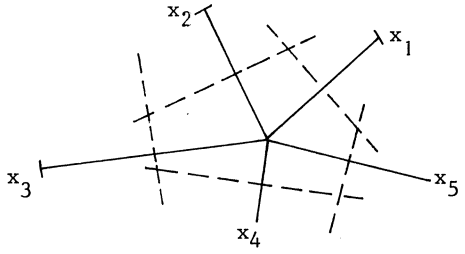


Fig. 1

Then a box is constructed around x_0 by using the midperpendiculars of x_0x_i. Their intersections are the vertices of a polygon which is called a box B with boundary B. Using Green's theorem now gives instead of (2.1)

$$\oint_{\partial B} \nabla u.dn = \iint_B f \, do$$

Then $u.dn$ is approximated by

$$\nabla u.dn = \frac{u(x_i)-u(x_0)}{|x_i-x_0|}$$

This way a difference scheme is constructed on a nonrectangular grid. The equation in x_0 has the form

$$\sum_{i=1}^{6} \frac{u(x_i)-u(x_0)}{|x_i-x_{i+1}|} \, |s_i-s_{i+1}|$$

For the continuity equation (see section 3) and triangular meshes this problem is treated in, e.g., [1] The same author needs the following geometrical condition for a convergence proof. The meshes must have a circumscribed circle with midpoint inside the mesh (see [2])
Also in practice it is well known that both obtuse triangles and arbitrary quadrilaterals can give problems. We have one particular example where we are solving a diode problem on a distorted quadrilateral mesh giving essentially erroneous results due to the distortion (see section 6).

3. THE CONTINUITY EQUATION

In this paper we shall concentrate on the use of box schemes for one of the equations [3] used in device modelling. This equation has the form

$$\nabla.(\nabla p+\nabla\psi p)=R \tag{3.1}$$

where p is the unknown and ψ supposed to be known in this paper. It should be noted here that R also is a function of p and possibly ψ that the function p varies enormously and that all the coefficients that depend on the unknowns have been set to one for the sake of this discussion. It is always combined with both Dirichlet and Neumann boundary conditions in the same problem. In this section we consider the discretisation of (3.1) without special attention to these boundary conditions. For practical purposes other forms of this equation are used but for this discussion (3.1) is simplest and essentially the same as the others.

Let us first consider the one dimensional case. Then the equation is

$$\left(\frac{d}{dx}\left(\frac{dp}{dx}+ap\right)\right)=R \tag{3.1'}$$

where $a=d\psi/dx$ with of course two boundary conditions. The function a varies $(0-10^5)$ in x. Where a is large (3.1') is singularly perturbed. In the limit it reduces to the ODE

$$\frac{d}{dx}(ap)=R \tag{3.2}$$

However for (3.1') we have two boundary conditions whereas (3.2) is an initial value problem. So for large a we almost have one boundary condition to many. This is found back as a transition layer in the solution. This can also be understood by considering the solution of (3.1') in the form

$$\alpha+\beta e^{-ax}+f(x) \tag{3.3}$$

where f(x) is a solution of the inhomogeneous problem not satisfying the boundary condition. For large a, $e^{-ax}\neq 0$ only in a very short interval. The difference scheme for (3.1) must be adapted to the fact that we are almost solving (3.2) for large a. If for instance we use

$$\delta_h(\delta_h+h\mu a)p=h^2R \tag{3.4}$$

for (3.1') (for definition of the operators δ_h and μ see e.g. [4]) we find

$$\delta_h h\mu ap=h^2R$$

for (3.2) it is obvious that this is not a stable ODE integrator. This leads to:

requirement (A): the difference scheme for (3.1') must reduce to a stable scheme for (3.2) if $a \to \infty$.

In [5] a scheme is derived for (3.1) which satisfies requirement (A). This scheme is essentially the same as the "Gummel" scheme [6] used in device modelling and derived in a totally different, more physics inspired way.
To understand this scheme we may reason as follows. The operator should be replaced by an operator resulting in backward Euler for (3.2). However this replacement depends on a. Let us define

$$M(a) = .5[(1+s(a))E_{\frac{1}{2}} + (1-s(a))E_{-\frac{1}{2}}]$$

Replacing μ by $M(a)$ and using the fact that $M(a) = \mu + .5s(a)\delta_h$ we find

$$\delta_h((1+.5\ s(a)h)\ \delta_h + h\mu a)p = R \tag{3.5}$$

We still have to find a suitable function for $s(a)$. For this purpose we use the fact that the transition layer is represented by the solution e^{-ax} of the homogeneous problem. We substitute this, writing $\gamma = 1+.5\ s(a)h$ in

$$\delta_h(\gamma\delta_h + h\mu a)p = 0 \tag{3.6}$$

giving $\gamma(a) = .5ah\ \mathrm{cotgh}(.5ah)$
The important observation is that we used the fact that the homogeneous scheme has two basic solutions, 1 and e^{-ax} in an essential way. This cannot be extended to two dimensions. However in two dimensions the reduced equation is

$$\nabla.((\nabla\psi)p) = R \tag{3.7}$$

This is a first order equation and therefore can be considered as a set of ODE's along the characteristics. This leads to:

requirement(B): the discrete scheme for the ODE's along the characteristics defined by (3.7) as $\|\nabla\psi\| \to \infty$.

In the following we briefly look at different schemes proposed in the literature for this problem in the light of requirement(B). All the investigated schemes satisfy (B) if $\nabla\psi$ is parallel to the x or the y axis. They all reduce to backward Euler along the characteristics which is a line parallel to the axis. However if a=(1,1) the situation is rather different.

For this problem we calculated the coefficients for one equation for the methods proposed in [6] - [8] for a square mesh. The coupling pattern is given in fig. 2

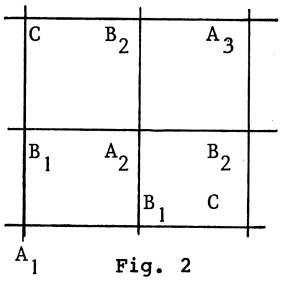


Fig. 2

The coefficients are now given as follows.

C	B_1	B_2	A_1	A_2	A_3	
1	-3	-1	-10	16	0	for Mitchell et al. [7]
1	-12	8	-10	16	0	for Brookes et al. [8]
0	1	1	0	-2	0	for Gummel (section 5)

It is easily understood that these schemes "spread" the solution perpendicular to the characteristics, a phenomenon also called cross wind diffusion. A pattern of the form

O	B	-B	O	A	-A

would have essentially less cross wind diffusion.
The new scheme proposed in section 4 gives exactly that.
It can be noticed e.g. that the B_1=8, B_2=-10 are closer to this then the B_1=-1, B_2=-10 which is symptomatic for the improvement claimed by the authors of [8].

4. A NEW CLASS OF BOX SCHEMES

In section 2 we have shown how classically a box scheme is constructed using midperpendiculars. In this section we generalise the box scheme idea.
Suppose we have a grid M consisting of points, straight segments connnecting neighbouring points and meshes as usual. No geometrical restrictions are made. Now a box point is chosen inside each mesh and a box point is chosen inside each mesh segment. Neither should coincide with a mesh point. By connecting each mesh box point with each of the segments box points in the same mesh a box mesh M* is obtained. Fig. 3 shows a very arbitrary example

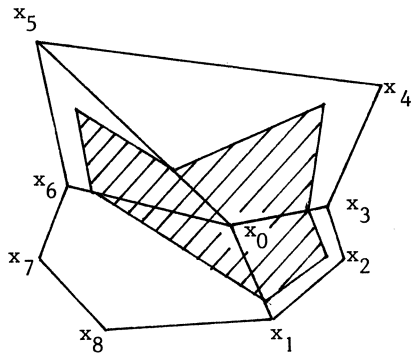


Fig. 3

We do not discuss the righthand side of (4.1) any further in this paper but concentrate on the left hand side.
For the discretisation of $\oint \nabla u.dn$ we first approximate u in terms of nodal values.
This defines $\nabla u.dn$ in each nonvertex point of the octogon in a unique way. So we now may choose some quadrature giving some difference scheme.
Summarising we may say that there are four steps in this discretisation:

- Choice of a mesh
- Choice of a box mesh
- Approximation of u in terms of nodal values
- Choice of quadrature

We might do this for instance with a triangular mesh by taking the baricentres and the segment midpoints.
In practice we use a quadrilateral mesh using baricentres and segment midpoints as shown in fig. 4 giving octoganal boxes.

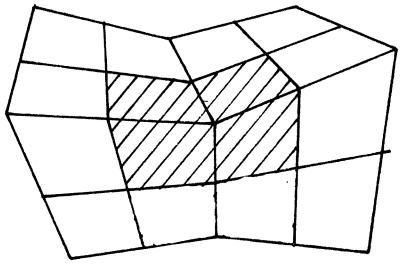


Fig. 4

A box again is used with a Green's theorem giving

(4.1) $$\oint \nabla u.dn = \iint f \, do$$
octogon

Now we approximate u in each quadrilateral by an isoparamatric bilinear function as is usual in FEM. A quadrature for the left hand side then transforms (4.1) into a nine point scheme.

5. A NEW BOX SCHEME FOR THE CONTINUITY EQUATION

Let us reconsider the continuity equation. In the classical box scheme, as used for the continuity equations in device modeling, the one dimensional reasoning is simply transferred to the segments x_0x_i as defined in section 2.
So we have to approximate

(5.1) $$\oint (\nabla p+(\nabla\psi)p).dn$$

we use $(\nabla p+(\nabla\psi)p).dn \cong \gamma\delta_h p+\mu((\delta_h\psi)p)$ and $\gamma = .5\delta_h\psi \coth(.5h\delta_h\psi)$

with the operators δ_h and μ applied on the segment x_0x_i giving

$$\sum_{i=1}^{5} [\delta_h(\gamma\delta_h p+|x_i-x_0|\psi\mu p)(t_i)]\,|s_{i+1}-s_i|$$

In section 3 we already saw that this results in a scheme with spurious cross wind diffusion perpendicular to the characteristics. The schemes presented in section 4 offer the possibility to introduce the factor γ only in the direction of the characteristic in a quadrature point.
For this purpose we choose a local coordinate system (x',y') in each quadrature point chosen for the evaluation of (5.1). The x' direction is tangential to the characteristics. So along $\nabla\psi$. Then (5.1) becomes

(5.2) $$\oint [(p_{x'},p_{y'})+(\psi_x p,0)].dn$$

As an approximation for this we use

$$\oint (\gamma\bar{p}_{x'},\bar{p}_{y'})+(\bar{\psi}_x\bar{p},0).dn$$

where the bar indicates using isoparametric bilinear quadrilaterals as in FEM.
Still the γ has only one dimensional meaning!

$$\gamma = .5h\bar{\psi}_x \coth(.5h\,\bar{\psi}_{x'})$$

and the remaining problem is the meaning of h. We take h, assuming the characteristic a straight line, the length of the segment along the x' axis inside the mesh (see fig. 5)

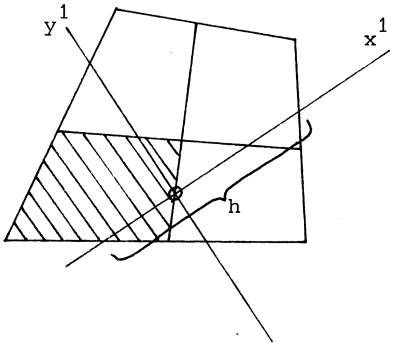


Fig. 5

It is probably very difficult to escape the inconsistency of a one dimensional and a two dimensional reasoning in the construction of these schemes. The essence of the problem is that the reduced two dimensional problem must be integrated in a stable way along the one dimensional characteristics.
Summarising we so far have chosen:

- a quadrilateral mesh
- octogonal boxes with the help of the mesh baricentres and segment midpoints.
- a bilinear isoparametric approximation per mesh.

and we have introduced a local coordinate system and a "fitting factor" γ in the direction of the field $\nabla\psi$.

Now we still have to choose a quadrature to approximate the loop integral. We choose eight quadrature points, the midpoints of the sides of the octogon and presume the integrant in (5.2) (constant along that side). This gives a nine point scheme in general.
For the reduced equation on a square mesh, as in section 3 we find the coefficients

C	B_1	B_2	A_1	A_2	A_3
0	-1	1	0	-2	2

In general, taking one quadrature point on a box segment dividing the segment with a ratio θ, $\theta=0$ on the vertex inside the mesh and 1 on the vertex on the mesh segment, we find the coefficients

$$0 \quad -\theta \quad \theta \quad 0 \quad 2\theta-4 \quad 4-2\theta$$

so for $\theta=0$ we find

$$0 \quad 0 \quad 0 \quad 0 \quad 2\theta \quad -2\theta$$

which means exactly backward Euler along the 45° characteristics. However for the pure Laplace equation this choice of quadrature might be less fortunate because a checker board scheme results. We do not have much practical experience to judge the importance of this phenomenon.

It is obvious that for any direction of $\nabla\psi$ the so constructed difference scheme is a sum of backward Euler type schemes along the characteristics through quadrature points for the reduced equation. So for one equation there is no cross wind diffusion. However neighbouring equations do not use the same characteristics, except for very special cases such as $\theta = 0$ and $\nabla\psi = (1,1)$.

6. EXAMPLE

In section 2 we remarked that, in practice, obfuse triangles and arbitrary quadrilaterals can give rise to essentially erroneaous results. To show this, we consider the following diode problem, which was taken from [9]: ($\mu = 10^{-4}$cm)

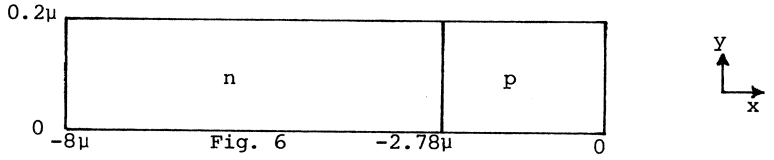


Fig. 6

The equations we have to solve are

$$-\mathrm{div}(\varepsilon\,\mathrm{grad}\psi) = q(p - n + D(x)) \tag{6.1}$$

$$\mathrm{div}\left(\frac{q\mu_p}{\alpha}\,\mathrm{grad}p + q\mu_p p\left[\mathrm{grad}\psi - \frac{1}{\alpha n_i}\,\mathrm{grad}n_i\right]\right) = qR \tag{6.2}$$

$$\mathrm{div}\left(\frac{q\mu_n}{\alpha}\,\mathrm{grad}n - q\mu_n n\left[\mathrm{grad}\psi + \frac{1}{\alpha n_i}\,\mathrm{grad}n_i\right]\right) = qR \tag{6.3}$$

with boundary conditions:

$$\psi(x=0) = -\frac{1}{\alpha}\log\left(\frac{-D(0)}{n_i}\right)$$

$$p(x=0) = -D(0)$$

$$n(x=0) = -n_i^2/D(0)$$

$$\psi(x=-8\mu) = V_A + \frac{1}{\alpha}\log\frac{D(-8\mu)}{n_i}$$

$$n(x=-8\mu) = D(-8\mu)$$

$$p(x=-8\mu) = n_i^2/D(-8\mu)$$

We have Neumann boundary conditions on the lines y=0 and y=0.2μ. The functions/constants appearing in (6.1)-(6.3) are given by:

$$\varepsilon = 11.7*\varepsilon_0 \; ; \; \varepsilon_0 = 8.854*10^{-14}$$

$$q = 1.6021*10^{-19}$$

$$D(x) = 6*10^{15} - 2.15*10^{18}*\exp\{-\left(\frac{x}{1.15*10^{-4}}\right)^2\}$$

$$+1.19*10^{19}*\exp\{-\left(\frac{x+8*10^{-4}}{1.3*10^{-4}}\right)^2\} \quad (cm^{-3};\ x \text{ in cm})$$

$$n_i = 1.22*10^{10}$$

$$\mu_p = 500$$

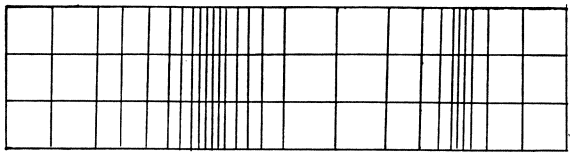
$$\mu_n = 500$$

$$R = 0$$

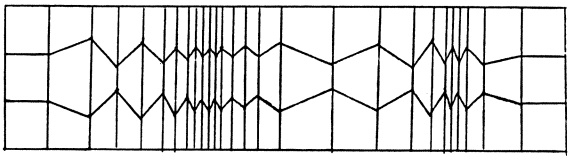
$$\alpha = q/KT;\ k = 1.38*10^{-23};\ T = 300$$

The constant V_A (applied voltage) in the boundary condition for ψ at $x=-8\mu$ is variable and was taken to be 0,5 and 20 respectively.

The above problem was solved on a rectangular mesh (to give the reference solution) and on a distorted mesh (see fig. 7).



(a) rectangular mesh



(b) distorted mesh

Fig. 7

In figure 8 we show the results of our calculations on a severely distorted mesh. We see that the solution on the distorted mesh differs significantly from the solution on the rectangular mesh. Physically, this is not acceptable (cf. [9]; the figures were also taken from this reference).

If we use the box scheme described in sections 4 and 5, however, we obtain an accuracy of two decimal places for the solution on the distorted mesh.

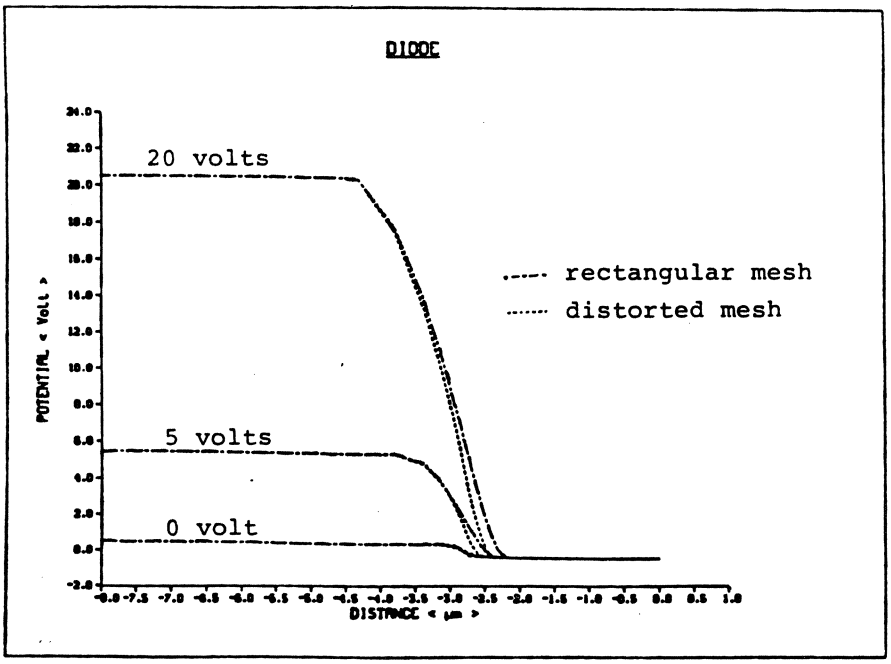


Fig. 8

REFERENCES

[1] M.S. Mock, "Analysis of a descretization algorithm for stationary continuity equations in semiconductor device models", COMPEL, vol. 2, no. 3 (1983).

[2] M.S. Mock, "Quadrilateral elements and the Scharfetter-Gummel method", in Proc. Int. Conf. on Simulation of Semiconductor Devices and Processes, K. Board, D.R.J. Owen (eds.), Pineridge Press, Swansea (1984).

[3] S. Polak, A. Wachters, H. Vaes, A. de Beer, C. den Heijer, "An algorithm for the calculation of Poisson's equation in 2-D semiconductor problems", Int. J. for Num. Meth. in Eng., vol. 17, no. 11 (1981).

[4] F.B. Hildebrand, "Introduction to numerical analysis", McGraw-Hill, 1956.

[5] A.M. Il'in, "A differencing scheme for a differential equation with a small parameter affecting the highest derivative", Math. Notes Acad. Sc. USSR, 6 (1969).

[6] J.W. Slotboom, "Analysis of bipolar transistors", Ph. D. Thesis Technical University of Eindhoven, (1977).

[7] A.R. Mitchell, D.F. Griffiths, A. Meiring, "Finite element Galerkin methods for convection-diffusion and reaction-diffusion", in Proc. of the Conference on Analytical and Numerical Approaches to Asymptotic Problems, Nijmegen, O. Axelsson, L.S. Frank, A. v.d. Sluis (eds), North Holland Math. Studies, vol. 47 (1981).

[8] A.N. Brooks, T.J.R. Hughes, "Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations", Comp. Meth. in Appl. Mech. and Eng., 32 (1982).

[9] D. Ming Der Huang, P.R.L. Sunnyvale, PRLS Trip Report 83-009 (1983).

ON THE NUMERICAL ANALYSIS OF WATER LUBRICATION IN OIL PIPELINES

N. PRAAGMAN, A. SEGAL

1. INTRODUCTION

In order to transmit a given quantity of very viscous oil through a long pipeline a large pressure gradient is required. This is mainly due to the friction with the pipe wall. For that reason investigations have been made to determine in which way this friction and with it the power required to transmit oil can be decreased. These investigations led to the conclusion that there is an advantage in adding a less viscous liquid for instance water, provided that the two liquids are immiscible. Experiments show that the less viscous fluid wets the complete inner pipe wall if the velocity of the flow is large enough. (See figure 1 and [5]). As a result the friction experienced

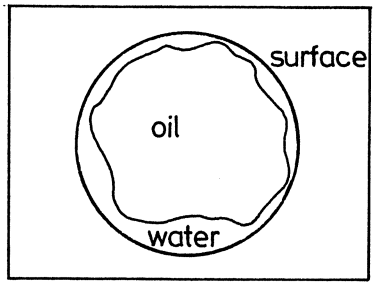


Figure 1. Oil-water pattern in a pipe if the flow velocity is large enough.

by the oil is considerably smaller and although power is needed to drive the added liquid, the total power requirement diminishes.

If, in the case of an oil water mixture, the velocity of the flow becomes too slow the oil touches the top of the pipe and hence the oil will experience friction from the wall. (See figure 2.) Especially during the start of the transport such a situation is encountered.

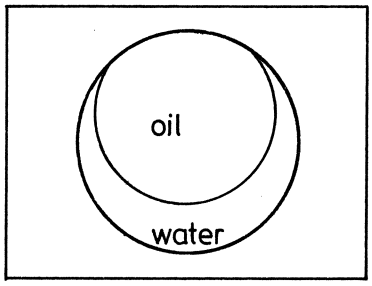


Figure 2. Oil-water pattern in a pipe if the flow velocity is low.

It is possible to compute the position of the interface in such a situation utilizing a mathematical formulation. This formulation is obtained by requiring that the potential energy has a minimum. In order to gain insight in the power requirements for that minimum the two-phase axial laminar flow pattern in the pipe has to be computed. This flow pattern can be used to compute numerical values for the power reduction factor and this can be done for several oil/water ratios.

In section 4 of this paper results of computations to obtain these values are given for some practical examples. First, in section 2, the mathematical formulations describing

(i) the position of the interface
(ii) the axial flow field
(iii) the power reduction

are given, while in section 3 the numerical methods are treated. Section 5 contains the concluding remarks.

The basic ideas for the mathematical description of the problem at hand are given in [1]. The numerical techniques for solving the mathematical problems can be found in [2], [3] and [4].

2. THE MATHEMATICAL FORMULATION

In order to obtain the mathematical formulations the following assumptions have to be made:

- the flow is laminar
- the fluids move only in the axial direction of the pipe
- the contact angle γ (see figure 3) between the inner pipe wall and the interface is an a priori known constant
- both fluids are incompressible

2.1. The interface oil-water

To determine the position of the interface the expression for the potential energy V has to be analyzed. In V the terms representing gravity, interface tension and surface friction are important. Using calculus of variations the following conditions under which V is minimized are obtained, (see figure 3)

(i) if $\beta + \gamma \geq \frac{\pi}{2}$,

the curve y(x) giving the position of the interface has to satisfy:

(1) $$\frac{dy}{dx} = -\sqrt{\frac{1}{f^2} - 1}$$

with

(2) $$y(-\sin\beta) = \cos\beta$$

and

(3) $$f(y) = -\tfrac{1}{2}Ay^2 + \gamma y - \cos(\beta+\gamma)$$

(ii) if $\beta + \gamma < \frac{\pi}{2}$, then

$$\frac{dy}{dx} = \text{isign}\sqrt{\frac{1}{f^2} - 1}$$

$$y(-\sin\beta) = \cos\beta, \text{ f as above.}$$

The value of isign depends on the situation treated. From the starting point P to the turning point T_p the value is 1, while from T_p to the symmetry point S_p isign = -1.

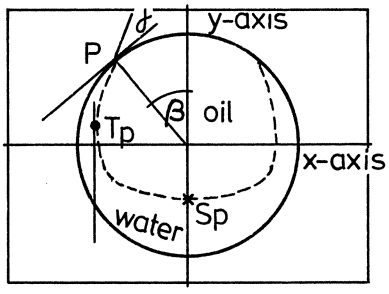


Figure 3. Definition sketch of β, γ, P, T_p and S_p.

- β determines the intersection point P of the pipe wall and the interface.
- γ is the angle between the tangent in P to the wall and the tangent in P to the interface.
- T_p is the point of the interface where the tangent is parallel to the y-axis.
- S_p is the point of the interface where the tangent is parallel to the x-axis.

The variable A in (3) is the so-called stratification parameter. A is dependent on the densities ρ_1 of the oil and ρ_2 of the water, the acceleration of gravity g, the radius of the pipe a and the interface tension T. A is computed using the formula

$$A = \frac{(\rho_2-\rho_1)ga^2}{T} \tag{4}$$

The variable λ in (3) is a parameter which has to be determined in such a way that the property of symmetry

$$\frac{dy}{dx}(x=0) = 0 \tag{5}$$

holds for the curve y(x).

2.2. The axial flow field.

For the given assumptions the flow field is described by the following partial differential equations (normalized for a pipe with radius 1, and a pressure gradient 1)

$$-\text{div}\,(\mu_{oil}\,\text{grad}\omega) = 1, \text{ section I, i.e. oil} \tag{6}$$

$$-\text{div}\,(\mu_{water}\,\text{grad}\omega) = 1, \text{ section II, i.e. water} \tag{7}$$

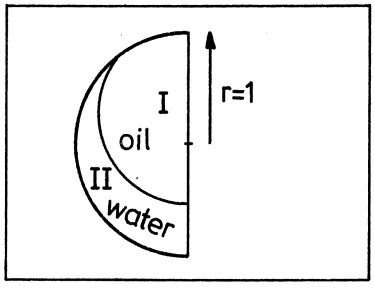


Figure 4. The domains I and II of the PDEs (6) and (7).

In these equations μ_{oil} and μ_{water} are the (constant) viscosities of the two liquids.

The boundary conditions for ω are:

(8) - on the pipe wall: $\omega = 0$ on $\partial\Omega_1$

(9) - on the symmetry axis: $\frac{\partial\omega}{\partial n} = 0$ on $\partial\Omega_2$

(10) - on the interface: ω is continuous on I

(11) - on the interface: $\mu_{oil} \left.\frac{\partial\omega}{\partial n}\right|_{oil} = \mu_{water} \left.\frac{\partial\omega}{\partial n}\right|_{water}$ on I.

For the definition of the domains and boundaries see figure 5.

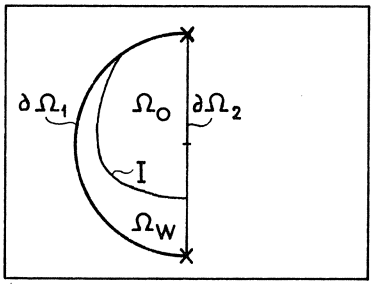


Figure 5. Definition of $\Omega_0, \Omega_\omega, \partial\Omega_1, \partial\Omega_2$ and interface I. ($\Omega = \Omega_0 \cup \Omega_\omega$.)

2.3. The power reduction factor.

The non-dimensional volumetric flow rate, if the pipe is filled with oil only, reads

$$\int_\Omega \omega dxdy = 2\pi \int_0^1 \tfrac{1}{4}(1-r^2)rdr = \frac{\pi}{8}.$$

Hence the normalized volumetric flow rates of oil and water are:

(12) $$Q_{oil} = \frac{8}{\pi} \int_{\Omega_{oil}} \omega dxdy$$

(13) $$Q_{water} = \frac{8}{\pi} \int_{\Omega_{water}} \omega dxdy$$

Since ω is proportional to the pressure gradient (eq. (6) and (7)) it follows that Q_{oil} is the ratio of the pressure gradients required to

transmit a given volumetric flow rate of oil without and with addition of water.

The reduction in the required pumping power is given by the factor

$$F_p = \frac{Q_{oil}{}^{*}Q_{oil}}{Q_{water}{}^{+}Q_{oil}}, \tag{14}$$

due to the proportionality of ω and the pressure gradient. (see [1]).

3. METHOD OF SOLUTION

For the solution of the three subproblems of section 2 numerical techniques have been used. These will be treated in the following subsections.

3.1. Determination of the interface.

Given the values of the angles β and γ and the ordinary differential equation (ODE) for the position $y(x)$ of the interface, an approximation of $y(x)$ is computed using the Runge-Kutta-Fehlberg method (see [3]) of second order. The formulae of this method read, in case of the standard ODE,

$$\dot{y} = F(y,t)$$

$$\begin{cases} y^{*}_{n+1} = y_n + h_n F(y_n,t_n) \\ y_{n+1} = y_n + \frac{h_n}{2}\{F(y_n,t_n) + F(y^{*}_{n+1},t_{n+1})\} \\ t_{n+1} = t_n + h_n \end{cases} \tag{15}$$

The computed approximation y_{n+1} of $y(t_{n+1})$ is accepted or rejected, depending on the local error:

$$\ell e_{n+1} = \| y_{n+1} - y^{*}_{n+1} \|. \tag{16}$$

A new stepsize, either to repeat the rejected step or to start a new step at t_{n+1}, is computed using the formula

$$h = 0.9 * h_n \sqrt{\frac{TOL}{\ell e_{n+1}}} \tag{17}$$

TOL is the required accuracy per unit step. Since the local curvature of the interface, especially near a turning point, changes rapidly, a rather high accuracy is required. The interface is computed for several values of λ. The value of λ is adjusted with respect to (5), using a bisection method.

3.2. Computation of the axial flow field.

The partial differential equations (6) and (7) with the boundary conditions (8), (9), (10) and (11) are easily solved using a standard Galerkin finite element approach.

First the problem is rewritten using a variational formulation (see [4], [2]):

Find $\omega \in H_E(\Omega)$ such that

$$(18)\qquad \begin{cases} \iint_\Omega \mu(\text{grad}\omega.\text{grad}\phi)dxdy = \iint_\Omega \phi dxdy, \ \forall\phi \in H_E(\Omega) \\ \text{with} \\ H_E(\Omega) = \{\nu \mid \nu \text{ is continuous on } \Omega,\ \nu|_{\partial\Omega_1}=0\} \end{cases}$$

and

$$\mu = \begin{cases} \mu_{oil} & \text{in } \Omega_0 \\ \mu_{water} & \text{in } \Omega_\omega \end{cases}$$

Problem (18) is discretized using the Finite Element Method. (F.E.M.). The domain Ω is divided in triangular elements e_j (see figure 6) and an approximating space $V_E(\Omega)$ of $H_E(\Omega)$ is defined:

$$V_E(\Omega) = \{\nu \mid \text{(i) -The restriction of } \nu \text{ to an element } e_j \text{ is a linear polynomial}$$
$$\text{(ii) -}\nu \text{ is continuous over } \Omega$$
$$\text{(iii)-}\nu|_{\partial\Omega_1}=0\}$$

Now the discretized version of (18) reads:

$$(19)\qquad \begin{cases} \text{Find } \nu^h \in V_E(\Omega) \text{ such that} \\ \iint_\Omega \mu(\text{grad}\nu^h.\text{grad}\phi^h)dxdy = \iint_\Omega \phi^h dxdy, \quad \forall\phi^h \in V_E(\Omega) \end{cases}$$

Observe that (9) and (11) are natural boundary conditions that have no impact on the functions of $V_E(\Omega)$.

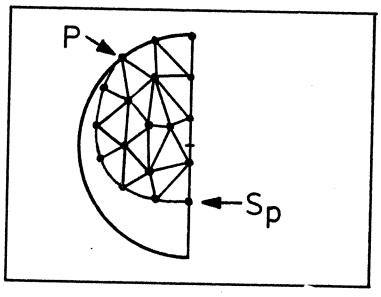


Figure 6. Triangularisation of the oil-domain Ω_0.

Problem (19) is transformed into a system of linear equations by the introduction of piecewise linear basisfunctions ψ_i satisfying

$$\psi_i(x_j,y_j) = \delta_{ij} \tag{20}$$

where j is the jth nodal point of the mesh. The approximating solution $\nu^h \in V_E$ then can be written as:

$$\nu^h(x,y) = \sum_{j=1}^{N} \nu_j \psi_j(x,y). \tag{21}$$

Substitution of $\phi^h = \psi_i$, (i=1,2,...,N) in (19) yields the following system of N linear equations in the unknowns $\nu_1, \nu_2, \ldots, \nu_N$:

$$\sum_{j=1}^{N} \nu_j \iint_{\Omega} \mu\left(\frac{\partial\psi_j}{\partial x}\frac{\partial\psi_i}{\partial x} + \frac{\partial\psi_j}{\partial y}\frac{\partial\psi_i}{\partial y}\right)dxdy = \iint_{\Omega} \psi_i dxdy, \quad (i=1,2,\ldots,N). \tag{22}$$

The resulting system of equations can be solved either with a direct method (Gaussian elimination) or an iterative solver. (For example a preconditioned conjugate gradient method). Because of the few equations that were needed to solve this problem with the accuracy required, about 100, a Cholesky factorization technique is prefered.

3.3. Calculation of the power reduction factor F_p.

In order to obtain quantitative results for F_p, equations (12) and (13) have been solved numerically. For the approximation of the integrals

the two dimensional trapezoïd rule has been used:

$$(23) \qquad \iint_{\Omega} \omega dxdy = \sum_{i=1}^{M} \iint_{e_i} \omega dxdy = \sum_{i=1}^{M} \tfrac{1}{3} O_i (\omega_{i_1} + \omega_{i_2} + \omega_{i_3})$$

with M the number of elements e_i in Ω, O_i the area of element e_i, and $\omega_{i_1}, \omega_{i_2}$ and ω_{i_3} the computed values of the axial velocity ω in the three nodal points i_1, i_2 and i_3 of element e_i.

4. RESULTS OF COMPUTATIONS

In this section results are given for three test examples. In order to determine the position of the interface with sufficient accuracy a value of TOL = 10^{-3} has been used, and λ has been computed with an accuracy of 10^{-5}. With these values the coordinates of the points of the computed interface are correct to three decimal places. The triangularization of Ω is also a parameter which can influence the results. Therefore several refinements have been computed. It turned out that with a coarse mesh as given in figure 6 the relative error in F_p is already less than 2%.

Table 1. Power reduction factor for several angles β and waterfractions.

Stratification parameter	- :	10.00
Oil viscosity	cp :	6.00
Water viscosity	cp :	1.00
Interface tension	N/M :	0.02
Contact angle γ	degree:	30.00

Water fraction	β	Power reduction
0.74	15^o	0.00
0.61	30^o	0.01
0.49	45^o	0.05
0.37	60^o	0.13
0.28	75^o	0.34
0.20	90^o	0.63
0.12	105^o	0.93
0.07	120^o	1.10
0.03	135^o	1.13

Table 2. Power reduction factor in case that the stratification parameter is 1, and all other variables are the same as in table 1.

Water fraction	β	Power reduction
0.73	10^{o}	0.00
0.66	15^{o}	0.01
0.59	20^{o}	0.04
0.50	30^{o}	0.10
0.37	45^{o}	0.30
0.27	60^{o}	0.57
0.20	75^{o}	0.82
0.14	90^{o}	1.00
0.09	105^{o}	1.18
0.06	120^{o}	1.18
0.03	135^{o}	1.15

Table 3. Power reduction factor, as in table 2 however the contact angle is 5^{o}.

Water fraction	β	Power reduction
0.52	15^{o}	0.11
0.34	30^{o}	0.55
0.24	45^{o}	0.90
0.16	60^{o}	1.22
0.10	75^{o}	1.41
0.06	90^{o}	1.38
0.04	105^{o}	1.30
0.02	120^{o}	1.18
0.01	135^{o}	1.17

The tabulated results show that there is an optimal choice indeed for the water-oil ratio in order to obtain the best possible power reduction factor. Comparison of tables 2 and 3 makes clear that the contact angle γ plays an important role. Comparison of tables 1 and 2 shows that the influence

of the stratification parameter A is less significant.

5. CONCLUSIONS

A mathematical model for the transport of oil has been treated. It has been shown that a quantitave solution for the power reduction factor can be obtained using a combination of the following numerical techniques:

- a Runga-Kutta-Fehlberg ODE integration technique
- the bisection method
- the Galerkin F.E.M.
- the Conjugate Gradient method or Gaussian elimination
- a numerical quadrature method.

6. REFERENCES

[1] BENTWICH. M., *Two-Phase axial laminar flow in a pipe with a naturally curved interface.* Chemical Engineering Science, 1976, Vol. 31, pp. 71-76.

[2] KAN. J.v., N. PRAAGMAN & A. SEGAL, *Numerical Analysis* CII/BIII *College Text Book*, Delft, University of Technology, 1982.

[3] LAMBERT, J.D., *Computational Methods in Ordinary Differential Equations*, John Wiley, London, 1973.

[4] MITCHELL, A.R. & R. WAIT, *The finite element method in partial differential equations*, John Wiley, Chichester, 1977.

[5] OOMS, G., A. SEGAL, A.J. v/d WEES, R. MEERHOFF & R.V.A. OLIEMANS, *A theoretical model for core-annular flow of a very viscous oil core and a water annulus through a horizontal pipe*, Journal of multi- phase flow, Vol. 10, no. 1, pp. 41-60, 1984.

REMARKS ABOUT A COMPUTATIONAL METHOD FOR SHALLOW WATER EQUATIONS THAT WORKS IN PRACTICE

G.S. STELLING, J.B.T.M. WILLEMSE

1. INTRODUCTION

In this contribution we will describe a numerical model for the simulation of flow in shallow seas, estuaries and rivers. Simulation models of this kind are in use quite extensively for civil engineering and water management problems. Examples will be given in the last chapter of this contribution. A simulation model involves various aspects. This work describes a number of important matters. Each subject however is treated very briefly. Details are given by Stelling [1]. The model will be described step by step. Therefore the second chapter only deals with purely initial value problems. The first part of the chapter treats linear problems, while the second part shows how the linear methods are extended to non-linear problems.

The third chapter is on boundary conditions. For hyperbolic problems the approximation of boundary conditions is of crucial importance. For a simple linear problem this remark will be illustrated. Then a few heuristic principles are introduced which are the basis of the boundary treatment given in this chapter. Also some practical aspects of the choice of boundaries and moving boundaries due to tidal flats will be described.

The fourth chapter describes applications. The applications involve flow in the North Sea, the Eastern Scheldt estuary, the river Rhine and a tidal flume.

2. INITIAL VALUE PROBLEMS

2.1 Introduction.

This chapter describes a numerical scheme for a purely initial value problem of shallow water equations.
The equations are given by:

$$(2.1\text{-}1a)\qquad u_t + uu_x + vu_y + g\zeta_x + g\frac{u(u^2+v^2)^{\frac{1}{2}}}{C^2H} - fv - \nu(u_{xx}+u_{yy}) = F^{(x)}$$

$$(2.1\text{-}1b)\qquad v_t + uv_x + vv_y + g\zeta_y + \frac{gv(u^2+v^2)^{\frac{1}{2}}}{C^2H} + fu - \nu(v_{xx}+v_{yy}) = F^{(y)}$$

$$(2.1\text{-}1c)\qquad \zeta_t + (Hu)_x + (Hv)_y = 0$$

where: u = velocity in x direction
v = velocity in y direction
ζ = waterelevation above some plane of reference
h = waterdepth below some plane of reference
H = h+ζ= total waterdepth
f = Coriolis parameter
g = acceleration due to gravity
C = Chezy coefficient for bottom roughness
$F^{(x,y)}$ = external forcing functions of windstress or barometric pressure.
ν = viscosity coefficient.

In order to clarify the scheme described in this chapter we start the treatment with the frozen coefficient equations derived from (2.1-1a). These equation are given by:

$$(2.1\text{-}2a)\qquad u_t + Uu_x + Vy_y + g\zeta_x = 0$$

$$(2.1\text{-}2b)\qquad v_t + Uv_x + Vv_y + g\zeta_y = 0$$

$$(2.1\text{-}2c)\qquad \zeta_t + U\zeta_x + V\zeta_y + Hu_x + Hv_y = 0$$

where U,V and H are constant coëfficients.

For (2.1-2) we have omitted the effects of Coriolis, bottom friction, viscosity and external forces.
In section (2.2) we describe the numerical scheme for the approximation of (2.1-2) and a stability analysis is added as well. In section (2.3) we describe a non-linear scheme for the approximations of (2.1-1). This scheme is similar to the linear scheme.

2.2 Linear aspects.

For the approximation of (2.1-2) in discrete points of the x,y space first a grid must be defined. The grid is defined by fig. 2-1.

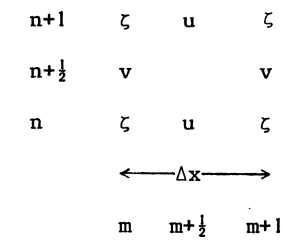


fig. (2-1) numerical grid.

In the "u" points the velocities in the x direction are approximated. In the "v" points the velocities in the y direction while in "ζ" points waterlevels are approximated.
The grid of fig. (2-1) is a socalled "staggered grid" which has been applied already since Hanssen [14]. For the approximation of (2.1-2a) the grid has several advantages such as efficiency and the absence of spurious roots. In case of implicit approximations for the time derivative the matrix conditions are much better then in case of non-staggered grids.
A discussion on the advantages of staggered grids is given by Stelling [1].

The spatial approximation of (2.1-2) is mainly based upon central differences. Only the approximations of Vu_y and Uv_x are based upon averaging of central differences and higher order upwind differencing.
The discretization in time is based upon the trapezoïdal rule. To simplify the implicit equations an ADI perturbation is applied.
The resulting scheme is given by:

stage 1:

(2.2-1a) $$(u^{k+\frac{1}{2}}-u^{k})/\tfrac{1}{2}\tau + U\overline{u^{k}}^{x}_{0x} + S_{0y}(V,u^{k}) + g\zeta^{k+\frac{1}{2}}_{0x} = 0, \text{ at } m+\tfrac{1}{2},n$$

(2.2-1b) $$(v^{k+\frac{1}{2}}-v^{k})/\tfrac{1}{2}\tau + V\overline{v^{k+\frac{1}{2}}}^{y}_{0y} + S_{+x}(U,v^{k+\frac{1}{2}}) + g\zeta^{k}_{0y} = 0, \text{ at } m,n+\tfrac{1}{2}$$

(2.2-1c) $$(\zeta^{k+\frac{1}{2}}-\zeta^{k})/\tfrac{1}{2}\tau + U\overline{\zeta^{k+\frac{1}{2}}}^{x}_{0x} + V\overline{\zeta^{k}}^{y}_{0y} + Hu^{k+\frac{1}{2}}_{0x} + Hv^{k}_{0y} = 0, \text{ at } m,n$$

stage 2:

$$(2.2\text{-}1d)\qquad (u^{k+1}-u^{k+\frac{1}{2}})/\tfrac{1}{2}\tau + U\overline{u_{0x}^{k+1}}^{x} + S_{+y}(V,u^{k+1}) + g\zeta_{0x}^{k+\frac{1}{2}} = 0, \text{ at } m+\tfrac{1}{2},n$$

$$(2.2\text{-}1c)\qquad (v^{k+1}-v^{k+\frac{1}{2}})/\tfrac{1}{2}\tau + V\overline{v_{0y}^{k+\frac{1}{2}}}^{y} + S_{0x}(U,v^{k+\frac{1}{2}}) + g\zeta_{0y}^{k+1} = 0, \text{ at } m,n+\tfrac{1}{2}$$

$$(2.2\text{-}1f)\qquad (\zeta^{k+1}-\zeta^{k+\frac{1}{2}})/\tfrac{1}{2}\tau + U\overline{\zeta_{0x}^{k+\frac{1}{2}}}^{x} + V\overline{\zeta_{0y}^{k+1}}^{y} + Hu_{0x}^{k+\frac{1}{2}} + Hv_{0y}^{k+1} = 0, \text{ at } m,n$$

where:

ζ_{0x} at $m+\frac{1}{2},n$ denotes $(\zeta_{m+1,n}-\zeta_{m,n})/\Delta x$,

ζ_{0y} at $m,n+\frac{1}{2}$ denotes $(\zeta_{m,n+1}-\zeta_{m,n})/\Delta y$,

$\overline{\zeta}^{x}$ at $m+\frac{1}{2},n$ denotes $(\zeta_{m+1,n}+\zeta_{m,n})/2$,

$\overline{\zeta}^{y}$ at $m,n+\frac{1}{2}$ denotes $(\zeta_{m,n+1}+\zeta_{m,n})/2$,

$S_{0y}(V,u)$ at m,n denotes $V(u_{m,n+2}+4u_{m,n+1}-4u_{m,n-1}-u_{m,n-2})/12\Delta y$

$$S_{+y}(V,u) \text{ at } m,n \text{ denotes} \begin{cases} V(3u_{m,n}-4u_{m,n-1}+u_{m,n-2})/2\Delta y & \text{if } V > 0 \\ V(-3u_{m,n}+4u_{m,n+1}-u_{m,n+2})/2\Delta y & \text{if } V \le 0 \end{cases}$$

u_{0x}, v_{0y}, $\bar{u}^{x}$, $\bar{v}^{y}$, $S_{0x}(U,v)$ and $S_{+x}(U,v)$ are defined similar to the definitions given above. For the functions S_{0y}, S_{0x}, S_{+x} and S_{+y} many alternatives are possible. The stability of the scheme (2.2-1) can be studied by substitution of:

$$(2.2\text{-}2)\qquad [\tilde{u}_{m,n}^{k},\tilde{v}_{m,n}^{k},\tilde{\zeta}^{k}]^{T} = [\hat{u}^{k},\hat{v}^{k},\hat{\zeta}^{k}]^{T} e^{i(\sigma_1 m\Delta x+\sigma_2 n\Delta y)} = \underline{\hat{w}}^{k} e^{i(\sigma_1 m\Delta x+\sigma_2 n\Delta y)}$$

After some derivation, see Stelling [1], this leads to

$$(2.2\text{-}3)\qquad \underline{\hat{w}}^{k+1} = \Lambda A^{-1}BC^{-1}D\Lambda^{-1}\underline{\hat{w}} = G\underline{\hat{w}}^{k}$$

Where G is the well known amplification matrix, see Richtmyer and Morton [15]. The matrices A,B,C,D and Λ are given by:

$$A = \begin{bmatrix} a & 0 & 0 \\ 0 & 1 & \frac{\tau}{2}\sqrt{gH}\,\hat{D}_{0y} \\ 0 & \frac{\tau}{2}\sqrt{gH}\,\hat{D}_{0y} & 1+\frac{\tau}{2}V\hat{D}_{1y} \end{bmatrix}, \quad B = \begin{bmatrix} 1 & 0 & \frac{-\tau}{2}\sqrt{gH}\,\hat{D}_{0x} \\ 0 & b & 0 \\ -\frac{\tau}{2}\sqrt{gH}\,\hat{D}_{0x} & 0 & 1-\frac{\tau}{2}U\,\hat{D}_{1x} \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 & \frac{\tau}{2}\sqrt{gH}\,\hat{D}_{0x} \\ 0 & c & 0 \\ \frac{\tau}{2}\sqrt{gH}\,\hat{D}_{0x} & 0 & 1+\frac{\tau}{2}U\hat{D}_{1x} \end{bmatrix}, \quad D = \begin{bmatrix} d & 0 & 0 \\ 0 & 1 & -\frac{\tau}{2}\sqrt{gH}\,\hat{D}_{0y} \\ 0 & -\frac{\tau}{2}\sqrt{gH}\,\hat{D}_{0y} & 1-\frac{\tau}{2}VD_{1y} \end{bmatrix}$$

$$\Lambda = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \sqrt{H/g} \end{bmatrix}$$

where

$$\hat{D}_{0x} = i \sin(\sigma_1 \tfrac{1}{2}\Delta x)/(\tfrac{1}{2}\Delta x),$$

$$\hat{D}_{0y} = i \sin(\sigma_2 \tfrac{1}{2}\Delta y)/(\tfrac{1}{2}\Delta y),$$

$$\hat{D}_{1x} = i \sin(\sigma_1 \Delta x)/\Delta x,$$

$$\hat{D}_{1y} = i \sin(\sigma_2 \Delta y)/\Delta y,$$

$$a = 1 + \frac{\tau}{2} U\hat{D}_{1x} + \frac{\tau}{2} V\hat{S}_{+y},$$

$$b = 1 - \frac{\tau}{2} V\hat{D}_{1y} - \frac{\tau}{2} U\hat{S}_{0x},$$

$$c = 1 + \frac{\tau}{2} V\hat{D}_{1y} + \frac{\tau}{2} U\hat{S}_{+x},$$

$$d = 1 - \frac{\tau}{2} U\hat{D}_{1x} - \frac{\tau}{2} V\hat{S}_{0y},$$

$$\hat{S}_{+x} = [(1-\cos(\sigma_1\Delta x))^2 + i \sin(\sigma_1\Delta x)(2-\cos(\sigma_1\Delta x))]/\Delta x,$$

$$\hat{S}_{0x} = i \sin(\sigma_1\Delta x)(2+\cos(\sigma_1\Delta x))/3\Delta x$$

$$\hat{S}_{+y} = [(1-\cos(\sigma_2\Delta y))^2 + i \sin(\sigma_2\Delta y)(2-\cos(\sigma_2\Delta y))]/\Delta y$$

and

$$\hat{S}_{0y} = i \sin(\sigma_2 \Delta y)(2+\cos(\sigma_2 \Delta y))/3\Delta y.$$

From (2.2-3) it follows that the following relation holds:

$$\|G^k\| \leq \|\Lambda A^{-1}\| \, \|BC^{-1}\|^k \, \|DA^{-1}\|^{k-1} \, \|D\Lambda^{-1}\| \tag{2.2-4}$$

For stability it is sufficient that: $\|BC^{-1}\| < 1$ and $\|DA^{-1}\| < 1$. To prove this we write A and D as follows:

$$A = \begin{bmatrix} a & 0 \\ 0 & A_s \end{bmatrix}, \qquad D = \begin{bmatrix} d & 0 \\ 0 & D_s \end{bmatrix} \tag{2.2-5}$$

where:

$$A_s = \begin{bmatrix} 1 & \frac{\tau}{2}\sqrt{gH}\,\hat{D}_{0y} \\ \frac{\tau}{2}\sqrt{gH}\,\hat{D}_{0y} & 1+\frac{\tau}{2}V\hat{D}_{1y} \end{bmatrix}, \qquad D_s = \begin{bmatrix} 1 & -\frac{\tau}{2}\sqrt{gH}\,\hat{D}_{0y} \\ -\frac{\tau}{2}\sqrt{gH}\,\hat{D}_{0y} & 1-\frac{\tau}{2}\sqrt{gH}\,\hat{D}_{1y} \end{bmatrix}$$

It follows that $D_s = A_s^H$ or:

$$DA^{-1} = \begin{bmatrix} d/a & 0 \\ 0 & A_s^H A_s^{-1} \end{bmatrix}$$

Since $A_s^H A_s^{-1}$ is a unitary matrix it follows that:

$$\| DA^{-1} \| \leq \text{Max}\,(|d/a|,1) \tag{2.2-6}$$

Because $|d/a| \leq 1$ it follows that $\|DA^{-1}\| \leq 1$.

Similarly one can prove that $\|BC^{-1}\| \leq 1$, which completes the proof of stability.

Note that despite of the unconditional stability the maximum timestep is limited based on considerations of accuracy. Especially the ADI structure can limit the maximum timestep as explained by Stelling [1].

Note that the Courant number, which is an often used dimensionless number, to indicate the size of the time step, is given by:

$$Cf = 2\tau\sqrt{gH}(1/\Delta x^2 + 1/\Delta y^2)^{\frac{1}{2}}. \tag{2.2-7}$$

The factor 2 of (2.2-7) is due to the fact that the gridsizes Δx (or Δy) are defined as the distance from a "u" point (or "v" point) to the nearest other "u" point (or "v" point), instead of to the nearest "ζ" point.

2.3 Nonlinear equations.

In this section we describe the approximation of (2.1-1). The frozen coefficient equation of this method was given already in the previous section. Based on this linear scheme a number of nonlinear schemes can be constructed. The grid is given by figure 2-2. The choice was based upon practical experiments. The version that proved to be stable for many situations was chosen.
The resulting scheme is an ADI/predictor corrector/iterative method which is given by:
stage 1:

$$u^{[0]} = u^k,\quad v^{[0]} = v^k,\quad \zeta^{[0]} = \zeta^k$$

For $p = 1,2,\ \ q = 1,2,\ldots,Q$:

$$(u^{[q]}-u^k)/\tfrac{1}{2}\tau + u^{[q]}\overline{u^k}^{x}_{0x} + S_{0y}(\overline{\overline{v}^{k+\frac{1}{2}}}^{y},u^k) - f\overline{v}^{k+\frac{1}{2}} + g\zeta^{[q]}_{0x}$$

$$+ gu^{[q]}[(\overline{\overline{v}^{k+\frac{1}{2}}}^{y})^2 + (u^k)^2]^{\frac{1}{2}}/(C^2H^k) - \nu(u^k_{0xx}+u^k_{0yy}) = 0,\ \text{at } m+\tfrac{1}{2},n \tag{2.3-1c}$$

$$(v^{[p]}-v^k)/\tfrac{1}{2}\tau + v^k\overline{v^{[p]}}^{y}_{0y} + S_{+x}[\overline{u^k}^{x},v^{[p]},\delta(p+p')] + f\overline{u^k}^{x}+g\zeta^k_{0y}$$

$$+ gv^{[p]}[(\overline{u^k}^{x})^2 + (v^k)^2]^{\frac{1}{2}}/(C^2H^k) - \nu(v^{[*]}_{0xx} + v^{[p]}_{0yy}) = 0,\ \text{at } m,n+\tfrac{1}{2} \tag{2.3-1b}$$

$$(\zeta^{[q]}-\zeta^k)/\tfrac{1}{2}\tau + (\overline{h}^{y}u^{[q]})_{0x} + \zeta^{[q-1]}u^{[q]}_{0x} + \overline{u^{[q-1]}\zeta^{[q]}}^{x}_{0x} + (H^kv^k)_{0y} = 0,\ \text{at } m,n$$

$$u^{k+\frac{1}{2}} = u^{[Q]},\quad v^{k+\frac{1}{2}} = v^{[2]},\quad \zeta^{k+\frac{1}{2}} = \zeta^{[Q]} \tag{2.3-1c}$$

where

$$\overline{\overline{v}} = \overline{v}^{xy},\quad \overline{\overline{u}} = \overline{u}^{xy}$$

$$S_{0y}(\overset{=k+\frac{1}{2}}{v},u^k) \text{ at } m+\tfrac{1}{2},n = \overset{=k+\frac{1}{2}}{v}_{m+\frac{1}{2},n}(u^k_{m+\frac{1}{2},n+2}+4u^k_{m+\frac{1}{2},n+1}-4u^k_{m+\frac{1}{2},n-1}-u^k_{m+\frac{1}{2},n-2})/12\Delta y$$

$$S_{+x}(\overset{=k}{u},v^{[p]},\delta) \text{ at } m,n+\tfrac{1}{2} = \begin{cases} \overset{=k}{u}_{m,n+\frac{1}{2}}(3v^{[p-\delta]}_{m,n+\frac{1}{2}}-4v^{[p-\delta]}_{m-1,n+\frac{1}{2}}+v^{[p-\delta]}_{m-2,n+\frac{1}{2}})/2\Delta x \text{ if } \overset{=k}{u}_{m,n+\frac{1}{2}} > 0 \\ \overset{=k}{u}_{m,n+\frac{1}{2}}(-3v^{[p-1+\delta]}_{m,n+\frac{1}{2}}+4v^{[p-1+\delta]}_{m+1,n+\frac{1}{2}}-v^{[p-1+\delta]}_{m+2,n+\frac{1}{2}})/2\Delta x \text{ if } \overset{=k}{u}_{m,n+\frac{1}{2}} < 0 \end{cases}$$

$$\delta(p+p') = \tfrac{1}{2}[1+(-1)^{p+p'}]$$

$$p' = \begin{cases} 0, \text{ if } \Sigma_{m,n}\, u^k > 0 \ (\Sigma_{m,n}u \text{ denotes the sum of } u \text{ over all grid points}) \\ 1, \text{ if } \Sigma_{m,n}\, u^k < 0 \end{cases}$$

$$\text{and } v^{[*]}_{0xx} \text{ at } m,n+\tfrac{1}{2} = (v^{[p-1+\delta(p+p')]}_{m+1,n+\frac{1}{2}} - 2v^{[p]}_{m,n+\frac{1}{2}} + v^{[p-\delta(p+p')]}_{m-1,n+\frac{1}{2}})/\Delta x^2$$

Stage 2:

$$u^{[0]} = u^{k+\frac{1}{2}},\ v^{[0]} = v^{k+\frac{1}{2}},\ \zeta^{[0]} = \delta^{k+\frac{1}{2}}$$

For $p = 1,2$ and $q = 1,\ldots,Q$:

(2.4-1d)
$$(u^{[p]}-u^{k+\frac{1}{2}})/\tfrac{1}{2}\tau + u^{k+\frac{1}{2}}\overline{u^{[p]}_{0x}}^{x} + S_{+y}[\overset{=k+\frac{1}{2}}{v},u^{[p]},\delta(p+p')] - f\overset{=k+\frac{1}{2}}{v} + g\zeta^{k+\frac{1}{2}}_{0x}$$
$$+ gu^{[p]}[(\overset{=k+\frac{1}{2}}{v})^2 + (u^{k+\frac{1}{2}})^2]^{\frac{1}{2}}/(C^2H^{k+\frac{1}{2}}) - \nu(u^{[p]}_{0xx}+u^{[*]}_{0yy}) = 0 \text{ at } m+\tfrac{1}{2},n$$

(2.4-1e)
$$(v^{[q]}-v^{k+\frac{1}{2}})/\tfrac{1}{2}\tau + v^{[q]}\overline{v^{k+\frac{1}{2}}_{0x}}^{x} + S_{0x}(\overset{=k+1}{u},v^{k+\frac{1}{2}}) + f\overset{=k+1}{u} + g\zeta^{[q]}_{0x}$$
$$+ gv^{[q]}[(v^{k+\frac{1}{2}})^2 + (\overset{=k+1}{u})^2]^{\frac{1}{2}}/(C^2H^{k+\frac{1}{2}}) - \nu(v^{k+\frac{1}{2}}_{0xx}+v^{k+\frac{1}{2}}_{0yy}) = 0 \text{ at } m,n+\tfrac{1}{2}$$

(2.4-1f)
$$(\zeta^{[q]}-\zeta^{k+\frac{1}{2}})/\tfrac{1}{2}\tau + (H^{k+\frac{1}{2}}u^{k+\frac{1}{2}})_{0x} + (\bar{h}^x v^{[q]})_{0y} + \zeta^{[q-1]}v^{[q]}_{0y} + \overline{v^{[q-1]}\zeta^{[q]}_{0y}}^{y} = 0, \text{ at } m,n$$

$$u^{k+1} = u^{[2]},\ v^{k+1} = v^{[Q]},\ \zeta^{k+1} = \zeta^{[Q]}$$

where

$$S_{+y}(\overset{=k+\frac{1}{2}}{v},u^{[p]},\delta) \text{ at } m+\tfrac{1}{2},n = \begin{cases} \overset{=k+\frac{1}{2}}{v}_{m+\frac{1}{2},n}(3u^{[p-\delta]}_{m+\frac{1}{2},n}-4u^{[p-\delta]}_{m+\frac{1}{2},n-1}+u^{[p-\delta]}_{m+\frac{1}{2},n-2})/2\Delta x, \text{ if } \overset{=k+\frac{1}{2}}{v}_{m+\frac{1}{2},n} > 0 \\ \overset{=k+\frac{1}{2}}{v}_{m+\frac{1}{2},n}(-3u^{[p-1+\delta]}_{m+\frac{1}{2},n}+4u^{[p-1+\delta]}_{m+\frac{1}{2},n+1}-u^{[p-1+\delta]}_{m+\frac{1}{2},n+2})/2\Delta x, \text{ if } \overset{=k+\frac{1}{2}}{v}_{m+\frac{1}{2},n} < 0 \end{cases}$$

$$\delta(p+p') = [1 + (-1)^{p+p'}]$$

$$p' = \begin{cases} 0, & \text{if } \Sigma_{m,n} v^{k+\frac{1}{2}} > 0 \quad (\Sigma_{m,n} v \text{ denotes the sum of v over all grid points}) \\ 1, & \text{if } \Sigma_{m,n} v^{k+\frac{1}{2}} < 0 \end{cases}$$

$$S_{0x}(\bar{u}^{k+1}, v^{k+\frac{1}{2}}) \text{ at } m,n+\tfrac{1}{2} = \bar{u}^{k+1}_{m,n+\frac{1}{2}}(v^{k+\frac{1}{2}}_{m+2,n+\frac{1}{2}} + 4v^{k+\frac{1}{2}}_{m+1,n+\frac{1}{2}} - 4v^{k+\frac{1}{2}}_{m-1,n+\frac{1}{2}} - v^{k+\frac{1}{2}}_{m-2,n+\frac{1}{2}})/12\Delta x$$

and

$$u^{[*]}_{0yy} \text{ at } m+\tfrac{1}{2},n = (u^{[p-1+\delta(p+p')]}_{m+\frac{1}{2},n+1} - 2u^{[p]}_{m+\frac{1}{2},n} + u^{[p-\delta(p+p')]}_{m+\frac{1}{2},n-1})/\Delta y^2$$

n=1	ζ	u	ζ	u	ζ
n+½	v	h	v	h	v
n	ζ	u	ζ	u	ζ

Figure 2.2. numerical grid.

3. BOUNDARY CONDITIONS

3.1 Introduction.

This chapter describes the boundary treatment of the simulation method in a concise way. The boundary treatment is of crucial importance; a wrong numerical treatment of boundaries could well lead to instabilities. Relevant articles on the stability of numerical boundary procedures are written by Kreiss [2], Gustafsson, Kreiss and Sundstrom [3], Goldberg and Tadmor [4] [5], Michelson [6] and others. The theory treated by these authors is difficult to apply to two -dimensional problems. Even for one-dimensional problems the analytical difficulties are often hard to come by. Yet the theory of boundary conditions is very important to improve the general insight into the numerical treatment of boundary conditions. From this theory we have derived a few heuristic principles that we apply for applications.
In section 3.2 we introduce the ideas behind our views by means of a simple example.
In section 3.3 the actual treatment of closed boundaries is given. In section 3.4 the treatment of open boundaries will be given.

3.2 Stability of numerical boundary procedures.

Numerical equations at the inner points of a numerical grid can often not be solved near or at the boundaries of this grid. This leads to the construction of special boundary schemes or extrapolation methods near boundaries for the calculation of "missing" points. We will illustrate this remark with a simple example; consider the following initial boundary value problem:

$$u_t + u_x = 0, \quad 0 < x \leq 1, \quad t \geq 0 \tag{3.2-1a}$$

$$u(0,t) = 1, \quad u(x,0) = 1-x, \quad 0 < x \leq 1 \tag{3.2-1b}$$

The solution of this equation is given by:

$$u(x,t) = \begin{cases} 1+t-x, & t-x \leq 0 \\ 1, & t-x > 0 \end{cases} \tag{3.2-2}$$

Note that only one boundary condition is given at the "inflow" boundary $x = 0$.

Eq. (3.2-1) will be approximated by central differences while the time is kept continuous. This yields:

$$(u_m)_t + (u_{m+1}-u_{m-1})/2\Delta x = 0, \quad m = 1,2,\ldots,M \tag{3.2-3a}$$

$$u_0(t) = 1, \quad u_m(0) = 1-m\Delta x, \quad m = 1,2,\ldots,M \tag{3.2-3b}$$

Where Δx denotes the distance between two gridpoints, $\Delta x = 1/M$.

The equations (3.2-3) are not complete because at $m = M+1$ an equation is missing. Suppose that at $m = M+1$ the following equation will be given:

$$u_{M+1} = 0. \tag{3.2-4}$$

Combined with (3.2-4), (3.2-3) is a complete set of equations. Its solution however is highly oscillatory, see Stelling [1] p. 43, and does not converge to the solution of (3.2-1). By the addition of sufficient numerical viscosity the oscillations will disappear. In that case a viscosity term will be added

to (3.2-3a) as follows:

$$(3.2\text{-}5) \qquad (u_m)_t + (u_{m+1}-u_{m-1})/(2\Delta x)-\varepsilon(u_{m+1}-2u_m+u_{m-1})/(\Delta x^2) = 0$$

Although for sufficiently large values of ε nonoscillatory and even convergent approximations of (3.2-1) can be obtained it is not a very satisfactory solution of the problem because it might imply very small values of Δx in order to obtain approximations of sufficient accuracy.
In fact the reason for large oscillatons is (3.2-4) which is basicly an overspecified boundary condition. The following outflow boundary condition:

$$(3.2\text{-}6) \qquad u_{M+1} = 2u_M - u_{M-1}$$

will reduce the oscillations drastically and moreover this condition will ensure convergence. Extrapolation formula up to first order will not destroy convergence as has been pointed out by Gustafsson [7].
Note that substitution of (3.2-6) into (3.2-3a) yields:

$$(3.2\text{-}7) \qquad (u_M)_t + (u_M-u_{M-1})/\Delta x = 0,$$

which is in fact a first order upwind differencing approximation. Yet with (3.2-6) the convergence of (3.2-3a) is of second order, see Stelling [1] p. 41. General theorems about the order of convergence related to the order of the boundary extrapolation formulas are given by Gustafsson [7].

The stability of numerical boundary procedures for 1-dimensional hyperbolic problems is studied by Gustafsson, Kreiss and Sudström [3]. For dissipative approximations this theory has been extended to multi-dimensional problems by Michelson [6].
For two-dimensional problems the application of such theories yields very complicated analytical problems while the results only hold for linear problems. By practical experience we found the numerical boundary procedures to be of crucial importance for stability or for the exclusion of wiggles especially with respect to the "advective" part of the shallow water equations.

With respect to the advective part we have adopted the following simple criteria for the construction of numerical boundary procedures:

(i) Avoid overspecification of zero order boundary conditions. Higher order boundary conditions are often less hazardous .

(ii) Viscosity terms require extra boundary conditions, also analytically. These conditions are only implemented within the boundary treatment of the viscosity terms. For advection terms the boundary treatment remains according to (i) as if viscosity is absent. This reduces oscillations if the amount of viscosity is only very small and the extra boundary conditions are of order zero.

(iii) Use schemes near bounaries that are stable when they are applied to a purely initial value problem.

3.3 A few aspects of numerical boundary procedures.

There are two types of boundaries: closed and open. Closed boundaries are physical, they are land-water boundaries. Open boundaries are non physical. They are drawn across the water to limit the domain of the problem. At closed boundaries the following boundary conditions are prescribed:

$$(3.3\text{-}1) \qquad \begin{cases} u_{\perp} = 0 \\ \dfrac{\partial}{\partial n} u_{\shortparallel} = 0, \quad \text{if} \quad \nu \neq 0 \end{cases}$$

where $u_{\perp}$ denotes the velocity normal to the boundary, $u_{\shortparallel}$ the velocity parallel to the boundary and $\frac{\partial}{\partial n}$ the derivative normal to the boundary. If $\nu \neq 0$ then (3.3-1) denotes a perfect slip boundary condition. Note that non-slip boundary conditions or partially non-slip conditions are possible as well. If the flows contains eddies this can influence the flowpattern significantly, see Stelling [1] and Stelling and Wang [13].

The location of closed boundaries has two possibilities: (i) the location is constant in time. (ii) the location is time varying due to flooding and drying of tidal flats, i.e. tidal flats are simulated numerically by moving closed boundaries, see Stelling [1] or Stelling, Wiersma and Willemse [8]. This second aspect complicates the treatment significantly. At arbitrary locations within the numerical domain the presence of closed boundaries must be recognized. This will cause some computational overhead. In order to keep this overhead as small as possible a certain strategy has been described by Stelling [1].

Open boundaries have a constant location in time. Often waterlevels or velocities are prescribed. For the number of boundary conditions see e.g. Daubert and Graffe [9]. For well posedness see e.g. Verboom Stelling and Officier [10] or Edwards and Preston [11]. The boundary conditions as implemented for our simulation model are somewhat different from the boundary conditions described in the afore mentioned literature and are based on practical experiments.

First we will treat some aspects of waterlevel boundary conditions. Waterlevel boundary conditions are treated in the following way:

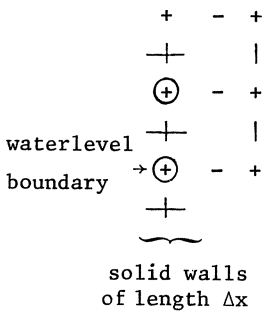


Figure 3.2 waterlevel boundaries.

Waterlevel boundaries are located on waterlevel points, see figure 3-1. The velocity points along the open boundary are set to zero. The zero velocity points along the open boundary together with the prescription of waterlevels basicly cause the following boundary conditions to be effective:

(3.3-2a) $\zeta = f^{\zeta}(t)$

(3.3-2b) $\frac{\partial}{\partial n}(u_{11}) = 0$, at inflow and or if $\nu \neq 0$

The boundary condition (3.3-2b) is a consequence of the treatment of advection terms near zero velocity points as described by Stelling [1]. It is to be noted that near waterlevel boundaries, if the flow field contains significant gradients due to large gradients of the bottom profile or due to eddies that reach the open boundary, the flow field is very sensitive to small disturbances. Even instabilities are likely to occur, although instabilities can be suppressed by adding a small non-reflective type of boundary condition to (3.2-2a), see Stelling [1] p. 152. According to some authors, see Edwards and Preston [11], waterlevel bounary

conditions do not yield well-posed problems. By addition of a small disturbance to the boundary condition this problem can be solved. The addition of a small non relfective part to (3.2-2c) yields:

$$\zeta + \varepsilon^{\zeta}\frac{\partial}{\partial t}\left[2(gH)^{\frac{1}{2}} \pm u\right] = f^{\zeta}(t) \qquad (3.3\text{-}3)$$

where ε^{ζ} denotes a small quantity (s^2).

Despite of (3.3-3) it is still strongly recommended to choose the location of the open boundary such that near or at the open boundary strong velocity gradients are not to be expected. For example in case of studies of the effect of jetties on the velocity distribution the open boundary has to be chosen at a sufficiently large distance from the jetty.

Similar to waterlevel boundaries velocity boundaries are prescribed. Velocity boundaries are given at velocity points as represented by fig.3-2

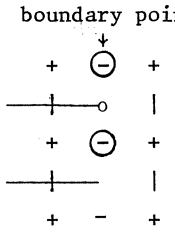


Figure 3.2 velocity boundaries

Again the velocities just outside the domain in the direction along the open boundary are set to zero.
Effectively this yields the following boundary conditions:

$$u = f^{u}(t) \qquad (3.3\text{-}4a)$$

$$\frac{\partial}{\partial n}(u_{11}) = 0, \quad \text{at inflow and/or} \quad \nu \neq 0. \qquad (3.3\text{-}4b)$$

Although the stability of velocity boundary conditions is much larger than of waterlevel boundary conditions it is still profitable to add a non-reflective part. This has a stabilizing effect. In case of stationary problems it sometimes increases the convergence to the steady state solution very significantly.
Addition gives:

$$u + \varepsilon^{u}\frac{\partial}{\partial t}\left(u \pm 2\sqrt{gH}\right) = f^{u}(t) \qquad (3.3\text{-}5)$$

where ε^u denotes a small quantity, (s).

Note that also discharges can be given as boundary conditions. For our model this type of boundary condition is given at velocity points.

Finally we want to stress the fact that in many cases computational difficulties are due to improper choices of the location of open boundaries or due to an inadequate numerical boundary procedure. Stabilizing these difficulties by addition of artificial viscosity, for which sometimes euphemistic terms are in use like "selective lumping" or "filtering", quite often affects the accuracy of the results. If really complicated flow pattern are to be studied then these inaccuracies could well be inacceptable.
It is our experience that by carefully chosen locations of the open boundaries and also by carefully constructed boundary procedures the addition of artificial viscosity can be minimized or even omitted. Another aspect that strongly contributes to the supression of spurious "wiggles" is the application of fully staggered grids, see Stelling [1].

4. EXAMPLES

In the section we will show some practical applications of the numerical method described in the previous sections. These examples differ in type: time varying problems as well as stationary, and a large range in the gridsizes. The applications are not described in detail, they merely serve as an illustration of the capability and robustness of the numerical method.

In the first example we will show some specific results of a simulation of the North Sea. This simulation was run with a timestep of τ = 600 sec., and with a gridsize of $\Delta x = \Delta y$ = 8000m. The model consists of approximately 11000 active calculation points. In figures 4.1 and 4.2 two computed timehistories are shown in stations at the Dutch and at the English coast, and in figure 4.3 a resulting flowpattern is given. For a detailed description of this model and of some of its applications we refer to Voogt [12].

The second example concerns a model of a small part of the Eastern Scheldt estuary, where a storm surge barrier is being built. This model belongs to a series of models (some of them are so-called nested models) whose gridsizes vary from Δx = 800m to Δx = 10m, and which are all situated in

the Eastern Scheldt estuary. The simulation we show was run with a timestep of τ = 60 sec and with a gridsize of Δx = 90m. In figures 4.4 and 4.5 two flowpatterns are given, at maximum ebb and at maximum flood. For a description of the effect of the timestep on the accuracy of the results (for applications on models with a complex geometry) we refer to Stelling, Wiersma and Willemse [8].

In the third model we show a steadystate calculation of a small part of the river Rhine. This simulation was performed with Δx = 50m, and τ = 60 sec. Here, the timestep can be interpreted as an iteration parameter for reaching the steady state. In figure 4.6 we show the resulting streamlines of the calculation.

The fourth and last example consists of a simulation of a laboratory flame, where the development of eddges behind a backstep has to be calculated. The simulation was run with a timestep of τ = 0.125 sec. and Δx = 0.025 m. In figures 4.7 up to 4.14 we give flow patterns at different moments. This example shows that the method is capable of handling difficult flow patterns. For a detailed description of the example, together with a comparison to measurements, we refer to Stelling and Wang [13].

REFERENCES

[1] STELLING, G.S., *On the construction of computational methods for shallow waterflow problems*, Ph. D. thesis, Delft University of Technology, Delft, The Netherlands 1983.

[2] KREISS, H.O., *Stability of difference approximations for mixed initial boundary value problems I*, Math. Comp. 22 (1968), pp. 703-714.

[3] GUSTAFSSON, B., H.O. KREISS & A. SUNDSTRÖM, *Stability theory of difference approximations for mixed initial boundary value problems*, II. Math. Comp. 26 (1972), pp. 649-686.

[4] GOLDBERG, M. & E. TADMOR, *Scheme independent stability criteria for difference approximations of hyperbolic initial boundary value problems, I*, Math. Comp. 32 (1978), pp. 1057-1107.

[5] GOLDBERG, M. & E. TADMOR, *Scheme independent stability criteria for difference approximations of hyperbolic initial boundary value problems, II*, Math. Comp. 36 (1981) pp. 603-625.

[6] MICHELSON, D., *Stability theory of difference approximations for multidimensional initial boundary value* 40 (1983), pp. 1-45.

[7] GUSTAFSON, B., *The convergence rate for difference approximations to mixed initial boundary value problems*. Math. Comp. 29 (1975) pp. 396-406.

[8] STELLING, G.S., A.K. Wiersma & J.B.T.M. WILLEMSE, *Some practical aspects of the accuracy of tidal computations*, to appear.

[9] DAUBERT, A. & O. GRAFFE, *Quelques aspects des écoutement presque horizontaux a deux dimensions en plan et non-paraments application aux estuaires*, La Houille Blanche 8, (1967), pp. 847-860.

[10] VERBOOM, G.K., G.S. STELLING & M.J. OFFICIER, *Boundary conditions for the shallow water equations*, in Computational Hydraulics ed Abbot and Cunge, Pitman Publishing, 1981.

[11] PRESTON, R.W. & N.A. EDWARDS, *On the number and type of boundary conditions required by the shallow water equations*, CERL memorandum TPRD/L/AP 0050/M82 1982.

[12] VOOGT, L., *2D mathematical model of the North Sea*, presented at the Jonsdapmeeting, july 1984, Bergen, Norway.

[13] STELLING, G.S. & L.X. WANG, *Experiments and computations of unsteady separating flow in an expanding flume*, Report no. 2-84, 1984, Laboratory of Fluid Mechanics, Department of Civil Engineering, Delft University of Technology, Delft, the Netherlands.

[14] HANSEN, W., *Theorie zur Errechnung des Wasserstandes und der Strömungen in Randmeeren nebst Anwendungen*, Tellus 8 (1956), pp. 289-300.

[15] RICHTMYER, R.D. & K.W. MORTON, *Difference methods for initial value problems*, Interscience Publishers, Wiley, New York, London, 1967.

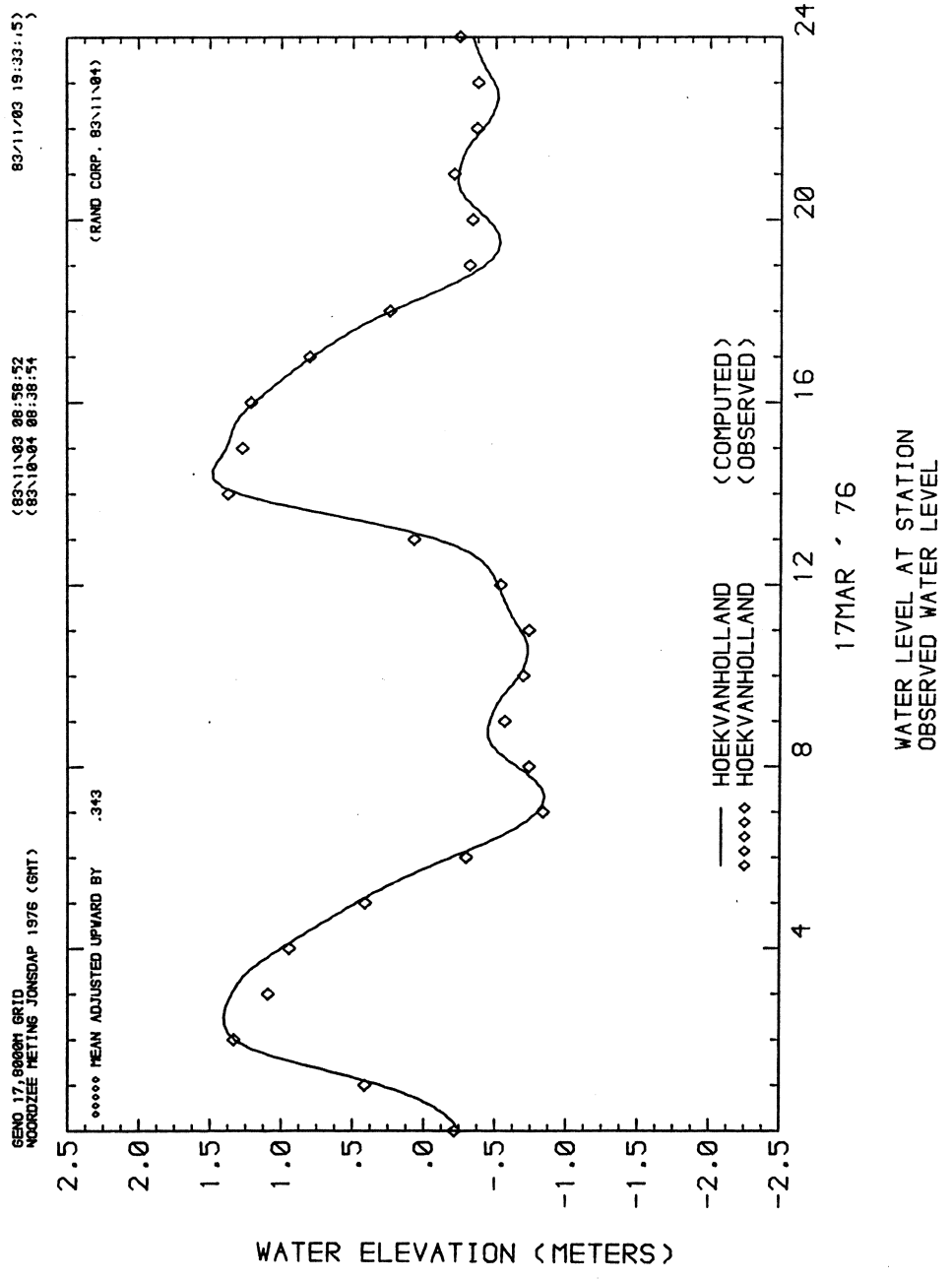

GEND 17,8000M GRID
NOORDZEE METING JONSDAP 1976 (GMT)
83/11/03 19:33:15)
(RAND CORP. 83\11\04)
(83\11\03 08:58:52
(83\10\04 08:38:54
◇◇◇◇◇ MEAN ADJUSTED UPWARD BY .343
HOEKVANHOLLAND (COMPUTED)
HOEKVANHOLLAND (OBSERVED)
WATER LEVEL AT STATION
OBSERVED WATER LEVEL
WATER ELEVATION (METERS)
17MAR ' 76
2.5
2.0
1.5
1.0
.5
.0
-.5
-1.0
-1.5
-2.0
-2.5
4
8
12
16
20
24

figure 4.1

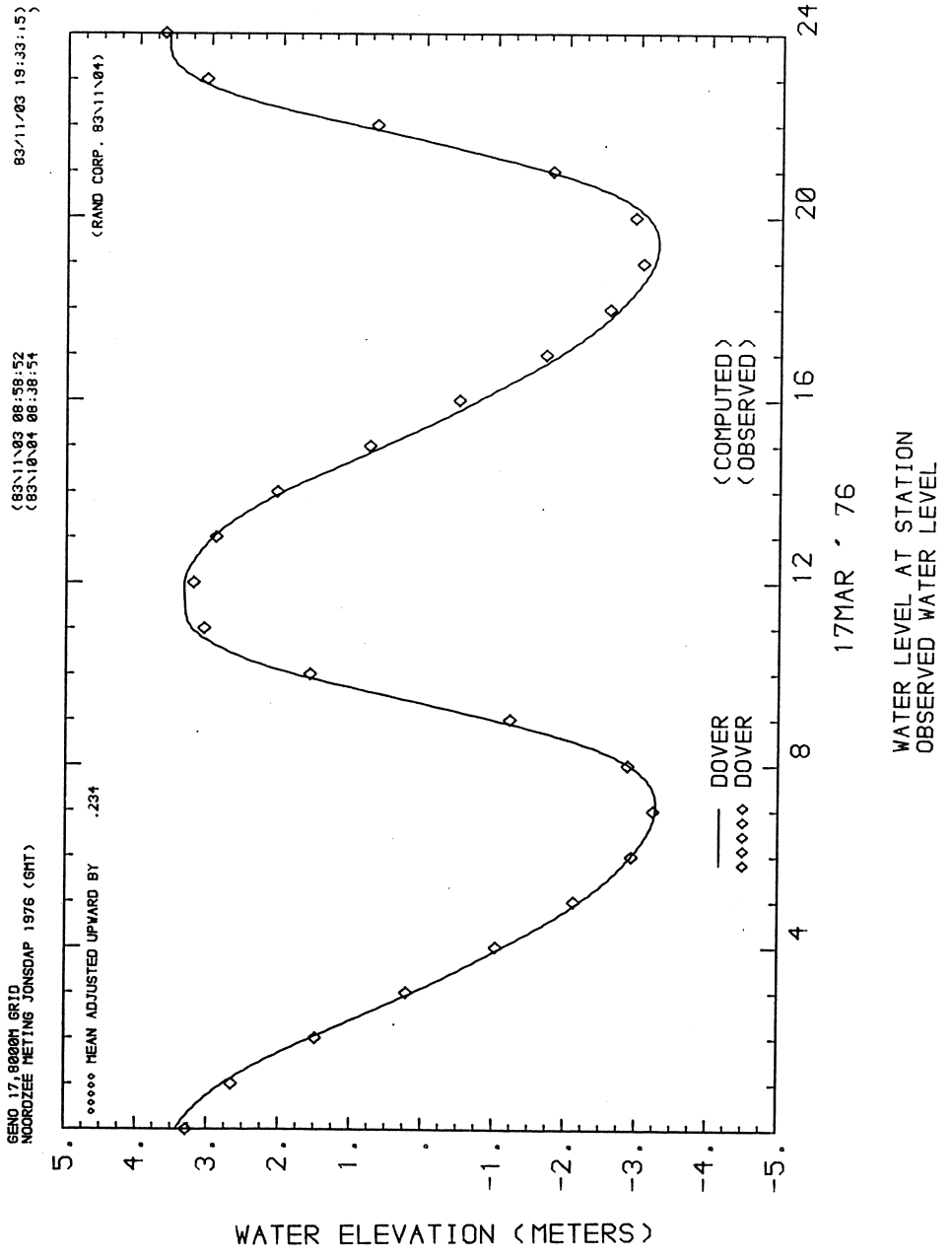

GENO 17,800M GRID
NOORDZEE METING JONSDAP 1976 (GMT)
(83\11\03 08:58:52
(83\10\04 08:38:54
83/11/03 19:33:15)
(RAND CORP. 83\11\04)
MEAN ADJUSTED UPWARD BY .234
(COMPUTED)
(OBSERVED)
DOVER
DOVER
WATER ELEVATION (METERS)
17MAR '76
WATER LEVEL AT STATION
OBSERVED WATER LEVEL

figure 4.2

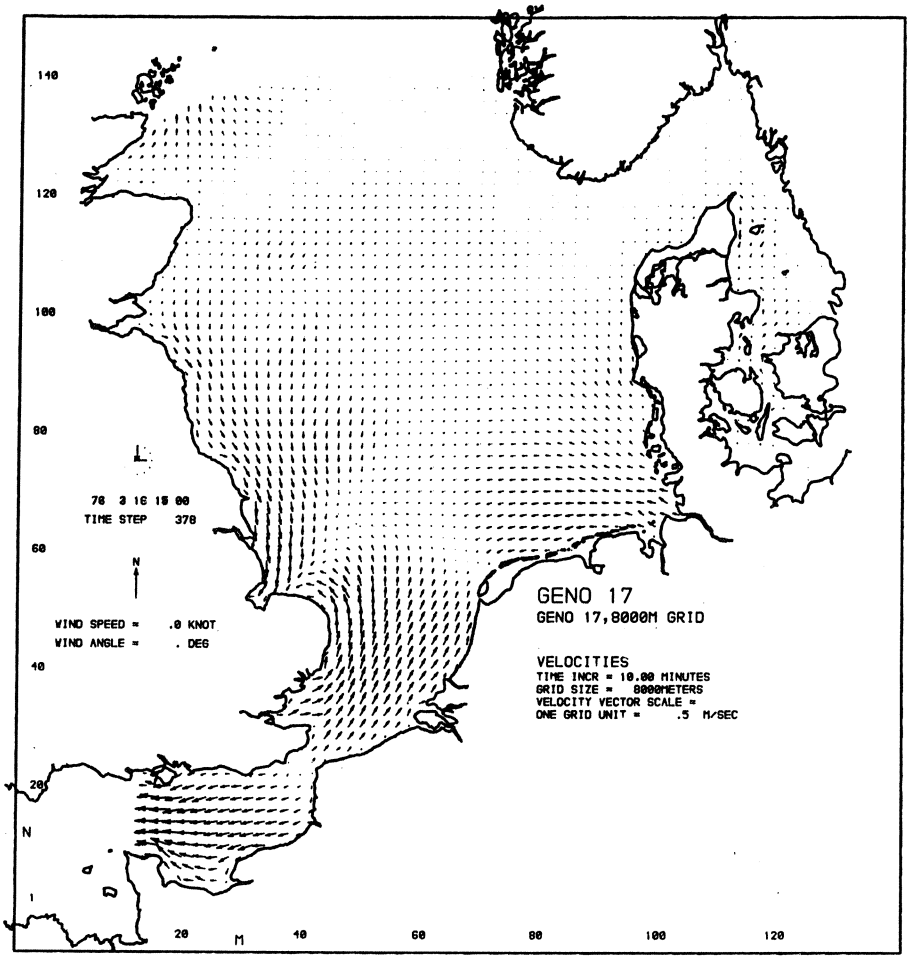

140
120
100
80
60
40
20
N
1
20
M
40
60
80
100
120
76 3 16 15 00
TIME STEP 378
N
WIND SPEED = .0 KNOT
WIND ANGLE = . DEG
GENO 17
GENO 17,8000M GRID
VELOCITIES
TIME INCR = 10.00 MINUTES
GRID SIZE = 8000METERS
VELOCITY VECTOR SCALE =
ONE GRID UNIT = .5 M/SEC

figure 4.3

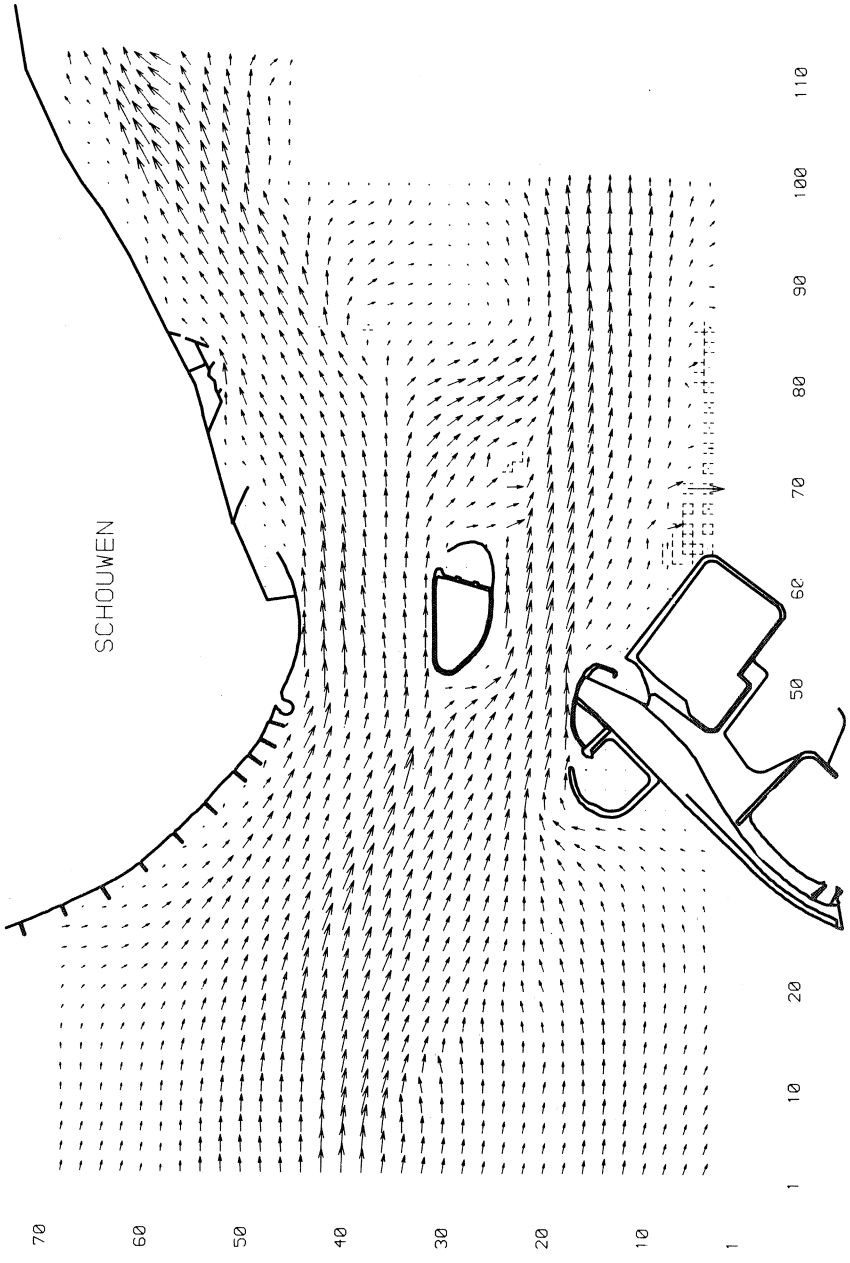


figure 4.4

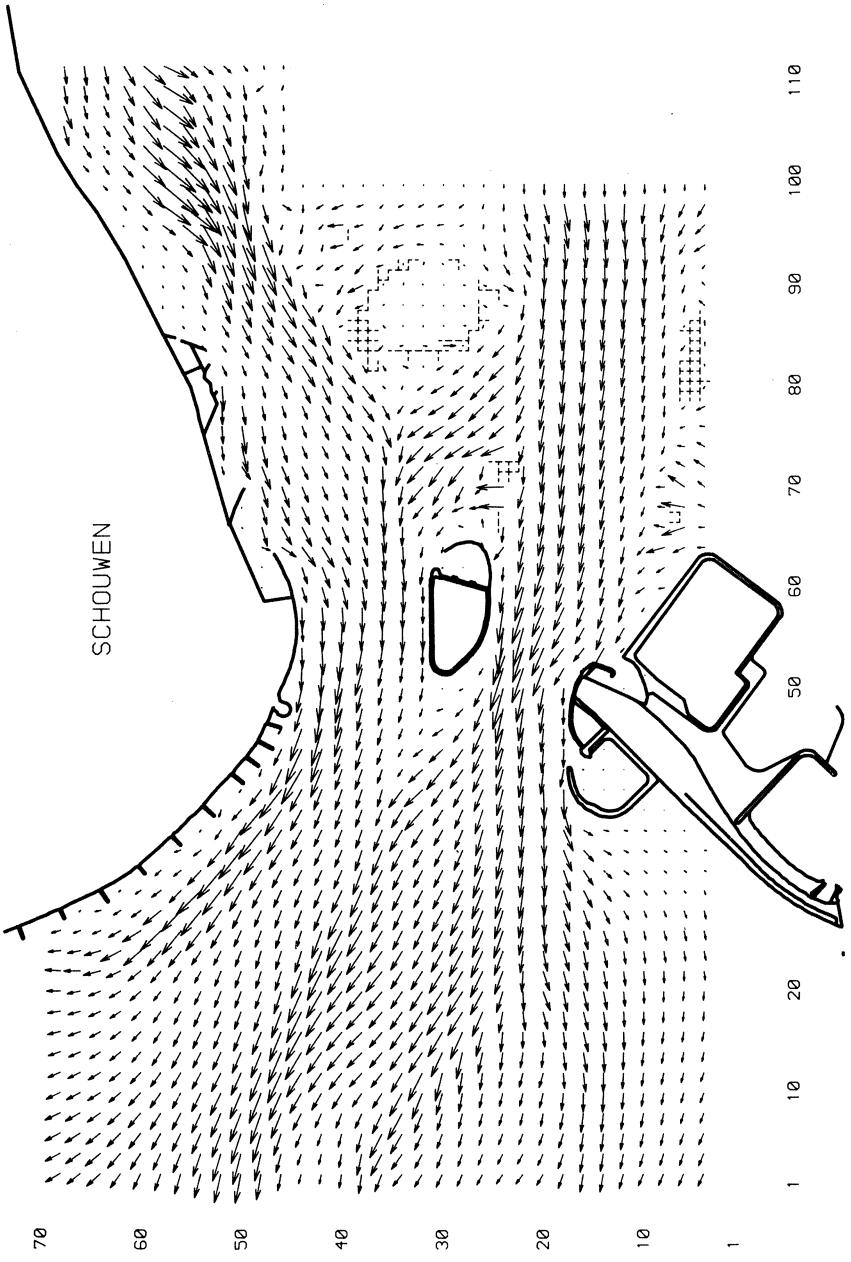


figure 4.5

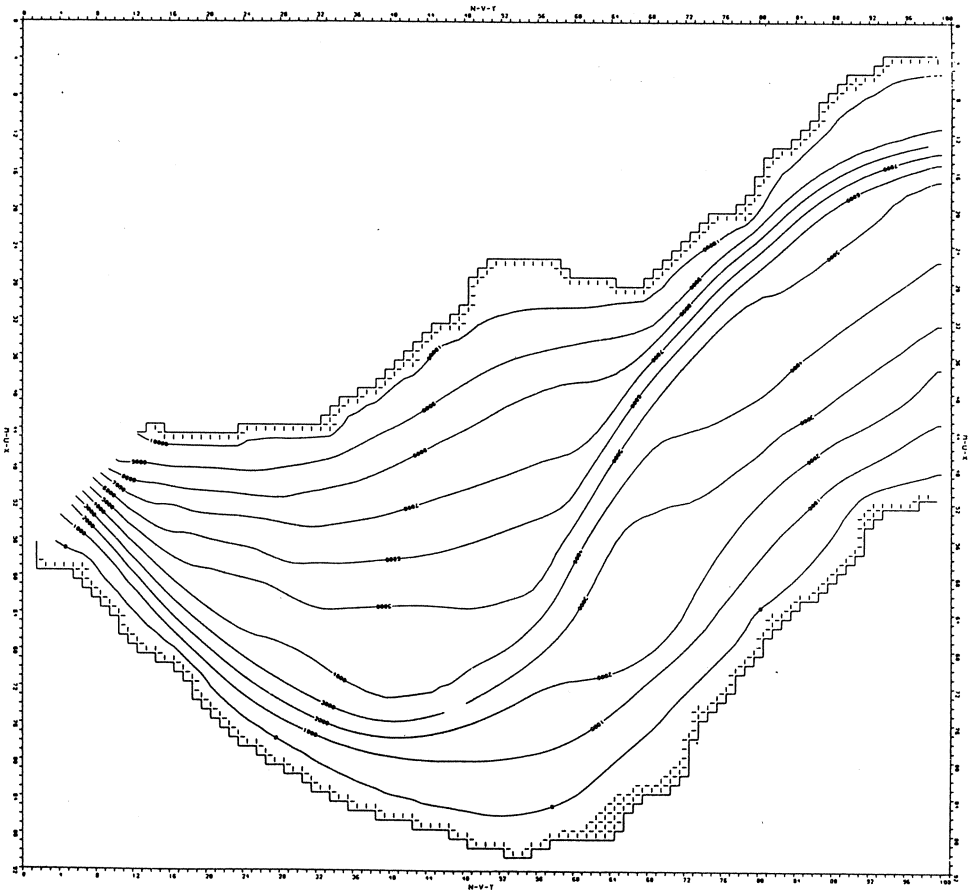


figure 4.6

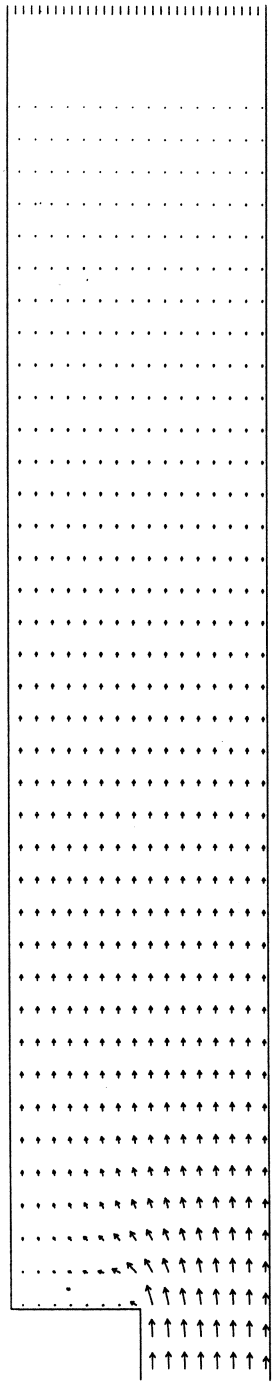


figure 4.7

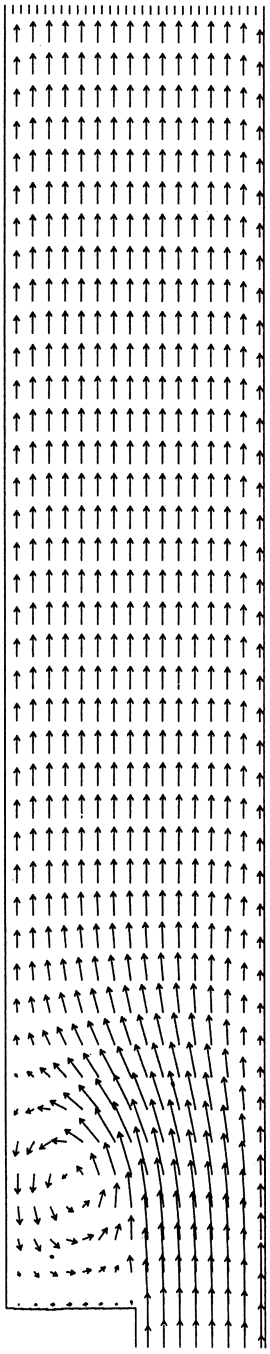


figure 4.8

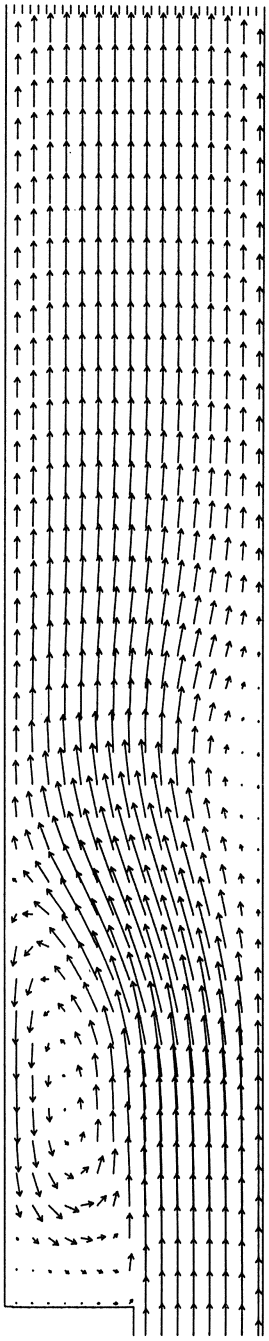


figure 4.9

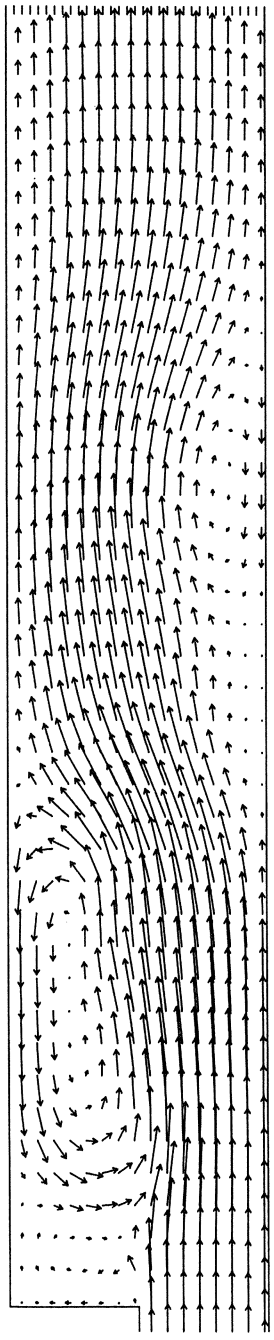


figure 4.10

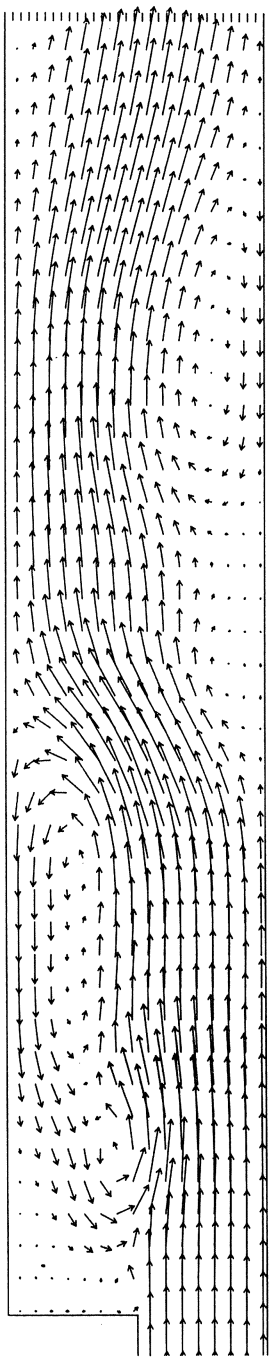


figure 4.11

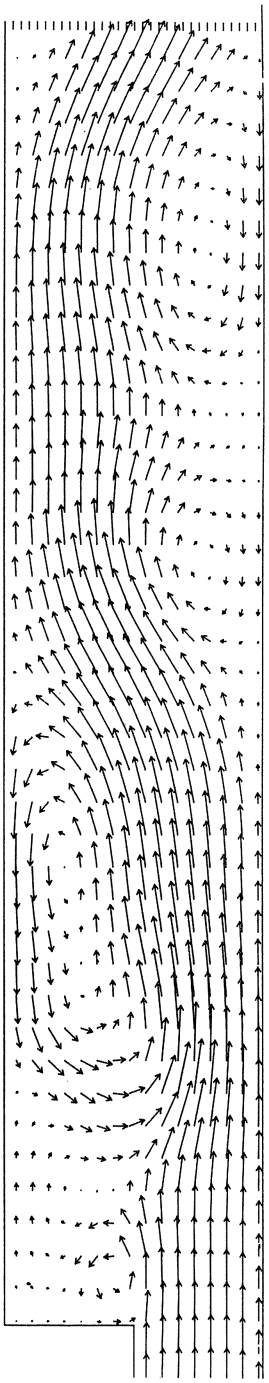


figure 4.12

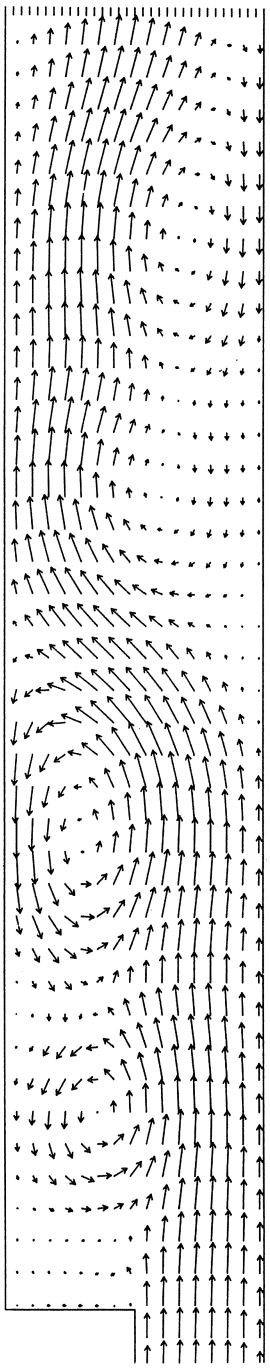


figure 4.13

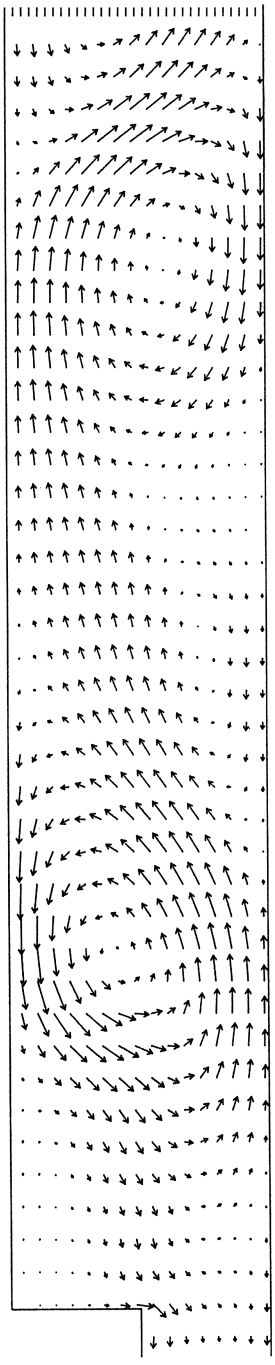


figure 4.14

STABILITY ANALYSIS OF THE MARCHING-ON-IN-TIME METHOD FOR ONE- AND TWO- DIMENSIONAL TRANSIENT ELECTROMAGNETIC SCATTERING PROBLEMS

A.G. TIJHUIS

The transient scattering of electromagnetic fields by one- and two-dimensional obstacles of finite extent is investigated with the aid of the time-domain integral equation technique. In solving such equations with the marching-on-in-time method, numerical instabilities form a major problem. These instabilities can be attributed to the errors in the discretization of the source-type integrals that occur in the equations. In this paper, we formulate two so-called stability criteria for such a discretization which, if they are met, guarantee that the instability can be controlled by reducing the discretization step. With the aid of these criteria, we analyze the solution of a number of electromagnetic scattering problems, namely the scattering of a pulsed plane wave by a one-dimensional, inhomogeneous, lossy dielectric slab, both in vacuum and in between two homogeneous, lossless dielectric halfspaces, and the scattering of such a pulse by a perfectly conducting or an inhomogeneous, lossy dielectric cylinder. Numerical results are presented and discussed.

1. INTRODUCTION

In this paper, a number of one- and two-dimensional electromagnetic transient-scattering problems are investigated with the aid of the time-domain integral equation technique. This technique has been applied to various configurations both in electromagnetics and in acoustics (see [1] and references cited therein). An important tool in the numerical solution of such equations is the so-called marching-on-in-time method. This method utilizes the property in the equation that the scattered field is expressed in terms of one or more integrals of field values at previous instants. The spatial domain of these integrals is either the boundary of the scattering obstacle (for homogeneous or impenetrable scatterers) or its interior (for inhomogeneous scatterers).

A limiting factor in the application of the marching-on-in-time method is the accumulation of the errors made in each step. In the one-dimensional case, this accumulation can be handled by discretizing the relevant space-time integrals in such a way that the error per step in the updating scheme is proportional to h^2, where h is the mesh size of a uniform space- (dimensionless) time grid. If this error is not amplified in the next step, the overall error towards the end of a finite interval will then at worst be proportional to h, since the number of updating steps increases linearly with 1/h (see also [2]). Hence, this error can be controlled by choosing h sufficiently small. For two- and three-dimensional scatterers, the accumulation of errors is more difficult to handle and may even lead to instabilities.

In the present paper, the one-dimensional technique is extended to a time-domain scattering problem in more than one dimension. In doing so, we first derive two so-called stability criteria for the discretization of the space-time integral which, if they are met, guarantee that the instability can be controlled by reducing the discretization step. Next, we illustrate the role of these criteria in the stability analysis of the marching-on-in-time method by reconsidering the one-dimensional problem discussed in [2, 3] of an inhomogeneous, lossy dielectric slab embedded in vacuum which is excited by a pulsed electromagnetic plane wave. Also, we generalize the solution scheme developed for that case to the related problem of a slab sandwiched between two homogeneous, lossless dielectric halfspaces.

Finally, we use the stability criteria to analyze the numerical solution of two two-dimensional electromagnetic scattering problems. For

the problem of a pulsed wave incident on a perfectly conducting cylinder, which was previously investigated in [4, 5], the integral equation can be discretized such that the stability criteria are met, with the exception of a possible systematic error for late times. For an inhomogeneous, lossy dielectric cylinder we have only been able to partly meet these criteria. Nevertheless, the behavior of the solutions obtained can, as far as their stability is concerned, be understood with their aid. For all configurations, representative numerical results are presented and discussed.

2. STABILITY CRITERIA

For a general, one-, two- or three-dimensional linear time-domain scattering problem, the real-valued field $\phi(\underline{x},t)$, here assumed to be a scalar, satisfies an integral relation of the shape

$$(2.1) \qquad \phi(\underline{x},t) = \phi^i(\underline{x},t) + \int_D d\underline{x}' \int_0^{t-R/c} dt' K(\underline{x},\underline{x}';t-t')\phi(\underline{x}',t'),$$

with $\underline{x}$ a Cartesian position vector and $R = |\underline{x}-\underline{x}'|$. In (2.1), D denotes a finite domain, $K(\underline{x},\underline{x}';t-t')$ a linear, time-invariant operator acting on $\phi(\underline{x}',t')$, $\phi^i(\underline{x},t)$ a known incident field that for $\underline{x}\in D$ vanishes for $t \leq 0$, and c a wave speed parameter. Further $K = 0$ when $\underline{x}'\in D'$ where D' is the complement of the closure of D in the space under consideration. When $\underline{x}\in D$, $0 \leq t < \infty$, (2.1) is an integral equation that allows the numerical determination of ϕ with the marching-on-in-time method. In this method, we discretize in space and time, approximate the second term on the right-hand side of (2.1) accordingly, and invoke the equality sign in (2.1) at the relevant space-time points. To this end, we construct a uniform spatial grid $\{\underline{x}_\alpha\}$ with mesh size h and take $t = \Delta t$, where $m = 0, 1, 2, \ldots, \infty$. The time step Δt is chosen such that $\Delta t = \min(R_{\alpha\alpha'})/c$, where $R_{\alpha\alpha'} = |\underline{x}_\alpha - \underline{x}_{\alpha'}|$. Then we end up with algebraic equations of the type

$$(2.2) \qquad \tilde{\phi}(\alpha,m) = \tilde{\phi}^i(\alpha,m) + \sum_{\alpha'} \sum_{m'=0}^{m} \tilde{K}(\alpha,\alpha';m-m')\tilde{\phi}(\alpha',m'),$$

where $\tilde{\phi}^i(\alpha,m) = \phi^i(\underline{x}_\alpha,m\Delta t)$. Furthermore, the upper limit of the time integration in (2.1) is always less than t, unless $R = 0$ which occurs only if $\alpha' = \alpha$. Hence, owing to the choice of Δt, the interpolations can be organized such that $\tilde{K}(\alpha,\alpha';0) = 0$ if $\alpha' \neq \alpha$, while for passive obstacles the property $\tilde{K}(\alpha,\alpha;0) \leq 0$ can be shown to hold. Then, (2.2) can be solved by

a step-by-step updating procedure, involving only the solution, at each space-time point, of a linear equation for the unknown field value.

Due to the discretization of the multiple integral in (2.1), $\tilde{\phi}(\alpha,m)$ will only be approximately equal to the actual field value $\phi(\underline{x}_\alpha,m\Delta t)$. Since in the numerical solution of (2.2), each field value $\tilde{\phi}(\alpha,m)$ is computed from field values $\tilde{\phi}(\alpha',m')$ at previous instants, the computational errors due to this discretization accumulate. As a consequence, the solution obtained may be unstable. We will now derive conditions under which such an instability can be controlled. To this end, we consider $\phi(\underline{x}_\alpha,m\Delta t) - \tilde{\phi}(\alpha,m)$. Combining (2.1) and (2.2), we arrive at

$$
\begin{aligned}
&[1 - \tilde{K}(\alpha,\alpha;0)][\phi(\underline{x}_\alpha,m\Delta t) - \tilde{\phi}(\alpha,m)] \\
&= \int_D d\underline{x}' \int_0^{m\Delta t - R/c} dt' K(\underline{x}_\alpha,\underline{x}';m\Delta t - t')\phi(\underline{x}',t') \\
&\quad - \sum_{\alpha'} \sum_{m'=0}^{m} \tilde{K}(\alpha,\alpha';m-m')\phi(\underline{x}_{\alpha'},m'\Delta t) + A(\alpha,m),
\end{aligned}
\tag{2.3}
$$

with

$$
A(\alpha,m) = \sum_{\alpha'} \sum_{m'=0}^{m-1} \tilde{K}(\alpha,\alpha';m-m')[\phi(\underline{x}_{\alpha'},m'\Delta t) - \tilde{\phi}(\alpha',m')].
$$

In (2.3), the difference of the first two terms on the right-hand side denotes the effect of the discretization error in the updating step; the term $A(\alpha,m)$ represents the accumulation of the errors made in previous updating steps. Now *suppose* that for the *exact* field $\phi(\underline{x},t)$, the discretization of the integral in (2.1) satisfies the criterion

$$
\begin{aligned}
&\int_D dx' \int_0^{m\Delta t - R/c} dt' K(\underline{x}_\alpha,\underline{x}';m\Delta t - t')\phi(\underline{x}',t') \\
&= \sum_{\alpha'} \sum_{m'=0}^{m} \tilde{K}(\alpha,\alpha';m-m')\phi(\underline{x}_{\alpha'},m'\Delta t) + O(h^2),
\end{aligned}
\tag{2.4}
$$

uniformly in α and m, and let $e_m = \max_{\{\alpha\}}|\phi(\underline{x}_\alpha,m\Delta t) - \tilde{\phi}(\alpha,m)|$. Using the triangle inequality, we then obtain from (2.3) and (2.4):

$$
|A(\alpha,m)| \leq [1 - \tilde{K}(\alpha,\alpha;0)]e_m + O(h^2),
\tag{2.5}
$$

since $\tilde{K}(\alpha,\alpha;0) \leq 0$. Next *suppose* that the variation of $\tilde{\phi}(\alpha',m)$ with α' and m' allows the estimate

$$|A(\alpha,m)| \leq \max_{\{\alpha\}}|A(\alpha,m-1)| + O(h^2), \tag{2.6}$$

uniformly in α and m. (2.6) implies that the increase in the accumulated error $A(\alpha,m)$ per updating step is at most of the same order of magnitude as the effect of the discretization error. Combining (2.5) and (2.6), we then find

$$|A(\alpha,m)| \leq [1 - \tilde{K}(\alpha,\alpha;0)]e_{m-1} + O(h^2). \tag{2.7}$$

Substituting (2.4) and (2.7) into (2.3), using the triangle inequality, and taking the maximum over $\{\alpha\}$, we finally end up with

$$e_m = e_{m-1} + O(h^2), \tag{2.8}$$

while $e_0 = 0$ in view of the initial conditions. From (2.8), it follows by induction that $e_m = O(mh^2)$. Because the time step Δt is proportional to h, the error $|\phi(\underline{x}_\alpha,m\Delta t) - \tilde{\phi}(\alpha,m)|$ at the end of a finite time interval $0 < t < T_{max}$ will then at most be proportional to $(T_{max}/h)h^2 = T_{max}h$ and, hence, it can be controlled by choosing h sufficiently small.

For scattering by passive obstacles, the field ϕ generally becomes negligible after some finite instant T_{max}. Then the time-domain field can be determined without instabilities by discretizing the multiple integral in (2.1) such that requirements (2.4) and (2.6) or, equivalently, requirements (2.4) and (2.7) are met, and by choosing h sufficiently small. It is noted that the analysis presented above is a worst-case analysis. Therefore, a marching-on-in-time scheme may be stable although the conditions (2.4) and (2.6) are violated. Upon variation of h, however, the stability of the scheme will not behave as predicted above. We will now analyze some specific time-domain scattering problems using the stability criteria (2.4) and (2.6). In doing so, we will denote (2.4) as *criterion I*, while *criterion II* refers to the condition that at least one of the equivalent requirements (2.6) and (2.7) is met.

3. SCATTERING BY AN INHOMOGENEOUS, LOSSY DIELECTRIC SLAB

In the first problem, we consider the scattering by a one-dimensional inhomogeneous lossy dielectric slab (Fig. 1) of thickness d, embedded in vacuum (see also [2, 3]). The permittivity $\varepsilon(x)$ and the conductivity $\sigma(x)$

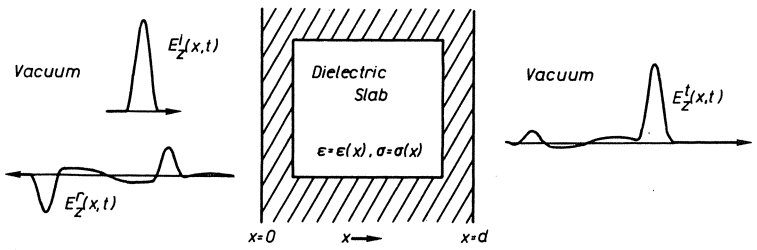


Figure 1. A pulsed plane wave normally incident on an inhomogeneous, lossy dielectric slab embedded in vacuum.

are assumed to be real-valued. Normally incident on the slab is an electromagnetic pulse of finite duration given by

$$(3.1) \qquad \underline{E}^i = F(t - x/c)\underline{i}_z \ , \ \underline{H}^i = -Y_0 F(t - x/c)\underline{i}_y$$

where $\underline{E}^i$ denotes the electric and $\underline{H}^i$ the magnetic field strength, $Y_0 = (\varepsilon_0/\mu_0)^{\frac{1}{2}}$, $c = (\varepsilon_0\mu_0)^{-\frac{1}{2}}$, with ε_0 and μ_0 being the permittivity and permeability in vacuo, respectively. The total electromagnetic field is then written as

$$(3.2) \qquad \underline{E} = E_z(x,t)\underline{i}_z \ , \ \underline{H} = H_y(x,t)\underline{i}_y .$$

In terms of the field components E_z and H_y, the source-free electromagnetic field equations in the slab are given by

$$(3.3) \qquad \begin{aligned} \partial_x H_y(x,t) &= [\varepsilon(x)\partial_t + \sigma(x)]E_z(x,t) \\ \partial_x E_z(x,t) &= \mu_0 \partial_t H_y(z,t). \end{aligned}$$

Elimination of H_y leads to

$$(3.4) \qquad \{\partial_x^2 - \mu_0[\varepsilon(x)\partial_t^2 + \sigma(x)\partial_t]\}E_z(x,t) = 0$$

which will be regarded as our fundamental differential equation. With the time-domain Green's function technique, the following integral relation is obtained, which is equivalent to (3.4):

$$(3.5)\qquad E_z(x,t) = E_z^i(x,t) - \frac{Z_0}{2}\int_0^d [\varepsilon_0\chi(x')\partial_t + \sigma(x')]E_z(x',t')dx',$$

where $\chi(x) = \varepsilon(x)/\varepsilon_0 - 1$ denotes the dielectric susceptibility, $Z_0 = (\mu_0/\varepsilon_0)^{\frac{1}{2}}$, and $t' = t - |x-x'|/c$. For $0 < x < d$ and $0 < t < \infty$, (3.5) is an integral equation for E_z. The second term on the right-hand side of (3.5) is identified with the reflected field $E_z^r(x,t)$ when $x < 0$, while the total field in $x > d$ is identified with the transmitted field $E_z^t(x,t)$. From (3.5), we then have the relations

$$(3.6)\qquad \begin{aligned} E_z^r(x,t) &= E_z^r(0,t + x/c) \qquad \text{for } x < 0, \\ E_z^t(x,t) &= E_z^t(d,t - (x-d)/c) \qquad \text{for } x > d, \end{aligned}$$

by which the field outside the slab can be obtained from the fields at the slab's interfaces.

In order to solve the integral equation (3.5) numerically, we discretize in space and time. When the space step is $h = d/N$ and the time step is $\Delta t = h/c$, this discretization results in:

$$(3.7)\qquad \begin{aligned} \tilde{E}_z(\ell,m) &= \tilde{E}_z^i(\ell,m) \\ &\quad -\tfrac{1}{2}\sum_{n=0}^{N} \alpha_n \frac{\tilde{\chi}(n)}{2h}\left[\frac{3}{2}\tilde{E}_z(n,m') - 2\,\tilde{E}_z(n,m'-2) + \tfrac{1}{2}\tilde{E}_z(n,m'-4)\right] \\ &\quad -\tfrac{1}{2}\sum_{n=0}^{N} \alpha_n \tilde{\sigma}(n)\tilde{E}_z(n,m') \end{aligned}$$

where $\ell = 0, 1, 2, \ldots, N$ and $m = 0, 1, 2, \ldots\infty$. In (3.7), we have $m' = m - |\ell - n|$, $\tilde{E}_z^i(\ell,m) = E_z^i(\ell h, m\Delta t)$, $\tilde{\chi}(n) = \chi(nh)$, $\tilde{\sigma}(n) = Z_0\sigma(nh)$, $\alpha_n = h$ for $0 < n < N$ and $\alpha_0 = \alpha_N = h/2$. The spatial integral in (3.7) has been approximated by a repeated trapezoidal rule and the time-derivative by a three-point backward interpolation formula with a time interval equal to the double time step. With the aid of the error estimates for these approximations (see [6, 7]), it follows directly that the discretization in (3.7) meets criterion I. If, as in [3], the time derivative were approximated by a two-point formula, the error in that approximation would be of $O(h)$ and, hence, criterion I would be violated.

In order to show rigorously that criterion II holds for the discretization in (3.7), we need additional information on the variation of $E_z(\ell h, m\Delta t) - \tilde{E}_z(\ell,m)$ with ℓ and m. Since such information is not available, only the following intuitive argument can be given. In the last term on the

right-hand side of (3.7), the total error is a superposition of N errors of $O(e_{m'})$, each of them multiplied by a factor h provided by the space integration. Therefore, it is at most proportional to $\max_{\{m'\}} e_{m'} = e_{m-1}$. In the second term on the right-hand side of (3.7), each error is multiplied by a factor 1/h due to the time differentiation as well as a factor h from the space integration. The total error is then a superposition of $O(N)$ errors of $O(e_{m-1})$ and will therefore depend on the sign distribution of these errors. Since the approximations leading to the discretization (3.7) are all based on interpolation in x or t, this sign distribution will be determined by the higher-order space and time derivatives occurring in the corresponding error estimates. Now $E_z(x,t)$ represents a pulsed wave which, due to repeated reflections at the slab's interfaces, travels backwards and forwards across the slab. For such a solution, the integrand in (3.5) and its derivatives vary with x'. As a consequence, the errors $E_z(nh,m'\Delta t) - \tilde{E}_z(n,m')$ will average out. With (2.5) at the previous instant, it then becomes plausible that (2.7), i.e. criterion II, is met.

The choice of the time interval of double length in the discretization of the time derivative in (3.7) allows a restriction of the numerical computation to space-time points $(\ell h, m\Delta t)$ where $\ell + m$ is even. A further reduction in computation time can be achieved by decomposing the summation in (3.7) according to $\sum_{n=0}^{N} = \sum_{n=0}^{\ell} + \sum_{n=\ell+1}^{N}$ and separating off the terms containing $\tilde{E}_z(\ell,m)$. This results into

$$
\begin{aligned}
\tilde{E}_z(\ell,m) = \{1 + \frac{\alpha_\ell}{2}[\frac{3}{4h}\tilde{\chi}(\ell) + \tilde{\sigma}(\ell)]\}^{-1} \times \\
\{\tilde{E}_z^i(\ell,m) + \tfrac{3}{2}[S_\chi^1(\ell-1,m-1) + S_\chi^2(\ell,m)] \\
- 2[S_\chi^1(\ell,m-2) + S_\chi^2(\ell,m-2)] \\
+ \tfrac{1}{2}[S_\chi^1(\ell,m-4) + S_\chi^2(\ell,m-4)] \\
+ S_\sigma^1(\ell-1,m-1) + S_\sigma^2(\ell,m)\}
\end{aligned}
\tag{3.8}
$$

where

$$
\begin{aligned}
S_\sigma^1(\ell,m) &= -\tfrac{1}{2}\sum_{n=0}^{\ell} \alpha_n \tilde{\sigma}(n)\tilde{E}_z(n,m-\ell+n), \\
S_\sigma^2(\ell,m) &= -\tfrac{1}{2}\sum_{n=\ell+1}^{N} \alpha_n \tilde{\sigma}(n)\tilde{E}_z(n,m+\ell-n);
\end{aligned}
\tag{3.9}
$$

similar definitions hold for S_χ^1 and S_χ^2. If $\tilde{E}_z(\ell',m')$ is known for

$0 \le \ell' \le N$ and $m' < m$, $\tilde{E}_z(\ell,m)$ can be determined by computing the right-hand side of (3.8). In the computation of the field at the next instant, the sums S_σ^1, S_σ^2, S_χ^1 and S_χ^2 need not be evaluated again but can be obtained from the recurrence relations

$$
\begin{aligned}
S_\sigma^1(\ell,m) &= S_\sigma^1(\ell-1,m-1) - \tfrac{1}{2}\alpha_\ell \tilde{\sigma}(\ell)\tilde{E}_z(\ell,m), \\
S_\sigma^2(\ell-1,m+1) &= S_\sigma^2(\ell,m) - \tfrac{1}{2}\alpha_\ell \tilde{\sigma}(\ell)\tilde{E}_z(\ell,m),
\end{aligned} \tag{3.10}
$$

and similar relations for S_χ^1 and S_χ^2. If these recurrence relations are used, the computation time is proportional to N^2 as $N\to\infty$, while a direct evaluation of (3.7) leads to a computation time of order N^3. Note that this approach differs from the related ones described in [2] and [3], where recurrence relations for $\tilde{E}_z(\ell,m)$ are used.

The computational scheme described above was implemented in PL/I on an Amdahl 470/V7B computer. Numerical results were obtained for several incident-pulse shapes $F(t)$ and various susceptibility and conductivity profiles. Numerical instabilities were not observed, even for rapid variations in the shape of the incident pulse and/or for small values of N,

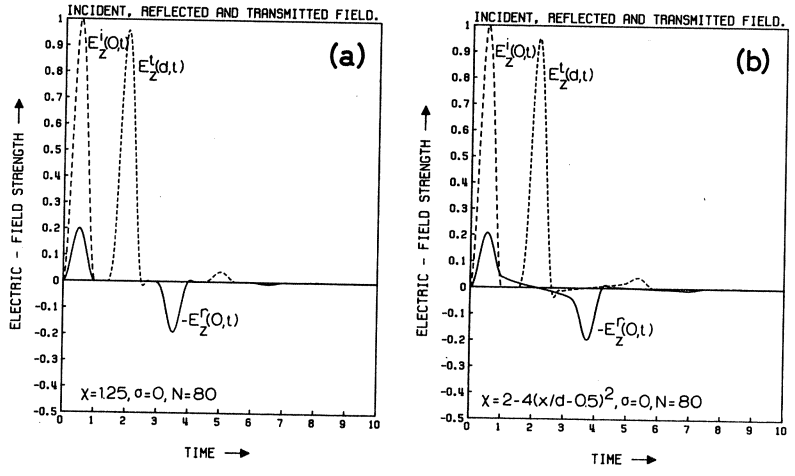


Figure 2. Incident, reflected and transmitted field in the case of (a) a lossless, homogeneous slab and (b) a lossless slab with a parabolic susceptibility profile. In both cases the incident field was given by $F(t)=\sin^2(\pi t/T)\,\mathrm{rect}(t-T/2;T)$ with $cT/d=1$. The time variable is ct/d.

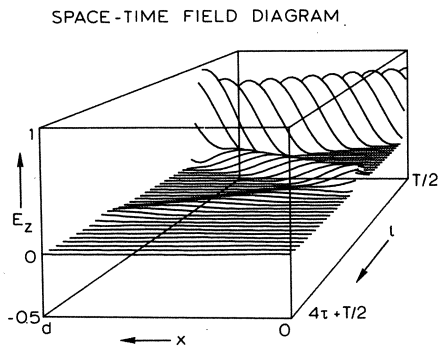


Figure 3. Three-dimensional space-time plot of the field inside the slab for 4 runs of the pulse specified in Fig. 2 through the slab specified in Fig. 2a. The travel time for a single traverse through the slab is denoted by τ with, in this case, $c\tau/d=1.5$.

for which cases the results become inaccurate. As an illustration, we present in Fig. 2a, Fig. 3 and in Table 1, results for a sine-squared pulse incident on a homogeneous, lossless slab. For that configuration, the solution is known in closed form in terms of transmitted and reflected waves. Fig. 2a shows the computed reflected and transmitted fields at the ends of the slab while Fig. 3 gives the field distribution inside the slab for the first four runs of the pulse across it. In Fig. 2a, a slight overshoot is observed at the instant where the directly transmitted wave has completely emerged from the slab. This can be explained by the fact that at that space-time point, the derivative $\partial_t^3 E_z(x,t)$, which occurs in the error estimate for the discretization of the time derivative, becomes unbounded. We have not taken special measures to prevent this effect since it turns out to vanish as N increases. In Table 1, we provide results of an accuracy test. For a lossless medium ($\sigma = 0$), we have the identity $E^{in} = E^{out}$, with

$$
\begin{aligned}
E^{in} &= Y_0 \int_0^\infty F^2(t)\,dt, \\
E^{out} &= Y_0 \int_0^\infty \{E_z^r(0,t)^2 + E_z^t(d,t)^2\}\,dt,
\end{aligned}
\tag{3.11}
$$

where E^{in} and E^{out} denote the total energy that, per unit surface, flows in and out of the slab, respectively. In Table 1, we have, for increasing N, checked this identity, compared the actual maximum values of $E_z(0,t)$ and $E_z(d,t)$ with the corresponding numerical values and listed the computation times for a fixed time interval. It is observed that the computation times are indeed proportional to N^2. Furthermore, the error in the computed fields decreases considerably faster than was predicted by the worst-case estimate of $O(1/N)$.

Table 1. Computational data from the numerical solution of the scattering problem specified in Figs. 2a and 3 for increasing N. The outgoing energy and the computation time correspond to the time interval $0<ct/d<20$.

	$(\varepsilon_0 d)^{-1} E^{out}$	error (%)	$E_z(d,\tau+T/2)$	$E_z(0,2\tau+T/2)$	CPU time
exact	0.375	-	0.96	0.192	-
N=10	0.291	28.7	0.665	0.116	0.38s
N=20	0.344	9.1	0.842	0.155	1.50s
N=40	0.370	1.4	0.946	0.186	6.05s
N=80	0.3743	0.18	0.958	0.1916	23.8 s
N=160	0.37492	0.02	0.9598	0.191993	96.0 s

Similar numerical experiments were carried out for the discretization presented in [3], where a two-point formula is employed to approximate the time derivative. It turns out that for incident pulses of short durations ($cT/d \leq 1$), computational problems arise that cannot be removed by increasing N. Finally, in Fig. 2b, results were plotted for an inhomogeneous, lossless slab. As in Fig. 2a, discrete reflected and transmitted pulses are observed originating from repeated reflections and transmissions at the slab's interfaces. In addition, a continuous reflected and transmitted field is observed in the time intervals in between, caused by the inhomogeneity of the slab.

4. SCATTERING BY AN INHOMOGENEOUS, LOSSY DIELECTRIC SLAB IN BETWEEN TWO HOMOGENEOUS, LOSSLESS HALFSPACES

In the second problem, we consider the dielectric slab discussed in Section 3 sandwiched between two homogeneous, lossless dielectric halfspaces. The configuration then consists of three domains $\mathcal{V}_i$ (with $i = 1, 2, 3$) as indicated in Table 2. As in Section 3, we want to determine the electric field strength in this configuration caused by a linearly polarized electromagnetic pulse of finite duration T which is normally incident from $\mathcal{V}_1$. In principle, this problem can be solved by forming an integral equation of the type (3.5), in which a homogeneous, lossless dielectric with $\varepsilon = \varepsilon_1$ is treated as the reference medium:

Table 2. Subdivision of the configuration into domains

domain	x-coordinate	permittivity	conductivity	permeability
$\mathcal{D}_1$	$-\infty<x<0$	$\varepsilon(x)=\varepsilon_1$	$\sigma(x)=0$	$\mu(x)=\mu_0$
$\mathcal{D}_2$	$0<x<d$	$\varepsilon(x)=\varepsilon_2(x)\geq\varepsilon_0$	$\sigma(x)=\sigma_2(x)\geq 0$	$\mu(x)=\mu_0$
$\mathcal{D}_3$	$d<x<\infty$	$\varepsilon(x)=\varepsilon_3$	$\sigma(x)=0$	$\mu(x)=\mu_0$

$$(4.1)\qquad E_z(x,t) = E_z^i(x,t) - \frac{Z_1}{2}\int_0^\infty \{[\varepsilon(x') - \varepsilon_1]\partial_t + \sigma(x')\}E_z(x',t')dx',$$

where $E_z^i(x,t) = F(t - x/c_1)$, $Z_1 = (\mu_0/\varepsilon_1)^{\frac{1}{2}}$ and $t' = t - |x-x'|/c_1$ with $c_1 = (\varepsilon_1\mu_0)^{-\frac{1}{2}}$. However, application of the marching-on-in-time method for the integral equation (4.1) clearly results in a solution for which the wave front propagates with a speed of at most c_1. As a consequence, the solution obtained will be incorrect if the actual wave speed $c(x) = [\varepsilon(x)\mu_0]^{-\frac{1}{2}}$ locally exceeds c_1, i.e. $\varepsilon(x) < \varepsilon_1$ for some x. In fact, numerical experiments for such cases produced results that increased exponentially with increasing t. Furthermore, for $\varepsilon_3 \neq \varepsilon_1$, the range of the space integration in (4.1) becomes semi-infinite, which requires special care if the electric field is to be computed for large t.

Both these problems can be circumvented by choosing for the reference medium a lossless, piecewise-homogeneous, three-layer medium with $\varepsilon(x) = \varepsilon_1, \varepsilon_0, \varepsilon_3$ for x in $\mathcal{D}_1, \mathcal{D}_2, \mathcal{D}_3$, respectively. With the time-domain Green's function for that configuration, the following integral relation is obtained

$$\begin{aligned}(4.2)\qquad E_z(x,t) = {} & E_z^a(x,t) \\ & - \frac{Z_0}{2}\int_0^d J_z\Big[x',t - |x-x'|/c\Big]dx' \\ & + \frac{Z_0}{2} R_1\sum_{n=0}^\infty (R_1R_3)^n\int_0^d J_z\Big[x',t - [x'+x+2nd]/c\Big]dx' \\ & + \frac{Z_0}{2} R_3\sum_{n=0}^\infty (R_1R_3)^n\int_0^d J_z\Big[x',t - [(d-x')+(d-x)+2nd]/c\Big]dx' \\ & - \frac{Z_0}{2} R_1R_3\sum_{n=0}^\infty (R_1R_3)^n\int_0^d\Big\{J_z\Big[x',t - [(d-x')+x+(2n+1)d]/c\Big] \\ & \qquad + J_z\Big[x',t - [x' + (d-x)+(2n+1)d]/c\Big]\Big\}dx',\end{aligned}$$

where $J_z(x,t)$ denotes the polarization current density

$$J_z(x,t) = [\varepsilon_0 \chi(x) \partial_t + \sigma(x)] E_z(x,t).$$

In (4.2) we have $R_1 = (N_1 - 1)/(N_1 + 1)$ with $N_1 = (\varepsilon_1/\varepsilon_0)^{\frac{1}{2}}$ and a similar definition for R_3, where R_1 and R_3 denote the plane-wave reflection coefficients for a boundary between the corresponding medium and vacuum. The auxiliary field $E_z^a(x,t)$ is the field that would result from the incident pulse in the reference medium:

$$(4.3) \qquad E_z^a(x,t) = \begin{cases} F(t - x/c_1) + R_1 F(t + x/c_1) \\ \qquad - T_1^+ R_3 T_1^- G(t + x/c_1 - 2d/c) & \text{in } \mathcal{D}_1, \\ T_1^+ G(t - x/c) - T_1^+ R_3 G\left(t - [(d-x) + d]/c\right) & \text{in } \mathcal{D}_2, \\ T_1^+ T_3^+ G\left(t - (x-d)/c_3 - d/c\right) & \text{in } \mathcal{D}_3, \end{cases}$$

with

$$G(t) = \sum_{n=0}^{\infty} (R_1 R_3)^n F(t - 2nd/c).$$

In (4.3), we have the plane-wave transmission coefficients $T_1^+ = 2N_1/(N_1+1)$, $T_1^- = 2/(N_1+1)$ and $T_3^+ = 2/(N_3+1)$ with the superscripts + and - referring to the direction of propagation. The integral equation given by (4.2) and (4.3) does not suffer from the same difficulties as the one given by (4.1) and, hence, does allow the application of the marching-on-in-time method. Although the equation obtained seems complicated, it can be solved by a recursive scheme similar to that employed in Section 3. Actually, the relations (3.8) and (3.10) still hold for $0 < \ell < N$. For $\ell = 0,N$, we have relations of the same kind involving the relevant reflection factors R_1 and R_3.

With the computational scheme resulting from (4.2) and (4.3), we have also performed a number of numerical experiments. The solutions obtained showed the same accuracy and stability behaviour that was observed for the case of the single slab. As an illustration, an example is shown in Fig. 4.

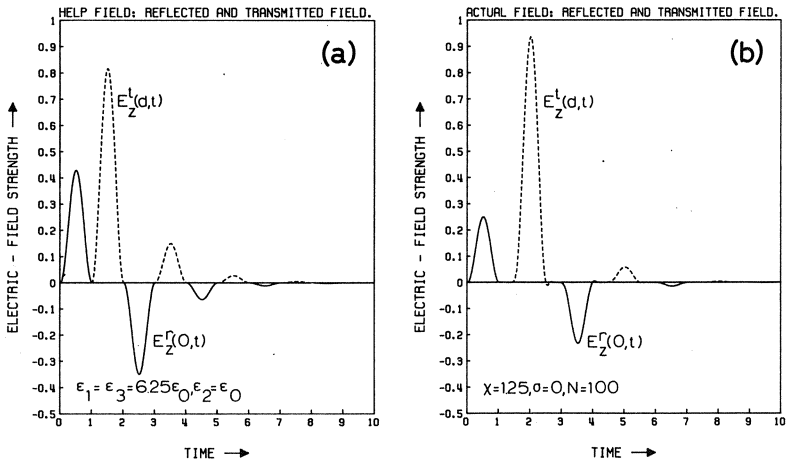


Figure 4. Reflected and transmitted field for the scattering of the incident field $F(t)=\sin^2(\pi t/T)\,\mathrm{rect}(t-T/2;T)$ with $cT/d=1$ in a three-media configuration with $\varepsilon_1=\varepsilon_3=6.25\varepsilon_0$, $\varepsilon_2=2.25\varepsilon_0$ and $\sigma(z)=0$. (a) auxiliary field as specified in (4.3); (b) actual field computed for N=100. The time variable is ct/d.

5. SCATTERING BY A PERFECTLY CONDUCTING CYLINDER

In the third problem, a pulsed electromagnetic plane wave of finite duration T is perpendicularly incident on a perfectly conducting cylinder (see Fig. 5). For this configuration, the magnetic field satisfies the boundary integral equation (see [4, 5])

$$\underline{J}(\underline{\rho},t) = 2\underline{n}(\underline{\rho}) \times \underline{H}^i(\underline{\rho},t) + \underline{n}(\underline{\rho}) \times \frac{1}{\pi}\int_C ds' \left\{ \nabla \times \int_0^{t-R/c} \frac{dt'}{[(t-t')^2 - R^2/c^2]^{\frac{1}{2}}}\, \underline{J}(\underline{\rho}',t') \right\}. \tag{5.1}$$

In (5.1), $\underline{J}(\underline{\rho},t)$ denotes the equivalent surface current $\underline{n}(\underline{\rho}) \times \underline{H}(\underline{\rho},t)$. In order to simplify this equation, we first carry out the curl operation according to

$$\nabla \times \int_0^{t-R/c} \frac{dt'}{[(t-t')^2 - R^2/c^2]^{\frac{1}{2}}}\, \underline{J}(\underline{\rho}',t') = -\underline{i}_R \times \frac{1}{c}\int_0^{t-R/c} \frac{dt'}{[(t-t')^2 - R^2/c^2]^{\frac{1}{2}}} \left\{ \frac{\underline{J}(\underline{\rho}',t')}{t-t'+R/c} + \partial_{t'}\underline{J}(\underline{\rho}',t') \right\}, \tag{5.2}$$

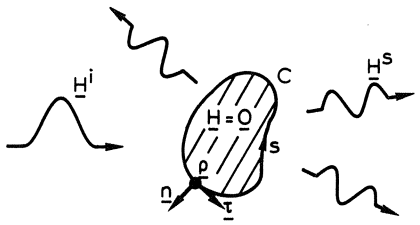


Figure 5. Scattering of pulsed electromagnetic plane wave by a perfectly conducting cylinder.

where $\underline{i}_R = (\underline{\rho}-\underline{\rho}')/R$. Next, we integrate by parts the first term on the right-hand side of (5.2). Assuming that $\underline{H}^i(\underline{\rho},t)$, and therefore also $\underline{H}(\underline{\rho},t)$, is a continuously differentiable function of t, we end up with:

$$\nabla \times \int_0^{t-R/c} \frac{dt'}{[(t-t')^2 - R^2/c^2]^{\frac{1}{2}}}\, \underline{J}(\underline{\rho}',t') = -\underline{i}_R \times \frac{1}{R}\int_0^{t-R/c} dt' \frac{t-t'}{[(t-t') - R^2/c^2]^{\frac{1}{2}}}\, \partial_{t'}\underline{J}(\underline{\rho}',t'), \tag{5.3}$$

with $\underline{i}_R = (\underline{\rho}-\underline{\rho}')/R$. Substitution of (5.3) and use of the definition of $\underline{J}(\underline{\rho},t)$ in (5.1) finally results in two scalar integral equations for the tangential components of $\underline{H}$:

$$\begin{aligned} H_z(\underline{\rho},t) &= 2H_z^i(\underline{\rho},t) + \frac{1}{\pi}\oint_C ds' \frac{(\underline{i}_R\cdot\underline{n}')}{R} I_z(\underline{\rho}';t,R), \\ H_\tau(\underline{\rho},t) &= 2H^i(\underline{\rho},t) + \frac{1}{\pi}\oint_C ds' \frac{(\underline{i}_R\cdot\underline{n})}{R} I_\tau(\underline{\rho}';t,R), \end{aligned} \tag{5.4}$$

with

$$I_{z,\tau}(\underline{\rho}';t,R) = \int_0^{t-R/c} dt' \frac{t-t'}{[(t-t')^2 - R^2/c^2]^{\frac{1}{2}}}\, \partial_{t'} H_{z,\tau}(\underline{\rho}',t').$$

These equations are now in a form that is suitable for the application of the marching-on-in-time method, as discussed in Section 2. Since both equations are solved by the same procedure, we restrict the discussion to the case where the magnetic field has only a z-component (H-polarization). Following Bennett [4], we restrict the space points to the finite set $\{\underline{\rho}_\ell\}$, $\ell = 1, \ldots, N$. The path length between two neighboring points is $h = L/N$, with L the total length of the contour C. The contour integral is approxi-

mated by a repeated rectangular rule. As a function of s', the integrand of (5.4) is a periodic, differentiable function with an integrable derivative. Therefore, it follows from discrete Fourier theory that the approximation error is proportional to h^2 (see [8]). This leaves us with the determination of the integrand to the same order of accuracy. For a specific obstacle, we assume the factor $(\underline{i}_R \cdot \underline{n}')/R$ to be known for $R > 0$, while for $R \to 0$, we have $(\underline{i}_R \cdot \underline{n}')/R \to -1/2a(\underline{\rho})$, with $a(\underline{\rho})$ the radius of curvature at the point of observation. To determine the time integral for $R > 0$, we restrict R to multiples of $c\Delta t$. For $t=m\Delta t$ and $R=kc\Delta t$, the time integral reduces to

$$I_z(\underline{\rho}';m\Delta t,kc\Delta t) = \sum_{m'=0}^{m-k-1} \int_{m'\Delta t}^{(m'+1)\Delta t} dt' \frac{m\Delta t - t'}{[(m\Delta t-t')^2 - k^2\Delta t^2]^{\frac{1}{2}}} \times \partial_{t'} H_z(\underline{\rho}',t'). \tag{5.5}$$

The integrals over the subintervals $m'\Delta t < t' < (m'+1)\Delta t$ are obtained by approximating $H_z(\underline{\rho}',t')$ by a quadratic interpolation polynomial in t', also using the field value at $t' = (m'-1)\Delta t$. The time differentiation and the subsequent integration are carried out analytically. By considering the interpolation error (see [6, 7]), it can be shown that this procedure leads to an approximation of order h^2 of the time integral $I_z(\underline{\rho}_\ell;m\Delta t, kc\Delta t)$. The integral $I_z(\underline{\rho}_{\ell'};m\Delta t,R_{\ell\ell'})$, which is required in the discretized contour integral, is obtained from the values at $R = kc\Delta t$ by linear interpolation in R. For small R, this may not seem justified since the approximation error is proportional to $h^2\partial_R^2 I_z$, with $\partial_R^2 I_z$ logarithmically singular at $R = 0$. However, the interpolation only needs to be carried out for $R > c\Delta t$, and in that region the interpolation error is of order $h^2\ln(h)$. Hence, the total error in the discretization is also of order $h^2\ln(h)$. Since the analysis presented in Section 2 can directly be generalized to include the factor of ln(h) in the discretization error, we may consider criterion I to be met.

As for the case of the dielectric slab, the complicated form of the integrals in (5.4) makes it hard to show that criterion II holds for the corresponding discretized form. Again, we have to be content with the following intuitive argument. In the time differentiation, an error of $O(e_{m'})$ is multiplied by a factor 1/h while integration over a single space coordinate provides a factor h. Such a factor h cannot be attributed to the time integral because of the singular behavior of its integrand near $t' = t-R/c$. In a sum of the type (2.2), the total error is then a super-

position of $O(N)$ errors of $O(e_{m-1})$ and will therefore depend on the sign distribution of these errors. Since all the approximations made above are based on interpolation in either s, R or t', this sign distribution will be determined by the higher-order space and time derivatives occurring in the corresponding error estimates. For a wave-like solution, these derivatives vary in space and time and, hence, the errors $\phi(\underline{x}_{\alpha'},m'\Delta t) - \tilde{\phi}(\alpha',m')$ will tend to average out. In that case, the discretization satisfies criterion II. On the other hand, if the solution is constant in space for a non-vanishing time interval, it may, at some instant, exhibit a systematic error at all space points. Such an error will be amplified in the next few time steps and, hence, a systematic instability in the solution will be observed.

The computational scheme described above was implemented in PL/I on an Amdahl 470/V7B computer for both polarizations. In Fig. 6, results are presented for a circular cylinder with radius a, illuminated by an E-polarized sine-squared incident pulse of duration cT/a = 2 for 32 and 64 points on the integration contour. It is observed that the instabilities can indeed be controlled by reducing $h = 2\pi a/N$. For large t, the tangential field

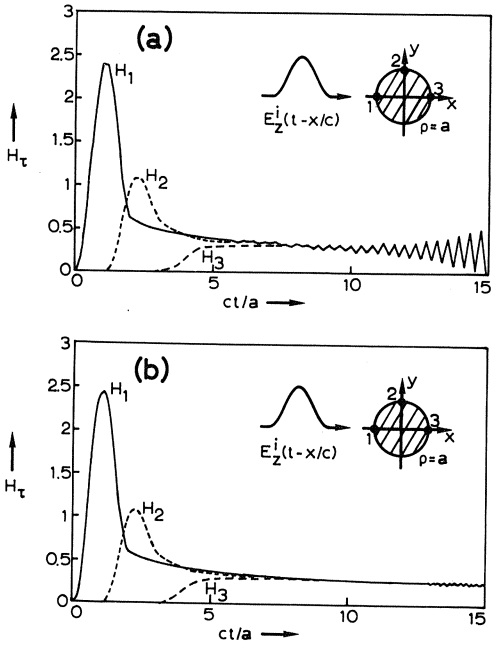


Figure 6. H_τ as a function of the normalized time ct/a for a perfectly conducting cylinder illuminated by the pulse specified in (3.1) for $F(t)=Z_0\sin^2(\pi t/T) \times$ $\times$ rect(t-T/2;T) with cT/a=2. Figs. (a) and (b) show the results computed with N=32 and N=64, respectively.

turns out to be constant around the contour. Obviously, the incident field excites a "stationary" current propagating in the z-direction and a corresponding "stationary" magnetic field around the cylinder. Note that the onset of this phenomenon coincides in time with the start of the - systematic - instabilities. For H-polarization, the incident field only induces a current propagating in the transverse direction, which becomes negligible after some finite instant. Until that instant, the induced current, and hence H_z, vary along the contour C. As a consequence, the errors average out and systematic errors as observed for E-polarization do not show up.

A disadvantage of the technique is the sharp increase in computation time upon reduction of h. For the configuration considered here, the number of operations is approximately proportional to N^4. For the field shown in Fig. 6, the computation times were 1, 10 and 130 seconds for N=16, 32 and 64, respectively. The results obtained are in agreement with a reference solution obtained by applying an FFT-algorithm to the frequency-domain Fourier-Bessel series solution (see [9]).

6. SCATTERING BY AN INHOMOGENEOUS, LOSSY DIELECTRIC CYLINDER

Finally, we turn our attention to the scattering of an E-polarized pulsed plane wave by an inhomogeneous, lossy dielectric cylinder (see Fig. 7), i.e. the two-dimensional equivalent of the problem discussed in Section 3. For this configuration we have, as in (3.4), the fundamental differen-

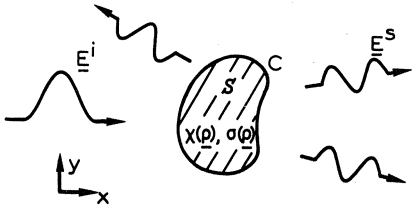


Figure 7. Scattering of a pulsed electromagnetic plane wave by an inhomogeneous, lossy dielectric cylinder.

tial equation

$$(6.1)\qquad \{\partial_x^2 + \partial_y^2 - \mu_0[\varepsilon(x,y)\partial_t^2 + \sigma(x,y)\partial_t]\}E_z(x,y,t) = 0.$$

Using the free-space Green's function, we end up with the following integral relation for the electric field:

$$(6.2)\qquad E_z(x,y,t) = E_z^i(x,y,t) - \frac{\mu_0}{2\pi}\iint_S dx'dy' \int_0^{t-R/c} dt' \frac{\partial_{t'} J_z(x',y',t')}{[(t-t')^2 - R^2/c^2]^{\frac{1}{2}}}.$$

In (6.2), $J_z(x,y,t)$ denotes the polarization current density

$$(6.3)\qquad J_z(x,y,t) = \sigma(x,y)E_z(x,y,t) + \varepsilon_0\chi(x,y)\partial_t E_z(x,y,t),$$

with σ the conductivity and χ the electric susceptibility. Equation (6.2) is an integral equation if $(x,y)\in S$. Compared to the integral equations discussed in the previous sections, we now have the additional difficulty of a logarithmic singularity in the space integration. This singularity shows up explicitly if we carry out a partial integration with respect to time. For continuous $\partial_t J_z$ in t, we arrive at

$$\begin{aligned}(6.4)\qquad E_z(z,y,t) = {} & E_z^i(x,y,t) + \frac{\mu_0}{2\pi}\iint_S dx'dy' \left\{\ell nR\ \partial_t J_z(x',y',t-R/c)\right\} \\ & - \frac{\mu_0}{2\pi}\iint_S dx'dy' \int_0^{t-R/c} dt' \left\{\ell n\left[t-t' + [(t-t')^2 - R^2/c^2]^{\frac{1}{2}}\right] \times \right. \\ & \left. \times\ \partial_{t'}^2 J_z(x',y',t')\right\},\end{aligned}$$

where in the second integral, the integrand remains bounded as $R\to 0$.

In discretizing the integrals in (6.2) or (6.4), the space time points were limited to: $x_k = kh$, $y_\ell = \ell h$, $t_m = m\Delta t$ with $\Delta t = h/c$. The boundary of the domain of integration was piecewise approximated by straight lines within square subdomains of width h. The time integrals were handled by approximating, as in Section 5, the field as a function of time by an interpolation polynomial of sufficiently high degree. Two different methods were attempted to determine the space integral. In method I, the conventional approach was followed and $J_z(x',y',t')$ was approximated by a piecewise-constant function:

$$(6.5)\qquad J_z(x,y,t) = [\sigma(k,\ell) + \varepsilon_0\chi(k,\ell)\partial_t]J_z(x_k,y_\ell,t)$$

for $(x,y)\in S\cap S_{k\ell}$, where $S_{k\ell}$ denotes the domain $\max(|x-x_k|,|y-y_\ell|) < h/2$ and where $\sigma(k,\ell)$ and $\chi(k,\ell)$ are taken at some point inside $S\cap S_{k\ell}$. Substitution of (6.5) in (6.2) results, for each combination of a point (x_k,y_ℓ) and a square $S_{k'\ell'}$, in an integral over $S_{k'\ell'}$ of a function of $R_{k\ell} = |\underline{\rho}'-\underline{\rho}_{k\ell}|$ only. For the self-patch domain $S_{k\ell}$, this function is integrated analytically; the contribution from the remaining patches is obtained by analytical integration of a linear approximation in $R_{k\ell}$.

In method II, we approximated the term $\partial_t J_z(x',y',t-R/c)$ in the first space integral of (6.4) and the complete integrand of the second one by the bilinear interpolation

$$\tilde{\delta}(x,y) = A_{k\ell} + B_{k\ell}(x - x_k) + C_{k\ell}(y - y_\ell) + D_{k\ell}(x - x_k)(y - y_\ell) \tag{6.6}$$

for $x_k < x < x_{k+1}$ and $y_\ell < y < y_{\ell+1}$ and determined the resulting space integrals analytically. In (6.6), the factors A, B, C and D are chosen such that $\tilde{\delta}(x_k,y_\ell) = \delta(x_k,y_\ell)$ for all k and ℓ. It is noted that neither of the discretizations meets criterion I. In method I, (6.5) already violates this criterion. In method II, we have the error estimate:

$$|\tilde{\delta}(x,y) - \delta(x,y)| < \frac{3}{8}\max\left(|\partial_x^2\delta(x',y')|,|\partial_y^2\delta(x',y')|\right)h^2, \tag{6.7}$$

where (x',y') varies over the range of the interpolation. Since the derivatives in (6.7) become unbounded as $R\to 0$, especially the contributions of surface patches near the point of observation are not computed with sufficient accuracy.

Numerical results have been obtained for both discretization schemes described above. The results obtained by scheme I exhibit a short-term instability. This instability is caused by interpolation errors at the space points next to the point of observation, where at $t' = t-R$ the time derivative must be determined by backward time interpolation. It can be removed by using, after each time step, the field obtained to determine a new approximation to those time derivatives by central time interpolation instead of backward time interpolation. The computed field values are subsequently improved by correcting the discretized version of the integral in (6.2) for half the difference of the two approximations of the relevant time-derivatives. With this correction procedure, the computational scheme yields stable results for a fairly wide range of incident-pulse durations and contrasts. An example is shown in Fig. 8. The computation times are

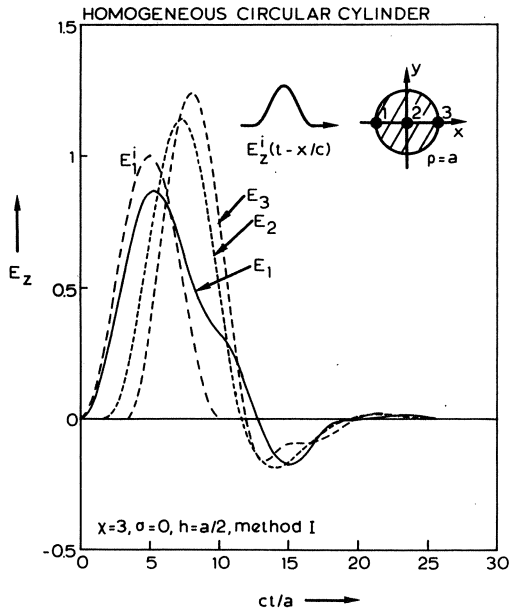


Figure 8. E_z as a function of the normalized time ct/a for a homogeneous, circular cylinder with $\chi=3$ and $\sigma=0$ illuminated by the pulse specified in (3.1) for $F(t)=\sin^2(\pi t/T) \times$ $\times$ rect$(t-T/2;T)$ with $cT/a=10$ at the points indicated in the inset (method I, $h=a/2$).

approximately proportional to N^5. For a not too small value of h, however, they are of the same order of magnitude as in the previous section. For a homogeneous circular cylinder, the results coincide with a reference solution obtained from the frequency-domain Fourier-Bessel series solution (see [9]). For too short incident-pulse lengths or too high contrasts, long-term instabilities are observed at late times. Since the discretization violates criterion I, a reduction in the discretization step does not remove these instabilities.

Scheme II has until now only been implemented for a square cylinder with diameter 2a. For $\chi = 0$, the results show, for a finite range of h, a stability behavior similar to that in the perfectly conducting case. For small σ, a stable solution can be obtained by reducing h. For larger σ, we again observe an almost stationary current $J_z = \sigma E_z$ which does not vary spatially within S. As in the previous section, it can be argued intuitively that such a solution may become unstable due to a systematic error at all space points. An example is shown in Fig. 9a. Similarly, it follows that for $\chi > 0$, instabilities can already be caused by a systematic error along a single line $x = x_k$. Both these systematic errors can be removed by using the second-order differential equation (6.1) to regularize the

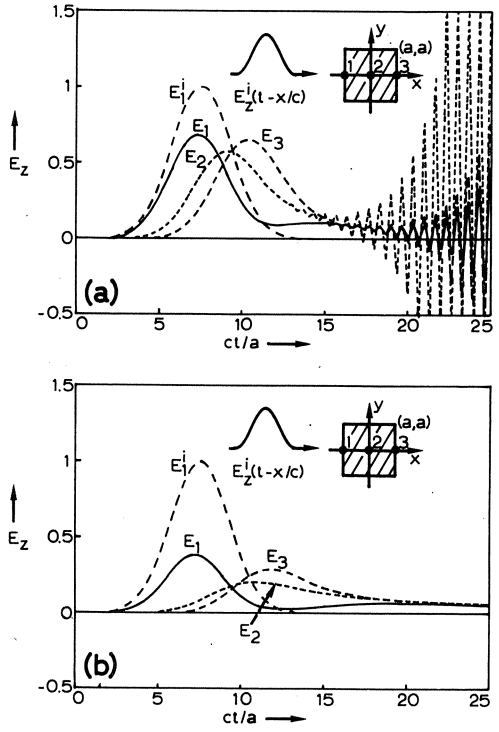


Figure 9. E_z as a function of the normalized time ct/a for a homogeneous square cylinder with $\chi=0$ illuminated by the pulse specified in (3.1) for $F(t)=\exp(-16(t-T)^2/T^2)$ with $cT/a=10$ computed with method II for $h=a/3$. (a): $Z_0\sigma d=2$, without regularization; (b): $Z_0\sigma d=10$, regularization with $p=q=1$.

marching-on-in-time scheme. This regularization proceeds as follows. If the mesh size h is chosen as $h = a/N$ and $\chi(x,y)$ and $\sigma(x,y)$ are continuous inside S, (6.1) can be discretized as

$$\begin{aligned}&\tilde{E}_z(k+1,\ell,m) + \tilde{E}_z(k-1,\ell,m) + \tilde{E}_z(k,\ell+1,m) + \tilde{E}_z(k,\ell-1,m) - 4\tilde{E}_z(k,\ell,m)\\ &- h\tilde{\sigma}(k,\ell)[\tfrac{3}{2}\tilde{E}_z(k,\ell,m) - 2\tilde{E}_z(k,\ell,m-1) + \tfrac{1}{2}\tilde{E}_z(k,\ell,m-2)]\\ &- [1 + \tilde{\chi}(k,\ell)][2\tilde{E}_z(k,\ell,m) - 5\tilde{E}_z(k,\ell,m-1) + 4\tilde{E}_z(k,\ell,m-2)\\ &\qquad - \tilde{E}_z(k,\ell,m-3)] = 0\end{aligned} \tag{6.8}$$

for $-N < k < N$, $-N < \ell < N$ and $m = 0, 1, 2, \ldots, \infty$. The equality sign in (6.8) holds up to $O(h^4)$ if $\tilde{E}_z(k,\ell,m)$ is replaced by $E_z(kh,\ell h,m\Delta t)$. For $E_z^i(x,y,t)$ and, hence, $E_z(x,y,t)$ three times differentiable with respect to t, we obtain, by quadratic extrapolation

$$\tilde{E}_z(k,\ell,m) = 3\tilde{E}_z(k,\ell,m-1) - 3\tilde{E}_z(k,\ell,m-2) + \tilde{E}_z(k,\ell,m-3) \tag{6.9}$$

for $-N \le k \le N$, $-N \le \ell \le N$ and $m = 0, 1, 2, \ldots, \infty$ to the same order of accuracy. Next, we introduce the squared error $D_m(p,q)$ as

$$D_m(p,q) = \sum_{k=-N}^{N} \sum_{\ell=-N}^{N} \delta_1(k,\ell,m)^2 + p \sum_{k=-N+1}^{N-1} \sum_{\ell=-N+1}^{N-1} \delta_2(k,\ell,m)^2 + pq\left\{\sum_{k=\pm N} \sum_{\ell=-N}^{N} \delta_3(k,\ell,m)^2 + \sum_{k=-N+1}^{N-1} \sum_{\ell=\pm N} \delta_3(k,\ell,m)^2\right\}, \tag{6.10}$$

where δ_1, δ_2 and δ_3 denote the deviations in the equality signs of the discretized form of (6.4) and of equations (6.8) and (6.9), respectively and p and q are nonnegative regularization parameters. $\tilde{E}_z(k,\ell,m)$ is then redefined as that combination of field values that, for m fixed, minimizes $D_m(p,q)$ for given field values at the previous instants. The product pq should be chosen as small as possible since (6.9) is not based on (6.1) or an equivalent integral equation. For the case $\chi = 0$, a representative example is shown in Fig. 9b. For large susceptibilities ($\chi > 4$), we again encounter long-term instabilities at late times. These errors cannot be removed by a further reduction of h, since for small h, the -inaccurate- contributions from the neighboring points are relatively more important.

Presently, an alternative discretization scheme that does meet criterion I is under investigation. Results are not yet available.

7. CONCLUSION

In this contribution, we have formulated two so-called stability criteria for the discretization of time-domain integral equations in the application of the marching-on-in-time method. If these criteria are met, the instability in the numerical solution can be controlled by reducing the discretization step. The relevant analysis of some simple electromagnetic scattering problems shows that generally, the integrals can be discretized with the required accuracy. The effect of errors made in previous updating steps can, until now, only be understood intuitively. In view of the results obtained, it seems worthwile to further investigate this matter. Also, it would be interesting to generalize the technique to more complicated scattering problems. For such cases, however, the computation time may well be a limiting factor, since the main disadvantage of the technique is the sharp increase in computation time upon reduction of the discretization step.

ACKNOWLEDGEMENTS. The author wishes to thank Professor H. Blok and Professor A.T. de Hoop for their helpful comments, and also Mr. P.C. Kempen for carrying out the numerical computations for the problem discussed in Section 5.

REFERENCES

[1] HERMAN, G.C., *Scattering of transient acoustic waves by fluids and solids*, Ph.D. Thesis, 183 pp., Delft University of Technology, 1981.

[2] TIJHUIS, A.G., *Iterative determination of permittivity and conductivity profiles of a dielectric slab in the time domain*, IEEE Trans. Antennas Propagat., AP-29, 239-245, 1981.

[3] BOLOMEY, J.Ch., Ch. DURIX & D. LESSELIER, *Time domain integral equation approach for inhomogeneous and dispersive slab problems*, IEEE Trans. Antennas Propagat., AP-27, 244-248, 1979.

[4] BENNETT, C.L., *A technique for computing approximate electromagnetic impulse response of conducting bodies*, Ph.D. Thesis, 146 pp., Purdue University, Lafayette, Ind., 1968.

[5] BENNETT, C.L. & W.L. WEEKS, *Transient scattering from conducting cylinders*, IEEE Trans. Antennas Propagat., AP-18, 627-633, 1970.

[6] HILDEBRAND, F.B., *Introduction to numerical analysis*, Chapter 3, McGraw-Hill, New York, 1956.

[7] STOER, J., *Einführung in die numerische Mathematik I*, pp. 41-44, Springer, Berlin, 1972 (German).

[8] TITCHMARSH, E.C., *The theory of functions* (second edition), pp. 399-404, Oxford University Press, London, 1950.

[9] JONES, D.S., *The theory of electromagnetism*, pp. 450-452, Pergamon Press, Oxford, 1964.

WEAKLY REFLECTIVE BOUNDARY CONDITIONS FOR TWO-DIMENSIONAL SHALLOW WATER FLOW PROBLEMS

G.K. VERBOOM & A. SLOB

In this paper weakly-reflective boundary conditions are derived for the two-dimensional shallow water equations, including bottom friction and Coriolis force. The essential aspects of the derivation are given. Zeroth and first order approximations are applied to the test problem of an initially Gaussian-shaped free surface elevation. For the numerical solution a finite element program is used and various aspect of the numerical implementation are discussed. For small scale practical problems a rather simple (one parameter) formulation might be sufficient. The influence of this parameter is discussed on the weakly-reflectiveness of the boundary condition.

1. INTRODUCTION

In the numerical solution of many hydraulic engineering problems waves play a dominant role. As the area of interest generally is just a small part of a much larger system artificial boundaries are introduced to obtain a limited domain. In nature waves can cross these artificial boundaries unhampered in both directions, but in our numerical model we must include this property explicitly. In literature so-called non- and weakly-reflective boundary conditions are derived for hyperbolic equations: Taylor 1975; Engquist and Majda 1977. For the quasi-linear one-dimensional shallow water equations Verboom 1982 derived upto second order weakly-reflective boundary conditions. For tidal wave problems reflection coefficients of only a few percent were realized with the Preissmann-scheme. For more general schemes numerical reflections limit the effect of higher order conditions.
In this paper we derive zeroth and first-order weakly-reflective boundary conditions for the two-dimensional shallow water equations including bottom friction and Coriolis force. The essential aspects of the derivation are given. Next the conditions are applied to the test problem of an initially

*) Paper presented at the 5th International Conference on Finite Elements in Water Resources, Vermont, June 18-22, 1984.

Gaussian-shaped free surface elevation. For the numerical solution we use a finite element program with bi-linear base functions and an explicit time integration scheme. For many practical problems in which the boundary conditions are derived from a larger area (coarser grid) model a much simpler formulation proposed by Stelling 1983 will do, Verboom et al. 1984.
This condition is discussed and results are reported for three values of the parameter included in that formulation.

2. THEORETICAL ASPECTS

Formulation of the problem.

The two-dimensional shallow water equations in primitive variables read

$$\vec{w}_t + A_1\vec{w}_x + A_2\vec{w}_y + A_3\vec{w} + \vec{F} = \vec{0}, \tag{1}$$

with $\vec{w} = (u,v,\zeta)^T$ and

$$A_1 = \begin{pmatrix} u & o & g \\ o & u & o \\ h+\zeta & o & u \end{pmatrix} \quad ; \quad A_2 = \begin{pmatrix} v & o & o \\ o & v & g \\ o & h+\zeta & v \end{pmatrix} \quad ; \quad A_3 = \begin{pmatrix} \lambda & -f & o \\ f & \lambda & o \\ 0 & o & o \end{pmatrix} .$$

The notation is explained at the end of the paper. The bottom friction parameter, λ, is generally given by

$$\lambda = \frac{g\sqrt{u^2+v^2}}{C^2(h+\zeta)} .$$

External forces, such as wind can be accounted for in $\vec{F}$. For most practical computations we solve System 1 with an ADI-finite difference program, but for specific relatively small scale problems we can use an explicit finite element program that solves the corresponding system written in conservative form. However, for the derivation of weakly-reflective boundary conditions it is advantageous to symmetrize and/or diagonalize matrix A_1 or A_2. As System 1 is a quasi-linear strictly hyperbolic system, A_1 and A_2 can be symmetrized simultaneously, but only one can be diagonalized at a time. Matrices A_1 and A_2 are symmetrized by the transformation

$$\psi = 2\sqrt{g(h+\zeta)}. \tag{2}$$

For the analysis in the following section we diagonalize matrix A_1 with the additional transformation

$$\vec{v} = V\,(u,v,\psi)^T$$

with

$$V = \begin{pmatrix} 1/\sqrt{2} & o & 1/\sqrt{2} \\ o & 1 & 0 \\ 1/\sqrt{2} & o & -1/\sqrt{2} \end{pmatrix} . \tag{3}$$

The system for $\vec{v}$ reads

$$\vec{v}_t + A\vec{v}_x + B\vec{v}_y + C\vec{v} + V\vec{F} = \vec{0}, \tag{4}$$

with $\vec{v} = (\frac{1}{\sqrt{2}}(u+\psi), v, \frac{1}{\sqrt{2}}(u-\psi))^T$ and

$$A = VA_1V^{-1} = \begin{pmatrix} u+\frac{1}{2}\psi & o & o \\ o & u & o \\ o & o & u-\frac{1}{2}\psi \end{pmatrix}$$

$$B = VA_2V^{-1} = \begin{pmatrix} o & \psi/2\sqrt{2} & o \\ \psi/2\sqrt{2} & o & -\psi/2\sqrt{2} \\ o & -\psi/2\sqrt{2} & o \end{pmatrix}$$

$$C = VA_3V^{-1} = \begin{pmatrix} \lambda/2 & -f/\sqrt{2} & \lambda/2 \\ f/\sqrt{2} & \lambda & f/\sqrt{2} \\ \lambda/2 & -f/\sqrt{2} & \lambda/2 \end{pmatrix} .$$

Derivation of weakly-reflective boundary conditions

The general solution of System 4 contains progressive waves which even in the limit of vanishing C and $\vec{F}$ are coupled (because A and B cannot be diagonalized simultaneously). Therefore, it is impossible to derive truly non-reflective (local) boundary conditions, i.e. a complete decoupling of ingoing and out-going waves. As a result, the boudary conditions to be derived are only weakly-reflective in some kind of approximation. For the one-dimensional shallow water equations, Verboom 1982, the approximation parameter is (λ/ω), where ω is the wave frequency. For the two-dimensional problem two additional parameters exist: (f/ω) and the angle of incidence of waves at the boundary.

To derive weakly-reflective boundary conditions for the half-space problem $x \leq 0$ one can proceed along the following lines:
i) freeze the coefficients in System 4 in order to arrive at a linear system,
ii) perform a Fourier transform in time and in y-direction, with dual variables ω and η, respectively, and write the transformed system as

$$\hat{v}_x = G\hat{v}, \tag{5}$$

with

$$G = -i\omega A_f^{-1}(I + \frac{\eta}{\omega} B_f + \frac{1}{i\omega} C_f), \tag{6}$$

where the index f refers to "frozen", and I is the identity matrix. In Equation 5 we neglected $\vec{F}$, but we will return to this matter later on. For A_f^{-1} to exist we must exclude the cases where $u_f = 0$, no flow across the boundary, and $u_f^2 = \frac{1}{2}\psi$, i.e. no critical flow.
iii) perform a transformation $\hat{w} = W\hat{v}$, such that $D = WGW^{-1}$, is a diagonal matrix. The components of $\hat{w}$ are decoupled and these quantities or more precisely an approximation to these quantities must be prescribed at the boundaries. The number of conditions to be prescribed at $x = 0$ is given by the number of positive eigen-values of D; similarly the number of boundary conditions for the right half-space problem, $x \geq 0$, equals the number of negative eigen-values of D.
iv) write the transformation W as a polynomial in (η/ω) and $(1/i\omega)$, and truncate this polynomial. A local weakly-reflective boundary condition is obtained by an inverse Fourier-transform.

The advantage of this formal derivation is that $\hat{w}$ still is an exact result: approximations are introduced in the final stage of the derivation only. Unfortunately, it is not possible to write the eigen-values and eigen-vectors of G, Equation 6, in an explicit form if f and λ are non-zero simultaneously, as they are in the general case. To solve this problem we can proceed along two lines.
Firstly, following Engquist and Majda 1977 instead of diagonalizing G exactly and then approximating W one can diagonalize G approximately with a much simpler transformation. Above that, it is sufficient to decouple the ingoing waves from the outgoing waves upto the desired order of approximation, whereas the outgoing waves still might depend more strongly on the ingoing waves. For the matrix WGW^{-1} this means that only some of the

off-diagonal terms need to be zero (in the order of approximation). The transformation matrix W can now be written as

$$W \simeq \sum_{p=o}^{\infty} \left(\frac{\eta}{\omega}\right)^p \left(\frac{1}{i\omega}\right)^q W_{pq}. \tag{7}$$

For System 4 W_{pq} will turn out to be rather simple for (p and q) ≤ 1. Secondly, one can by-pass all matrix manipulations and substitute a solution of the form

$$\vec{v} = \vec{\hat{v}} \exp(i\omega t + ik_x x + ik_y y), \tag{8}$$

in the frozen coefficient problem, and look for boundary conditions of the form

$$\alpha\hat{u} + \beta\hat{v} + \gamma\hat{\psi} = \hat{f},$$

where α, β, γ and $\hat{f}$ are functions of ω, k_x and k_y, such that only ingoing waves are prescribed. Weakly-reflective boundary conditions are obtained by approximating the coefficients α, β and γ in terms of ω, k_x, and k_y. The number of boundary condition required equals the number of ingoing waves and is found from the analysis. However, additional information is required on what kind of combinations of $\hat{u}$, $\hat{v}$, and ψ might be physically relevant. A drawback of both methods is that the boundary conditions derived do not guarantee a stable computation, see also Verboom et al. 1982. If, as a rule of thump, the boundary condition contains a term like y_t + by, then b must be negative irrespective of the other terms.

We proceed with a derivation along the lines indicated by Engquist and Majda 1977: apply the transformation given by Equation 7 to Equation 5 and 6. In a rather straightforward analysis weakly-reflective boundary conditions can be derived for inflow and outflow boundaries. The results for the left half space problem, $x \leq 0$, subcritical flow, and (p and q) ≤ 1 are summarized in Table 1.
The boundary conditions given in Table 1 apply (as well) for a boundary at x=L for a problem defined at x=[0,L], L $\in \mathbb{R}$, of course.
For p=1 and q=1 two conditions instead of one are derived without introducing a second order derivative in space or time. Set B differs from set A in that it is a higher order approximation in $(\lambda/i\omega)$-terms; for $(f/i\omega)$-terms

this was found to be not possible. A similar result was found by Pakvis 1983, for the one-dimensional shallow water equations. The one-dimensional weakly-reflective boundary condition, equivalent to set B, with p=1 and q=1, reads

$$(9) \qquad (u-\psi)_t + \frac{\lambda_f}{4}\{(3+\beta+\frac{2u_f}{\psi_f})u - (1+\beta-\frac{2u_f}{\psi_f})\psi\} = g_1.$$

The parameter β can be chosen arbitrarily, but Equation 9 is a one order higher approximation in $(\lambda/i\omega)$-terms, if $\beta=0$. For the linear equations Pakvis proved that the reflection coefficient depends rather strongly on β. The optimal value is about zero for small values of (λ/ω) and approaches -0.8 if (λ/ω) is of the order of 100. For these extremely high values of (λ/ω), which can easily be encountered in, for instance, flood wave problems, the reflection coefficient with the optimal value of β was found to be less than half the reflection coefficient with $\beta=0$.

Table 1 Weakly-reflective boundary conditions for subcritical in- and outflow boundaries for the left half-space problem, $x \leq 0$.

Subcritical inflow ($u_f < 0$ and $u_f < \sqrt{\frac{1}{2}\psi_f}$).

p=0, q=0:
$$\begin{pmatrix} u-\psi \\ v \end{pmatrix} = \vec{g}$$

p=1, q=0:
$$\begin{pmatrix} (u-\psi)_t \\ v_t+\frac{1}{2}(u_f+\frac{1}{2}\psi_f)(u+\psi)_y \end{pmatrix} = \vec{g}$$

p=1, q=1 A:
$$\begin{pmatrix} (u-\psi)_t+\frac{\lambda_f}{4}(1+\frac{2u_f}{\psi_f})(u+\psi) \\ v_t+\frac{1}{2}(u_f+\frac{1}{2}\psi_f)(u+\psi)_y+\frac{f}{2}(1+\frac{2u_f}{\psi_f}) \end{pmatrix} = \vec{g}$$

B:
$$\begin{pmatrix} (u-\psi)_t+\frac{\lambda_f}{4}\{(3+\frac{2u_f}{\psi_f})u-(1-\frac{2u_f}{\psi_f})\psi\}-\frac{1}{4}(u_f+\frac{1}{2}\psi_f)v_y \\ v_t+\frac{1}{2}(u_f+\frac{1}{2}\psi_f)(u+\psi)_y+v_f v_y+\frac{f}{2}(1+\frac{2u_f}{\psi_f}) \end{pmatrix} = \vec{g}$$

Subcritical outflow ($0 < u_f < \sqrt{\frac{1}{2}\psi_f}$)

p=0, q=0 $\quad u-\psi = g_1$

p=1, q=0 $\quad (u-\psi)_t - u_f v_y = g_1$

p=1, q=1 A: $(u-\psi)_t + \frac{\lambda_f}{4}(1+\frac{2u_f}{\psi_f})(u+\psi) - \frac{2u_f}{\psi_f} fv - u_f v_y = g_1$

B: $(u-\psi)_t + \frac{\lambda_f}{4}\{(3+\frac{2u_f}{\psi_f})u-(1-\frac{2u_f}{\psi_f})\psi\} - \frac{2u_f}{\psi_f} fv - u_f v_y = g_1.$

For the two-dimensional linear shallow water equations without friction and Coriolis force Wagatha 1983 derived a set of parameters to perturbe the boundary conditions. He could decrease the reflection by optimizing these parameters, through the influence was much less compared to the factor reported by Pakvis. As Wagatha obtained his results from numerical computations it might be that spurious reflections due to the numerical scheme blurred the results. From these results it is expected that the boundary conditions summarized in Table 1 can be optimized by introducing certain parameters. However, this is left for the future.

The external influences, driving forces, are included in the right hand sides of the boundary conditions. As the components of $\vec{g}$ are unknown in general and cannot be taken from field measurements one must construct $\vec{g}$ from previous knowledge on u, v, ψ and their derivatives if these occur in the boundary conditions. For practical problems this information can be obtained from a larger area (coarser grid) model only. To get rid of short wave disturbances due to the initial condition Stelling 1983 proposed a much simpler formulation. For a velocity controlled inflow boundary at x=0 this formulation is given by

$$u + \alpha(u+\psi)_t = g, \tag{10}$$

where α is constant, $\alpha > 0$. The influence of the second term readily follows from an analysis of a linear one-dimensional problem given by

$$u_t + g\zeta_x = 0$$
$$\zeta_t + hu_x = 0.$$

The reflection coefficient at x=0 is given by

$$Re = 1/\sqrt{1+(4\pi\alpha/T)^2}, \tag{11}$$

here T is the wave period. For short waves Re goes to zero, but Re $\simeq$ 1 for long (tidal) waves. It is to be noted that Equation 10 has some resemblance with Equation 9 for $\beta = -1$ and $\alpha \sim 2/\lambda$; the main difference is, of course, the right hand side. Equation 10 has been applied successfully in many so-called nested model applications, Verboom et al. 1984.

Steady-state solution and external forces

In the derivation of the preceeding sections we neglected not only $\vec{F}$ (and external forces) but also the steady-state solution of System 4. The latter can be accounted for by substituting $\vec{v}' = \vec{v} - \vec{v}_o$ in the frozen coefficient formulation, where $\vec{v}_o$ is the steady-state solution. In the final result $\vec{v}$ is replaced by $\vec{v} - \vec{v}_o$. In actual computations the (approximate) steady-state of the original problem is used with good results. The influence of steady external forces such as wind can be accounted for in the same way, through it may require a larger area model to get a reliable quess for $\vec{v}$. Time dependent winds must be accounted for in the boundary forcing function $\vec{g}$. Apart of nested models no general procedure seems to be available to solve this problem.

NUMERICAL RESULTS

The first example concerns the application of a finite element program with weakly-reflective boundary conditions to the evolution of a Gaussian-shaped free surface elevation, Figure 1.

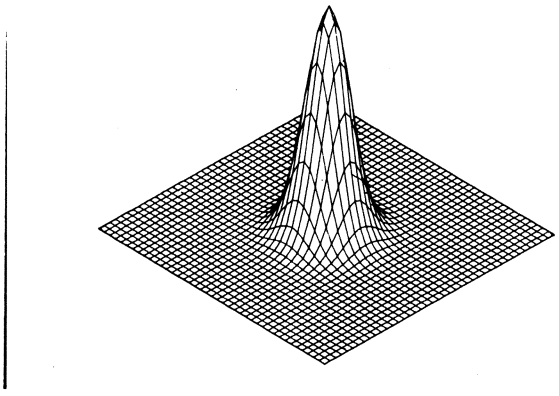


Figure 1. Initially Gaussian-shaped free surface elevation.

The relevant parameters are: h = 10 m, $\zeta(0,x,y) = \exp(-x^2+y^2)/L^2$, L = 200 m, bottom friction and Coriolis force are neglected. System 2 but expressed in conservative form with variables $(h+\zeta)u$, $(h+\zeta)v$, and $(h+\zeta)$ instead of u, v and ζ, respectively, is solved with a Galerkin finite element method. Square elements with bi-linear base functions are used; the time discretization is basically explicit and characterized by

$$
\begin{aligned}
U^{n+1} &= f_1(U^n, V^n, H^n) \\
V^{n+1} &= f_2(U^{n+1}, V^n, H^n) \qquad (12) \\
H^{n+1} &= f_3(U^{n+1}, V^{n+1}, H^n),
\end{aligned}
$$

with $U = u(h+\zeta)$, $V = v(h+\zeta)$, and $H = h+\zeta$.
The numerical parameters are $\Delta x = 50$ m and $\Delta t = 5$ s. The results at four points in time are given in Figures 2A-D. At the boundaries zeroth order conditions are prescribed, i.e. p=0, q=0.

To study the reflective properties in more detail these results are subtracted from a larger area solution and a reflection coefficient is defined as

$$Re = \frac{\max_t(\Delta\zeta)}{\max_t \zeta}.$$

Though Re is not a proper reflection coefficient in that its value is bounded to the domain [-1,1], it provides an indication of the degree of reflections.

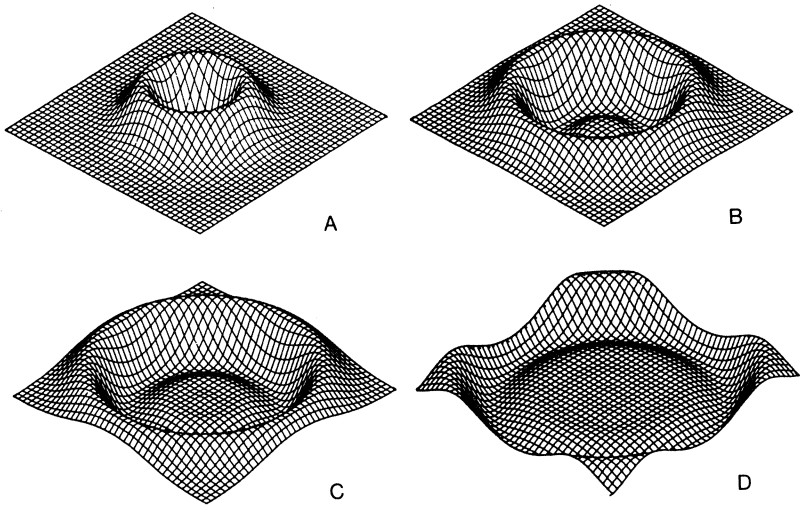


Figure 2. Time evolution of an initially Gaussian-shaped free surface elevation after 30, 60, 80 and 110 s.

Figure 3 shows the results for a zeroth order, p=0 and q=0, and a first order,

p=1 and q=0, boundary conditions, respectively. The reflection coefficient is given as a function of the angle of incidence of the wave, which is proportional to the coordinate along a boundary.

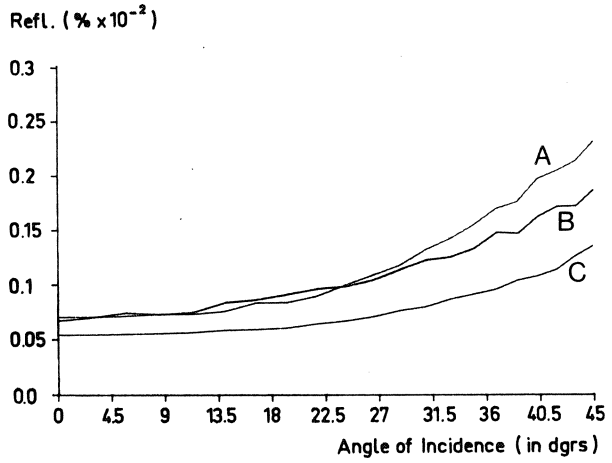


Figure 3. Reflection coefficient as a function of the angle of incidence and the boundary conditions. A: zeroth order, B: first order, C: first order, Δx and Δt halved.

The reflection coefficient depends rather strongly on the angle of incidence and increases about a factor of 3 in the range $(0\text{-}45)^{o}$. The reflections at normal incidence are solely due to numerical reflections; they especially confuse the results of the first order condition.
Concerning the numerical implementation there is a lot of freedom in a finite element program, because the program itself even does not demand a specific number of boundary conditions. For the zeroth order condition we used the following implementation at outflow.

$$\frac{\bar{u}^{n+1}+u^{n}}{2} - \psi^{n} = u_{o} - \psi_{o}$$

where $\bar{u}^{n+1}$ is a temporary value of u^{n+1} and is used to find v^{n+1} and $\bar{\psi}^{n+1}$, and $\bar{U}^{n+1} = \bar{u}^{n+1}H^{n}$. The definite values of u^{n+1} and ψ^{n+1} are found from

$$u^{n+1} - \psi^{n+\frac{1}{2}} = u_{o} - \psi_{o}$$

$$u^{n+1} + \psi^{n+\frac{1}{2}} = \bar{u}^{n+1} + \bar{\psi}^{n+\frac{1}{2}}$$

$$\psi^{n+\frac{1}{2}} = 2\sqrt{g(H^{n+1}+H^{n})/2}.$$

Finally, the new values of U^{n+1} is found from $U^{n+1} = u^{n+1}H^{n+1}$. Several other formulations were tested, but the best results were obtained if u and ψ were shifted one half time step, as follows from Equation 12 and $u+\psi$ was kept constant.

The second example concerns the influence of the parameter α in Equation 10. This boundary condition is used in an ADI-finite difference program with a space staggered grid. For a schematized river section (2 km long and 5 m depth) the waterlevel is prescribed and kept constant at the downstream end and the velocity is abruptly prescribed at 1 m/s at the upstream end. The numerical parameters used are; Δx = 100 m and Δt = 100 s.
Figure 4A shows the velocity at the downstream boundary as a function of time for α = 100, 500 and 1.10^4. For α = 100, Figure 4A-II, strong eigen-oscillations are generated which are damped in time by bottom friction. The initial period is about 1.5 hours, whereas the wave transition time is only 300 s. For α = 500, Figure 4A-I, the boundary condition is also transparant for the eigen-frequencies and hardly any eigen-oscillation is generated. The initial period is about halved. If α is increased about 1000 the initial period starts to increase again, not because of oscillations but because the variables now approach the final solution in the limit of large time (like a super-critically damped resonator), Figure 4B-I.

As the transition time is only about 300 s the initial period should drop well below one half hour if a weakly-reflective boundary condition of Table 1 is used.

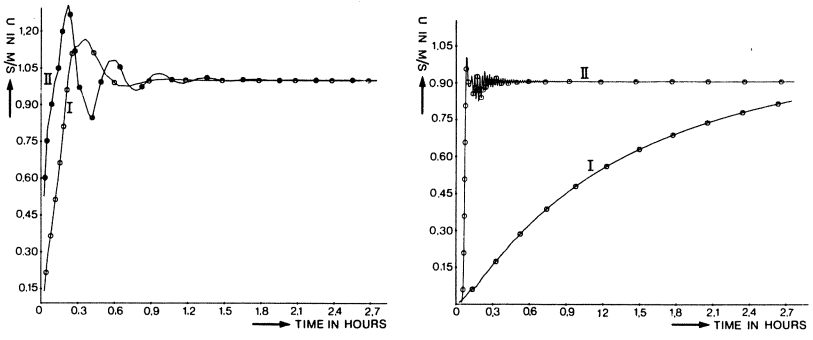


Figure 4. Velocity as a function of time. Influence of α, A-I, A-II, and B-I, and of a zeroth-order boundary conditions, B-II, on the initial period.

Figure 4B shows the result if the velocity boundary is replaced by a zeroth order condition. Indeed, no eigen-oscillations are generated and the initial period is about 0.3 hour. The short ($2\Delta t$) waves are due to the abrupt starting condition only: they do not occur if the boundary condition is increased to its final value in one or two wave transition times. This boundary condition, with α about 100, is succesfully used in many practical applications, Verboom et.al. 1984.
A final remark concerns the influence of the parameter α on the eigen-frequencies of a problem area. If $\alpha > 0$ the eigen-frequencies are lowered: for the example discussed above the oscillation period was increased by about 27% for $\alpha = 100$ and by about a factor of 3 for $\alpha = 500$. These figures are confirmed by an analysis for a simple one-dimensional formulation.

CONCLUSION

Weakly-reflective boundary conditions have been derived for the two-dimensional shallow water equations, including bottom friction and Coriolis-force. With a finite element formulation reflection coefficients of only a few precent were obtained. The reflection coefficient increases by about a factor of three if the angle of incidence increases from zero to 45^{o}. The effect of higher order conditions is blurred by reflections generated by the numerical scheme itself. For many practical applications a much simpler formulation proposed by Stelling 1983, suppresses effectively short wave disturbances and eigen oscillations of the problem area. Initial periods of only a few wave transition times can be realized by using the weakly-reflective boundary conditions derived in this paper.

REFERENCES

[1] ENGQUIST, B. & A. Majda 1977, *Absorbing conditions for the numerical simulation of waves.* Math. Comp. 31, 629-651.

[2] PAKVIS, J. 1983, *Weakly-reflective boundary conditions for the shallow water equations (in Dutch).* Ms. Sc. thesis, Delft University of Technology, Dep. Num. Math., Delft Hydraulics Laboratory, Rep. S 545-I.

[3] SLOB, A. 1983, *Weakly-reflective boundary conditions for the two-dimensional shallow water equations (in Dutch)*. Ms. Sc. thesis, Delft University of Technology, Dep. Num. Math., Delft Hydraulics Laboratory, Rep. S545-II.

[4] SLOB, A., G.K. VERBOOM & G. SEGAL 1984, *Weakly-reflective boundary conditions for the two-dimensional shallow water equation*. To appear.

[5] STELLING, G.S. 1983, *On the construction of computational methods for shallow water flow problems*. Ph.D Thesis Delft University of Technology.

[5] TAYLOR, M.E. 1975, *Reflection of singularities of solutions to systems of differential equations*. Comm. Pure Applied Math., 28, 457-478.

[6] VERBOOM, G.K. 1982, *Weakly-reflective boundary conditions for the shallow water equations*. Presented at 4th International Conference on Finite Elements in Water Resources, Hannnover, June 21-25. Not included in the proceedings. Available as Delft Hydraulics Publication No. 266.

[7] VERBOOM, G.K., G.S. STELLING & M.J. OFFICIER 1982, *Boundary conditions for the shallow water equations*. In Engineering Applications of Computational Hydraulics; Homage to Alexandre Preissmann (Ed. M.B. Abott and J.A. Cunge) Pitman, London.

[8] VERBOOM, G.K., H.J. DE VRIEND, G.J. AKKERMAN, R.A.H. THABET & J.C. WINTERWERP 1984, *Nested models, applications to practical problems*. This conference.

[9] WAGATHA, L. 1983, *Approximation of pseudo-differential operators in absorbing boundary conditions for hyperbolic equations*. Num. Math. 42, 51-64.

NOTATION

$\vec{w}$: $(u,v,\zeta)^T$

u,v : velocity components in x and y direction, respectively

ζ : free surface elevation above a reference plan

h : bottom below a reference plane

λ : bottom friction parameter

C : de Chézy-coefficient

f : Coriolis parameter

g : gravitational acceleration

x,y,t : space and time coordinates

$\vec{F}$: vector, includes external forces

A_j : coefficient matrix

ψ : $2\sqrt{g(h+\zeta)}$

$\vec{v}$: $(1+\sqrt{2}\ (u+\psi),\ v,\ 1/\sqrt{2}(u-\psi))^T$

A,B,C : coefficient matrices

V : transformation defined in Equation 3

ω : $2\pi/T$

T : wave period

$\hat{\vec{v}}$: Fourier transform of $\vec{v}$ in t and y

G : matrix defined in Equation 6

g_j : boundary forcing function

Re : reflection coefficient

TWO-DIMENSIONAL SPECTRAL ANALYSIS IN THE EVALUATION OF IMAGE QUALITY OF IMAGING SYSTEMS

J. VRANCKX

1. INTRODUCTION

As a manufacturer of imaging systems one of our main concerns at Agfa-Gevaert is the "image quality" achieved by our systems. This image quality is determined by objective parameters such as sharpness, graininess, contrast resolution etc.. All these objective parameters can be measured and are combined in the ultimate image quality criterion: the signal-to-noise ratio. The problem with the separate measurement of these parameters is that the measurements are very often performed in non-comparable situations. To avoid this, we developed a method where signal-to-noise ratio is determined in one single measurement.

The method was developed for the Medical Imaging Department of the Diagnostic Imaging Systems Division. The examples shown will all refer to specific problems of this department so maybe a brief introduction into the world of medical radiography is needed. It should however be stressed that our method is not at all restricted to this specific type of imaging systems.

Radiographs are made by exposing a patient to a beam of X-rays; the patient's body modulates the beam. This modulated beam is the input signal for the imaging system. A conventional imaging system in medical radiography consists of a light sensitive film sandwiched between two intensifying screens. The role of these screens is to absorb the X-rays and to convert the absorbed energy into light photons. These light photons are then absorbed in the film, where they produce optical density upon development. The modulated X-ray beam carries the input signal.
There are several sources of noise in such a system: the most important one is the statistical fluctuation of the X-ray absorption. Other sources are inhomogeneities in screen and film.

Since our method for measuring signal-to-noise ratios is closely related to the measurement of noise we will, after a short review of the properties of Fourier-tranforms, discuss the noise measurement and show some experimental results. Then we will discuss the method for the signal to-noise measurement. We will show some experimental results for different screen-film systems and show the effect of quantisation caused by the digitisation needed for digital image processing.

2. FOURIER TRANSFORMS (Ref. 1-4,9)

Let $f(p,q,r...)$ be a function of the variables $p,q,r,....$ These variables can be space, time or other coordinates. The Fourier Transform of $f(p,q,r,...)$ is then defined as

$$(1) \qquad F(\mu,\nu,...) = \int_{p=-\infty}^{\infty} \int_{q=-\infty}^{\infty} f(p,q,...)e^{2\pi i(p\mu+q\nu)}dpdq...$$

with $i = \sqrt{-1}$.
The inverse Fourier Transform is defined as

$$(2) \qquad f(p,q,...) = \int_{\mu=-\infty}^{\infty} \int_{\nu=-\infty}^{\infty} F(\mu,\nu,...)e^{-2\pi i(p\mu+q\nu+...)}d\mu d\nu...$$

The functions $f(p,q,r,...)$ and $F(\mu,\nu,...)$ are Fourier Transform pairs which we will denote as

$$(3) \qquad f(p,q,r...) \overset{FT}{\longleftrightarrow} F(\mu,\nu,...)$$

The Fourier Transform is a complex function, which can be seen when we rewrite the exponential of eq 1 as

$$(4) \qquad e^{2\pi ix} = \cos 2\pi x + i \sin 2\pi x$$

Fourier transforming a function is thus decomposing it into sine and cosine functions with different frequencies.
The Fourier Transform pairs shown in fig. 1 are then easily understood.
We will now give some important properties of Fourier Transforms, properties we will rely on in the later parts of this paper.

The first property is LINEARITY. Let $f(p,q,...)$ and $g(p,q,...)$ be two

functions with their respective Fourier Transforms $F(\mu,\nu,\ldots)$ and $G(\mu,\nu,\ldots)$. The functions

$$(5) \qquad a.f(p,q,\ldots) + b.g(p,q,\ldots) \xleftrightarrow{FT} a.F(\mu,\nu,\ldots) + b.G(\mu,\nu,\ldots)$$

also form a Fourier Transform pair.

We now give some SYMMETRY relations:
- the Fourier Transform of a real even function is real even
- the Fourier Trnasform of a real uneven function is imaginary and uneven.

The symmetry relations for complex functions will not be considered here. The interested reader can find them in ref. 3.

The third property is CONVOLUTION and CORRELATION. The convolution of two functions is given by

$$(6) \qquad y(\xi,\eta,\ldots) = \int_{p=-\infty}^{\infty} \int_{q=-\infty}^{\infty} f(p,q,\ldots)g(p-\xi,q-\eta,\ldots)dpdq\ldots$$

Let the functions

$$(7) \qquad f(p,q,\ldots) \xleftrightarrow{FT} F(\mu,\nu,\ldots)$$

be Fourier Transform pairs. The functions

$$(8) \qquad y(\xi,\eta,\ldots) \xleftrightarrow{FT} F(\mu,\nu,\ldots).G(\mu,\nu,\ldots)$$

are then also Fourier Transform pairs. A convolution can thus be calculated by Fourier transforming the two functions, multiplying the Fourier Transform and taking the inverse Fourier Transform of this product. Correlation is defined as

$$(9) \qquad k(\xi,\eta,\ldots) = \int_{p=-\infty}^{\infty} \int_{p=-\infty}^{\infty} f^{*}(p,q,\ldots)g(p+\xi,q+\eta,\ldots)dpdq$$

where * denotes the complex conjugate.
The functions

$$(10) \qquad k(\xi,\eta,\ldots) \xleftrightarrow{FT} F^{*}(\mu,\nu,\ldots).G(\mu,\nu,\ldots)$$

are the Fourier Transform pairs. Correlation can thus be calculated in the same way as convolution.

All these equations stand for continuous analytical functions. In practice that type of function is not available: what we have are discrete samples lying in some finite interval. The integrals in the previous equations are to be replaced by summations. The finite discrete Fourier Transform has the same properties as the continuous Transform (linearity, symmetry). The main difference lies in the fact that the discretisation and the truncation in the data results in discretisation and truncation in the resulting Fourier Transform. The Nyquist criterium states that for N samples taken with a sampling distance of Δp the maximum frequency of the Fourier Transform is

$$\nu_{max} = 1/(2\Delta p) \tag{11}$$

and the frequency-interval is

$$\Delta\nu = 1/(N\Delta p) \tag{12}$$

3. ANALYSIS OF IMAGE NOISE: WIENER SPECTRUM AND AUTOCORRELATION (Ref. 1,5-8)

An image is a two-dimensional pattern of optical density D(x,y). In an image made without an external input signal the pattern is only noise. One first way to describe this pattern is by its mean and by its standard-deviation:

$$\bar{D} = \lim_{X,Y\to\infty} \frac{1}{2X}\frac{1}{2Y}\int_{-X}^{X}\int_{-Y}^{Y} D(x,y)\,dxdy \tag{13}$$

$$\sigma_D^2 = \lim_{X,Y\to\infty} \frac{1}{2X}\frac{1}{2Y}\int_{-X}^{X}\int_{-Y}^{Y} [D(x,y)-\bar{D}]^2\,dxdy \tag{14}$$

More information about the density pattern can be extracted from the autocorrelation function:

$$AC(\xi,\eta) = \lim_{X,Y} \frac{1}{2X}\frac{1}{2Y}\int_{-X}^{X}\int_{-Y}^{Y} \Delta D^*(x,y)\Delta D(x+\xi,y+\eta)\,dxdy \tag{15}$$

with $\Delta D(x,y) = D(x,y) - \bar{D}$. From equations 14 and 15 we see that

$$AC(0,0) = \sigma_D^2.$$

The Wiener spectrum is defined as

$$(16) \qquad W(u,v) = \lim_{X,Y\to\infty} \frac{1}{2X}\frac{1}{2Y} F^*(u,v).F(u,v)$$

with F(u,v) being the Fourier Transform of $\Delta D(x,y)$. The Wiener spectrum is the Fourier Transform of the autocorrelation function. This is written as

$$(17) \qquad AC(\xi,\eta) = \int\int_{-\infty}^{\infty} W(u,v)e^{-2\pi i(u\xi+v\eta)}dudv.$$

From equations 15 and 17 we see that the standard deviation of the density pattern is

$$(18) \qquad \sigma_D^2 = AC(0,0) = \int\int_{-\infty}^{\infty} W(u,v)dudv.$$

The Wiener spectrum is thus equivalent to the autocorrelation function. It contains more information than the standard deviation since it describes the frequency dependence of the noise. The standard deviation only gives a global appreciation of the noise. It can be shown that under some conditions (which are most often satisfied in imaging systems) the Wiener spectrum not only contains the first two moments (mean and standard deviation) but also all higher moments of the density distribution in the image.

We will now consider some methods to measure the Wiener spectrum. Conceptually the simplest method is the two-dimensional scan method using a circular aperture. The resulting two-dimensional spectrum W'(u,v) is, for small density fluctuations giving a linear relation between density and transmission fluctuations:

$$(19) \qquad W'(u,v) = W(u,v) \times T^2(u,v) \times T_M^2(u,v)$$

where W(u,v) is the true spectrum and T(u,v) the transfer function of the circular aperture:

$$(20) \qquad T(u,v) = 2J_1(\pi\omega d)/(\pi\omega d)$$

where J_1 = a Bessel function of the first kind (the two-dimensional

counterpart of sin (x)/x)

$\omega = \sqrt{u^2+v^2}$

d = the diameter of the circular aperture.

and $T_M(u,v)$ = the transfer function of the measuring system. In the frequency range we consider it sufficiently close to unity to be ignored.

It is easy to calculate the true spectrum from the measured one.

One of the difficulties of this method is the positional accuracy of the microdensitometer. The newer computer-controlled microdensitometers are sufficiently accurate to scan a film image homogeneously in a rectangular grid. Another problem is the computing power required for the two-dimensional Fourier Transform. The 16-bit minicomputers which are often used to control the microdensitometers do not have the necessary computing power. It is probably due to these two difficulties that up to now only a few two-dimensional noise power spectra have been published (Ref. 5).

A second method consists of a one-dimensional scan using a circular aperture. The resulting spectrum is integrated along one of the frequency axes:

$$W'(u) = \int_{-\infty}^{\infty} W(u,v) \times T^2(u,v)dv. \tag{21}$$

De Belder (Ref. 8) gives an equation to calculate the true spectrum from the measured one.

In the third method, which is widely used, the sample is scanned in one dimension with a long narrow slit. Basically equation 21 holds also for this case. Due to the adequate choice of the aperture and under some assumptions (Ref. 6) this equation can be reduced to:

$$W'(u) = \mathrm{sinc}^2(\pi a u) \times W(u,0) \times l \tag{22}$$

where sinc(πau) stands for sin(πau)/(πau)

a = the slit width

l = the slit length

The measured spectrum then is a section through the real two-dimensional spectrum. The main drawback of this method is that working with too short a slit leads to an underestimation of the noise power at low spatial frequencies (Ref. 6-8). This method gives indeed biased results in the low frequency range of the spectrum. It is possible to synthesize a longer slit from a two-dimensional scan with a short slit to lower the bias.

At Agfa-Gevaert we implemented the two-dimensional method because it gives unbiased results at low frequencies, very important in the study of radiographic systems, and because it makes it possible to find directional effects and moreover because of its inherent simplicity. The main problem to solve was the problem of computing power. The measuring method we used can be described as follows: a film sample of uniform density (usually about density 1 above fog) is scanned two-dimensionally with a circular aperture of diameter d on a Perkin Elmer PDS model 1010A microdensitometer. For the two-dimensional Fourier Transform we used the Fortran subroutine for multidimensional Fast Fourier Transform written by Norman Brenner (Ref. 10). To protect against aliasing the sampling distance is chosen to be half the aperture diameter. For white noise this would give maximally about 40% aliasing at the Nyquist frequency and about 10% at 0.75 times the Nyquist frequency. For radiographic systems the noise is lower at higher frequencies so that the aliasing is lower than the just mentioned values. We mostly work with an array of 256 × 256 measuring points. To have smoother spectra we take an ensemble average over 49 blocks, each block being 64 × 64 points wide. The blocks overlap 32 points in the x and in the y direction (Ref. 9). For an aperture diameter of d we have:

the sampling distance : $\Delta x = \Delta y = d/2$

the Nyquist frequency : $u_{max} = v_{max} = 1/d$

the bandwith for 64 × 64 data blocks: $\Delta u = \Delta v = 1/(32d)$

Due to the microdensitometer optics the spectral values are expressed in instrument density and should be transformed into diffuse density values (Ref. 6). The correction for the aperture diameter can then be applied according to equations 19 and 20.

The noise power spectrum of Agfa-Gevaert's CURIX RP1 film-CURIX UNIVERSAL screen system has been measured with circular apertures of

.4,.2,.1 and .05 mm diameter. All spectra where calculated from 256 × 256 data points. As an example fig. 2 shows the two-dimensional spectrum for the 0.2 mm aperture, the z-axis being logarithmic, in units of mm^2. The spectrum is in instrument density and uncorrected for the aperture transfer function. The spectrum is obviously quite rough, the spread being about 14% of the spectral value.

Since it is easier to compare one-dimensional spectra we take an average over the two-dimensional spectrum at constant circular frequency $\omega = \sqrt{u^2+v^2}$, after we made sure that the noise was isotropic. Fig. 3 shows the resulting uncorrected one-dimensional spectra. The spectra are now rather smooth, the spread being some 4% of the spectral value. It is very interesting to see that the spectra overlap at low frequencies: the spectra are, by definition, unbiased at these frequencies. The discrepancies at higher frequencies can easily be removed by correcting for the aperture transfer function, as is shown in fig. 4. The corrected spectra are totally independent of the aperture diameter.

4. ANALYSIS OF SIGNALS IN NOISE (Ref. 3,9)

We will now describe two methods for measuring signal-to-noise ratios. In the first method we use a one-dimensional signal as input. The Wiener spectrum of the combined signal and noise is measured. The second method can use any low contrast signal as input. The Wiener spectra of signal and noise are calculated from the cross-correlation of two images containing the same signal.

4.1. Analysis of one-dimensional signals

We use here an image which consist of a one-dimensional sine-wave superposed on the noise. The Wiener spectrum has a peak at the frequency of the sine-wave, as is shown in fig. 5. Taking the spectral value along a circle with the frequency of the sine-wave as radius gives a curve as fig. 6. It is then possible to estimate the signal (peak value) and the noise (the median value) at that frequency. We used the method for measuring Wiener spectra as described previous section. The aperture diameter is .2 mm and the sampling distance is .125 mm. We use this sampling distance to have an integer number of signal periods in a $32\Delta x$ distance.

4.2. Analysis of unconstrained signals

The cross-correlation of two density patterns $\Delta D_1(x,y)$ and $\Delta D_2(x,y)$ is given by

$$CC(\xi,\eta) = \iint_{-\infty}^{\infty} \Delta D_1^*(x,y)\Delta D_2(x+\xi,y+\eta)\,dxdy. \tag{23}$$

This function describes how the first patterns looks like the second one. One can define a cross-Wiener spectrum as the Fourier Transform of the cross-correlation:

$$\begin{aligned} &\Delta D_1(x,y) \xleftrightarrow{FT} F_1(u,v) \\ &\Delta D_2(x,y) \xleftrightarrow{FT} F_2(u,v) \\ &CW(u,v) = F_1^*(u,v).F_2(u,v) \end{aligned} \tag{24}$$

Contrary to the normal Wiener spectrum this is a complex function. If R and I are the real and the imaginary part of the Fourier Transform then the cross-Wiener spectrum can be written as:

$$CW(u,v) = R_1R_2 + I_1I_2 + i(R_1I_2 - R_1I_2). \tag{25}$$

The normal Wiener spectrum can be seen as a special case where both density patterns are the same so that $R_1 = R_2$ and $I_1 = I_2$.

Suppose now that we have two density patterns with the same signal superposed on noise. When the signal is small, signal and noise can be regarded as independent. Two subsequent realisations of the noise (mainly statistical fluctuations of the photon flux) can also be regarded as independent. The two images are:

$$\begin{aligned} \Delta D_1(x,y) &= \Delta D_S(x,y) + \Delta D_{N1}(x,y) \\ \Delta D_2(x,y) &= \Delta D_S(x,y) + \Delta D_{N2}(x,y) \end{aligned}$$

with $\Delta D(x,y)$ being the signal and $\Delta D_{Ni}(x,y)$ the noise. Let F_{SNi} be the Fourier Transform of the i-th image, F_S the Fourier Transform of the signal and F_{Ni} the transform of the i-th noise realisation. Using the

linearity of the Fourier Transform and equation 26 we then can write that:

$$\begin{aligned} F_{SN1}(u,v) &= F_S(u,v) + F_{N1}(u,v) \\ F_{SN2}(u,v) &= F_S(u,v) + F_{N2}(u,v) \end{aligned} \tag{27}$$

The cross-Wiener spectrum is then:

$$\begin{aligned} CW(u,v) &= F^*_{SN1}(u,v).F_{SN2}(u,v) \\ &= F^*_S(u,v).F_S(u,v) + F^*_{N1}(u,v).F_s(u,v) + F^*_S(u,v).F_{N2}(u,v) \\ &\quad + F^*_{N1}(u,v).F_{N2}(u,v) \end{aligned} \tag{28}$$

Taking the inverse Fourier Transform of the cross-Wiener spectrum gives us the cross-correlation:

$$\begin{aligned} CC(\xi,\eta) &= CC_S(\xi,\eta) + CC_{N1,S}(\xi,\eta) + CC_{S,N2}(\xi,\eta) \\ &\quad + CC_{N1,N2}(\xi,\eta) \end{aligned} \tag{29}$$

We used the following Fourier Transform pairs to derive equation 29:

$$\begin{aligned} CC(\xi,\eta) &\xleftrightarrow{FT} CW(u,v) \\ CC_S(\xi,\eta) &\xleftrightarrow{FT} F^*_S(u,v).F_S(u,v) = W_S(u,v) \\ CC_{N1,S}(\xi,\eta) &\xleftrightarrow{FT} F^*_{N1}(u,v).F_S(u,v) \\ CC_{S,N2}(\xi,\eta) &\xleftrightarrow{FT} F^*_S(u,v).F_{N1}(u,v) \\ CC_{N1,N2}(\xi,\eta) &\xleftrightarrow{FT} F^*_{N1}(u,v).F_{N2}(u,v). \end{aligned} \tag{30}$$

Since image and noise are said to be independent there is no correlation between them. The same is true for the two realisations of the noise. The sole non-zero term in equation 29 is the cross-correlation between the signal and itself: this is its autocorrelation. The cross Wiener spectrum is nothing but the Wiener spectrum of the signal. It is easy to calculate now the Wiener spectrum of the noise by subtracting the signal spectrum from the (signal+noise) spectrum. Signal-to-noise

versus frequency can be calculated by dividing the signal-spectrum by the noise spectrum. A global appreciation of signal-to-noise is given by integrating the spectra and by dividing the resulting standard-deviations. Fig. 7 shows the noise spectrum after the signal spectrum is subtracted. The total spectrum (signal+noise) was shown in fig. 5. We see that the signal is almost completely removed. When giving numerical results we will use the integrated form of the signal-to-noise ratio.

The resulting signal-to-noise ratios are dependent on the input signal and on the imaging system. Imaging systems can be compared by using the same signal as input or by dividing the measured signal-to-noise by the input signal-to-noise ratio. In principle there are no constraints on the kind of signal that is used as input signal. As a result of our method of averaging the spectra over 64×64 point data blocks with an overlap of 32 points (to have smoother spectra), we introduce the restriction that the signal should have an integer number of periods in a $32\Delta x$ distance. Non periodic signals should have the same statistical characteristics in every data block.

The signal-to-noise ratios where measured by the method described in 4.1 and by the cross-correlation method using the same sinewave input signal. Since the first method gives the signal-to-noise ratio at the fixed signal frequency and the second method gives a global value (integrated over the whole spectrum) the results are not the same. Both methods have been used to measure the signal-to-noise ratios for five different screen film systems. There is a 98% correlation between the two sets of results.

In the introduction we mentioned the inhomogeneities of the intensifying screen as a possible source of noise. It is possible to evaluate the importance of this source of noise by using the screen inhomogeneities as an input signal. By making two images of exactly the same part of the screen we can extract the screen noise from the other (random) types of noise. This is important for us to evaluate the quality of the screens. Fig. 8 shows the Wiener spectrum of the screen noise for a screen that was made so as to have a great amount of inhomogeneities. We see here that the Wiener spectrum is not isotropic. Inspection of the screen indeed shows an alignment of the inhomogeneities. In this screen the amount of screen noise is as great at the other sources of noise together. In commercial screens the situation is much better,

screen noise being some 20 to 30% of the other noise sources. Since independent sources of noise are to be added squared, screen noise adds only a negligible amount to the total system noise.

We will now investigate the influence of digitisation on the signal to-noise ratio. Measuring the spectra already requires a digitisation phase: the PDS microdensitometer uses a logarithmic amplifier and a 12 bit A/D converter. This results in 4096 different grey levels equally spaced along the optical density axis (from D=0 to D=5.12). This is optimal for the small density variations under investigation. Fig. 9 shows how signal and noise for a screen-film system are affected by a change of the number of bits used in the A/D conversion. Also shown is the grey level spacing. Fig. 10 shows the influence of the number of A/D bits on the signal-to-noise ratio. For an 8 bit A/D conversion the S/N is lowered some 3% as compared to the 12 bit conversion. The spacing between the grey levels is then equal to the standard deviation of the density fluctuations caused by the noise. Digitisation with less than 8 bit results in a severe loss of signal-to-noise ratio. This important result sets a lower level on the number of grey levels required when digitising images.

5. CONCLUSION

We developed methods to evaluate the noise and the signal-to-noise ratio of imaging systems. These methods were used with good results in the study of radiographic screen-film systems. This field however is not the only field where our methods are applicable. Every imaging system or parts thereof can be evaluated using this method. In image processing it is possible, by measuring signal to-noise ratios before and after processing, to evaluate the effectiveness of the so-called 'noise-cheating' algorithms. It is also possible to evaluate lenses, photographic materials, camera's, CRT-display's and so on.

Another field where our methods are of great importance is the field of statistical image enhancement. In this type of enhancement discrimination between image and noise is sought on statistical grounds (maximum likelihood, maximum entropy, Bayes-criterion). All these methods require a good estimate of the noise power spectrum. Our method gives that estimate and what more is, it also gives an estimate of the signal power spectrum.

REFERENCES

[1] DAINTY, J.C. & R. SHAW, *Image Science*, Academic Press (1974).

[2] JENNISON, R.C., *Fourier Transforms and Convolutions for the Experimentalist*. Pergamon Press (1961).

[3] BRIGHAM, E.O., *The fast Fourier Transform*. Prentice Hall Inc (1974).

[4] ANDREWS, H.C., W.K. PRATT & K. CASPARI, *Computer Techniques in Image Processing*, Academic Press (1970).

[5] SANDRIK, J.M., R.F. WAGNER & K.M. HANSON, *Applied Optics*, 21, 3597 (1982).

[6] SANDRIK, J.M. & R.F. WAGNER, *Applied Optics*, 20, 2795 (1981).

[7] DOI, K., G. HOLJE, L. LOO, H. CHAN, J.M. SANDRIK, R.J. JENNINGS & R.F. WAGNER, *MTF's and Wiener spectra of Radiographic Screen-Film Systems*, FDA 82/8187. US-Department of Health and Human Service.

[8] DE BELDER, M. & J. DE KERF, *Photogr. Sci. Eng.*, 11(6), 371 (1967).

[9] SCHWARTZ, M. & L. SHAW, *Signal Processing*. Mc Graw Hill (1975).

[10] BRENNER, N., *Fortran subroutine for multidimensional FFT*, MIT Lincoln Laboratory (January 1969). See also, *IEEE Audio Transactions*, Special issue on FFT (June 1967).

FIG.1-EXAMPLES OF FOURIER TRANSFORM PAIRS

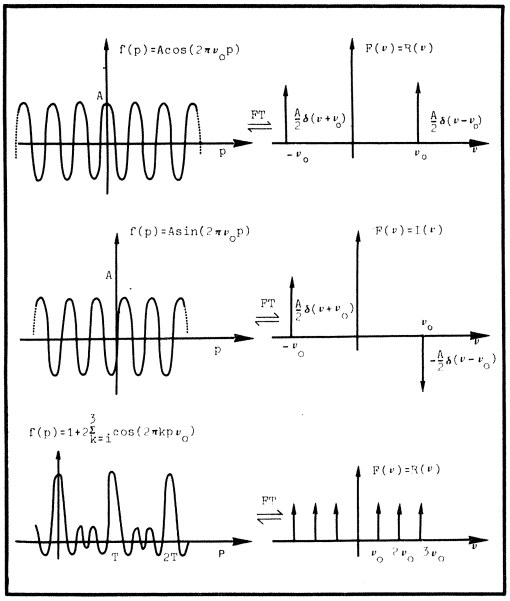


FIG.2-TWO-DIMENSIONAL NOISE POWER SPECTRUM
CURIX RP1 FILM + CURIX UNIVERSAL SCREEN

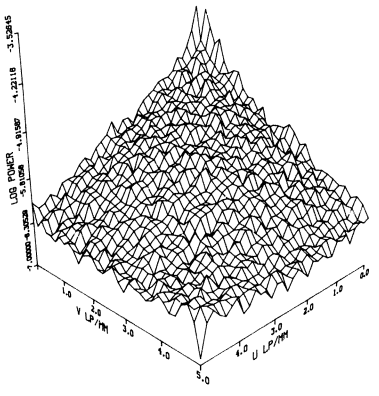


FIG.3-UNCORRECTED NOISE POWER SPECTRUM
CURIX RP1 FILM + CURIX UNIVERSAL SCREEN

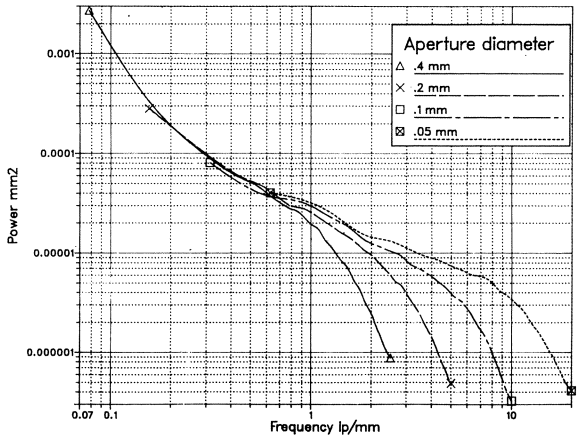


FIG.4-CORRECTED NOISE POWER SPECTRUM
CURIX RP1 FILM + CURIX UNIVERSAL SCREEN

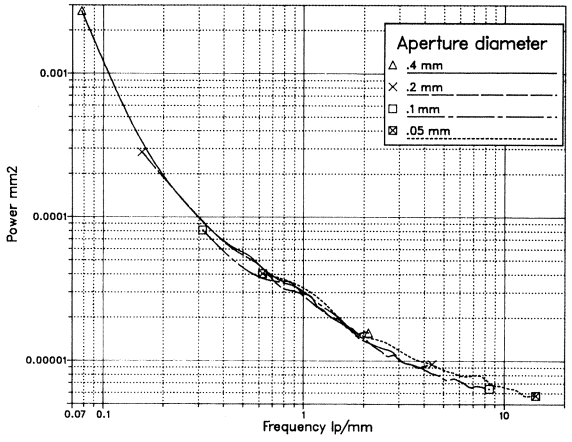


FIG.5-CURIX RP1 FILM + CURIX UNIVERSAL SCREEN
SINE-WAVE : TOTAL SPECTRUM

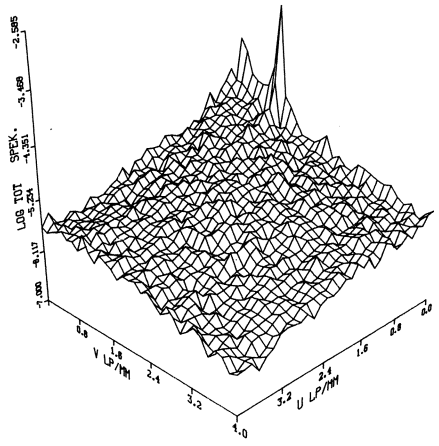


FIG.6-CURIX RP1 FILM + CURIX UNIVERSAL SCREEN
SINE-WAVE : TOTAL SPECTRUM

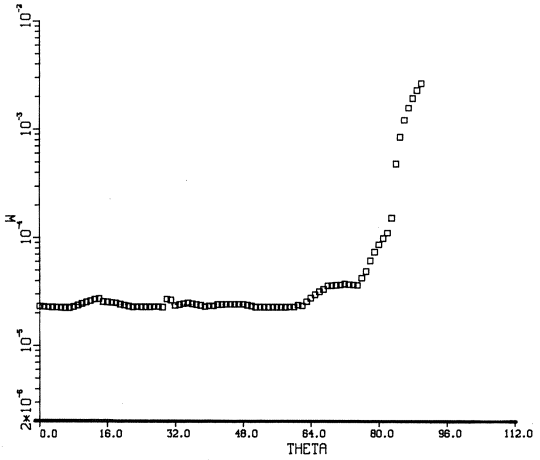


FIG.7-CURIX RP1 + CURIX UNIVERSAL SCREEN
TOTAL SPECTRUM MINUS SIGNAL SPECTRUM

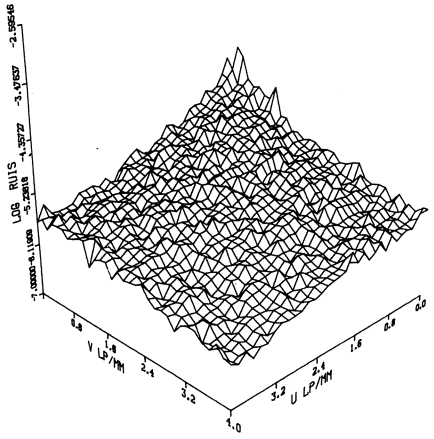


FIG.8-SCREEN NOISE FOR EXPERIMENTAL SCREEN

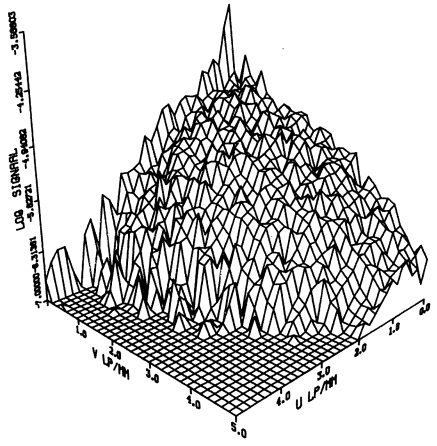


FIG.9-INFLUENCE OF QUANTISATION ON SIGNAL AND NOISE

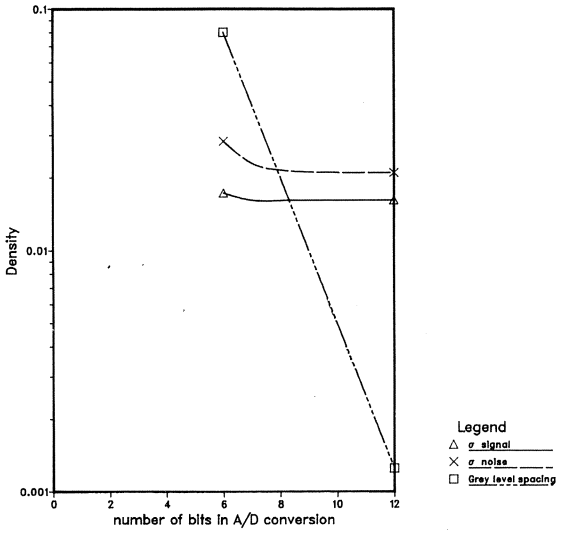
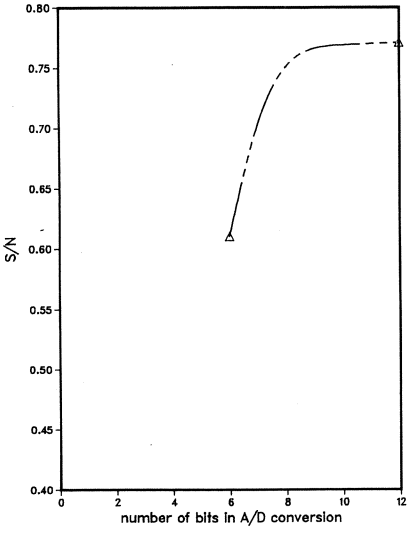


FIG.10-INFLUENCE OF QUANTISATION ON SIGNAL-TO-NOISE RATIO



ROBUST CALCULATION OF 3D TRANSONIC POTENTIAL FLOW BASED ON THE NON-LINEAR FAS MULTI-GRID METHOD AND A MIXED ILU/SIP-ALGORITHM

A.J. van der WEES

A mixed incomplete lower upper decomposition/strongly implicit procedure (ILU/SIP) relaxation algorithm is investigated within a non-linear FAS multi-grid research code. The algorithm is designed to be fast, robust (stable convergence for all local flow directions) and insensitive (good short wave damping for all possible mesh ratios). The algorithm involves only one free parameter.
Numerical results will be presented for the Laplace equation and for a transonic model disturbance equation solved for the flow in "a windtunnel with a bump on the bottom".

NOMENCLATURE

A, A^*, B	matrix-vector operators, equations (20), (21)
a,b,c	sensitivity constants, simulating mesh ratios, equation (29)
e	error vector, equation (27)
F^N	right-hand side in G^N-problem, equation (1)
$\hat{F}^K$	right-hand side in G^K-problem, equation (2)
G^K	grid of level K
G	complex reduction-factor, equation (28)
$\hat{g}$	right-hand side in Von-Neumann boundary condition, equation (17)
g^{ij}	i,j = 1,2,3, contravariant metric tensor
h^K	mesh size on grid G^K
I_K^{K-1}	restriction operator from grid G^K to grid G^{K-1} acting upon dependent variables
$\bar{I}_K^{K-1}$	restriction operator in Von-Neumann boundary condition acting upon dependent variables
I_{K-1}^K	prolongation operator from grid G^{K-1} to grid G^K
i_K, j_K	grid-point indices on grid G^K in two dimensions
i,j,k	grid-point indices; also grid-dimensions

*Part of this research has been performed under contract with the Netherlands Agency for Aerospace Programs (NIVR).

$\hat{\imath}$	imaginary unit
J	Jacobian
K	Level index in multi-grid method, K=1...N
L^K	general non-linear operator on grid G^K
$\tilde{L}^K$	correction operator on grid G^K
L,U	lower and upper matrices
1^1	number of iteration sweeps on grid G^1
M_∞	freestream Machnumber
m^K,n^K,p^K	numbers of iteration sweeps on grid G^K,K > 1
N	number of levels in multi-grid method
n	iteration count
Q	auxiliary constant, equations (31), (35), (36)
R^K	residual on grid G^K, equation (4)
v_K,v	amplification factor per cycle in a 2-level multi-grid method
W_K^{K-1}	restriction operator from grid G^K to grid G^{K-1} acting upon residuals
$\bar{W}_K^{K-1}$	restriction operator in Von-Neumann boundary condition acting upon residuals
w_K	amplification factor per cycle in a K-level multi-grid method
x,y,z	orthogonal coordinates
α	ILU/SIP parameter
β_i	i = 3,4...2(N-2), coefficients in equation (16)
γ	= 1,4, ratio of specific heats
Δx_K	mesh size in x-direction on G^K
$\Delta\phi$	correction to ϕ after one iteration sweep, equation (21)
ε	small parameter, introduced in tables 1,2,3
ϕ^K	exact solution of G^K-problem (here disturbance potential)
ϕ^K,ϕ_o^K	approximate solution of G^K problem
$\psi^K,\psi_{appr.}^{K-1}$	exact value on G^K and approximate value on G^{K-1} of $(\varphi^K-\phi^K)$
κ	positive constant, equation (13)
λ^K	wavelength on grid G^K
λ	reduction-factor per work unit in the multi-grid method
μ,θ,ω	"frequencies" of Fourier components in the error
ν	integer defining a fixed recursive strategy withing the multi-grid method
$\rho,\bar{\rho}$	reduction-factor, respectively maximum reduction-factor in the high-frequency part of the error spectrum, obtained from a local-mode analysis

$\overleftarrow{\partial}_x, \overleftarrow{\partial}_t$ first order backward finite difference operators

1. INTRODUCTION

Finite difference methods for the calculation of 3D transonic potential flow are of growing importance in aerodynamic design. However, much work has yet to be done before they become a mature capability. This involves the extension of the methods to complicated geometries (finite volume approach, zonal approach), the coupling with boundary layer calculation methods and ultimately the incorporation in (interactive) design processes. Quite a different aspect is the obvious need for finer grids and better convergence levels. The common necessary requirements here with respect to the solution algorithms are "fast" and "robust" (stable convergence for all local flow directions). Though rather reliable, the Successive Line Over Relaxation (SLOR) schemes that are still being used today in a good many routinely used computer codes satisfy neither of these requirements.

For a number of years, now, efforts are under way to obtain faster algorithms. In mono-grid methods the Approximate-Factorization (AF) schemes are promising[1,2,3,4].Another interesting development is the Strongly Implicit Procedure (SIP)[5].A breakthrough, however, seems to have been the multi-grid(MG) method [6,7].
An important ingredient in any multi-grid method is the relaxation algorithm used to smooth the errors. So far, SLOR, ADI (Alternating Direction Implicit) as well as SIP have been used succesfully as the smoothing algorithmin transonic applications of the multi-grid method [8,9,10,11, 12,13,14,15,16,17]. The "best" smoothing algorithm is obviously determined by the balance that exists between its damping characteristics for short-wave errors, its computational complexity and its robustness and insensitivity (good short wave damping for all possible mesh ratios).

The aim of the paper is to present research that has been carried out at NLR to construct a robust and insensitive smoothing algorithm within the multi-grid method that performs (hopefully) faster than SLOR. To that end, a suitable candidate for the smoothing algorithm, the Incomplete Lower Upper (ILU) decomposition scheme [18,19,20,21], has been thoroughly investigated. In its most general form, ILU can be looked upon as a general scheme that includes SIP[14] as a special case. It will be

demonstrated in the paper that a particular form of this general scheme, denoted by ILU/SIP, is a promising insensitive and robust candidate for the smoothing algorithm in multi-grid methods for transonic applications. Its performance as a smoothing algorithm within a so-called non-linear Full Approximation Storage (FAS) multi-grid method will be demonstrated by solving a form of the transonic small-disturbance equation on a rectangular domain.

Most results presented in this paper are taken from [22], which was presented at the AIAA 6th Computational Fluid Dynamics Conference, Danvers, Massachusetts, USA (July 1983). Some more recent results have been included also to show the current capacilities of the ILU/SIP algorithm within the FAS multi-grid method.

The results of the present investigation have provided the basis for the fast solvers that are being implemented at NLR in the production codes for the calculation of transonic flow about airplane configurations.

2. MULTI-GRID METHOD

The concept underlying the multi-grid method is to eliminate efficiently each Fourier component of the error spectrum on the coarsest possible grid. This concept relies on the use of relaxation algorithms that are very efficient in damping those components of the error whose wavelength, in at least one of the coordinate-directions, is comparable to the mesh size. For non-linear equations, the so-called non-linear Full Approximation Storage (FAS) [6,7] has been used by many investigators [8,9,10,11,12,13,14, 15,16] and is now widely accepted. A brief outline of the FAS multi-grid method follows below.

Consider the discretized non-linear boundary value problem

$$L^N \varphi^N = F^N \tag{1}$$

on the finest grid G^N of a sequence of grids G^K, $K = 1,2,\ldots,N$, of decreasing mesh size in the computational domain. Here, G^{K-1} is constructed from G^K by leaving out every other gridpoint and hence the mesh size in any coordinate-direction satisfies $h^{K-1} = 2h^K$.
Since the FAS multi-grid method is a recursive process, it is sufficient to explain the relationship between the problems that must be solved on

the grids G^K and G^{K-1}.
Suppose that the G^K-problem is (note that $\hat{F}^N = F^N$)

$$L^K \varphi^K = \hat{F}^K. \tag{2}$$

where φ^K is a yet unknown approximation of φ^N on G^K. Note, that ϕ^K can only contain Fourier components for which $\lambda^K \geq 2h^K$ in each coordinate-direction. Further, let ϕ^K be a given approximation of φ^K. Then the necessary correction $\psi^K = \varphi^K - \phi^K$ can be solved from the correction equation

$$\tilde{L}^K \psi^K = L^K(\phi^K + \psi^K) - L^K \phi^K = R^K, \tag{3}$$

where the residual R^K is defined by

$$R^K = \hat{F}^K - L^K \phi^K. \tag{4}$$

As it is more efficient to solve those components of ψ^K, which are smooth on G^{K-1} ($\lambda^K \geq 2h^{K-1}$ in each coordinate-direction), as much as possible on the coarser grid G^{K-1}, it is worthwhile to approximate equation (3) on the grid G^{K-1}. In the FAS multi-grid method this approximation is

$$\begin{aligned} \tilde{L}^{K-1} \psi^{K-1} \equiv\ & L^{K-1}(I_K^{K-1} \phi^K + \psi^{K-1}) - \\ & + L^{K-1}(I_K^{K-1} \phi^K) = W_K^{K-1} R^K. \end{aligned} \tag{5}$$

Here I_K^{K-1} and W_K^{K-1} are restriction operators (not necessarily the same) that assign (smoothed) values of ϕ^K and R^K to each gridpoint of G^{K-1}. By the cut-off character of the restriction operators, ψ^{K-1} can only contain Fourier components which are smooth on G^{K-1}.
A convenient way to solve equation (5) is to introduce

$$\varphi^{K-1} = I_K^{K-1} \phi^K + \psi^{K-1} \tag{6}$$

and consequently write the G^{K-1} problem in the form

$$L^{K-1} \varphi^{K-1} = \hat{F}^{K-1}, \tag{7}$$

where

$$\hat{F}^{K-1} = W_K^{K-1} R^K + L^{K-1}(I_K^{K-1} \phi^K). \tag{8}$$

Since L^{K-1} and L^K approximate the same differential equation they can be taken (and will be taken in this investigation) identical. This has the advantage that the same relaxation algorithm can be used on each grid. Now φ^{K-1}, or rather an approximation ϕ^{K-1} of it, can be solved from equation (7) by doing a number of relaxation sweeps, starting from $I_K^{K-1}\phi^K$ as the initial guess. Then ψ^{K-1} is obviously approximated by

$$\psi^{K-1}_{appr.} = \phi^{K-1} - I_K^{K-1}\phi^K, \tag{9}$$

see equation (6).
It follows that the original approximation ϕ^K to φ^K can be improved to ϕ^K_o by putting

$$\phi^K_o = \phi^K + I^K_{K-1}\psi^{K-1}_{appr.} = \phi^K + I^K_{K-1}(\phi^{K-1} - I_K^{K-1}\phi^K), \tag{10}$$

where I^K_{K-1} is a prolongation operator that assigns values of $\psi^{K-1}_{appr.}$ to each gridpoint of G^K by interpolation.
This way, mainly the short wave components of ψ^K ($2h^K \leq \lambda^K < 2h^{K-1}$ in at least one coordinate-direction) need to be determined from the G^K-problem, equation (2) or (3). As the relaxation algorithm was required to be very efficient in damping error components of precisely that part of the spectrum, the short wave components of φ^K can be determined doing only a few relaxation sweeps.

Fixed strategies: V-cycles versus W-cycles

The relationship between the G^K-problem and the G^{K-1}-problem as explained above is depicted in Fig. 1. It remains to define a strategy by which this relationship is employed recursively on the grid sequence $G^K, K = 1,2,\ldots,N$. Here, only fixed strategies will be considered.

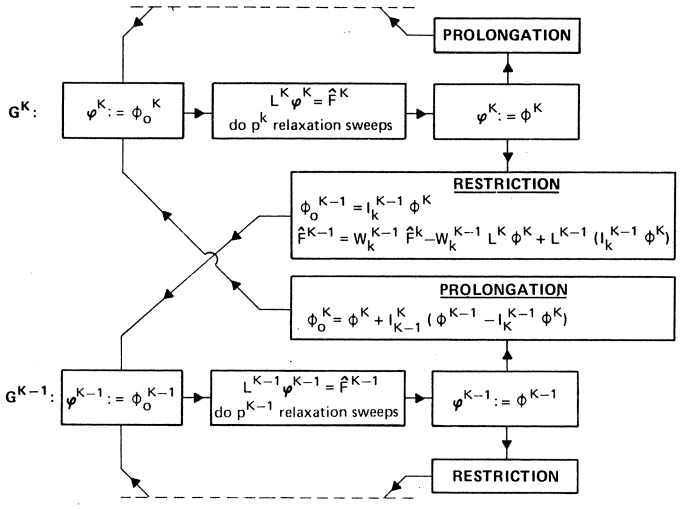


Fig. 1 Relationship between the G^K-problem and the G^{K-1}-problem in the non-linear FAS multi-grid method

The simplest fixed strategy by far is obviously the repeated application of a so-called V-cycle (see Fig. 2). In transonic calculations this simple form of fixed strategy has been used successfully by many authors [9,11, 13,14,15,16,17]. It will be shown in this paper that a somewhat more complicated fixed strategy, called a W-cycle (see Fig. 3), is in fact more promising. A theoretical result of Hackbush [23] substantiates this point of view and will be summarized below.

Consider a 2-level-multi-grid method to solve the G^K-problem in which the G^{K-1}-problem is solved exactly in each cycle. After one cycle, the original error ψ^K is then reduced to ψ^K_{new} according to the relation

$$\|\psi^K_{new}\| = \nu_k \|\psi^K\| . \tag{11}$$

A necessary and sufficient condition for convergence is, of course, that the amplification factor ν_K satisfies $0 \le \nu_k < 1$.
Next consider a N-level multi-grid method to solve the G^N-problem according to the following fixed recursive strategy:

"Within each cycle to approximately solve a G^K-problem, $K = 3,\ldots,N-1$, the cycle to approximately solve a G^{K-1}-problem is ν times repeated. Furthermore, all G^1-problems are solved exactly".

Obviously, $\nu = 1$ represents a V-cylce (see again Fig. 2), while $\nu = 2$ represents W-cycles of the type shown in Fig. 3.

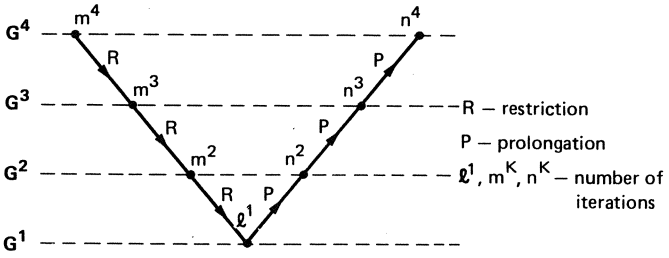


Fig. 2 Example of a V-cycle on four grids

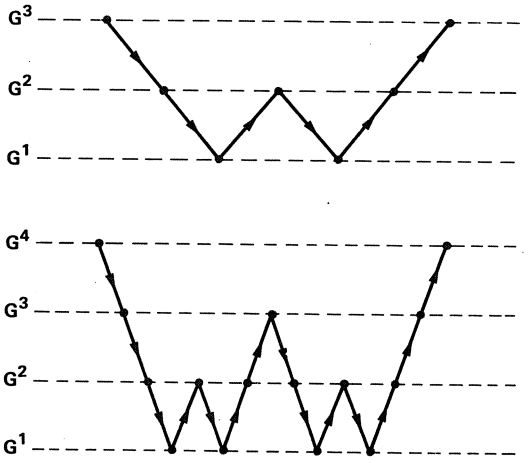


Fig. 3 Examples of a W-cycle on three and on four grids

If w_K is the amplification factor of one such cycle to approximatley solve a G^K-problem, there must hold

$$\|\psi^K_{new}\| = w_K \|\psi^K\| . \tag{12}$$

For a rather large class of non-linear elliptic problems, Hackbush [23] has derived the following recursive inequality,

$$w_K \le v_K + \kappa (w_{K-1})^\nu, \quad w_1 = 0, \quad \kappa > 0, \tag{13}$$

for this particular strategy.

Now suppose, for example, $v_K = v$ for $K \geq 2$ and $\kappa = 1$. Then, following the equality sign in (13), there is obtained for the V-cycle (ν=1),

$$w_N = (N-1)v. \tag{14}$$

This indicates that the V-cycle strategy can diverge for large enough N, so that in general the convergence of V-cycle strategies is not guaranteed. Similarly, the W-cycle strategy corresponding with $\nu = 2$ leads to

$$w_N = v_N + (v_{N-1}+(\ldots(v_5+(v_4+(v_3+v_2^2)^2)^2)^2\ldots)^2)^2. \tag{15}$$

For $v_K = v < 1$, $K \geq 2$ this is a finite series of the form

$$w_N = v + v^2 + \beta_3 v^3 + \beta_4 v^4 + \ldots \beta_{2(N-2)} v^{2(N-2)}. \tag{16}$$

Comparison of equations (14) and (16) indicates that the prospects of obtaining $w_N < 1$ are in general better for the W-cycle strategy. In fact, Hackbush [21] has proved that the W-cycle strategy converges if the coarsest grid is fine enough. W-cycle strategies have been used successfully by several authors [25,26,27]. Recently, Braess [24] has proved that the V-cycle strategy also converges under the additional assumption that at least one relaxation sweep has to be performed before and after the coarse grid correction on each grid ($m^k > 0$ and $n^k > 0$ in Fig. 2).

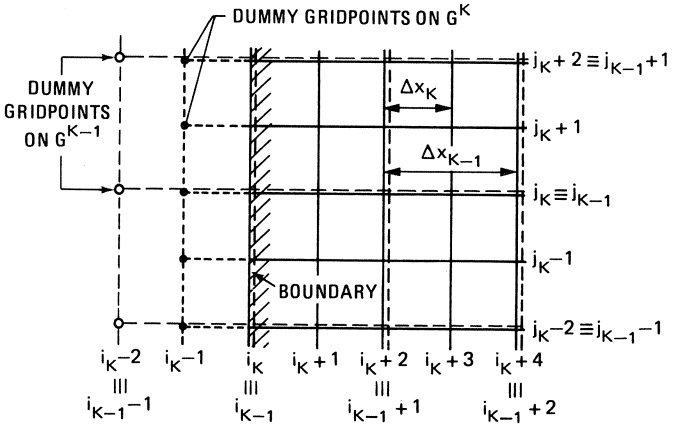


Fig. 4 Example of dummy gridpoints on G^K and G^{K-1} for a two-dimensional situation

Von Neumann boundary conditions

In applying multi-grid methods, the treatment of Von-Neumann boundary conditions requires careful attention. In this paper, these conditions will be enforced using dummy gridpoints. The procedure is best explained for a simple two-dimensional example of a G^K-problem (see Fig. 4),

$$\frac{\varphi^K_{i_K+1,j_K} - \varphi^K_{i_K-1,j_K}}{2\Delta x_K} = \hat{g}_{i_K,j_K}, \tag{17}$$

where (i_K-1,j_K) is the dummy grid point on grid G^K. Equation (17) is in fact the "boundary-equivalent" of equation (2) and hence will be treated in a similar fashion as much as possible. Therefore, if ϕ^K is again a given approximation of φ^K, the restriction of equation (17) to the G^{K-1}-problem is formally,

$$\begin{aligned}\frac{\varphi^{K-1}_{i_{K-1}+1,j_{K-1}} - \varphi^{K-1}_{i_{K-1}-1,j_{K-1}}}{2\Delta x_{K-1}} &= \\ = \bar{W}^{K-1}_K\left(\hat{g}_{i_K,j_K} - \frac{\phi^K_{i_K+1,j_K} - \phi^K_{i_K-1,j_K}}{2\Delta x_K}\right) &+ \\ + \frac{(I^{K-1}_K\phi^K)_{i_{K-1}+1,j_{K-1}} - (I^{K-1}_K\phi^K)_{i_{K-1}-1,j_{K-1}}}{2\Delta x_{K-1}}&.\end{aligned} \tag{18}$$

In equation (18), the boundary-restriction operator $\bar{W}^{K-1}_K$ can involve only boundary gridpoints of G^K. On the other hand, the restriction operator I^{K-1}_K is the same as defined before. The difficulty with equation (18) is that the second term in the right-hand side involves the gridpoint $(i_{K-1}-1,j_{K-1}) = (i_K-2,j_K)$ where the operation $I^{K-1}_K\phi^K$ is undefined. Hence, equation (18) is approximated as follows,

$$\begin{aligned}\frac{\varphi^{K-1}_{i_{K-1}+1,j_{K-1}} - \varphi^{K-1}_{i_{K-1}-1,j_{K-1}}}{2\Delta x_{K-1}} &= \\ = \bar{W}^{K-1}_K\left(\hat{g}_{i_K,j_K} - \frac{\phi^K_{i_K+1,j_K} - \phi^K_{i_K-1,j_K}}{2\Delta x_K}\right) &+ \\ + \bar{I}^{K-1}_K\left(\frac{\phi^K_{i_K+1,j_K} - \phi^K_{i_K-1,j_K}}{2\Delta x_K}\right),&\end{aligned} \tag{19}$$

where $\bar{I}_K^{K-1}$ is now a boundary-restriction operator involving only boundary gridpoints of G^K. Also here, the boundary-restriction operators $\bar{W}_K^{K-1}$ and $\bar{I}_K^{K-1}$ need not necessarily be the same.

3. ILU/SIP-ALGORITHM

General description

An extensive treatment of ILU and SIP can be found in Meijerink and Van der Vorst [19] and Stone [18] respectively. Here, only a brief description will be presented.
Quasi-linearization of the G^K-problem, equation (2), results in the matrix-vector equation

$$A^K[\varphi^K]\varphi^K = f^K. \tag{20}$$

If ϕ^K is a given approximation to φ^K, an iteration process (note that the irrelevant grid-level index K has been dropped) to solve this equation can be described as

$$\begin{cases} A^*[\phi^n]\Delta\phi^n = f - A[\phi^n]\phi^n, \\ A^*[\phi^n] \equiv A[\phi^n] + B[\phi^n], \\ \Delta\phi^n = \phi^{n+1} - \phi^n. \end{cases} \tag{21}$$

This results in the modified equation

$$B[\phi^n]\Delta t\, \hat{\partial}_t \phi^{n+1} = f - A[\phi^n]\phi^{n+1}, \tag{22}$$

where the error matrix B should be chosen such that the iteration matrix A^* is easily invertible. In both ILU and SIP the error matrix B is derived from an incomplete decomposition [18,19] of the system matrix A, viz. the LU-decomposition of a sparse matrix approximating A. This results in sparse lower and upper matrices L and U. The easily invertible product LU defines the iteration matrix A^* and the errormatrix B according to the relation

$$A^* = LU = A + B. \tag{23}$$

As an illustration of the matrices A and B, their structure is sketched in Fig. 5 for the case that a form of ILU is applied to the 7-point discretization of the Laplace-operator.

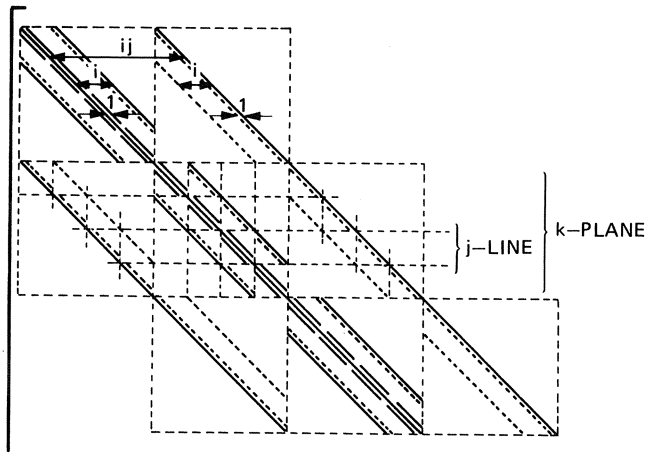


Fig. 5 Upper left corner of the patterns of the system matrix A (drawn lines) and the error matrix B (dotted lines) for an ILU-decomposition of the 7-point discretization of the Laplace-operator on an i*j*k-grid.

In the literature, several versions of ILU and SIP algorithms can be found. Incomplete Crout-decomposition (or Cholesky-decomposition, if the matrix is symmetric) is most commonly used.
The algorithms differ in the treatment of the entries of the errormatrix B. The following versions can be distinguished:

1. The term with $\overleftarrow{\partial}_t \phi^{n+1}$ in the modified equation (22) is left untouched, meaning that the entries of B need not be computed. This version is known as ILU.

2. Elimination of the term with $\overleftarrow{\partial}_t \phi^{n+1}$ from the modified equation (22) by what is generally called "lumping": during the incomplete decomposition of A, all entries in each row of B are computed and added to the main diagonal of A. In terms of finite-difference approximations this corresponds to a zeroth-order Taylor-expansion of the entries in the error matrix B.

3. Elimination of the term with $\overleftarrow{\partial}_t \phi^{n+1}$ from the modified equation (22) by a first-order Taylor-expansion of the entries in the error matrix B.
The basic interpolation-formula for an entry $B_{i+p,j+q,k}$ of B is on a uniform grid

$$B_{i+p,j+q,k} = B_{i+p,j,k} + B_{i,j+q,k} - B_{i,j,k}, \tag{24}$$

where the entries on the right-hand side are assumed to be contained within the matrixpattern of A.

Equation (24) is used as follows. If, as a result of the incomplete decomposition of the matrix A, there appears an entry $B_{i+p,j+q,k}$ in a row of the error matrix B, it is added to the off-diaonal entries $A_{i+p,j,k}$, $A_{i,j+q,k}$ and substracted from the diagonal entry $A_{i,j,k}$ of the systemmatrix A. This version is known as SIP.

In most cases the entries of the matrix A are small compared to the largest entries of the systemmatrix A. However, they are not insignificant. They can be helpful (as is assumed with ILU, version 1), especially in elliptic problems. In other cases, the choice of Taylor-expansion (versions 2 and 3) can be crucial in order to obtain a fast and insensitive algorithm.

The ILU/SIP-algorithm is obtained by taking the convex combination of SIP (method 3) and ILU (method 1),

$$\text{ILU/SIP}(\alpha) \equiv \alpha*\text{SIP} + (1-\alpha)*\text{ILU}, \quad 0 \le \alpha \le 1. \tag{25}$$

This combination has, implicitly, been used earlier [5,14,18], but it has never been recognized as such. This explains why experience with ILU [20,21] has not been used in the past within SIP.

Two important aspects of the ILU/SIP-algorithm are its full implicitness and the absence of a preferred sweep-direction. As a consequence local flows in all possible directions are solved equally well. The price for the full implicitness is the need to store the entire upper triangular matrix U, (see equation (23)). Under "Coding aspects" it will be shown that the disk-storage required is not restrictive. The alternative is, of course, to use an algorithm which is restricted to being implicit per plane in order to allow incomplete LU decomposition per plane in core. Such an algorithm, called plane-ILU[†] is in fact a three-dimensional generalization of the two-dimensional SLOR-algorithm, in which the solution of the tridiagonal system per line has been replaced by the incomplete LU decomposition per plane. This method will be analyzed for an elliptic test-problem and compared with (fully implicit) ILU/SIP.

[†]This plane-ILU algorithm is not to be confused with the line-ILU algorithm developed by Meijerink and presented by Kettler [28], which is in fact a block-ILU algorithm.

Multiple applications [20,21,27] have shown that ILU within the multi-grid method is a robust and insensitive tool for the solution of a wide variety of elliptic problems, which, on a uniform grid, is at least two times faster than SLOR within the multi-grid method. Application of SIP within the multi-grid method for the solution of transonic flow [14] has led to results which are promising, but which leave room for improvement. In what follows, the good results obtained by Wesseling [21] for two-dimensional elliptic problems using ILU within the multi-grid method will be generalized to the three-dimensional mixed type problems of transonic flow by using ILU/SIP within the multi-grid method.

Local mode analysis

In particular the damping of the high-frequency components in the error spectrum and the insensitivity of the ILU/SIP-algorithm will be analyzed by means of the local mode (Von Neumann) analysis. This is an effective tool to investigate locally the damping of the various (high-frequency) Fourier components of the error. As the local mode analysis is not valid near shocks and sonic surfaces, the analysis will be restricted to the relaxation of strictly elliptic and hyperbolic finite-difference schemes.

The local mode analysis consists of two steps. First the equations

$$(26)\qquad \begin{cases} (A\varphi)_{ijk} = f_{ijk} \\ e_{ijk} = \phi^{n}_{ijk} - \varphi_{ijk} \\ \Delta\phi^{n}_{ijk} = \phi^{n+1}_{ijk} - \phi^{n}_{ijk} = e^{n+1}_{ijk} - e^{n}_{ijk} \end{cases}$$

for gridpoint i,j,k are combined with equation (21) to give

$$(27)\qquad (A+B)(e^{n+1}-e^{n})\big|_{ijk} = -Ae^{n}\big|_{ijk}.$$

Next, the Fourier component $e^{n}_{ijk} = (G[\mu,\theta,\omega])^{n} e^{\hat{\imath}(i\mu+j\theta+k\omega)}$ is substituted into this expression, leading to the reduction-factor

$$(28)\qquad \rho[\mu,\theta,\omega] = \left|G[\mu,\theta,\omega]\right|,$$

which can be analyzed as a function of the "frequencies" $0 < \mu,\theta,\omega \leq \pi$. For the use within the multi-grid method, in particular the high-frequency modes of the error (at least one "frequency" is high, e.g. $\frac{\pi}{2} < \mu \leq \pi$) should be damped efficiently. A good measure for this is the maximum value $\bar{\rho}$ of $\rho(\mu,\theta,\omega)$ over the high-frequency part of the (μ,θ,ω)-domain.

Elliptic testproblem

For the testproblem the commonly used seven-point discretization of the elliptic equation

$$a\varphi_{xx} + b\varphi_{yy} + c\varphi_{zz} = 0, \quad a,b,c > 0 \tag{29}$$

will be taken.
The following relaxation algorithms will be compared:
-SLOR-x (successive relaxation of x-lines),
-SLOR-xyz (three subsequent sweeps of SLOR-x, SLOR-y and SLOR-z),
-y-plane ILU (incomplete LU-decomposition of planes of constant y),
-ILU (version 1 in "General description"),
-ILU with lumping (version 2 in "General description"),
-SIP (version 3 in "General description"),
-ILU/SIP(α) (convex combination: α*SIP + (1-α)*ILU, $0 \leq \alpha < 1$).
The sensitivity of each algorithm is investigated by employing various values of a, b and c, simulating various mesh ratios in a computational domain.

If $\Delta x = \Delta y = \Delta z$, discretization of equation (29) by central differences yields for the systemmatrix A:

$$\begin{aligned}(A\phi)_{000} \equiv & -a\phi_{-100} - b\phi_{0-10} - c\phi_{00-1} + \\ & + (2a+2b+2c)\phi_{000} - a\phi_{100} - b\phi_{010} - c\phi_{001} = 0,\end{aligned} \tag{30}$$

where ϕ_{pqr} is short for $\phi_{i+p,j+q,k+r}$.
For the operator B, the following expressions can be found:
-SLOR-x: $(B\phi)_{000} = b\phi_{010} + c\phi_{001}$,
-SLOR-xyz: three subsequent sweeps of SLOR-x, SLOR-y and SLOR-z,
-y-plane ILU: $(B\phi)_{000} = b\phi_{010} + \frac{ac}{Q}(\phi_{-101}+\phi_{10-1})$,

$$Q = a + b + c + \sqrt{(a+b+c)^2 - (a^2+c^2)},$$

-ILU: $(B\phi)_{000} = \frac{1}{Q}[ac(\phi_{-101}+\phi_{10-1}) + ab(\phi_{-110}+\phi_{1-10}) + bc(\phi_{01-1}+\phi_{0-11})],$

$$Q = a + b + c + \sqrt{(a+b+c)^2 - (a^2+b^2+c^2)},$$

-ILU with lumping: $(B\phi)_{000} = (B_{ILU}\phi)_{000} - \frac{2}{Q}(ac+ab+bc)\phi_{000}$,

-ILU/SIP(α): $(B\phi)_{000} = (B_{ILU}\phi)_{000} + \frac{\alpha}{Q}[ac(-\phi_{-100}-\phi_{001}-\phi_{100}-\phi_{00-1}+2\phi_{000}) +$

$$+ ab(-\phi_{-100}-\phi_{010}-\phi_{100}-\phi_{0-10}+2\phi_{000}) +$$

$$+ bc(-\phi_{010}-\phi_{001}-\phi_{0-10}-\phi_{00-1}+2\phi_{000})],$$

-SIP: $(B\phi)_{000} = (B_{ILU/SIP}\phi)_{000}$ with $\alpha=1$.

Substitution of equations (30) and (31) into equation (27) leads to the results presented in table 1, which have been obtained by numerical evaluation of equation (27) for the frequency values $\mu,\theta,\omega = \pm\frac{2\pi}{K}$, K = 2,3,4,8,16,32,64. These values are considered to be representative for the entire frequency-domain. Due to the symmetry of the problem, only the values of a,b and c mentioned in the table need be considered.

Table 1 Maximum reduction-factors $\bar{\rho}$ of high-frequency modes for various algorithms in case of the elliptic testproblem, equation (29)

algorithm / (a,b,c)	SLOR--x	SLOR--xyz	y-plane ILU	ILU with lumping	ILU/SIP (α)		
					ILU $\alpha = 0$	ILU/SIP (.7) $\alpha = .7$	SIP $\alpha = 1$
(1,1,1)	.50	.48	.41		.22	.27	.27
$\varepsilon = .1$				U			
$(\varepsilon,1,1)$	.90	.85	.82	N	.72	.66	.24
$(1,\varepsilon,1)$	.83		.72	S			
$(1/\varepsilon,1,1)$	.49	.69	.44	T	.34	.32	.17
$(1,1/\varepsilon,1)$	.84		.83	A			
$(1,\varepsilon,1/\varepsilon)$	.97	.90	.81	B	.81	.76	.15
$(\varepsilon,1/\varepsilon,1)$	.97		.97	L			
$(1/\varepsilon,1,\varepsilon)$	.49		.77	E			

Table 1 (continued)

algorithm / (a,b,c)	SLOR--x	SLOR–xyz	y-plane ILU	ILU with lumping	ILU/SIP (α) ILU α = 0	ILU/SIP (.7) α = .7	SIP α = 1
(1,1,1)	.50	.48	.41		.22	.27	.27
ε = .001				U			
(ε,1,1)	.99	.98	.98	N	.97	.95	.38
(1,ε,1)	.98		.97	S			
($^1/\varepsilon$,1,1)	.16	.53	.17	T	.28	.15	.03
(1,$^1/\varepsilon$,1)	.99		.99	A			
(1,ε,$^1/\varepsilon$)	.99	.46	.18	B	.18	.17	.02
(ε,$^1/\varepsilon$,1)	.99		.99	L			
($^1/\varepsilon$,1,ε)	.10		.10	E			

Table 2 Maximum reduction-factors $\bar{\rho}$ of high-frequency modes for various algorithms in case the elliptic testproblem, equation (29), is reduced to the two-dimensional case

algorithm / (a,b)	SLOR–x	SLOR–xyz	y-plane ILU	ILU with lumping	ILU/SIP (α) ILU α = 0	ILU/SIP (α) α = .7	SIP α = 1
(1,1)	.45	.39	see SLOR–x	.63	.20	.21	.23
ε = .1							
(ε,1)	.83	.61		>1	.46	.26	.14
(1,ε)	.45						
ε = .001							
(ε,1)	.99	.67		.42	.17	.09	.02
(1,ε)	.45						

From table 1 it can be concluded that SLOR-x and y-plane ILU are very sensitive to variation of mesh ratios. For ε=.1 (a mesh ratio of $\sqrt{10}$ = 3.16) as well as ε=.001 (a mesh ratio of $\sqrt{1000}$ = 31.6), almost unacceptable maximum reduction-factors in the range $\bar{\rho}$ = .97-.99 can occur. SLOR - xyz, ILU and ILU/SIP(.7) are moderately sensitive. SIP is most intensitive. ILU with lumping is unstable. The best damping is obviously provided by ILU/SIP(α), where the insensitivity improves with increasing α. Except for (a,b,c) = (1,1,1), where the damping is nearly independent of α, the maximum reduction-factor $\bar{\rho}$ shows a similar tendency. The most insensitive character of SIP (α=1) is questionable, however, because this property is not preserved for values of α slightly lower than one. For this reason,

this most insensitive character of SIP will probably not show up in practical numerical experiments, where the basic assumptions of the local mode analysis (constant coefficients and periodic boundary conditions) are violated anyway. The final conclusion from table 1 is, that all ILU/SIP(α) - algorithms are expected to be in the order of two times faster than SLOR-x, SLOR-xyz and y-plane ILU within the multi-grid method on a uniform grid. The insensitivity is probably best served by taking ILU/SIP(α) with α closer to one, e.g. α = 0.7. As a consequence of this conclusion, the analysis for the hyperbolic testproblem that follows hereafter will be restricted to ILU/SIP(α) and SLOR-x, the latter being used as the reference case.

For purposes of comparison, the results for the two-dimensional case are presented in table 2. Here, SLOR-x (or y-plane ILU) is the only very sensitive algorithm. All other algorithms, expect ILU with lumping, which can be unstable, are relatively insensitive.

Hyperbolic testproblem

A representative hyperbolic testproblem is

$$a\varphi_{xx} + b\varphi_{yy} + c\varphi_{zz} = 0, \quad a,b,c > 0. \tag{32}$$

The discretization will use central-differencing in the y- and z-direction and upwind-differencing in the x-direction. This results for the system matrix A in:

$$\begin{aligned}(A\phi)_{000} \equiv\ & a\phi_{-200} - 2a\phi_{-100} + (a+2b+2c)\phi_{000} + \\ & - b\phi_{010} - b\phi_{0-10} - c\phi_{001} - c\phi_{00-1} = 0.\end{aligned} \tag{33}$$

The SLOR-x algorithm is stabilized by constructing the term $\Delta t\overleftarrow{\partial}_t(2a\overleftarrow{\partial}_x + b\overleftarrow{\partial}_y + c\overleftarrow{\partial}_z)\phi^{n+1}$ in the left-hand side of the modified equation (22) by choosing the error matrix B according to

$$(B\phi)_{000} \equiv b(\phi_{010}-\phi_{000}) + c(\phi_{001}-\phi_{000}) - a(\phi_{-200}-\phi_{000}). \tag{34}$$

This algorithm will be compared with the ILU/SIP(α)-algorithm for two cases:

- without adding any stabilizing terms;

$$(B\phi)_{000} = -\frac{1}{Q}[ac\phi_{-201} + ab\phi_{-210} - 2a\phi_{-101} + \\ - 2ab\phi_{-110} - bc(\phi_{0-11} + \phi_{01-1})] + \\ -\frac{\alpha}{Q}[-ac(\phi_{-200} + \phi_{-101} - \phi_{-100}) + \\ + 2ac(\phi_{-100} + \phi_{001} - \phi_{000}) + \\ - ab(\phi_{-200} + \phi_{-110} - \phi_{-100}) + \\ + 2ab(\phi_{-100} + \phi_{010} - \phi_{000}) + \\ + bc(\phi_{0-10} + \phi_{010} + \phi_{001} + \phi_{00-1} - 2\phi_{000}]$$

$$Q = b+c+\tfrac{1}{2}a+\sqrt{(b+c+\tfrac{1}{2}a)^2-(b^2+c^2)}. \qquad (35)$$

- with an extra stabilizing term $\Delta t\overleftarrow{\delta}_t(2a\overleftarrow{\delta}_x)\phi^{n+1}$ in the left-hand side of the modified equation (22) by choosing the errormatrix B according to

$$(B\phi)_{000} = -a(\phi_{-200}-\phi_{000}) + \frac{1}{Q}[2ab\phi_{-110} + \\ 2ac\phi_{-101} + bc(\phi_{0-11} + \phi_{01-1})] \\ + \frac{\alpha}{Q}[2ab(-\phi_{-100} - \phi_{001} + \phi_{000}) + \\ + 2ac(-\phi_{-100} - \phi_{001} + \phi_{000}) + 2\phi_{000}] \\ + bc(-\phi_{0-10} - \phi_{001} - \phi_{010} - \phi_{00-1} + 2\phi_{000}]$$

$$Q = a+b+c+\sqrt{(a+b+c)^2-(b^2+c^2)}. \qquad (36)$$

The choice of sensitivity parameters will be limited to $(a,b,c) = (\frac{1}{\varepsilon},1,1)$, $\varepsilon = .1$, because this is representative for a situation in the supersonic part of a highly transonic flow away from shocks and sonic surfaces. Numerical evaluation of the reduction-factors for the same frequencies as with the elliptic testproblem leads to the results presented in table 3. (Note in particular that all ILU/SIP(α)-algorithms are exact, $\bar{\rho} \equiv 0$, if the testproblem is reduced to the two-dimensional case.) Of each algorithm only the stability characteristics are given in the table, because these are considered to be the most important property.

Table 3 Convergence characteristics for various algorithms applied to the hyperbolic testproblem $-\frac{1}{\varepsilon}\phi_{xx} + \phi_{yy} + \phi_{zz} = f$, $\varepsilon = .1$

algorithm	stable for high frequencies	stable for low frequencies	$\overleftarrow{\partial}_t \phi^{n+1}$ in modified equation (22)
SLOR–x stabilized	yes	yes	no
ILU ($\alpha = 0$)	yes	no	yes
ILU stabilized ($\alpha = 0$)	yes	no	yes
SIP ($\alpha = 1$)	no	no	no
SIP stabilized ($\alpha = 1$)	yes	yes	no
ILU/SIP (.7)	yes	no	yes
ILU/SIP (.7) stabilized	yes	slightly unstable ($\bar{\rho} \approx 1.01$)	yes

The table shows that a non-vanishing $\overleftarrow{\partial}_t \phi^{n+1}$-term in the modified equation (22) always results in an algorithm that is unstable for the low-frequency components of the error spectrum. The reason is that the modified equation in this case is an unstable differential equation. Reversely, however, the absence of $\overleftarrow{\partial}_t \phi^{n+1}$ in the modified equation (22) does not guarantee a stable algorithm, because the CFL-criterion has to be satisfied. Obviously, this is not the case with SIP. Stabilizing SIP by adding $\overleftarrow{\partial}_x \overleftarrow{\partial}_t \phi^{n+1}$ appears to be possible, however. Summarizing, both SLOR-x and SIP can be stabilized adding $\overleftarrow{\partial}_x \overleftarrow{\partial}_t \phi^{n+1}$ to the modified equation (22). The two versions of ILU and ILU/SIP(.7) are only unstable for low frequencies, but can probably be used within the multi-grid method to reduce high-frequency error components. Note that table 3 has to be interpreted with caution with reference to the mixed elliptic/hyperbolic problems of transonic flow. Also, boundary-conditions, shocks and sonic surfaces are not included in the local mode analysis. The practical use of the local mode analysis for the hyperbolic testproblem is therefore of limited value here, although it gives useful theoretical information. In practical transonic experiments, however, it will appear that even the presence of $\overleftarrow{\partial}_t \phi^{n+1}$ in the modified equation can still produce convergence, as will be shown in the results.

Robustness and insensitivity

Due to its full implicitness, ILU/SIP has the desirable property that it can, in principle, converge for all local flow directions (robustness)

without the need to choose a sweep-direction. Also the convergence characteristics within the multi-grid method are not severely affected on stretched grids (insensitivity). This is in contrast to SLOR-algorithms, where the sweep-direction influences the robustness as well as the insensitivity. This makes ILU/SIP a suitable candidate for use within the multi-grid method, especially in those cases where the local flow-direction varies strongly (e.g. in air-intakes).

A further useful property of ILU/SIP is its flexibility with respect to the choice of the matrix pattern for the lower and upper triangular matrices L and U. For instance, less sparse difference-molecules than the 7-point Laplace-discretization can be fully accommodated within the ILU/SIP-algorithm by a simple extension of these matrix patterns. Extension of the matrix pattern can however also be applied for sparse difference-molecules to improve the convergence and insensitivity characteristics of the ILU/SIP-algorithm within the multi-grid method. The price to be paid is, of course, that a less sparse upper matrix U has to be stored.

Coding aspects

An important demand at NLR with respect to the choice of a relaxation-algorithm within the multi-grid method has been that it should be possible to implement it on a computer of large, but limited, addressable storage capacity (.5-2 Mwords). This has been translated to the basic requirement that the algorithm should have a plane-by-plane structure, enabling a code to be set up requiring only a limited number of planes in core at the same moment. Vectorizability is desirable but not a pacing item.

The evaluation of residuals has proved to be expensive in finite volume codes. Because residuals have to be evaluated in each relaxation-sweep, even a rather expensive algorithm can become cost-effective, provided its reduction-rate is high. For this reason the reduction-factors in table 1 and 2 have not been corrected by the different amount of work that has to be done in each algorithm (if the evaluation of residuals becomes very expensive, these amounts of work will become almost equal).

By its full implicitness, the ILU/SIP-algorithm requires that the entire computational domain be updated simultaneously. Hence the whole upper triangle matrix U must be stored, requiring (for the test-problem

in this paper) storage-capacity for four large vectors (of length equal to the solution-vector). As in general this upper matrix U cannot be stored in core, IO-transfers have to be made in each relaxation-sweep. These transfers can be carried out plane-by-plane. The absence of a preferred sweep-direction (see under "Robustness and insensitivity") can be used to advantage by choosing the sweep-direction and the number of planes to be transferred simultaneously, in such a way that an optimal balance between in-core storage and usage of IO-time is obtained.

The vectorizability of the ILU/SIP-algorithm has not yet been thorougly investigated. Although intrinsically difficult to vectorize, the algorithm can be modified [29] to fullvectorizability in the case of a linear problem. It is expected, though, that the robustness and insensivity of the algorithm will be more or less damaged by this modification.

4. RESULTS

Description of test-problem

The transonic small-disturbance equation

$$(37) \qquad [(1-M_\infty^2)\varphi_x - \tfrac{1}{2}(\gamma+1)M_\infty^2\varphi_x^2]_x + [\varphi_y]_y + [\varphi_z]_z = 0,$$

is solved on a rectangular domain using a finite-volume type fully conservative finite-difference scheme. Roughly speaking, this envolves central differencing in elliptic areas ($M < 1$), upwind differencing in hyperbolic areas ($M > 1$), while special switch operators are used at the interfaces [30]. The configuration is depicted in Fig. 6 and it is in fact "a windtunnel with a bump (simulating a non-lifting airfoil) on the bottom". In the x-direction, the "bump" is a doubly-coupled semi-airfoil whose thickness varies in the y-direction. The maximum thickness is 5 % of the chord. A sample pressure coefficient and Machnumber distribution showing a strong shock wave, are presented in Fig. 7. The physical domain is covered either by a uniform or by a stretched grid (see Fig. 8); in the latter case the physical domain is larger. In most cases calculated the multi-grid uses a sequence of four grids G^4, G^3, G^2, G^1, which is constructed by a repeatedly leaving out every other gridpoint; occasionally the coarsest grid G^1 will be discarded. In most of the results that will be presented the finest grid G^4 employs 64*16*16 meshes. However, some results

on a finer grid of 96*24*16 will also be shown. In many cases, calculations have been carried out using the so-called full multi-grid method, in which the physical problem is subsequently solved on the grids G^1, G^2, G^3 (each time using the multi-grid method) before the multi-grid process on G^4 is actually started (see Fig. 9). This generally has a favourable effect on the (gradual) build-up of supersonic zones.

Various experiments using MG(non-linear FAS) ILU/SIP

The relations between the calculations on the various grids $G^1 \ldots G^N$ in the non-linear FAS multi-grid method are summarized in Fig. 1. The restriction process involves the two restriction operators W_K^{K-1} (working on residuals) and I_K^{K-1} (working on dependent variables; here, disturbance potentials φ). Both injection operators and the smoothing operator defined in Fig. 10 will be tested and compared. The prolongation process involves the prolongation operator I_{K-1}^{K}. Both tri-linear interpolation and four point cubic interpolation in the three coordinate directions, compare Shmilovich and Caughey [11], will be tested and compared.

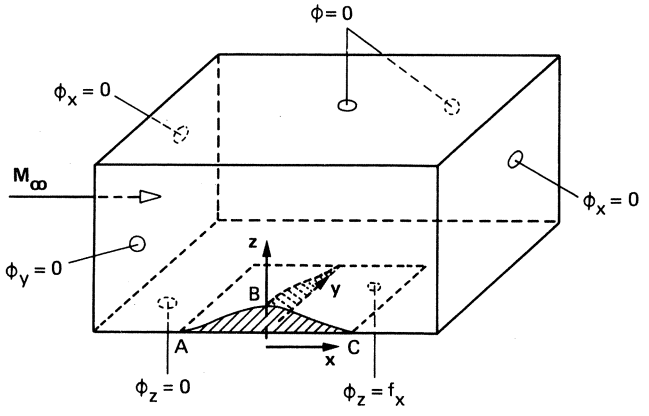


Fig. 6 Description of testproblem: "Windtunnel with a bump on the bottom"

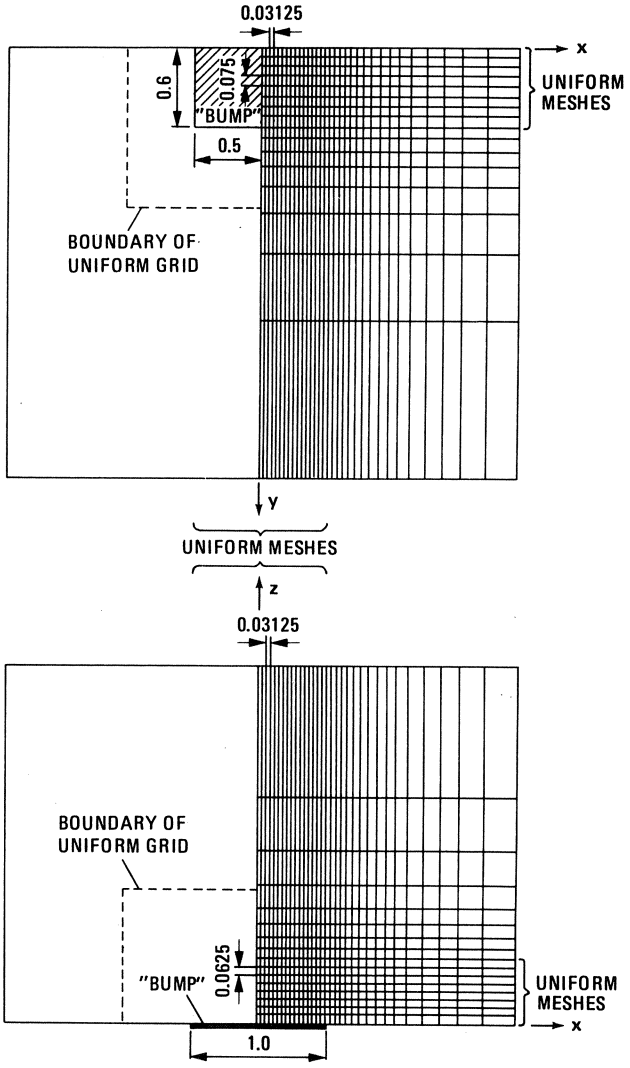


Fig. 8 The 64*16*16 stretched grid

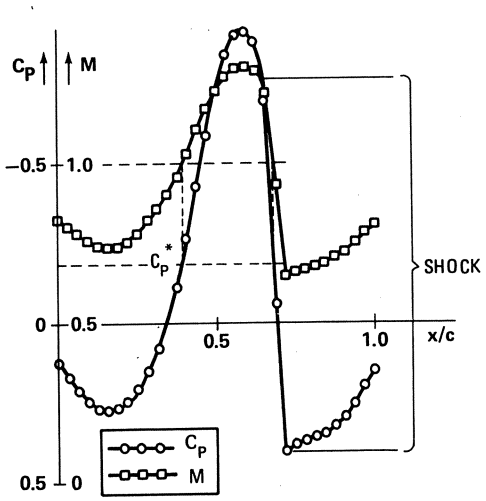


Fig. 7 Pressure coefficient and Machnumber distribution for M_∞=0.9 along the line ABC (see Fig. 6).

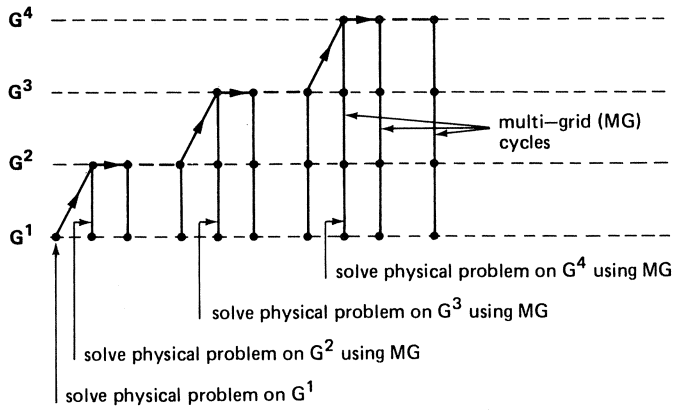


Fig. 9 Schematic of full multi-grid method

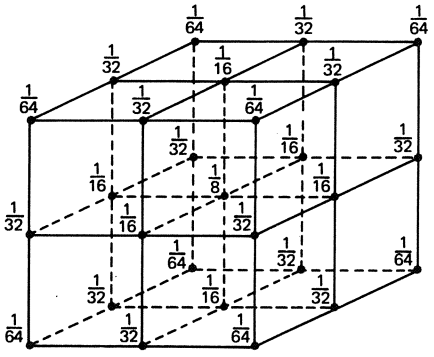


Fig. 10 Schematic of smoothing operator for W_K^{K-1}, I_K^{K-1}.

On the grid G^K the "finite-volume" operator is L^K. This operator generally requires the contravariant metric tensor g^{ij} and the Jacobian J. The required geometric data on G^K can be determined numerically in two ways. One way is to determine them directly from the coordinates of the G^K grid points. The second way is by restriction (injection or smoothing) of the available geometric data (g^{ij},J) on G^{K+1}. As the second way requires far more storage space and is less accessible to recalculation of geometric data, the first way seems preferable for complicated three-dimensional problems. In this paper, therefore all geometric data in L^K are determined directly from the coordinates of the G^K grid points. Hence, L^K can be and is taken identical on each grid G^K.

With respect to fixed multi-grid strategies the relative merits of V- and W-cycles will be investigated. The choice of V-cycle will be limited to the N-level class (N=4,3) defined by, compare Fig. 2,

$$V_N[m,1^1,n]: - m^k = m, \quad n^k = n \quad \text{for} \quad K = 2 \ldots N-1$$

$$- \text{if } m \geq n \text{ then } m^N = m, \quad n^N = 0$$

$$\text{else } m^N = 0, \quad n^N = n.$$

The choice of W-cycles will be limited to the 4-level recursive class $W_4[1^1,n]$ depicted in Fig. 11.

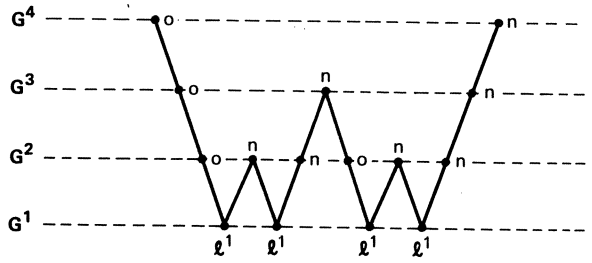


Fig. 11 Schematic of the four level recursive W-cycle class W_4 [ℓ^1, n]

In the testproblem a uniform computational space ξ, η, ζ is introduced through the simple stretching functions

$$x = x(\xi), \quad y = y(\eta), \quad z = z(\zeta). \tag{38}$$

The transonic small-disturbance equation (37) then transforms identically into

$$\xi_x\eta_y\zeta_z\{[(1-M_\infty^2)\frac{\zeta_x}{\eta_y\zeta_z}\varphi_\xi + \tfrac{1}{2}(\gamma+1)M_\infty^2\frac{\xi_x^2}{\eta_y\zeta_z}\varphi_\xi^2]_\xi + \tag{39}$$
$$+ [\frac{\eta_y}{\xi_x\zeta_z}\varphi_\eta]_\eta + [\frac{\zeta_z}{\xi_x\eta_y}\varphi_\zeta]_\zeta\} = 0.$$

In all experiments of which convergence histories are shown, the mean (l_1) norm (MEANRES) will be plotted against so-called equivalent work-units (WU). One work-unit equals the amount of work done for one relaxation sweep on the finest grid. In the calculation of the number of work-units needed for one multi-grid cycle, the work invested in restriction and prolongation is neglected. For example, the W-cycle strategy $W_4(4,2)$ requires $2 + \frac{4}{8} + \frac{8}{64} + \frac{16}{512} = 2{,}65625$ work-units per cycle. Because the plots enable the comparison of "mathematical convergence rates" only, it must be realized that the best convergence rate does not automatically correspond to the fastest computer runtime. Computer runtimes are the product of the balance that exists between good mathematical convergence, the time needed for restriction and prolongation and the magnitude of the work-unit which is determined by the choice of relaxation algorithm. It is emphasized here, that the work-unit of the SLOR algorithm and the ILU/SIP algorithm are different. In a few cases, also plots of the number of

the number of supersonic points (NRSUP) as a function of the work-unit will be presented.

Performance of MG (non-linear FAS) - ILU/SIP

In the experiments that will be presented both the residuals and the dependent variables will be smoothed in the restrictions. Preliminary experiments have shown that this choice pays off when the ILU/SIP algorithm is used within the multi-grid method. For the same reason, unless stated otherwise, cubic interpolation will be used in the prolongations. In all transonic runs the gradual build-up of the supersonic zone is ensured as much as possible by applying the full multi-grid method (Fig. 9). Three versions of the ILU/SIP(α)-algorithm will be compared, viz. ILU(α=0), ILU/SIP(.7) and SIP(α=1). The choice α = .7 is reasonably optimal [31,32].

Elliptic testproblem

First the performance of the three different ILU/SIP(α) versions within the multi-grid method will be demonstrated for the case $M_\infty = 0$, corresponding to the elliptic testproblem, equation (29), for a=b=c=1. The resulting convergence histories on the uniform 64*16*16 grid are shown in Fig. 12 for the W4[3,1]-strategy. Obviously the ILU/SIP(.7) algorithm leads to the fastest convergence with an reduction-factor per work unit $\lambda \cong .21$. In view of the results in table 1, it is a bit unexpected that ILU/SIP(.7) leads to slightly better results than ILU for a purely elliptic problem. The rather striking slow-down of the convergence rate in case of the SIP-algorithm is caused by slow damping of the boundary-conditions at the far-field boundaries. This phenomenon has also been reported by Schneider and Aziz [31,32].
The resulting convergence histories on the stretched 64*16*16 grid are shown in Fig. 13. Again, the ILU/SIP(.7)-algorithm converges fastest with an initial reduction-factor per work unit $\lambda \cong .30$. Use of the SIP-algorithm, however, has caused instability (due to the unstable relaxation of boundary-conditions). The curves for ILU and ILU/SIP(.7) show a distinct kink after which the convergence rate slows down considerably.

A close examination of the numerics has revealed that the kink in the MEANSRES-plot corresponds to the shift of the maximum residual from a position close to the "bump (airfoil) on the bottom of the wind tunnel"

(Fig. 6) to a position close to the far field boundary of the computational domain. This has led to the following explanation. Near the "airfoil" the grid is uniform and has modest mesh ratios (Fig. 8), whence the ILU/SIP algorithm is able to smooth the local errors efficiently. At the far-field boundary of the computational domain the grid is highly stretched and exhibits some rather nasty mesh ratios (Fig. 8). As can be learned from a local mode analysis, this causes the ILU/SIP algorithm to be far less efficient in the smoothing of local errors. So, the convergence at the far-field boundary is lagging behind and this shows up eventually in a dominating maximum error. Hence, the multi-grid process is ultimately determined by the relatively inefficient ILU/SIP-damping near this far-field boundary. On more regular grids than in the present test problem, the same phenomenon will show up as a more gradual stalling of the rate of convergence. This has been observed e.g. by Raj [15].

A way to overcome this effect, at least partially, will be briefly discussed hereafter.

Local Richardson extrapolation

In mono-SLOR algorithms, a well known technique to accelerate convergence is the so-called (global) Richardson extrapolation [33]. Here the same idea is applied to the afore-mentioned local errors near the far-field boundary of the computational domain, which are lagging behind in convergence rate due to the fact that the highly stretched grid exhibits some nasty mesh ratios (Fig. 8), causing less effective damping of local error modes. Figure 13 shows a preliminary result if this technique is used as the basis of a fixed extrapolation strategy.

Transonic testproblem

The numerical computations for the transonic testproblem have mainly been performed using the ILU/SIP(α)-algorithm without extra stabilizing terms in the supersonic zone, equation (35). Only for the case $M_\infty = 0.90$ on the stretched 96*24*16 grid was it necessary to add extra stabilizing terms in the supersonic zone, equation (36).

First the robustness of the ILU/SIP(.7)-algorithm within the multi-grid method will be demonstrated on the stretched 64*16*16 grid at $M_\infty = 0.90$. The results for the W4[9,3]-strategy and two different

V4[0,9,3]-strategies are shown in Fig. 14. In one V-cycle strategy, pure injection in the restriction and tri-linear interpolation in the prolongation has been used. This choice was motivated by the fact that this combination proved to be unreliable in the MG-SLOR experiment, compare Fig. 15. The other V-cycle strategy uses smoothing and cubic interpolation as in all W-cycle strategies. The figure shows reliable convergence for all three strategies. Apparently, the unreliable convergence of MG-SLOR experiments with pure injection and tri-linear interpolation is not encountered here. The figure also shows the tremendous effect of smoothing (residuals as well as dependent variables) and cubic interpolation. It can also be observed that the V4[0,9,3]-strategy and the W4[9,3]-strategy perform almost equally well, beit that the W-cycle is slightly better initially.

A comparison of ILU(α=0), ILU/SIP(.7) and SIP(α=1) on the stretched 64*16*16 grid at M_∞ = 0.95, employing the W4[6,2]-strategy is shown in Fig. 16. It appears that the SIP-algorithm leads to unstable multi-grid performance, compare also Fig. 13, due to the unstable relaxation of the boundary conditions at the far-field boundaries. Furthermore, ILU/SIP(.7) leads to much faster convergence than ILU(α=0). The reason is probably that ILU/SIP(.7) contains less $\overleftarrow{\delta}_t \phi^{n+1}$ in the modified equation (22), see also under "Hyperbolic testproblem".

Fig. 17 shows an application of the local Richardson extrapolation technique for the highly transonic case M_∞ = 0.95 on the stretched 64*16*16 grid. The algorithm used within the multi-grid method is ILU/SIP(.7); the strategy is W4[9,3]. It can be observed that, after the Richardson extrapolation is switched on, the convergence speed becomes indeed comparable to the convergence speed before the kink occurs. Thus, at the cost of only a few extra work units highly converged results on stretched grids can be obtained within a reasonable number of work units.

Finally, the ILU/SIP algorithm will be tested within the multi-grid method on the much finer 96*24*16 stretched grid. First, the case M_∞ = .95 will be considered. Fig. 18 shows that in this case the MG-ILU/SIP(.7) method leads to a limit cycle after one order of magnitude recution of the mean residual. Apparently, the formally unstable character of the modified equation (22), which contains a non-vanishing $\overleftarrow{\delta}_t \phi^{n+1}$-term, manifests itself (see under "Hyperbolic testproblem"). Therefore, the ILU/SIP(.7)-algorithm has been used in the stabilized version, equation (36), which

adds a stabilizing $\overleftarrow{\partial}_x\overleftarrow{\partial}_t\phi^{n+1}$-term to the modified equation (22) in the hyperbolic region. Fig. 18 shows that this stabilized method indeed converges, although the convergence rate is rather slow. The fastest convergence is obtained if the $\overleftarrow{\partial}_t\phi^{n+1}$-term is eliminated from the modified equation by using ILU/SIP(1.) in the hyperbolic region, thus eliminating the formal instability of the modified equation. Moreover, the added $\overleftarrow{\partial}_x\overleftarrow{\partial}_t\phi^{n+1}$-term has to be kept as small as possible by adding a $\varepsilon\overleftarrow{\partial}_x\overleftarrow{\partial}_t\phi^{n+1}$-term with $\varepsilon < 1$ (usually $\varepsilon = .4$ is taken). This way, the algorithm is unconditionally stable and fast and reliable convergence is obtained (Fig. 18). The above ILU/SIP(.7)-version, with ILU/SIP(1.) plus a stabilizing $\varepsilon\overleftarrow{\partial}_x\overleftarrow{\partial}_t\phi^{n+1}$-term in the hyperbolic region, will now be compared to the SLOR-algorithm within the multi-grid method. Fig. 19 (M_∞=.90) shows that the initial reduction rate ($\lambda\cong.74$) is indeed much better than the one for SLOR ($\lambda\cong.84$). Two orders of magnitude reduction in the error of the residual are already achieved at 14.5 work units, where SLOR requires 7.5 work units more. Fig. 19 also shows that the Richardson extrapolation is an effective tool to obtain fast convergence after the "kink" has manifested itself ($\lambda\cong.82$).

In Fig. 20 the more severe test case $M_\infty = .95$ is shown. In this case, the initial convergence rates of MG-SLOR and MG-ILU/SIP are almost equal ($\lambda\cong.81$). Two orders of magnitude reduction in the residual are obtained at approximately 30 work units. The comparison to SLOR on four successive grids shows that this mono-grid method has not achieved the two orders of magnitude reduction in the error level even after 100 work units. In Fig. 21 it is shown that the build-up of the number of supersonic points is better for MG-ILU/SIP(.7) than for MG-SLOR. The mono-grid SLOR method is significantly worse than the multi-grid methods. A crossplot of Fig. 20 and Fig. 21 shows the above two effects in an even more illuminating way. At about 1.5 orders of magnitudc reduction in the residual, MG-ILU/SIP(.7) has already nearly reached the final number of supersonic points, while MG-SLOR is still about 1 % away from this final value. The mono-grid SLOR method is, however, still far from the final value. If the number of supersonic points (development of the supersonic zone) is interpreted as a measure for the "quality of the solution", it is obvious from Fig. 22 that multi-grid methods (MG-SLOR and MG-ILU/SIP provide (at a certain error level) solutions of far better quality than the corresponding mono-grid methods. This is explained by a more efficient

approximation of the long-wave contents of the solution. Additionally, the MG-ILU/SIP method provides a solution which is better than the one provided by the MG-SLOR method.

5. CONCLUDING REMARKS

Two relaxation-algorithms, viz. SLOR and a mixed Incomplete Lower Upper decomposition/Strongly Implicit Procedure (ILU/SIP), have been investigated for use within the non-linear FAS multi-grid method in transonic applications. The well-understood SLOR-algorithm was used primarily for reasons of comparison.

The main conclusions of the research presented can be summarized as follows:

- The combination of pure injection in the restrictions and tri-linear interpolation in the prolongations is far from being optimal. Smoothing of the residuals in the restrictions combined with cubic interpolation in the prolongations, contributes the most to the improvement of the (mathematical) convergence rates of an otherwise fixed multi-grid strategy.
- Both V-cycle and W-cycle fixed strategies can lead to reliable multi-grid convergence. However, the prospects of fixed W-cycle strategies are better from a theoretical viewpoint.
- For transonic applications, and even for subsonic (purely elliptic) applications, the ILU/SIP(.7)-algorithm performs better than ILU. The use of SIP within the multi-grid method can easily lead to divergence.
- ILU/SIP is a serious candidate for the error-smoothing algorithm within the multi-grid method in transonic applications. The algorithm is unconditionally stable in supersonic (hyperbolic) regions of the flow and is a more efficient smoothing algorithm than SLOR. Its full implicitness and insensitivity, but also the absence of a preferred sweep-direction in the coding, are especially of value if complicated configurations involving strongly varying local flow directions and highly stretched grids are involved (e.g. air intakes).
- At a certain (specified) reduction of the error level, MG-SLOR as well as MG-ILU/SIP provide solutions of better quality than the corresponding

mono-grid algorithms. This can probably be used to advantage by specifying a lower convergence level.

- An explanation has been provided for the deterioration of the initial convergence rate of the multi-grid method on highly stretched grids. This stresses the requirement of smoothly stretched grids, not only from a viewpoint of approximation accuracy, but also of solution efficiency.
- A local form of the well-known Richardson extrapolation has been put forward as a possible means to partically overcome the deterioration of the initial multi-grid convergence rate on highly stretched grids. A more consistent way to avoid deterioration of the convergence rate is possibly the use of a grid which is less efficient in the number of computational points, but which has a smoother stretching.

6. ACKNOWLEDGEMENT

The authors wish to express their thanks to their colleague Dr. H. Schippers for some very helpful discussions on the favourable properties of the W-cycle strategy and his support in summarizing some important theoretical results of Hackbush.

7. REFERENCES

[1] BALLHAUS, W.F., A. JAMESON & J. ALBERT, *Implicit approximate-factorization schemes for the efficient solution of steady transonic flow problems*. Paper presented at the AIAA Computational Fluid Dynamics Conference, Albuquerque, New Mexico, June 1977.

[2] BAKER, T.J. & C.R. FORSEY, *A fast algorithm for the calculation of transonic flow over wing/body combinations*. AIAA Paper 81-1015, 1981.

[3] BENEK, J., J. STEINHOFF & A. JAMESON, *Application of approximate factorization to three-dimensional transonic flow calculations*. AIAA Paper 81-1026-CP, 1981.

[4] HOLST, T., *Numerical solution of transonic wing flow fields*. AIAA Paper 82-0105, 1982.

[5] SANKAR, N.L., J. MALONE & Y. TASSA, *A strongly implicit procedure for steady three-dimensional transonic potential flows.* AIAA Paper 81-0385, 1981.

[6] BRANDT, A., *Multi-level adaptive solutions to boundary-value problems.* Mathematics of Computation, Volume 31, nr. 138, April 1977.

[7] BRANDT, A., *Multi-level adaptive techniques (MLAT) for singular-perturbation problems.* ICASE Report Number 78-18, October 12, 1978.

[8] SOUTH, J.C. & A. BRANDT, *Applications of a multi-level grid method to transonic flow calculations.* ICASE Report Number 76-8, March 19, 1976.

[9] JAMESON, A., *Acceleration of transonic potential flow calculations on arbitrary meshes by the multiple grid method.* AIAA Paper 79-1458 CP, 1979.

[10] McCARTHY, D.R. & R.A. REYHNER, *A multi-grid code for three-dimensional transonic potential flow about axisymmetric inlets at angle of attack.* AIAA Paper 80-1365, 1980.

[11] SHIMILOVICH, D.A. CAUGHEY, *Application of the multi-grid method to calculations of transonic potential flow about wing-fuselage combinations.* NASA Conference Publication 2202, October 1981.

[12] BROWN, J.J., *A multigrid mesh-embedding technique for three-dimensional transonic potential flow analysis.* NASA Conference Publication 2202, October 1981.

[13] BOERSTOEL, J.W., *A multigrid algorithm for steady transonic potential flows around aerofoils using Newton iteration.* NASA Conference Publication 2202, October 1981.

[14] SANKAR, N.L., *A multigrid strongly implicit procedure for two-dimensional transonic potential flow problems.* AIAA Paper 82-0931, 1982.

[15] RAJ, P., *A multigrid method for transonic wing analysis and design.* AIAA Paper 83-0262, 1983.

[16] CAUGHEY, D.A., *Multi-grid calculation of three-dimensional transonic potential flows.* AIAA Paper 83-0374, 1983.

[17] BOERSTOEL, J.W. & A. KASSIES, *Integrating multi-grid relaxation into a robust fast-solver for transonic potential flow around lifting aerofoils.* AIAA Paper 83-1885 CP, 1983.

[18] STONE, H.L., *Iterative solution of implicit approximations of multi-dimensional partial difference equations.* SIAM J. Numer. Anal., Volume 5, Number 3, pp. 530-558, 1968.

[19] MEIJERINK, J.A. & H.A. VAN DER VORST, *An iterative solution method for linear systems of which the coefficient matrix is a symmetric M-matrix.* Math. Comp., Volume 31, Number 137, pp. 148-162, 1977.

[20] WESSELING, P. & P. SONNEVELD, *Numerical experiments with a multiple grid and a preconditioned Lanczos method.* Approximation Methods for Navier-Stokes Problems, Proceedings, Paderborn 1979, R. Rautman, ed., Lecture Notes in Math. 771, Springer-Verlag, Berlin etc., pp. 534-562, 1980.

[21] WESSELING, P., *A Robust and Efficient Multigrid Method.* Lecture Notes in Mathematics 960, Multi-grid Methods, Proceedings, Köln-Porz, 1981. Edited by W. Hackbush and U. Trottenberg, Springer-Verlag.

[22] WEES, A.J. VAN DER, J. VAN DER VOOREN & J.H. MEELKER, *Robust calculation of 3D transonic potential flow based on the non-linear FAS multi-grid method and incomplete LU decomposition,* AIAA Paper 83-1950, 1983.

[23] HACKBUSH, W., *Multi-grid convergence theory.* Lecture Notes in Mathematics 960, Multigrid Methods, pp. 177-219, Proceedings, Köln-Porz, 1981. Edited by W. Hackbush and U. Trottenberg, Springer-Verlag.

[24] BRAESS, D. & W. HACKBUSH, *A new convergence proof for the multigrid method including the V-cycle,* SIAM J. Numer Anal., Vol. 20, No. 5, October 1983.

[25] SCHAFFER, J., *High order multi-grid methods to solve the Poisson equation.* NASA Conference Publication 2202, October 1981.

[26] SCHIPPERS, H., *Application of multigrid methods for integral equations to two problems from fluid dynamics.* NASA Conference Publication 2202, October 1981.

[27] STÜBEN, K. & U. TROTTENBERG, *Multigrid methods: fundamental algorithms, model problem analysis and applications.* Lecture Notes in Mathematics 960, Multigrid Methods, pp. 1-176, Proceedings Köln-Porz, 1981. Edited by W. Hackbush and U. Trottenberg. Springer-Verlag.

[28] KETTLER, R., *Analysis and comparison of relaxation schemes in robust multigrid and preconditioned conjugate gradient methods.* Lecture Notes in Mathematics 960, Multigrid Methods, pp. 502-534, Proceedings Köln-Porz, 1981. Edited by W. Hackbush and U. Trottenberg, Springer-Verlag.

[29] VORST, VAN DER H.A., *A vectorizable variant of some ICCG methods.* SIAM J. Sci. Stat. Comput., Vol. 3, No. 30, September 1982.

[30] VOOREN, J. VAN DER, G.H. HUIZING & A. VAN ESSEN, *A finite difference method for the calculation of transonic flow about a wing, based on small perturbation theory,* NLR TR 81031 L, 1981.

[31] ZEDAN, M. & G.E. SCHNEIDER, *3-D Modified Strongly Implicit Procedure for Finite Difference Heat Conduction Modelling,* AIAA Paper 81-1136, 1981.

[32] SCHNEIDER, G.E. & M. ZEDAN, *Investigation into the Stability Characteristics of Modified Strongly Implicit Procedures,* ASME Paper 82-HT-023, 1982.

[33] CAUGHEY, D.A. & A. JAMESON, *Accelerated Iterative Calculation of Transonic Nacelle Flowfields,* AIAA paper 76-100, 1976.

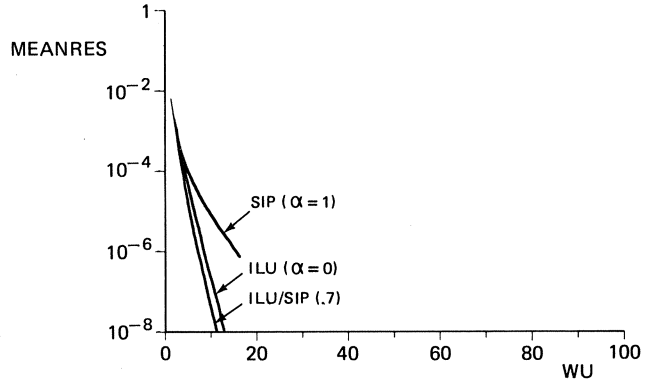


Fig. 12 Performance of ILU/SIP(α) within the multi-grid method on the uniform 64*16*16 grid for the case $M_\infty = 0$, using a W4[3,1]-strategy

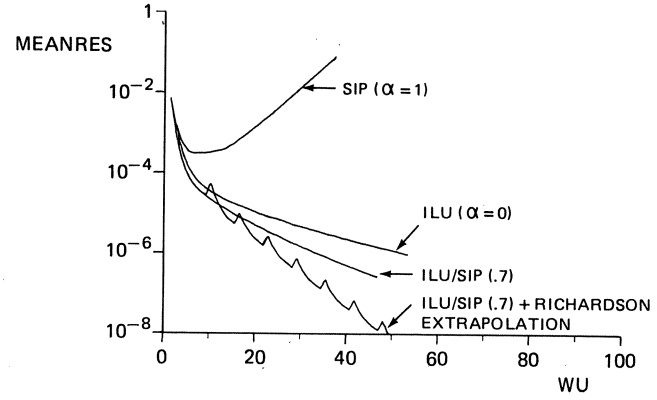


Fig. 13 Performance of ILU/SIP(α) within the multi-grid method on the stretched 64*16*16 grid for the case $M_\infty = 0$, using a W4[3,1]-strategy

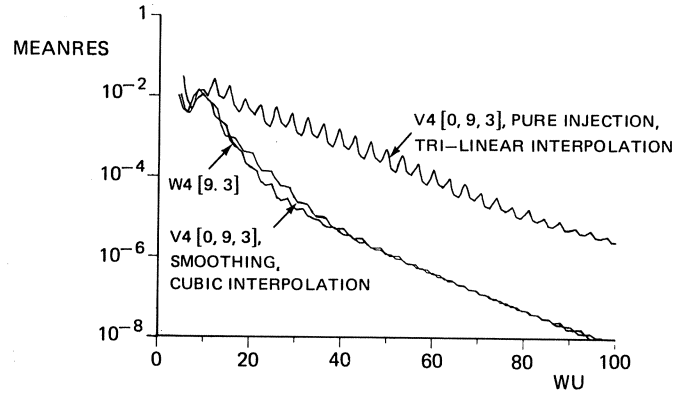


Fig. 14 Reliable convergence of ILU/SIP(.7) within the multi-grid method on the stretched 64*16*16 grid for the case $M_\infty = 0.90$, using the W4[9,3]-strategy and two different V4[0,9,3]-strategies

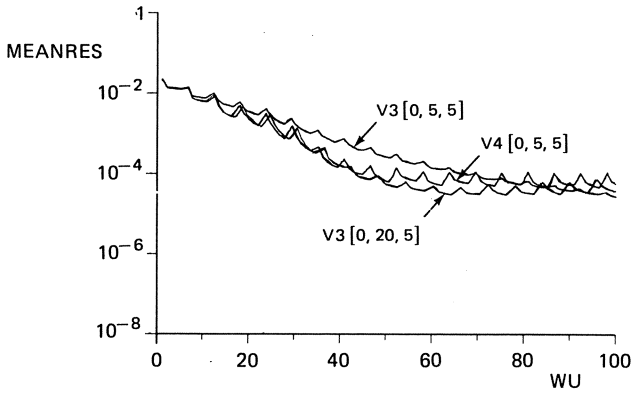


Fig. 15 Unpredictable V-cycle convergence for SLOR on the stretched 64*16*16 grid at M_∞ = 0.90 employing pure injection and tri-linear interpolation

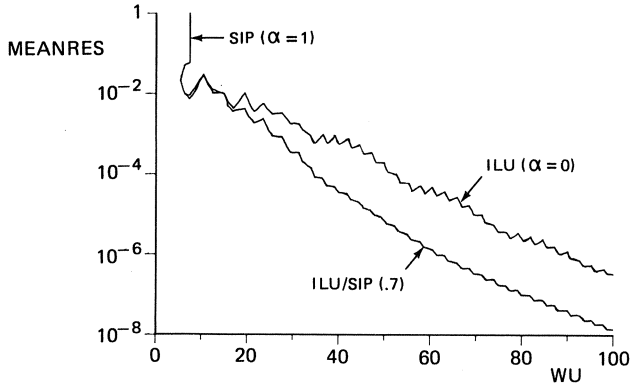


Fig. 16 Comparison of ILU ($\alpha = 0$), ILU/SIP(.7) and SIP ($\alpha = 1$) within the multi-grid method on the stretched 64*16*16 grid at $M_\infty = 0.95$

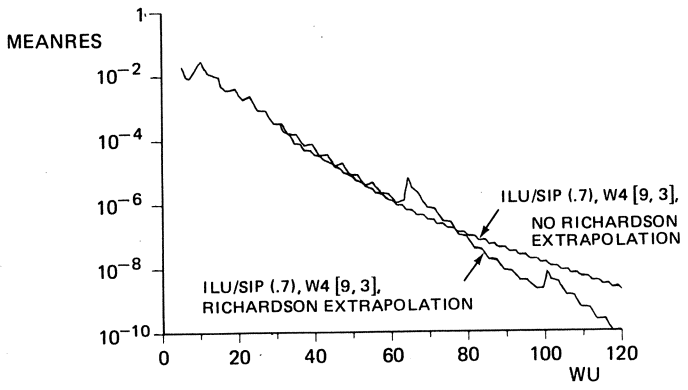


Fig. 17 The effect of local Richardson extrapolation for the case $M_\infty = 0.95$ on the stretched 64*16*16 grid

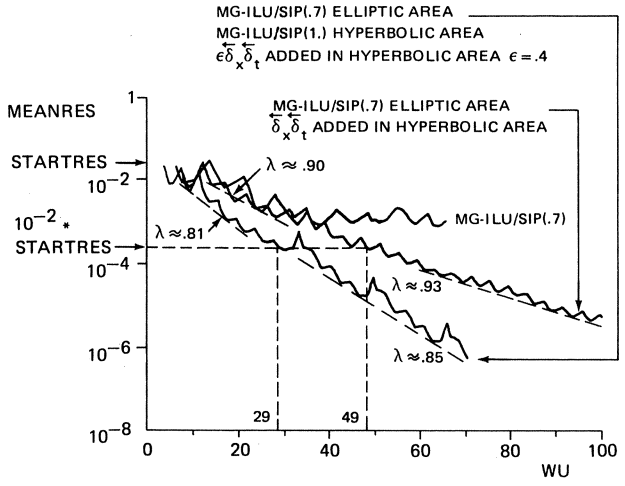


Fig. 18 Performance of several versions of the ILU/SIP(.7) algorithm within the multi-grid method on the stretched 96*24*16 grid for the case $M_\infty = .95$

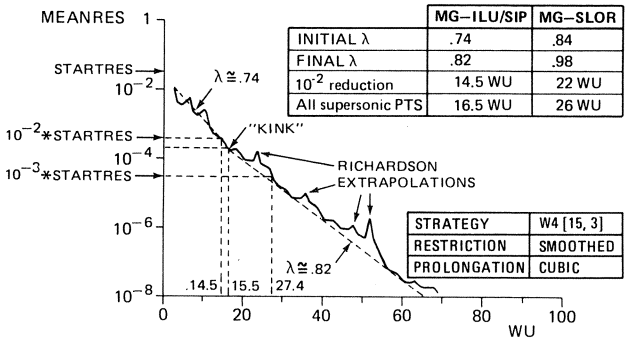


Fig. 19 Performance of ILU/SIP (.7) within the multi-grid method on the stretched 96*24*16 grid for the case M_∞ = .90 (2% supersonic points)

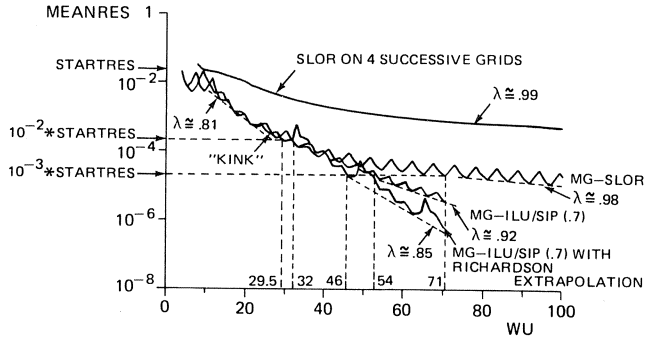


Fig. 20 Performance of ILU/SIP (.7) and SLOR within the multi-grid method on the stretched 96*24*16 grid for the case M_∞ = .95 (11% supersonic points)

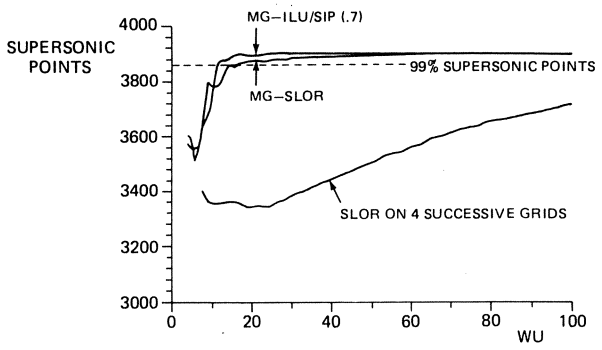


Fig. 21 Comparison of build up of number of supersonic points on 96*24*16 grid for the case M_∞ = .95 for several algorithms

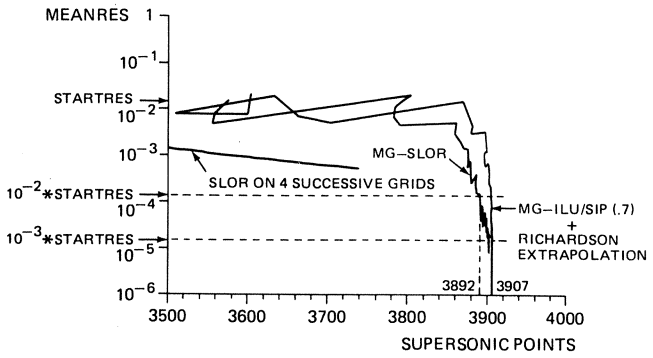


Fig. 22 Crossplot of figure 20 and figure 21, showing the build up of the number of supersonic points at a certain error level for several algorithms

TRANSPORT OF WASTE HEAT OR POLLUTANTS IN THE SUBSOIL

W. ZIJL

1. INTRODUCTION

The last decade has witnessed rapid progress in the development of computer-based models for the simulation of subsurface fluid motion. Well-known are reservoir simulation models for economically predicting the response of an oil or gas producing reservoir to a variety of operating conditions or development plans.

The subsurface environment is increasingly involved in water and energy supply, and also in waste disposal problems.

In many of these cases, numerical simulation is indispensable to obtain reasonably quantitive insight in economical and environmental effects, and to weigh various alternatives against each other.

The recent advent of supercomputers and attached array processors now makes possible, technically and economically, advanced three-dimensional simulations of subsurface transport processes.

In this way, transport of pollutants and waste heat in a subsurface flow system can be predicted, being an important tool for environmental impact assessment and licensing purposes (1).

The classical theory of porous media is devoted to the description of flow and transport through soils consisting of sand, clay, peat, etc. Typical applications are in the fields of petroleum reservoir engineering and groundwater hydrology. In these fields, a porous medium is defined as a solid structure containing a multiply connected void space through which a fluid can flow; see Fig. 1. The constituents of the solid structure (i.e. the sand, clay, peat) may be distributed randomly or in a regular way. The fluids (oil, water, gas) spread through the void space, thereby causing a pressure gradient which acts as a force on the constituents of the solid structure. For a good understanding of transport processes in

porous media, it might be helpful to consider the many non-classical examples of fluid motion conforming to this definition. For instance, agricultural products are stored to be available over longer periods, and the container with these products can be considered as a porous medium for the cooling air flow; see Fig. 1. Other examples include filtration, chemical reactions using solid catalysts, adsorption, and mass transfer in packed columns. Finally, flow in the core structure of nuclear power reactors and their components like shield rod arrays, heat exchangers, and steam generators can be considered as flow in a porous medium (2), (3). Non-classical porous media are often refered to as generalized porous media.

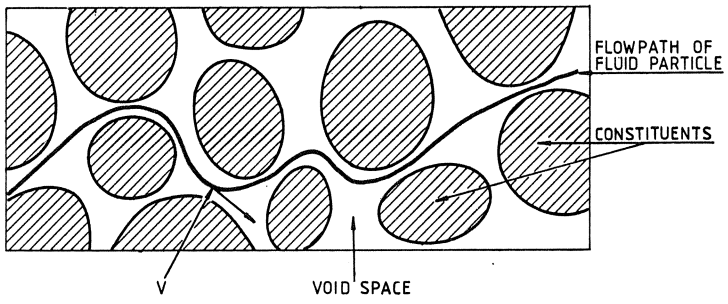


Fig. 1. Fluid motion in void space of porous medium. The constituents may be sand, clay, peat, etc. (classical porous medium) or potatoes, eggs, tubes etc. (generalized porous medium).

Fluid motions in (generalized) porous media are governed by the fundamental laws based on conservations of mass, momentum and energy. However, from a practical standpoint, it is hopeless to try to apply these basic laws directly to the problems of (generalized) porous media. Instead a semiempirical approach is used where the concept of a fluid-structure continuum is employed. An important parameter in a porous medium is the porosity defined as a fraction of the control volume not occupied by the solid matrix, or solid structure.

We can see that, if the control volume is of the size of a pore, the porosity would be either one or zero. As we increase the size of the control volume, the porosity value will fluctuate before reaching a representative value. The value of porosity associated with a point P is the representative value for a control volume of sufficiently large size containing P. Other physical properties are defined as a mean value at a

point P in the porous medium for the same control volume or representative elementary volume. This is the so-called continuum approximation, where the actual porous medium is replaced by a fictitious continuum.

It is simple to prove that the surface porosity, that is the fraction of a sufficiently large control surface not covered by the solid structure, is equal to the earlier defined (volume) porosity (4).

The resulting equations for the underground motion of oil, gas and water are conventionally solved with finite difference methods using one-, two-, or three-dimensional grids, and vector computers or attached array processors are used to solve the resulting system of large matrix equations! To establish the advantage of vector computers (or super computers) over conventional scalar computers, the program SWIP has been run on two scalar computers and on the vector computers Cray-1S and CYBER-205, simulating a two-dimensional, axi-symmetric subsurface heat storage cycle. While no significant difference was found between performance of the two super computers, it appeared that their application is already attractive if problem size exceeds some 800 grid blocks. In the near future this turn-over point will be lower as a result of software written specially for vector-computers, such as the program DARTEX.

In this paper much emphasis will be laid upon appropriate mathematical modeling of these above-described geohydrologic problems, and especially the problems encountered when considering anisotropy are discussed in more than conventional detail, since these problems have serious consequences for the numerical analysis.

2. BASIC EQUATIONS

2.1. The fluid-structure continuum

The basic equations to be solved are well-established; they are the classical partial differential equations expressing conservation of mass, linear momentum and energy for a Newtonian fluid, the so-called Navier-Stokes equations (5). To obtain a well-posed partial differential problem, initial and boundary conditions must be prescribed.

The initial condition for the fluid velocity $\underline{v}$ is that $\underline{v}(\underline{r},0)$ must be prescribed at time $t = 0$, and the boundary condition at a boundary completely enclosing the fluid is that $\underline{v}$ must be prescribed for all times $t > 0$. For the (generalized) porous medium under consideration, this means that $\underline{v} = \underline{0}$

must also be prescribed on the individual constituents of the solid structure (e.g., on the grains of sand).

Although the equations and boundary conditions needed for the prediction of the flow pattern in a porous medium are well-established, it is, however, also a well-known fact that the Navier-Stokes equations can hardly be solved. Even for simple geometries, analytical and numerical solutions can only be obtained for relatively low Reynolds numbers only.

Furthermore, the requirement that $\underline{v} = \underline{0}$ on all components of the solid structure is prohibitive even for low Reynolds number flow. Consequently, from a practical standpoint, it is hopeless at this time to try to apply the basic Navier-Stokes equations directly to the problem of flow in porous media.

For that reason, it is necessary to describe the flow distribution in a porous medium approximately by partial differential equations describing a so-called fluid-structure continuum.

Such a continuum approximation is well-known in petroleum reservoir engineering and groundwater hydrology, where the actual porous medium (sand, clay, peat, etc.) is replaced by a fictitious continuum to any point of which we can assign mean variables and parameters which are continuous functions of the space and time co-ordinates. In these classical fields, the equations describing flow are the continuity equation and Darcy's Law.

A complete treatment of the dynamics and statics of fluids in porous media, where most of the problems considered are oriented towards groundwater hydrology is presented in (4). A derivation of the continuum equations for generalized porous media (a tube bundle in heat exchangers) has been presented in (2).

2.2. The continuum equations

Starting from the fundamental conservation equations, or Navier-Stokes equations, the fluid-structure continuum equations are derived and the resulting equations are given by:

$$\frac{\partial}{\partial t}(\rho\phi) + \nabla.(\rho\phi\bar{\underline{v}}) = 0 \quad \text{(continuity eqn.)} \tag{2.1}$$

$$\rho\dot{\underline{v}} = -\nabla\bar{p} + \rho\underline{g} + \rho\underline{Q} \quad \text{(momentum eqn.)} \tag{2.2}$$

(see (2)).

In Eqs. (2.1), (2.1), $\underline{\bar{v}}$ is the mean fluid velocity in the void space between the constituents of the solid structure, $\underline{\dot{\bar{v}}}$ is the mean fluid motional acceleration given by $\underline{\dot{\bar{v}}} = \partial\underline{\bar{v}}/\partial t + \underline{\bar{v}}.\nabla\underline{\bar{v}}$, $\bar{p}$ is the mean fluid pressure, ρ is the fluid density, ϕ is the porosity representing the ratio of the volume occupied by the fluid and the total volume, $\underline{Q}$ represents the frictional force distribution, and $\underline{g}$ is the gravitational acceleration. The force distribution $\underline{Q}$ is an unknown for which an additional expression must be found.

At low Reynolds number flow (based on the hydraulic diameter of the void space), the advective acceleration in Eq. (2.2) may be neglected, and $\underline{Q}$ is linear in $|\underline{\bar{v}}|$. If, in addition, momentum transport may be considered as quasi-steady (this is allowed on time scales where sound propagation is negligible) the momentum equation (2.2) simplifies to Darcy's law:

$$(2.3) \qquad \underline{u} = \phi\underline{\bar{v}} = -\frac{k}{\mu}(\nabla\bar{p} - \rho\underline{g})$$

where $\underline{u}$ is the volumetric flow rate in the space containing both the fluid and the solid structure (in the literature $\underline{u}$ is often denoted as the Darcy velocity), μ is the fluid dynamic viscosity, and k is the permeability depending on the geometrical properties of the solid structure and on the flow direction. In this way, momentum equation (2.2) may be considered as a generalization of Darcy's Law (2.3).

The equations describing motion in the fluid-structure continuum have essentially the same character as the well-known Euler equations describing inviscid fluid dynamics; the only differences are the porosity ϕ representing the ratio of the volume occupied by the fluid and the total volume, and the continously distributed force term $\underline{Q}$ (or permeability k) accounting for the flow resistance of the structure.

The principal difference between the Navier-Stokes equations and the Euler equations is the absence of the second-order viscosity term $\mu\nabla^2\underline{v}$ in the Euler equations. One of the consequences of this absence is a simplification of the boundary conditions. Instead of boundary conditions for the three components of $\underline{v}$ for the Navier-Stokes equations, only the boundary condition for the normal component of the fluid velocity, $\underline{v}.\underline{n} = 0$, holds for the Euler equations. This latter mathematical feature makes Euler-like equations especially well-suited for the description of a

fluid-structure continuum in contrast to Navier-Stokes-like equations, as will be shown in the following discussion.

As an example, let us assume that the porous medium is a container filled with potatoes, and that the fluid is a cooling air stream.

From a physical point-of-view, the boundary conditions in this fluid-structure continuum are that no fluid is flowing out of the impermeable walls of the container, i.e., $\underline{\bar{v}}.\underline{n} = 0$ on the walls. Of course, in a fluid-structure continuum no boundary conditions may be prescribed on the individual constituents of the structure (the potatoes).

A mean volumetric flow rate of fluid is passing across a unit area containing the solid constituents (the potatoes), and the local fluid velocity, as it passes through the clearances between the individual solid constituents, will not be considered. Additional boundary conditions, e.g., for the velocity components parallel to the walls, for the so-called turbulent tangential stress, or for the vorticity, may not be prescribed since, if they are prescribed, there is no reason why the same conditions are not applied on the boundaries of the individual constituents of the solid structure.

In conclusion, from a physical point-of-view the boundary conditions in a fluid-structure continuum approximation should have an Euler-like character, and "improvement" of the momentum equation by adding a second-order term is non-physical.

Of course, this point has great consequences for the numerical approximation method applied to solve the equations.

2.3. The permeability or resistance force

In this section only low Reynolds number flow with quasi-steady momentum transfer will be considered, since this type of flow is commonly encountered in petroleum reservoir engineering and groundwater hydrology. However, many of the conclusions also hold for generalized porous media.

For this type of flow, acceleration of fluid does not play a part in the momentum balance, and the continously distributed vicous drag force is proportional to the mean velocity $\underline{\bar{v}}$:

$$(2.4) \qquad \rho \underline{Q} = - \frac{\mu}{k} \phi \underline{\bar{v}} = - \frac{\mu}{k} \underline{u},$$

where μ is the fluid dynamic viscosity, and k is the permeability of the solid structure. In general, the solid structure is anisotropic, which means that the permeability depends on the flow direction (i.e., $k = k(\underline{\bar{v}})$ with $\underline{\bar{v}} . \partial k / \partial \underline{\bar{v}} = 0$).

For instance, in many cases the subsoil has a structure such that in the horizontal flow direction the permeability is larger than in the vertical flow direction. If the flow is in the vertical direction, the following relationship holds:

$$\rho \underline{Q} = - \frac{\mu}{k_v} \phi \underline{\bar{v}}, \tag{2.5}$$

also, for flow in the horizontal direction the following relationship holds:

$$\rho \underline{Q} = - \frac{\mu}{k_h} \phi \underline{\bar{v}}, \tag{2.6}$$

where $k_v < k_h$, k_v and k_h are constants

One of the many possible combinations of (2.5), (2.6) is:

$$\rho \underline{Q} = - \frac{\mu}{k_h \left(\frac{\bar{v}_h}{|\underline{\bar{v}}|}\right)^2 + k_v \left(\frac{\bar{v}_v}{|\underline{\bar{v}}|}\right)^2} \phi \underline{\bar{v}},$$

or, equivalently,

$$k(\underline{v}) = k_h \left(\frac{\bar{v}_h}{|\underline{\bar{v}}|}\right)^2 + k_v \left(\frac{\bar{v}_v}{|\underline{\bar{v}}|}\right)^2. \tag{2.7}$$

Similarly, also the following choice is possible:

$$\frac{1}{k(\underline{\bar{v}})} = \frac{1}{k_h} \left(\frac{\bar{v}_h}{|\underline{\bar{v}}|}\right)^2 + \frac{1}{k_v} \left(\frac{\bar{v}_v}{|\underline{\bar{v}}|}\right)^2, \tag{2.8}$$

where $\bar{v}_h$ and $\bar{v}_v$ are the horizontal and vertical velocity components.

In fact, there is an infinity of ways in which expressions (2.5) and (2.6) can be satisfied, and only experiments can justify the ultimate choice of a friction model. For tube bundles some experimental work has been performed in this area (6). For an isotropic medium, where $k_h = k_v = k$, expressions (2.7) and (2.8) simplify to $k(\underline{\bar{v}}) = k$ independent of $\underline{\bar{v}}$.

In Eq. (2.4) it is assumed that the frictional force has always a direction apposite to the flow direction. However, in petroleum reservoir engineering and groundwater hydrology it is common practise to assume that relations (2.5) and (2.6) hold simultaneously for the x, y and z-components of the velocity, i.e.:

$$(2.9) \qquad \underline{Q} = (Q_x, Q_y, Q_z) = -\mu\phi\left(\frac{\bar{v}_x}{k_x}, \frac{\bar{v}_y}{k_y}, \frac{\bar{v}_z}{k_z}\right).$$

This latter expression leads to a linear problem and, thus, to numerically simpler formulations (no iterations), but the consequence is that the solid structure feels a force component normal to the direction of flow.

This implies, for instance, that a solid structure (e.g. a tube bundle) dropped in a lake will sink to the bottom along a path not parallel with the direction of gravitational acceleration, but will instead follow a path with lateral displacement (see Fig. 2), which seems to be non-physical.

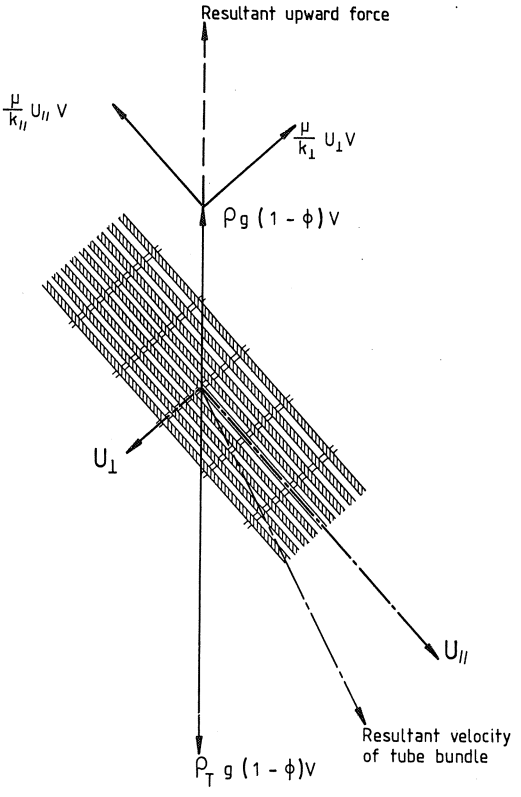


Fig. 2. Tube bundle dropped in a pool of water. According to the tensor model of permeability (2.9), the tube bundle will sink with a lateral velocity component; with the scalar model of permeability (2.7) or (2.8), the tube bundle will sink in the vertical direction.

Also, flow in a pipe filled with an anisotropic porous medium (e.g. a sand-clay mixture) will exert a force on the pipe normal to the flow direction (see Fig. 3), which, again is non-physical.

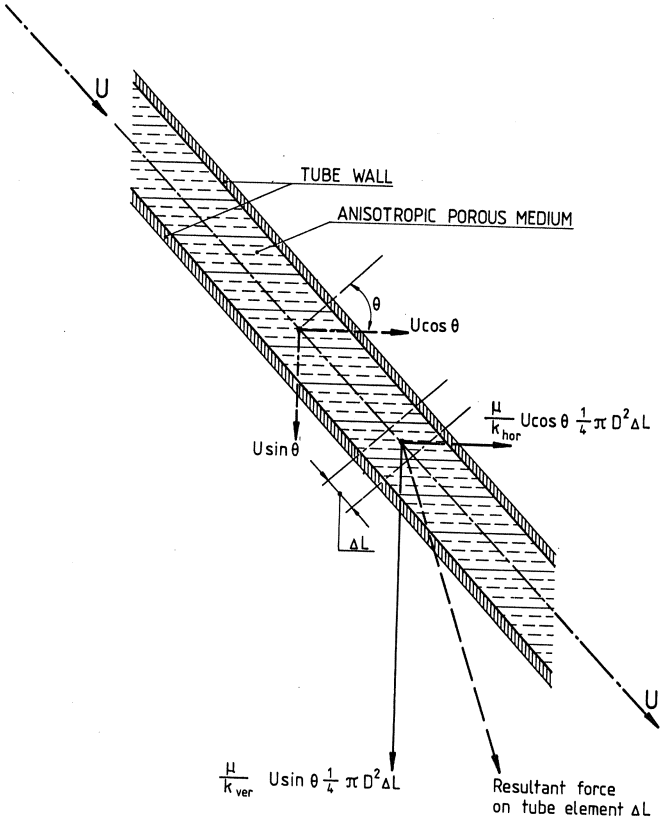


Fig. 3. Pipe filled with anisotropic medium. According to the tensor model of permeability (2.9), the tube will feel a lateral force; with the scalar model of permeability (2.7) or (2.8), there is no lateral force.

Another type of argument against the tensor model (2.9) is that in a tube bundle the permeability has different values of the azimuthal orientation θ of the flow path in radial direction (see Fig. 4), and it is not clear how this can be incorporated in a dyadic which has a maximum of three principal directions.

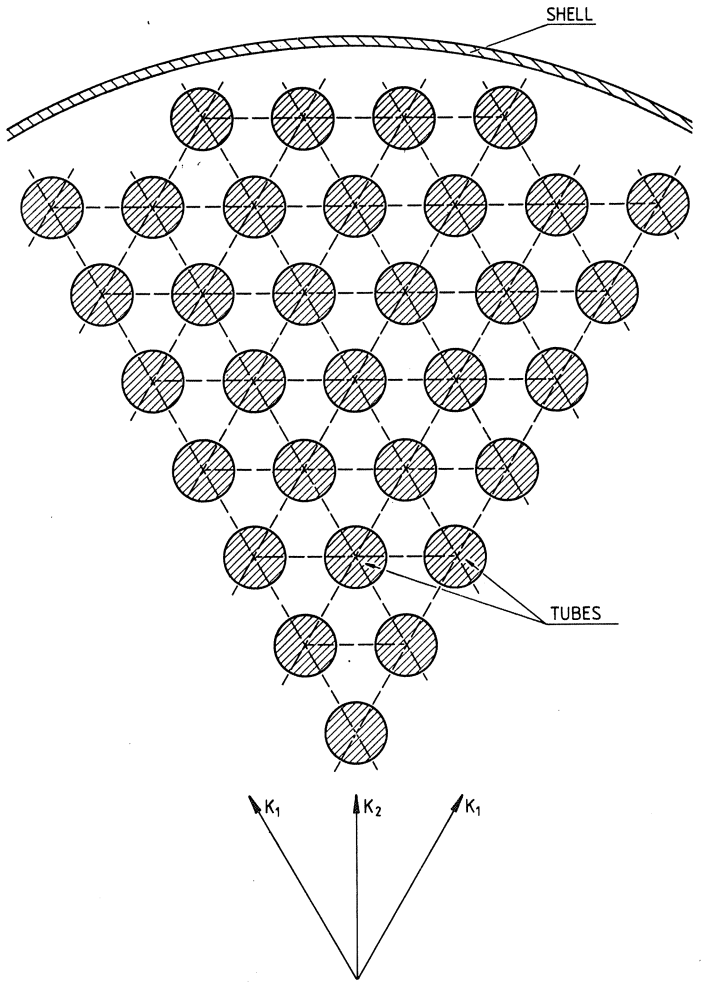


Fig. 4. Arrangement of tubes in a heat exchanger. The permeability in the direction of K_1 is larger than the permeability in the direction of K_2. The tensor model of permeability does not allow for this.

Also, let $\underline{F}(\bar{v})$ be the force on a piece of porous medium caused by the velocity $\bar{v}$. From the Darcy equation (2.3), where k is assumed to have tensor character and k is independent of $\bar{v}$, it follows that $\underline{F}(\bar{v}) = -\underline{F}(-\bar{v})$. In other words, whether one forces (pumps) water to flow from the front or from the end of the piece of porous medium, the drag is the same. From hydrodynamics we know that this is non-physical for higher-Reynolds number flow in a porous medium with non-symmetric constituents; see Fig. 5.

It is interesting to note that this pradox is similar to the Olmstead and Gautesen paradox for Oseen flow (7).

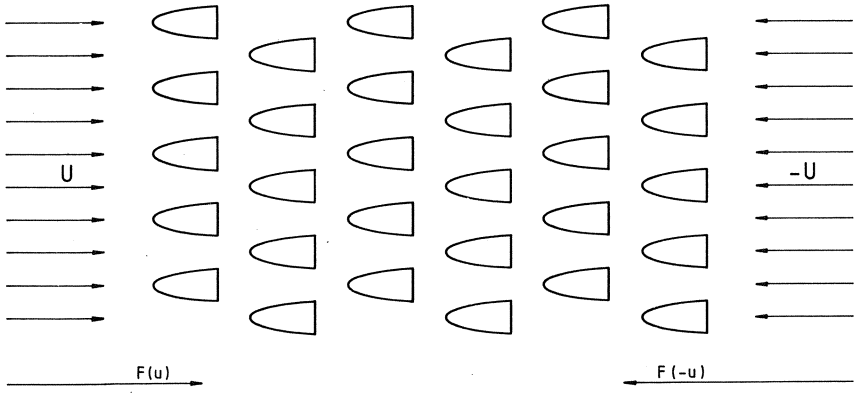


Fig. 5. Arrangement of specially shaped constituents. According to the tensor model of permeability $F(U) = -F(-U)$; the scalar model of permeability allows $F(U) \neq -F(-U)$.

Also arguments based on hydrodynamic theory (lift-theory, Magnus effect) can be applied to show that the existence of a sensible lateral force is very unlikely (18).

In conclusion, even in classical anisotropic porous media, the equations to be solved are always non-linear. Consequently only numerical approximation methods will lead to a solution.

For a discussion of the permeability in non-Darcy flow see (6), (8), (9).

3. NUMERICAL SIMULATION

3.1. Why simulation

Planned interventions in the subsoil, as production of hydrocarbons (oil, gas) and groundwater, heat injection and extraction (heat storage in aquifers), underground coal gassification, geothermal heat production, and the establishment of waste disposal sites, should more and more be judged with respect to their economic and environmental impacts.

As an example, waste disposal sites are a source of slowly slinking

groundwater contamination. Precipitation (rain, snow) causes infiltration of water into the waste disposal site. This water absorps contaminants during percolation through the waste disposal site until it enters a subsoil regional flow system. Finally this contaminated water comes at an other place (e.g. in a water winning region) where its presesence might not be desired; see Fig. 6.

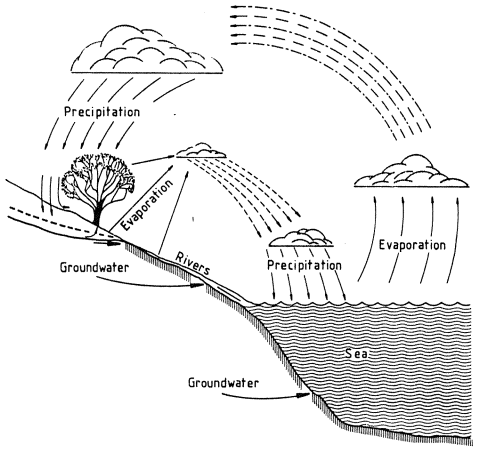


Fig. 6. Hydrological cycle. Groundwater enters the underground by infiltration (e.g. precipitation) and leaves the underground in surface water (e.g. rivers) and water winning wells.

From this example it will be appreciated that execution of a planned intervention without an environmental impact assessment is no longer justified.

The transport processes in the subsoil can be simulated numerically with reasonable accuracy. Simulation in the energy technology (thermal energy storage, geothermal energy) is necessary to predict feasibility and efficiency beforehand, and simulation of transport of pollutants coming from planned waste disposal sites is necessary for licensing procedures.

Environmental impact assessment deals with the prediction of the impact of a planned intervention (e.g. the planned establishment of a waste disposal site) and the prediction of the impact of some reasonable alternatives (e.g. other locations for the waste disposal site). It will be clear that, in the case of waste disposal sites, experiments or demonstration

projects are impossible. Simulation of the transport to be expected is the only possibility to judge the evironmental impact. Even if a waste disposal site is fully contained, e.g. with plastic foil or asphalt, it remains necessary to determine the impact of a hypothetical crack in the containment.

In other fields of technical sciences, especially in the fields of nuclear reator techniques and offshore techniques, the application of numerical simulations (stress analyses) for licensing authorities is already common practise. Also for that reason it may be expected that the importance of numerical simulations in environmental impact assessment studies will increase considerably in the next few years.

3.2. Computer programs based on the pressure and transport velocity representations

Conventional numerical analysis of flow in porous media is based on the pressure as primary variable, i.e., the mass flow rate is eliminated by substitution of (2.3) into (2.1) yielding a diffusion-type equation for the pressure:

$$(3.1)\qquad \frac{\rho\phi}{\kappa}\frac{\partial p}{\partial t} - \nabla.(\frac{\rho}{\mu}k\nabla p) = -\nabla(\frac{\rho}{\mu})^2.(kg\nabla z) - \frac{\rho}{\mu}^2\nabla - (kg\nabla z) - S,$$

where $\kappa = \rho\phi dp/d(\rho\phi)$ is the combined bulk modulus of liquid and porous medium, $\underline{g} = g\nabla z$ is the gravitational acceleration and S is an additional source term in the continuity equation (2.1).

Provided that ρ, μ, κ, k and ϕ are known as a function of space and time, the pressure can be calculated from (3.1). Having obtained the pressure field, the transport velocity field is determined from (2.3) by numerical differentiation of p.

However, numerical differentiation often leads to a degradation of accuracy and, therefore, it is better to avoid it. For that reason approaches where the transport velocity is calculated directly are presented in the literature; see (10), (11), (12).

One possibility is to "differentiate" equation (3.1) to obtain an expression for the mass flow rate $\underline{q} = \rho\phi\underline{\bar{v}}$ (12):

$$(3.2)\qquad \frac{\rho\phi\beta}{\kappa}\frac{\partial \underline{q}}{\partial t} - \nabla^2\underline{q} = \nabla x\underline{\Omega} + \nabla(\frac{\rho\phi}{\kappa})\frac{\partial p}{\partial t} + \frac{\rho\phi}{\kappa}(\frac{\partial\rho}{\partial t}g\nabla z - \frac{\partial\beta}{\partial t}\underline{q}),$$

where the so-called Darcy vorticity $\underline{\Omega}$ is defined as:

$$(3.3) \qquad \underline{\Omega} = \nabla x \underline{q} + \frac{1}{\beta}\nabla x(-\beta \underline{q} + \rho g \nabla z) = \frac{1}{\beta}[\nabla(\rho g) x \nabla z - \nabla \beta x \underline{q}],$$

and where $\beta = \mu/(\rho k)$ is the drag coefficient.

From a physical point-of-view, the following boundary conditions are possible:

i) The normal transport velocity component $\underline{n}.\underline{q}$ is prescribed. For example, at an impermeable base or at a water divide $\underline{n}.\underline{q} = 0$. With the aid of equation (2.3) this condition can be replaced by a boundary condition for the pressure.

ii) The pressure is prescribed. With the aid of equation (2.3) this condition can be replaced by boundary conditions for the tangential transport velocity components.

In order to obtain a well-posed problem equivalent to the system (2.1), (2.3), the following auxiliary boundary conditions must be added to the above-mentioned physical boundary conditions and equation (3.2); see (12).

i) If the normal component $\underline{n}.\underline{q}$ is prescribed, then the tangential components $\underline{n}\ x(\nabla x \underline{q}) = \underline{n} x \underline{\Omega}$, must be prescribed in addition.

ii) If the pressure p or the tangential components $\underline{n}\ x\ \underline{q}$ are prescribed, then the continuity equation $\nabla.\underline{q} = -\partial(\rho\phi)/\partial t$ must be also prescribed.

The computer code SWIP (Survey Waste Injection Program) is based on the pressure representation (3.1). The code has been developed by INTERCOMP Resource Development and Engineering, Inc. Houston (USA) by the direction of the United States Geological Survey, Water Resources Divisions, Denver.

SWIP was originally put together, in part, from petroleum reservoir simulation codes.

In SWIP also the energy and material balance equations are solved. These balance equations are:

(3.4) $\nabla.[\frac{\rho k}{\mu} H(\nabla p-\rho g\nabla z)] + \nabla.(\underline{\underline{\lambda}}.\nabla T) - S_L$

Net energy advection — Conduction — Heat loss to surrounding strata

$- SH - S_H$

Enthalpy in with fluid source S — Energy in without fluid input

$=\frac{\partial}{\partial t}[\phi\rho U+(1-\phi)U_R]$

Accumulation

where H is the fluid enthalpy, $\underline{\underline{\lambda}}$ is the combined thermal conductivity of liquid and porous medium, U is the fluid internal evergy $=H-p/\rho$, and U_R is the internal energy of the solid structure.

(3.5) $\nabla.[\rho C\frac{k}{\mu}(\nabla p-\rho g\nabla z)] + \nabla.(\rho\underline{\underline{D}}.\nabla C) - SC$

Net advection — Diffusion (including micro dispersion) — Sources

$-\lambda_d\phi\rho K_e C = \frac{\partial}{\partial t}(\phi\rho K_e C)$

Reaction/decay — Accumulation

where C is the concentration, $\underline{\underline{D}}$ is the combined diffusion and micro-dispersion coefficient, λ_d is the reaction constant and K_e is the equilibrium adsorption coefficient. Diffusion is the phenomenon that contaminants in a stagnant fluid spread out occupying an ever increasing portion of the flow domain, and micro dispersion is a similar phenomenon for a moving fluid with mean velocity $\underline{\bar{v}}$. From a physical point-of-view micro dispersion is advection on the level of the Navier-Stokes flow with velocity $\underline{v}$ in the void space between the constituents of the solid structure. This micro dispersion should not be confused with macro dispersion, which is advection around pieces of porous medium with a

permeability constrasting with the permeability of the surroundings porous medium. This latter type of dispersion should not be described by the introduction of a (non-physical) dispersion coefficient, but by an accurate representation of the permeability field in equations (3.1) or (3.2).

The computer program DARTEX (DARcy vorTEX) is based on the transport velocity representation (3.2). It has been developed by the TNO Institute of Applied Geoscience (formerly Groundwater Survey TNO).

At present the program is in the phase of being a prototype, but it is planned to transform it into a robust code in 1984.

In DARTEX the flow field $\underline{q}$ is calculated for steady states only (negligible accumulation). In contradistinction with the SWIP code the temperature dependencies of viscosity and density are not accounted for. However, not only the linear tensor model (2.9), but also the non-linear scalar models (2.7), (2.8) are implemented to account for anistropy. From a numerical point-of-view this means that a suitable iteration method must be found.

3.3. Numerical approximation methods

As is common practise in fluid dynamics and petroleum reservoir engineering, the resulting sets of equations are solved using the finite difference approximation.

The partial differential equations are replaced by difference equations by dividing the region of interest into a three-dimensional grid and developing finite-difference approximations for this grid. Once the region of interest is divided into grid blocks, finite-difference equations are developed whose solution closely approximates the solution of the original equations.

Both for SWIP and DARTEX block-centered grids have been used, which are second-order consistent if equidistant spacing is applied (12).

In SWIP the resulting pentadiagonal (2D) or septadiagonal (3D) matrix equations are solved either directly by D^4-ordering and LU factorization, or iteratively by Line Successive Overrelaxation (3).

In DARTEX the resulting septadiagonal matrix is symmetric and positive definite, and the system of linear equations is solved by preconditioned conjugate gradients (14).

3.4. Supercomputers

Due to the limited capacity of the past generation of computers, it was customary to perform simulations with two-dimensional models, i.e., the quantities to be determined like concentration, temperature and velocity were, in most cases, considered as a function of the horizontal coordinates x and y, but not of the vertical coordinate z. In this way, mean values over the vertical coordinate were obtained. However, due to the layered structure of the underground, taking the mean value over the vertical direction is an unreliable procedure since, in that case, preferential flow paths are neglected. The preferential flow paths make that the actual dispersion of contaminants is completely different from the dispersion predicted by mean values.

The right answer is to make predictions with three-dimensional models (3D models). However, using 3D models the number of arithmetic operations and the memory requirements increase drastically with several orders of magnitude. And there is still another complicating factor. The detailed structure of the permeability- and porosity fields must be introduced in the model. However, this requires a finer grid causing again a drastic increase in arithmetic operations and memory requirements.

For that reason, supercomputers (or vector computers) or attached array processors should be used.

4. SWIP AND DARTEX ON SUPERCOMPUTERS

4.1. Heat storage

As a test example, a heat storage problem was chosen. In an aquifer with a thickness of 30 m between impervious layers, hot water with a temperature of 110°C is injected with a mass flow rate of 16.7 $kg.s^{-1}$ (volumetric flow rate 63.1 $m^3.h^{-1}$). The initial temperature in the aquifer is 20°C. In the whole aquifer the porosity is 0.3; however, the isotropic aquifer is layered with respect to the permeability. The upper layer of 10 m thickness has a permeability of $1.5 \times 10^{-12} m^2$, the middle layer with a thickness of 10m has a permeability of $0.3 \times 10^{-12} m^2$, and the lower layer of 10m thickness has again a permeability of $1.5 \times 10^{12} m^2$. (Hydraulic conductivities of 4.5m day^{-1}, 0.9m.day^{-1} and 4.5m.day^{-1} respectively at 20°C); see Fig. 7. Though the upper- and undersides are

impervious, heat transfer to the over- and underburden clay layers is possible. Both viscosity and density of the groundwater are temperature dependent, possible solutes are not accounted for.

The time of injection was 30 days; then there was neither injection nor production, and finally, there was production of the stored hot water during 30 days. Of course, not all the injected heat is recovered.

For a contour plot of the temperatures see Fig. 7. We will not go further into the physical aspects of the problem, but discuss the computational results. Since the solution is axi-symmetric around the well, the axi-symmetric option of SWIP was used. The extension of the region was limited to 60m.

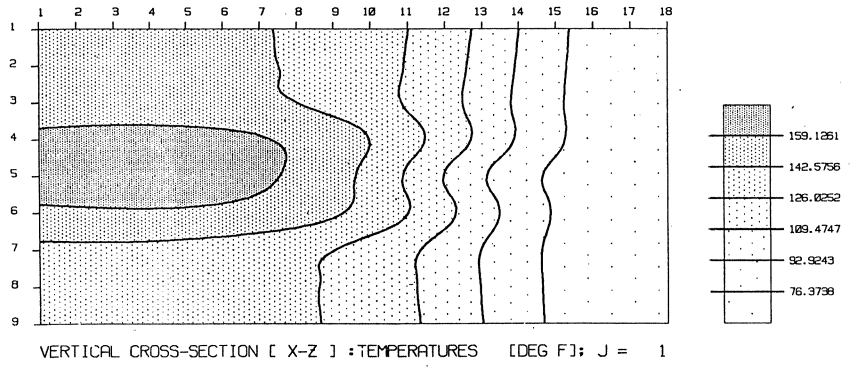


Fig. 7. Contour plot of temperatures 30 days after injection. Injection takes place at the axis of symmetry (the left-hand vertical line).

4.2. Evaluation of test results

A number of test problems was run on the VAX 11/780 of the TNO Institute of Applied Geoscience, on the CYBER 175/100 of the Academic Center Utrecht, on the Cray -1S/1000 of the University of London Computing Centre, and on the CYBER 205 of Control Data Corporation (Arden Hills, USA). The data regarding operation system, compiler and precision are presented in table 1.

Table 1. Overview of used hardware and system software

	operating system	operating system level	compiler	compiler level	compiler option	precision (bits)
VAX 11/780	VMS	2.4	F 77	V2.4-64	optimize	32
CYBER 175/100	NOS/BE 1.5	538	FTN 4	4.8+564	OPT=2	60
Cray-1S/1000	COS 1.11		CFT	1.10	OFF=CTPV OFF=CTP	64
CYBER 205	VSOS 1	L 575	FORTRAN 2.0	R 20C	0=BLOUV 0=BLOU	64

The discretization in grid blocks consisted of equidistant intervals to maintain second-order consistency all over the flow field. Three levels of refinement in discretization were used for the same physical problem: 1st) 18 × 9 grid blocks, 2nd) 36 × 18 grid blocks, and 3rd) 54 × 27 grid blocks. For these three problems the CPU time was determined on the VAX-11/780; these times were 183.9 sec, 1364 sec, and 6186 sec. The problem 18 × 9 has a CPU time of 43.52 sec on the CYBER-175/100, i.e., more than a factor 4 faster than running on the VAX-11/780.

Running on the Cray-1S/1000 under the condition that the compiler did not generate vector code (i.e. with pure scalar arithmetic) resulted in CPU-times of 8.632 sec, 67.77 sec, and 330.1 sec respectively, i.e., a factor 21, 20 and 19 faster than on the VAX-11/780. After having used the vectorization option of the compiler, and after having made slight changes in the most time-consuming DO loops, the CPU times were 7.60 sec, 47.08 sec, and 179.9 sec respectively, i.e. respectively 14%, 44% and 83% faster than with pure scalar arithmetic, and respectively a factor 24, 29 and 34 faster than the VAX 11/780. It is noted that the acceleration increases by increasing vector lengths, which were respectively 9, 18 and 27 for the three problems.

On the CYBER 205, under the condition that the compiler did not generate vector code, the results in CPU time are 10.75 sec for the 18 × 9 problem, and 488.2 sec for the 54 × 27 problem. With respect to the VAX 11/780 the accelerations are 17 and 13 respectively, i.e. the scalar performance of the CYBER 205 is decreasing with respect to the VAX 11/780 for increasing problem size. After vectorization by the

compiler, the CPU times of the 18 × 9 and the 54 × 27 problems were 9.822 sec and 266.9 sec respectively.

That is to say, vectorization resulted respectively in a 10% and 115% decrease of the CPU time, with respect to pure scalar arithmetic, and in a factor 19 and 27 with respect to the VAX-11/780. It is noted that the effects of vectorization of the CYBER 205 are more pronounced than these effects of the Cray-1S, in such a way that the decrease in scalar performance of the CYBER 205 is compensated by an increase in vector performance (for this particular problem). The results are summarized in the tables 2, 3 and 4.

Table 2. CPU-time (sec)

	18 × 9	36 × 18	54 × 27
VAX-11/780	183.86	1364.47	6185.85
CYBER-175/100	43.52		
Cray-1S (scalar)	8.63	67.77	330.11
Cray-1S (vectorized)	7.60	47.08	179.94
CYBER-205 (scalar)	10.75		488.15
CYBER-205 (vectorized)	9.82		226.90

Table 3. Acceleration with respect to VAX 11/780

	18 × 9	36 × 18	54 × 27
CYBER-175/100	4		
Cray-1S (scalar)	21	20	19
Cray-1S (vectorized)	24	29	34
CYBER-205 (scalar)	17		13
CYBER-205 (vectorized)	19		27

Table 4. Acceleration of vectorization with respect to pure scalar arithmetic.

	18 × 9	36 × 18	54 × 27
Cray-1S	1.14	1.44	1.83
CYBER-205	1.095		2.15

From table 3 it follows that the acceleration V of the Cray-1S and the CYBER 205 (where the compilers generate vector code) are related to the problem size N by the following expressions (for this particular problem with the code SWIP):

$$V_{cray} = 5(\frac{N}{162})^{\frac{1}{2}} + 19$$

$$V_{cyber} = 0.8 \times V_{cray}.$$

For instance, it is not unrealistic to state the the price of computing time amounts *f*. 0.10 per CPU-second on the VAX, and *f*. 3.00 per CPU-second on the Cray. In that case N should be larger than 800 to make the Cray competitive with the VAX.

In practical situations problems are always three-dimensional and the number of grid blocks commonly encountered is 20 × 20 × 10 = 4000 or larger. In this example is $V_{cray} \cong 45$, $V_{cyber} \cong 35$ and computations with the VAX would be 50% more expensive than calculations with a supercomputer. Furthermore, with some additional efforts to vectorize SWIP the accelerations presented here can certainly be doubled, making the use of supercomputers even more cost effective (15).

This cost effectiveness becomes even clearer from our experiences with the program DARTEX. For a relatively small test problem with 7 × 7 × 5 = 245 grid blocks the acceleration on the Cray-1S with respect to the VAX was 37 times. Furthermore, it turned out that the process of preconditioning, which was not yet vectorized, consumed 49% of the CPU-time. That is to say: only by vectorizing the preconditioning (e.g. by the use of Neumann Series (14)) the program can be made approximately 70 times faster on a supercomputer than on the VAX.

CONCLUSIONS

- The equations describing fluid flow in the anisotropic subsoil are non-linear, which means that numerical approximation methods must be applied.
- Since actual problems are always three-dimensional with large spatial variations in permeability, supercomputers and attached array processors provide a promising way to solve problems.
- It is expected that the demand for simulation of transport phenomena in the underground will increase considerably when environmental impact assessment studies are required by a licensing authority.

REFERENCES

[1] BACHMAT, Y., J. BREDEHOEFT, B. ANDREWS, D. HOLTZ & S. SEBASTIAN, *Groundwater Management: the Use of Numerical Models*, American Geophysical Union, Washington D.C. (1980).

[2] ZIJL, W. & H. DE BRUIJN, *Continuum equations for the prediction of shell-side flow and temperature patterns in heat exchangers*, Int. J. Heat Mass Transfer, 26, 3, pp. 411-424 (1983).

[3] DE BRUIJN, H., *Least squares numerical analysis of the steady-state and transient thermal-hydraulic behaviour of* LMFBR *heat exchangers*. See this CWI Syllabus.

[4] BEAR, J., *Dynamics of Fluids in Porous Media*, American Elsevier, New York (1972).

[5] BIRD, R.B., W.E. STEWART & E.N. LIGHTFOOT, *Transport Phenomena*, Wiley International, New York (1960).

[6] IDEL'CHIK, I.E., *Handbook of Hydraulic Resistance: Coefficients of Local Resistance and Friction*, AEC-tr-6630. The U.S. Atomic Energy Commission and The National Science Foundation, Washington, D.C.; Israel Program for Scientific Translations Ltd. (1966).

[7] SHINBROD, M., *Lectures on Fluid Mechanics*, Gordon and Breach, New York (1973).

[8] HUBBERT, M.K., *Darcy's Law and the Field Equations of the Flow of Underground Fluids*, Trans. SPE of AIME, 207, pp. 222-239 (JPT) (1956).

[9] FORSCHHEIMER, P., *Wasserbewegung durch Boden*, Z. Ver. Deut. Ing., 45, pp. 1782-1788 (in German) (1901).

[10] SEGOL, G., G.F. PINDER & W.G. GRAY, *A Galerkin-finite element technique for calculating the transient position of the saltwater front*, Water Resources Research, II, 2, pp. 343-347 (1975).

[11] DE BRUIJN, J.G.M. & W. ZIJL, *Least squares finite element solution of several thermal-hydraulic problems in a heat exchanger*, in Proc. 2nd Int. Conf. on Numerical Methods in Thermal Problems, pp. 752-763 (1981).

[12] ZIJL, W., *Finite element methods based on a transport velocity representation for groundwater motion*, accepted for publication in Water Resources Research (1984).

[13] AZIZ, K. & A. SETTARI, *Petroleum Reservoir Simulation*, Applied Science Publishers Ltd., London 1979.

[14] VAN DER VORST, H.A., *Preconditioning by Incomplete Decompositions*, Ph.D-thesis, Utrecht State University (1982).

[15] METCALF, M., *FORTRAN OPTIMIZATION*, Academic Press, London, (1982).

[16] LAMB, H., *Hydrodynamics*, Cambridge University Press, London, (1974).

[17] BATCHELOR, G.K., *An Introduction to Fluid Dynamics*, Cambridge University Press, London (1974).

[18] DE BRUIJN, J.G.M. & W. ZIJL, *Numerical Simulation of Shell-Side Flow and Temperature Distributions in Heat Exchangers*, accepted for: Handbook for Heat and Mass Transfer Operations, ed. N.P. Cheremisinoff, Gulf Publishing, West Orange, N.J. (1984).

MC SYLLABI

1.1 F. Göbel, J. van de Lune. *Leergang besliskunde, deel 1: wiskundige basiskennis.* 1965.

1.2 J. Hemelrijk, J. Kriens. *Leergang besliskunde, deel 2: kansberekening.* 1965.

1.3 J. Hemelrijk, J. Kriens. *Leergang besliskunde, deel 3: statistiek.* 1966.

1.4 G. de Leve, W. Molenaar. *Leergang besliskunde, deel 4: Markovketens en wachttijden.* 1966.

1.5 J. Kriens, G. de Leve. *Leergang besliskunde, deel 5: inleiding tot de mathematische besliskunde.* 1966.

1.6a B. Dorhout, J. Kriens. *Leergang besliskunde, deel 6a: wiskundige programmering 1.* 1968.

1.6b B. Dorhout, J. Kriens, J.Th. van Lieshout. *Leergang besliskunde, deel 6b: wiskundige programmering 2.* 1977.

1.7a G. de Leve. *Leergang besliskunde, deel 7a: dynamische programmering 1.* 1968.

1.7b G. de Leve, H.C. Tijms. *Leergang besliskunde, deel 7b: dynamische programmering 2.* 1970.

1.7c G. de Leve, H.C. Tijms. *Leergang besliskunde, deel 7c: dynamische programmering 3.* 1971.

1.8 J. Kriens, F. Göbel, W. Molenaar. *Leergang besliskunde, deel 8: minimaxmethode, netwerkplanning, simulatie.* 1968.

2.1 G.J.R. Förch, P.J. van der Houwen, R.P. van de Riet. *Colloquium stabiliteit van differentieschema's, deel 1.* 1967.

2.2 L. Dekker, T.J. Dekker, P.J. van der Houwen, M.N. Spijker. *Colloquium stabiliteit van differentieschema's, deel 2.* 1968.

3.1 H.A. Lauwerier. *Randwaardeproblemen, deel 1.* 1967.

3.2 H.A. Lauwerier. *Randwaardeproblemen, deel 2.* 1968.

3.3 H.A. Lauwerier. *Randwaardeproblemen, deel 3.* 1968.

4 H.A. Lauwerier. *Representaties van groepen.* 1968.

5 J.H. van Lint, J.J. Seidel, P.C. Baayen. *Colloquium discrete wiskunde.* 1968.

6 K.K. Koksma. *Cursus ALGOL 60.* 1969.

7.1 *Colloquium moderne rekenmachines, deel 1.* 1969.

7.2 *Colloquium moderne rekenmachines, deel 2.* 1969.

8 H. Bavinck, J. Grasman. *Relaxatietrillingen.* 1969.

9.1 T.M.T. Coolen, G.J.R. Förch, E.M. de Jager, H.G.J. Pijls. *Colloquium elliptische differentiaalvergelijkingen, deel 1.* 1970.

9.2 W.P. van den Brink, T.M.T. Coolen, B. Dijkhuis, P.P.N. de Groen, P.J. van der Houwen, E.M. de Jager, N.M. Temme, R.J. de Vogelaere. *Colloquium elliptische differentiaalvergelijkingen, deel 2.* 1970.

10 J. Fabius, W.R. van Zwet. *Grondbegrippen van de waarschijnlijkheidsrekening.* 1970.

11 H. Bart, M.A. Kaashoek. H.G.J. Pijls, W.J. de Schipper, J. de Vries. *Colloquium halfalgebra's en positieve operatoren.* 1971.

12 T.J. Dekker. *Numerieke algebra.* 1971.

13 F.E.J. Kruseman Aretz. *Programmeren voor rekenautomaten; de MC ALGOL 60 vertaler voor de EL X8.* 1971.

14 H. Bavinck, W. Gautschi, G.M. Willems. *Colloquium approximatietheorie.* 1971.

15.1 T.J. Dekker, P.W. Hemker, P.J. van der Houwen. *Colloquium stijve differentiaalvergelijkingen, deel 1.* 1972.

15.2 P.A. Beentjes, K. Dekker, H.C. Hemker, S.P.N. van Kampen, G.M. Willems. *Colloquium stijve differentiaalvergelijkingen, deel 2.* 1973.

15.3 P.A. Beentjes, K. Dekker, P.W. Hemker, M. van Veldhuizen. *Colloquium stijve differentiaalvergelijkingen, deel 3.* 1975.

16.1 L. Geurts. *Cursus programmeren, deel 1: de elementen van het programmeren.* 1973.

16.2 L. Geurts. *Cursus programmeren, deel 2: de programmeertaal ALGOL 60.* 1973.

17.1 P.S. Stobbe. *Lineaire algebra, deel 1.* 1973.

17.2 P.S. Stobbe. *Lineaire algebra, deel 2.* 1973.

17.3 N.M. Temme. *Lineaire algebra, deel 3.* 1976.

18 F. van der Blij, H. Freudenthal, J.J. de Iongh, J.J. Seidel, A. van Wijngaarden. *Een kwart eeuw wiskunde 1946-1971, syllabus van de vakantiecursus 1971.* 1973.

19 A. Hordijk, R. Potharst, J.Th. Runnenburg. *Optimaal stoppen van Markovketens.* 1973.

20 T.M.T. Coolen, P.W. Hemker, P.J. van der Houwen, E. Slagt. *ALGOL 60 procedures voor begin- en randwaardeproblemen.* 1976.

21 J.W. de Bakker (red.). *Colloquium programmacorrectheid.* 1975.

22 R. Helmers, J. Oosterhoff, F.H. Ruymgaart, M.C.A. van Zuylen. *Asymptotische methoden in de toetsingstheorie; toepassingen van naburigheid.* 1976.

23.1 J.W. de Roever (red.). *Colloquium onderwerpen uit de biomathematica, deel 1.* 1976.

23.2 J.W. de Roever (red.). *Colloquium onderwerpen uit de biomathematica, deel 2.* 1977.

24.1 P.J. van der Houwen. *Numerieke integratie van differentiaalvergelijkingen, deel 1: eenstapsmethoden.* 1974.

25 *Colloquium structuur van programmeertalen.* 1976.

26.1 N.M. Temme (ed.). *Nonlinear analysis, volume 1.* 1976.

26.2 N.M. Temme (ed.). *Nonlinear analysis, volume 2.* 1976.

27 M. Bakker, P.W. Hemker, P.J. van der Houwen, S.J. Polak, M. van Veldhuizen. *Colloquium discretiseringsmethoden.* 1976.

28 O. Diekmann, N.M. Temme (eds.). *Nonlinear diffusion problems.* 1976.

29.1 J.C.P. Bus (red.). *Colloquium numerieke programmatuur, deel 1A, deel 1B.* 1976.

29.2 H.J.J. te Riele (red.). *Colloquium numerieke programmatuur, deel 2.* 1977.

30 J. Heering, P. Klint (red.). *Colloquium programmeeromgevingen.* 1983.

31 J.H. van Lint (red.). *Inleiding in de coderingstheorie.* 1976.

32 L. Geurts (red.). *Colloquium bedrijfsystemen.* 1976.

33 P.J. van der Houwen. *Berekening van waterstanden in zeeën en rivieren.* 1977.

34 J. Hemelrijk. *Oriënterende cursus mathematische statistiek.* 1977.

35 P.J.W. ten Hagen (red.). *Colloquium computer graphics.* 1978.

36 J.M. Aarts, J. de Vries. *Colloquium topologische dynamische systemen.* 1977.

37 J.C. van Vliet (red.). *Colloquium capita datastructuren.* 1978.

38.1 T.H. Koornwinder (ed.). *Representations of locally compact groups with applications, part I.* 1979.

38.2 T.H. Koornwinder (ed.). *Representations of locally compact groups with applications, part II.* 1979.

39 O.J. Vrieze, G.L. Wanrooy. *Colloquium stochastische spelen.* 1978.

40 J. van Tiel. *Convexe analyse.* 1979.

41 H.J.J. te Riele (ed.). *Colloquium numerical treatment of integral equations.* 1979.

42 J.C. van Vliet (red.). *Colloquium capita implementatie van programmeertalen.* 1980.

43 A.M. Cohen, H.A. Wilbrink. *Eindige groepen (een inleidende cursus).* 1980.

44 J.G. Verwer (ed.). *Colloquium numerical solution of partial differential equations.* 1980.

45 P. Klint (red.). *Colloquium hogere programmeertalen en computerarchitectuur.* 1980.

46.1 P.M.G. Apers (red.). *Colloquium databankorganisatie, deel 1.* 1981.

46.2 P.G.M. Apers (red.). *Colloquium databankorganisatie, deel 2.* 1981.

47.1 P.W. Hemker (ed.). *NUMAL, numerical procedures in ALGOL 60: general information and indices.* 1981.

47.2 P.W. Hemker (ed.). *NUMAL, numerical procedures in ALGOL 60, vol. 1: elementary procedures; vol. 2: algebraic evaluations.* 1981.

47.3 P.W. Hemker (ed.). *NUMAL, numerical procedures in ALGOL 60, vol. 3A: linear algebra, part I.* 1981.

47.4 P.W. Hemker (ed.). *NUMAL, numerical procedures in ALGOL 60, vol. 3B: linear algebra, part II.* 1981.

47.5 P.W. Hemker (ed.). *NUMAL, numerical procedures in ALGOL 60, vol. 4: analytical evaluations; vol. 5A: analytical problems, part I.* 1981.

47.6 P.W. Hemker (ed.). *NUMAL, numerical procedures in ALGOL 60, vol. 5B: analytical problems, part II.* 1981.

47.7 P.W. Hemker (ed.). *NUMAL, numerical procedures in ALGOL 60, vol. 6: special functions and constants; vol. 7: interpolation and approximation.* 1981.

48.1 P.M.B. Vitányi, J. van Leeuwen, P. van Emde Boas (red.). *Colloquium complexiteit en algoritmen, deel 1.* 1982.

48.2 P.M.B. Vitányi, J. van Leeuwen, P. van Emde Boas (red.). *Colloquium complexiteit en algoritmen, deel 2.* 1982.

49 T.H. Koornwinder (ed.). *The structure of real semisimple Lie groups.* 1982.

50 H. Nijmeijer. *Inleiding systeemtheorie.* 1982.

51 P.J. Hoogendoorn (red.). *Cursus cryptografie.* 1983.

CWI SYLLABI

1 Vacantiecursus 1984 *Hewet - plus wiskunde.* 1984.

2 E.M. de Jager, H.G.J. Pijls (eds.). *Proceedings Seminar 1981-1982. Mathematical structures in field theories.* 1984.

3 W.C.M. Kallenberg, et.al. *Testing statistical hypotheses: worked solutions.* 1984.

4 J.G. Verwer (ed.). *Colloquium topics in applied numerical analysis, volume 1.* 1984.

5 J.G. Verwer (ed.). *Colloquium topics in applied numerical analysis, volume 2.* 1984.