Centrum voor Wiskunde en Informatica
Centre for Mathematics and Computer Science

J.H.M. ten Thije Boonkkamp

The odd-even hopscotch pressure correction scheme for the computation of free convection in a square cavity

Department of Numerical Mathematics Report NM-R8701 January

The Centre for Mathematics and Computer Science 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.).

The Odd-Even Hopscotch Pressure Correction Scheme for the Computation of Free Convection in a Square Cavity

J.H.M. ten Thije Boonkkamp
Centre for Mathematics and Computer Science
P.O. Box 4079, 1009 AB Amsterdam, The Netherlands

The odd-even hopscotch scheme is an integration scheme applicable to large classes of time-dependent partial differential equations. In this paper we examine it for the computation of free convection in a square cavity. The odd-even hopscotch scheme is combined with the pressure correction method in order to decouple the computation of the pressure from that of the velocity and temperature. The resulting scheme is called the odd-even hopscotch pressure correction scheme, and when combined with a suitable space discretization this scheme proves to be efficient regarding computing time and storage requirements. In order to test the accuracy of our scheme, the solution of the free convection problem computed with this scheme is compared with a very accurate reference solution computed by de Vahl Davis.

1980 Mathematics Subject Classification: 65M20, 65N05, 76D05
Keywords & Phrases: free convection, Navier-Stokes equations in Boussinesq approximation, odd-even hopscotch method, pressure correction method.
Note : This report will be submitted for publication elsewhere.

1. Introduction

In this paper we consider the free convection of a fluid in a square cavity, i.e. the flow in a cavity caused by a temperature gradient. For this problem we use the primitive variable formulation (velocity, pressure, temperature) and the governing equations are the Navier-Stokes equations in Boussinesq approximation [1]. A very accurate steady state benchmark solution for this problem is computed by the de Vahl Davis [15].

We consider the fully transient Navier-Stokes equations in Boussinesq approximation. For the time-integration of these equations one can choose an explicit scheme, an implicit scheme or a combination of both. Explicit schemes are very cheap (per time step), but stability of these schemes is subject to severe time step restrictions. Implicit schemes are usually unconditionally stable, but are much more expensive since they require the solution of large sets of algebraic equations at each time step. The time-integration technique we use is the odd-even hopscotch (OEH) method, which is a combination of the explicit and implicit Euler rule [2, 3, 6]. When combined with a suitable space discretization, the OEH scheme is almost as cheap(stepwise) as the explicit Euler rule, but has much better stability properties.

In order to decouple the computation of the pressure from the computation of the velocity and the temperature, the OEH scheme is combined with the pressure correction method [7]. The scheme thus obtained is called the odd-even hopscotch pressure correction (OEH-PC) scheme. The OEH-PC scheme we will describe in this paper is an extension of the OEH-PC scheme for the incompressible Navier-Stokes equations described in [12].

The purpose of this paper is to give a description of the OEH-PC scheme for the free convection problem and to demonstrate that it is a feasible time-integration technique for this problem. To that end we apply the OEH-PC scheme to a free convection problem due to de Vahl Davis [15], who has computed a very accurate steady state solution for this problem.

The contents of the paper is the following. In Section 2 a description of the free convection problem is given. The OEH-PC scheme is introduced in Section 3 and the space discretization combined

Report NM-R8701
Centre for Mathematics and Computer Science
P.O. Box 4079, 1009 AB Amsterdam, The Netherlands

with this scheme is given in Section 4. Section 5 is devoted to the pressure computation and Section 6 gives a survey of the stability results for the OEH scheme based on linear stability theory. Finally in Section 7 the computational results are presented and compared with the benchmark solution of de Vahl Davis.

2. Physical Problem and Equations

Consider the free convection of a viscous fluid in a 2-dimensional square cavity of width l as shown in Fig. 1. [9,15]. The direction of the gravitational acceleration g is along the negative y-axis and the physical boundary conditions are:

- no slip conditions for the velocity on all 4 walls
- constant temperatures T_1 and $T_2 (T_1 > T_2)$ on respectively the left and right vertical walls
- perfectly insulating upper and lower walls.

The equations governing the fluid motion, in the Boussinesq approximation [1], are:
equation of continuity

$$u_x + v_y = 0 \tag{2.1}$$

equations of motion (Navier-Stokes equations)

$$u_t + uu_x + vu_y = -\frac{1}{\rho_0} p_x + \nu \nabla^2 u \tag{2.2a}$$

$$v_t + uv_x + vv_y = -\frac{1}{\rho_0} p_y + \nu \nabla^2 v - g(1 - \alpha(T - T_0)) \tag{2.2b}$$

temperature equation

$$T_t + uT_x + vT_y = a \nabla^2 T. \tag{2.3}$$

In the above equations u and v are the components of the velocity in x- and y-direction, respectively, p is the pressure and T the temperature. The unknown quantities u, v, p and T are all functions of x, y and t, whereas ρ_0 is the (constant) density at some properly chosen mean temperature T_0. In this problem $T_0 = \frac{1}{2}(T_1 + T_2)$. The coefficients ν, α and a are respectively the kinematic viscosity, the coefficient of volume expansion and the coefficient of thermal conductivity.

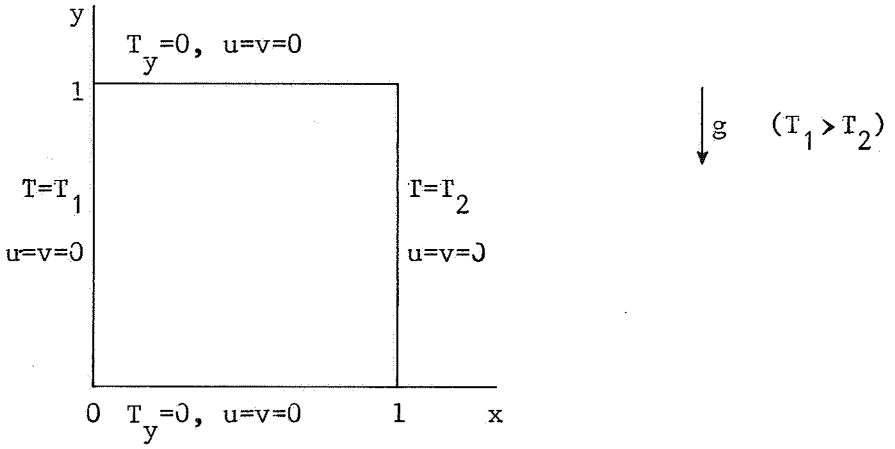

Fig.1. Geometry and boundary conditions.

In order to make the above equations dimensionless, we introduce the following (dimensionless) quantities:

$$x' = \frac{x}{l},\, y' = \frac{y}{l},\, t' = \frac{a}{l^2}t,\, u' = \frac{l}{a}u,\, v' = \frac{l}{a}v, \tag{2.4}$$

$$p' = \frac{l^2}{\rho_0 a^2}(p - p_{hydr}),\, T' = \frac{T-T_2}{T_1-T_2}.$$

Note that p' is the dimensionless deviation from the hydrostatic pressure $p_{hydr} = const. - \rho_0 gy$. After substitution of these variables, the governing equations take the following form (drop the primes)

$$u_x + v_y = 0 \tag{2.5}$$

$$u_t + uu_x + vu_y = -p_x + Pr\nabla^2 u \tag{2.6a}$$

$$v_t + uv_x + vv_y = -p_y + Pr\nabla^2 v + RaPr(T - \tfrac{1}{2}) \tag{2.6b}$$

$$T_t + uT_x + vT_y = \nabla^2 T, \tag{2.7}$$

where Pr and Ra are the Prandtl number and the Rayleigh number respectively, defined by

$$Pr = \frac{\nu}{a},\, Ra = \frac{g\alpha l^3(T_1 - T_2)}{\nu a}. \tag{2.8}$$

In the dimensionless variables the computational space domain is $\Omega = [0,1]\times[0,1]$ and the boundary conditions we consider read

$$x = 0:\, u = v = 0,\, T = 1 \tag{2.9}$$

$$x = 1:\, u = v = 0,\, T = 0$$

$$y = 0,1:\, u = v = 0,\, T_y = 0.$$

Initial conditions will be specified later.

In what follows we will consider the dimensionless equations and refer to them, for convenience, as the Navier-Stokes equations in Boussinesq approximation. Finally we note that the equations (2.6a)-(2.7) can be rewritten as

$$u_t + (u^2)_x + (uv)_y = -p_x + Pr\nabla^2 u \tag{2.6a$'$}$$

$$v_t + (uv)_x + (v^2)_y = -p_y + Pr\nabla^2 v + RaPr(T - \tfrac{1}{2}) \tag{2.6b$'$}$$

$$T_t + (uT)_x + (vT)_y = \nabla^2 T. \tag{2.7$'$}$$

The equations (2.6a)-(2.7) are written in the so-called convective form and the equations (2.6a')-(2.7') in the conservative form.

3. The Odd-Even Hopscotch Pressure Correction Scheme

In this section we consider the odd-even hopscotch (OEH) scheme for the time-integration of the Navier-Stokes equations in Boussinesq approximation. For a detailed discussion of the OEH scheme, the reader is referred to [2,3,6,14]. The OEH scheme is combined with the pressure correction method, which is a predictor-corrector method for the decoupling of the pressure computation. The resulting scheme will be referred to as the odd-even hopscotch pressure correction (OEH-PC) scheme, and is an extension of the scheme given in [12]. A description of the pressure correction method can be found in [7].

The Navier-Stokes equations in Boussines approximation in d space dimensions ($d=2$ or $d=3$) can in general be written as:

$$\mathbf{u}_t = \mathbf{f}(\mathbf{u},T)-\nabla p \tag{3.1}$$

$$T_t = h(\mathbf{u},T) \tag{3.2}$$

$$\nabla\cdot\mathbf{u} = 0, \tag{3.3}$$

where **u** is the velocity, p the pressure and T the temperature. For the time being, the exact form of $\mathbf{f}(\mathbf{u},T)$ and $h(\mathbf{u},T)$ (convective/conservative) is not important. The partial differential equations (PDEs) (3.1)-(3.3) are defined on a connected space domain Ω with boundary Γ, on which conditions for the velocity **u** and the temperature T are specified. Notice that the boundary conditions for **u** must satisfy

$$\oint_\Gamma \mathbf{u}\cdot\mathbf{n}ds = \iint_\Omega \nabla\cdot\mathbf{u}dS = 0, \tag{3.4}$$

where **n** is the unit outward normal on Γ (conservation of mass).

We present the OEH-PC scheme for the PDEs (3.1)-(3.3) following the method of lines approach [6]. Thus suppose first that by a suitable finite difference space discretization the PDEs (3.1)-(3.3) are replaced by the following set of ordinary differential equations (ODEs) and algebraic equations

$$\dot{\mathbf{U}} = \mathbf{F}(\mathbf{U},T)-GP \tag{3.5}$$

$$\dot{T} = H(\mathbf{U},T) \tag{3.6}$$

$$D\mathbf{U} = B. \tag{3.7}$$

In (3.5)-(3.7) the variables $\mathbf{U},P$ and T are grid functions defined on a space grid covering Ω, and $\mathbf{F}(\mathbf{U},T)$ and $H(\mathbf{U},T)$ are the finite difference replacements of $\mathbf{f}(\mathbf{u},T)$ and $h(\mathbf{u},T)$, respectively. The operators G and D are the finite difference replacements of the gradient- and divergence-operators and B is a term containing boundary values for the velocity **u**.

First consider the ODEs (3.5), (3.6) and suppose for the time being that GP is a known forcing term. Let $\mathbf{x}_j$ be a gridpoint corresponding to the multi-index $j = (j_1,...,j_d)$ and let $\mathbf{U}_j^n$ denote the approximation to $\mathbf{u}(\mathbf{x}_j,t_n)$ (and likewise for $P,T,\mathbf{F}$ and H), then the OEH scheme for (3.5), (3.6) reads [3]

$$\mathbf{U}_j^{n+1}-\tau\theta_j^{n+1}(\mathbf{F}_j(\mathbf{U}^{n+1},T^{n+1})-(GP)_j^{n+1}) = \mathbf{U}_j^n+\tau\theta_j^n(\mathbf{F}_j(\mathbf{U}^n,T^n)-(GP)_j^n) \tag{3.8a}$$

$$T_j^{n+1}-\tau\theta_j^{n+1}H_j(\mathbf{U}^{n+1},T^{n+1}) = T_j^n+\tau\theta_j^nH_j(\mathbf{U}^n,T^n). \tag{3.8b}$$

Here $\tau = t_{n+1}-t_n$ is the time step and θ_j^n is the so-called odd-even function defined by

$$\theta_j^n = \begin{cases} 1 \text{ if } n+\sum_i j_i \text{ is odd (odd points)} \\ 0 \text{ if } n+\sum_i j_i \text{ is even (even points).} \end{cases} \tag{3.9}$$

Note that in the odd points the OEH scheme reduces to the forward Euler rule and in the even points to the backward Eules rule.

An alternative form of (3.8a), (3.8b) is

$$\mathbf{U}^{n+1} = \mathbf{U}^n+\tau\mathbf{F}_O(\mathbf{U}^n,T^n)+\tau\mathbf{F}_E(\mathbf{U}^{n+1},T^{n+1})-\tau(GP^n)_O-\tau(GP^{n+1})_E \tag{3.10a}$$

$$T^{n+1} = T^n+\tau H_O(\mathbf{U}^n,T^n)+\tau H_E(\mathbf{U}^{n+1},T^{n+1}), \tag{3.10b}$$

where $\mathbf{F}_O$ is the restriction of **F** to the odd points (etc.). Note that $\mathbf{F}_O+\mathbf{F}_E = \mathbf{F}$ and $H_O+H_E = H$. We shall use this formulation in the remainder of the section. It is customary to write down two successive steps of (3.10a), (3.10b) with stepsize $\tau/2$, where the order of implicit and explicit calculations alternate [14]

$$\tilde{\mathbf{U}} = \mathbf{U}^n+\frac{1}{2}\tau\mathbf{F}_O(\mathbf{U}^n,T^n)+\frac{1}{2}\tau\mathbf{F}_E(\tilde{\mathbf{U}},\tilde{T})-\frac{1}{2}\tau GP^n \tag{3.11a}$$

$$\tilde{T} = T^n + \frac{1}{2}\tau H_O(\mathbf{U}^n, T^n) + \frac{1}{2}\tau H_E(\tilde{\mathbf{U}}, \tilde{T}) \tag{3.11b}$$

$$\mathbf{U}^{n+1} = \tilde{\mathbf{U}} + \frac{1}{2}\tau \mathbf{F}_E(\tilde{\mathbf{U}}, \tilde{T}) + \frac{1}{2}\tau \mathbf{F}_O(\mathbf{U}^{n+1}, T^{n+1}) - \frac{1}{2}\tau G P^{n+1} \tag{3.11c}$$

$$T^{n+1} = \tilde{T} + \frac{1}{2}\tau H_E(\tilde{\mathbf{U}}, \tilde{T}) + \frac{1}{2}\tau H_O(\mathbf{U}^{n+1}, T^{n+1}). \tag{3.11d}$$

This is a second order accurate integration formula for the numerical integration of the ODE systems (3.5), (3.6) using stepsize τ. The variables $\tilde{\mathbf{U}}$ and $\tilde{T}$ can be interpreted as results from the intermediate time level $(n+\frac{1}{2})\tau$, like in a Runge-Kutta formula. Note that in (3.11a) P is set at time level $t_n = n\tau$ and in (3.11c) at time level $t_{n+1} = (n+1)\tau$. An alternative for maintaining second order is to compute P at time level $n+\frac{1}{2}$ both stages. However the choice made in (3.11a), (3.11c) is better adapted to the pressure correction approach.

Consider (3.11a)-(3.11d) coupled with the (time discretized) set of algebraic equations

$$D\mathbf{U}^{n+1} = B^{n+1}. \tag{3.11e}$$

The computation of $\mathbf{U}^{n+1}, P^{n+1}$ and T^{n+1} requires the simultaneous solution of (3.11c)-(3.11e). In order to avoid this, we follow the known pressure correction approach [7] in which the computation of P^{n+1} is decoupled in the predictor-corrector fashion. Substitution of P^n for P^{n+1} in (3.11c) defines the predicted velocity $\overset{\approx}{\mathbf{U}}$ and the predicted temperature $\overset{\approx}{T}$. The corrected velocity, pressure and temperature (which we hereafter also denote by $\mathbf{U}^{n+1}, P^{n+1}$ and T^{n+1} and hence should not be mixed up with the approximations in (3.11c), (3.11d)) are then defined by replacing $\mathbf{F}_O(\mathbf{U}^{n+1}, T^{n+1})$ in (3.11c) and $H_O(\mathbf{U}^{n+1}, T^{n+1})$ in (3.11d) by respectively $\mathbf{F}_O(\overset{\approx}{\mathbf{U}}, \overset{\approx}{T})$ and $H_O(\overset{\approx}{\mathbf{U}}, \overset{\approx}{T})$:

$$\mathbf{U}^{n+1} = \tilde{\mathbf{U}} + \frac{1}{2}\tau \mathbf{F}_E(\tilde{\mathbf{U}}, \tilde{T}) + \frac{1}{2}\tau \mathbf{F}_O(\overset{\approx}{\mathbf{U}}, \overset{\approx}{T}) - \frac{1}{2}\tau G P^{n+1} \tag{3.12a}$$

$$T^{n+1} = \tilde{T} + \frac{1}{2}\tau H_E(\tilde{\mathbf{U}}, \tilde{T}) + \frac{1}{2}\tau H_O(\overset{\approx}{\mathbf{U}}, \overset{\approx}{T}), \tag{3.12b}$$

together with the discrete continuity equation (3.11e). From (3.12a), (3.12b) and the modified equations (3.11c), (3.11d) one can easily see that

$$\mathbf{U}^{n+1} - \overset{\approx}{\mathbf{U}} = -\frac{1}{2}\tau G Q^n, \; Q^n = P^{n+1} - P^n \tag{3.13a}$$

$$T^{n+1} - \overset{\approx}{T} = 0. \tag{3.13b}$$

The trick of the pressure correction method is now to multiply (3.13a) by D and to write, using (3.11e),

$$LQ^n = \frac{2}{\tau}(D\overset{\approx}{\mathbf{U}} - B^{n+1}), L = DG. \tag{3.14}$$

Since $L = DG$ is a discretization of the Laplace operator $\nabla\cdot(\nabla)$, the correction Q^n for the pressure can be obtained by applying a Poisson solver. Once Q^n is known, the new velocity can be directly determined from (3.13a).

To sum up, the OEH-PC scheme for the semi-discrete system (3.5)-(3.7), reads

$$\tilde{\mathbf{U}} = \mathbf{U}^n + \frac{1}{2}\tau \mathbf{F}_O(\mathbf{U}^n, T^n) + \frac{1}{2}\tau \mathbf{F}_E(\tilde{\mathbf{U}}, \tilde{T}) - \frac{1}{2}\tau G P^n \tag{3.15a}$$

$$\tilde{T} = T^n + \frac{1}{2}\tau H_O(\mathbf{U}^n, T^n) + \frac{1}{2}\tau H_E(\tilde{\mathbf{U}}, \tilde{T}) \tag{3.15b}$$

$$\overset{\approx}{\mathbf{U}} = \tilde{\mathbf{U}} + \frac{1}{2}\tau \mathbf{F}_E(\tilde{\mathbf{U}}, \tilde{T}) + \frac{1}{2}\tau \mathbf{F}_O(\overset{\approx}{\mathbf{U}}, T^{n+1}) - \frac{1}{2}\tau G P^n \tag{3.15c}$$

$$T^{n+1} = \tilde{T} + \frac{1}{2}\tau H_E(\tilde{\mathbf{U}}, \tilde{T}) + \frac{1}{2}\tau H_O(\overset{\approx}{\mathbf{U}}, T^{n+1}) \tag{3.15d}$$

$$LQ^n = \frac{2}{\tau}(D\overset{\approx}{\mathbf{U}}-B^{n+1}), P^{n+1} = P^n+Q^n \tag{3.15e}$$

$$\mathbf{U}^{n+1} = \overset{\approx}{\mathbf{U}}-\tfrac{1}{2}\tau GQ^n. \tag{3.15f}$$

When combined with a suitable space discretization, the OEH-PC scheme possesses various advantageous features. We shall discuss this in greater detail in the next section for symmetric finite differences on a staggered grid.

We conclude this section with two remarks. Firstly, the second stage (3.15c), (3.15d) can be economized using its equivalent fast form (cf. [2,3])

$$\overset{\approx}{\mathbf{U}}_E = 2\tilde{\mathbf{U}}_E-\mathbf{U}_E^n,\ T_E^{n+1} = 2\tilde{T}_E-T_E^n \tag{3.16a}$$

$$\overset{\approx}{\mathbf{U}}_O = \tilde{\mathbf{U}}_O+\tfrac{1}{2}\tau\mathbf{F}_O(\overset{\approx}{\mathbf{U}},\ T^{n+1})-\tfrac{1}{2}\tau(GP^n)_O,\ T_O^{n+1} = \tilde{T}_O+\tfrac{1}{2}\tau H_O(\overset{\approx}{\mathbf{U}},T^{n+1}). \tag{3.16b}$$

Our implementation is based on this fast form. Secondly, in the derivation of scheme (3.15) no use has been made of the particular definition of $\mathbf{F}_O,\mathbf{F}_E,H_O$ and H_E, except that $\mathbf{F}_O+\mathbf{F}_E = \mathbf{F}$ and $H_O+H_E = H$. Consequently, pressure correction schemes using other splittings of $\mathbf{F}$ and H, such as ADI, can also be described by (3.15).

4. Space Discretization

Consider the 2-dimensional Navier-Stokes equations in Boussinesq approximation (see section 2)

$$u_t = f_1(u,v,T)-p_x, \text{ with } f_1(u,v,T) = -(u^2)_x-(uv)_y+Pr(u_{xx}+u_{yy}) \tag{4.1a}$$

$$v_t = f_2(u,v,T)-p_y, \text{ with } f_2(u,v,T) = -(uv)_x-(v^2)_y+Pr(v_{xx}+v_{yy})+RaPr(T-\tfrac{1}{2}) \tag{4.1b}$$

$$T_t = h(u,v,T), \text{ with } h(u,v,T) = -uT_x-vT_y+T_{xx}+T_{yy} \tag{4.2}$$

$$u_x+v_y = 0. \tag{4.3}$$

Note that the equations of motion are written in conservative form while the temperature equation is written in convective form. The reason for this will become clear later. Boundary conditions for the velocity and the temperature are specified. Also note that there are no pressure boundary conditions available although we have to solve a Poisson equation for the pressure. We will return to this point later in the section.

For the space discretization we use the staggered grid of Fig.2. (see also [12,8]). The application of standard, second order central differences converts (4.1a)-(4.2) into (Cf.(3.5), (3.6))

$$\dot{U}_{ij} = F_{1,ij}(U,V,T)-d_xP_{ij} \quad i = 1(1)N-1,\ j = 1(1)M \ (\textit{interior} \times -\textit{points}) \tag{4.4.a}$$

$$\dot{V}_{ij} = F_{2,ij}(U,V,T)-d_yP_{ij} \quad i = 1(1)N,\ j = 1(1)M-1 \ (\textit{interior} \bigcirc -\textit{points}) \tag{4.4b}$$

$$\dot{T}_{ij} = H_{ij}(U,V,T) \quad i = 1(1)N-1,\ j = 1(1)M-1 \ (\textit{interior} \star -\textit{points}) \tag{4.5}$$

where

$$F_{1,ij}(U,V,T) = -\frac{1}{2h}(U_{i+i,j}^2-U_{i-1,j}^2)-\frac{1}{2k}(U_{i,j+1}\overline{V}_{i,j+1}-U_{i,j-1}\overline{V}_{i,j-1}) + \tag{4.6a}$$
$$\frac{Pr}{h^2}(U_{i+1,j}-2U_{ij}+U_{i-1,j})+\frac{Pr}{k^2}(U_{i,j+1}-2U_{ij}+U_{i,j-1})$$

$$F_{2,ij}(U,V,T) = \frac{-1}{2h}(\overline{U}_{i+1,j}V_{i+1,j}-\overline{U}_{i-1,j}V_{i-1,j})-\frac{1}{2k}(V_{i,j+1}^2-V_{i,j-1}^2) + \tag{4.6b}$$
$$\frac{Pr}{h^2}(V_{i+1,j}-2V_{ij}+V_{i-1,j})+\frac{Pr}{k^2}(V_{i,j+1}-2V_{ij}+V_{i,j-1})+RaPr(T-\tfrac{1}{2})$$

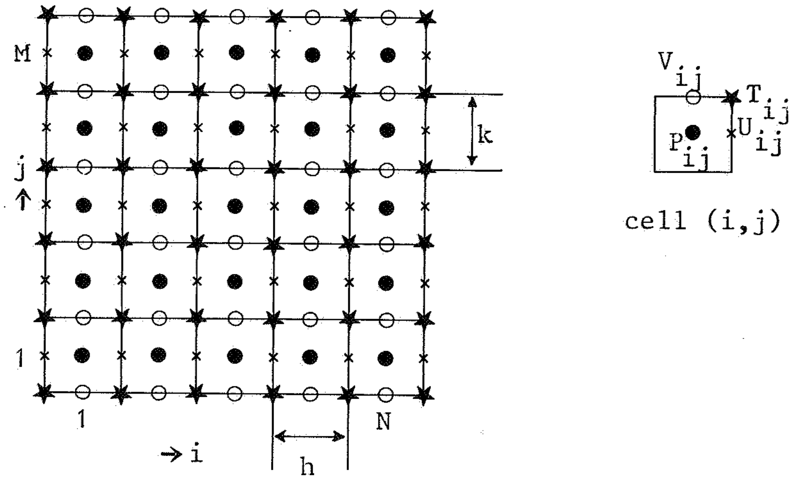

Fig.2. The staggered grid.

$$H_{ij}(U,V,T) = \frac{-1}{2h}\bar{\bar{U}}_{ij}(T_{i+1,j}-T_{i-1,j}) - \frac{1}{2k}\bar{\bar{V}}_{ij}(T_{i,j+1}-T_{i,j-1}) + \tag{4.6c}$$

$$\frac{1}{h^2}(T_{i+1,j}-2T_{ij}+T_{i-1,j}) + \frac{1}{k^2}(T_{i,j+1}-2T_{ij}+T_{i,j-1})$$

$$d_xP_{ij} = \frac{1}{h}(P_{i+1,j}-P_{ij}) \tag{4.6d}$$

$$d_yP_{ij} = \frac{1}{k}(P_{i,j+1}-P_{ij}). \tag{4.6e}$$

In (4.6a) $\bar{V}_{ij}$ represents an approximation to V in the $\times$-points (points where U is defined); likewise $\bar{U}_{ij}$ and $\bar{T}_{ij}$ represent approximations to U and T in the O-points. The values of $\bar{U}_{ij}, \bar{V}_{ij}$ and $\bar{T}_{ij}$ are determined by averaging over neighbouring values of respectively U_{ij}, V_{ij} and T_{ij} in such a way that the odd-even coupling between the variables is preserved. This means that the variables in an odd cell are only coupled with variables in even cells and vice versa. This leads to

$$\bar{U}_{ij} = \frac{1}{2}(U_{ij}+U_{i-1,j+1}), \bar{V}_{ij} = \frac{1}{2}(V_{ij}+V_{i+1,j-1}), \bar{T}_{ij} = \frac{1}{2}(T_{ij}+T_{i-1,j}). \tag{4.7}$$

The same argument leads to the definitions

$$\bar{\bar{U}}_{ij} = \frac{1}{2}(U_{ij}+U_{i,j+1}), \bar{\bar{V}}_{ij} = \frac{1}{2}(V_{ij}+V_{i+1,j}), \tag{4.8}$$

where $\bar{\bar{U}}_{ij}$ and $\bar{\bar{V}}_{ij}$ are approximations to U and V in the $\star$-points. In case we use the convective form of the equations of motion or the conservative form of the temperature equation, one can see that the odd-even coupling between the variables is lost. This explains our choice of writing the equations of motion in conservative form and the temperature equation in convective form. Remark that the gradient operator G (Cf.(3.5)) is of course defined by $GP_{ij} = (d_xP_{ij}, d_yP_{ij})^T$.

Concerning the boundary conditions for the velocity we note the following. Consider e.g. equation (4.1a) in the $\times$-points $(i,1)(i=1(1)N-1)$. Discretization of the derivatives $(uv)_y$ and u_{yy} would require values outside the computational domain. Therefore we replace the central difference approximations to $(uv)_y$ and u_{yy} by the following noncentered first order differences [8], which preserve the odd-even coupling between the variables

$$((uv)_y)_{i1} = \frac{2}{3k}(U_{i2}\overline{V}_{i2} - u(ih, 0)v(ih, 0)) \tag{4.9a}$$

$$(u_{yy})_{i1} = \frac{4}{3k^2}(U_{i2} - 3U_{i1} + 2u(ih, 0)). \tag{4.9b}$$

Second order non-centered approximations to $(uv)_y$ and u_{yy} would destroy the odd-even coupling.

Space discretization of equation (4.3) in all ·-points (using central differences) yields

$$(D\mathbf{U})_{ij} := \frac{1}{h}(U_{ij} - U_{i-1,j} + \beta(V_{ij} - V_{i,j-1})) = 0, \tag{4.10}$$

where $\beta = h/k$. One should note that boundary values for U or V occurring in (4.10) are written in the right hand side B (Cf. (3.7)). For example for $j = 1$, equation (4.3) is discretized as

$$(D\mathbf{U})_{i1} := \frac{1}{h}(U_{i1} - U_{i-1,1} + \beta V_{i1}) = B_{i1} = \frac{1}{k}V_{i0}. \tag{4.10'}$$

Having defined the operators G and D, one can easily deduce the following expression for the operator L

$$(LQ)_{ij} = D(GQ)_{ij} = \frac{1}{h}(d_x Q_{ij} - d_x Q_{i-1,j} + \beta(d_y Q_{ij} - d_y Q_{i,j-1})) = \tag{4.11}$$
$$\frac{1}{h^2}(\beta^2 Q_{i,j-1} + Q_{i-1,j} - (2+2\beta^2)Q_{ij} + Q_{i+1,j} + \beta^2 Q_{i,j+1}),$$

which is the standard 5-point molecule for the Laplace operator. Near a boundary (4.11) takes a different form, because of the different definition of the operator D. For example, for $j = 1$ one finds

$$(LQ)_{i1} = D(GQ)_{i1} = \frac{1}{h}(d_x Q_{i1} - d_x Q_{i-1,1} + \beta d_y Q_{i1}) = \tag{4.11'}$$
$$\frac{1}{h^2}(Q_{i-1,1} - (2+\beta^2)Q_{i1} + Q_{i+1,1} + \beta^2 Q_{i2}).$$

Now consider equation (3.15e) at the ·-points $(i, 1)$ $(i = 1(1)N)$. Using (4.10), (4.10′), (4.11) and (4.11′) it is easy to see that $\frac{1}{k}(Q_{i0} - Q_{i1}) = \frac{2}{\tau}(V_{i0}^{n+1} - \overset{\approx}{V}_{i0}) = 0$, which is the (central difference) approximation to $\frac{\partial Q}{\partial n}((i - \frac{1}{2})h, 0) = 0$, where n is the outward unit normal on $x = 0$. Hence we see that a Neumann condition for the pressure (-increment) is automatically involved in the scheme.

We conclude this section with a few remarks. The essential feature of the OEH scheme is the alternating use of the forward and backward Euler rule. Consider the equations (3.15a)-(3.15d) of the OEH-PC scheme (3.15). The order of computation is:

$$\tilde{\mathbf{U}}_O = \mathbf{U}_O^n + \frac{1}{2}\tau\mathbf{F}_O(\mathbf{U}^n, T^n) - \frac{1}{2}\tau(GP^n)_O \tag{4.12a}$$

$$\tilde{T}_O = T_O^n + \frac{1}{2}\tau H_O(\mathbf{U}^n, T^n) \tag{4.12b}$$

$$\tilde{\mathbf{U}}_E = \mathbf{U}_E^n + \frac{1}{2}\tau\mathbf{F}_E(\tilde{\mathbf{U}}, \tilde{T}) - \frac{1}{2}\tau(GP^n)_E \tag{4.12c}$$

$$\tilde{T}_E = T_E^n + \frac{1}{2}\tau H_E(\tilde{\mathbf{U}}, \tilde{T}) \tag{4.12d}$$

$$\overset{\approx}{\mathbf{U}}_E = \tilde{\mathbf{U}}_E + \frac{1}{2}\tau\mathbf{F}_E(\tilde{\mathbf{U}}, \tilde{T}) - \frac{1}{2}\tau(GP^n)_E = 2\tilde{\mathbf{U}}_E - \mathbf{U}_E^n \tag{4.12e}$$

$$T_E^{n+1} = \tilde{T}_E + \frac{1}{2}\tau H_E(\tilde{\mathbf{U}}, \tilde{T}) = 2\tilde{T}_E - T_E^n \tag{4.12f}$$

$$\overset{\approx}{\mathbf{U}}_O = \tilde{\mathbf{U}}_O + \frac{1}{2}\tau\mathbf{F}_O(\overset{\approx}{\mathbf{U}}, T^{n+1}) - \frac{1}{2}\tau(GP^n)_O \tag{4.12g}$$

$$T_O^{n+1} = \tilde{T}_O + \frac{1}{2}\tau H_O(\approx{\mathbf{U}}, T^{n+1}). \tag{4.12h}$$

Because of the odd-even coupling between the variables and the alternating use of the forward and backward Euler rule, the algorithm (4.12) is only diagonally implicit. This means that per cell a 3×3 system of linear equations has to be solved, which is of course very cheap. Hence the OEH-PC scheme is almost as fast as the explicit Euler rule, but has much better stability properties as we shall see in Section 6. From (4.12) one can also see that only one array of storage is required for each variable (a known feature of the OEH scheme [2,3]), which is especially of interest for multi-dimensional problems.

5. Computation of the Pressure

For the computation of the pressure (-increment) we have to solve the Poisson equation

$$LQ^n = \frac{2}{\tau}c, \; c = D\overset{\approx}{\mathbf{U}} - B^{n+1}, \tag{5.1}$$

where L is the operator defined in (4.11) and (4.11′). Considered as a matrix, L has a few attractive properties such as symmetry, positive definiteness and a pentadiagonal structure. However, L is singular with $Le = 0$, where $e = (1,...,1)^T$, and therefore the set of equations (5.1) has only a solution if $(e,c) = 0$. The condition $(e,c) = 0$ implies that

$$\sum_{j=1}^{M} k(U_{Nj}^{n+1} - U_{0j}^{n+1}) + \sum_{i=1}^{N} h(V_{iM}^{n+1} - V_{i0}^{n+1}) = 0, \tag{5.2}$$

which is a second order approximation to (3.4) at time level $t_{n+1} = (n+1)\tau$. In our case (5.2) is trivially satisfied due to the zero boundary values for the velocity. For arbitrary boundary values $\mathbf{u}_\Gamma$, it may be necessary to make small adjustments in the right hand side c in order to satisfy (5.2).

There are many methods available for the solution of (5.1). Since the OEH scheme is very cheap per time step, it is essential that we combine it with a fast Poisson solver in order to obtain a fast OEH-PC scheme. In our computations we used the multigrid method MGD5V [4,11]. The multigrid method MGD5V is a saw tooth multigrid iterative process (i.e. one relaxation sweep after each coarse grid correction) for the solution of linear second order elliptic boundary value problems. It uses incomplete line *LU*-decompposition as relaxation method, a 7-point prolongation and restriction, and a Galerkin approximation for the coarse grid matrices. The multigrid process is repeated until the l_2-norm of the residual is less than 10^{-4}. We wish to emphasize that MGD5V was designed for more general elliptic problems than our simple Poisson equation. Consequently, the computation of the pressure-increment, which is considerable anyhow, can probably be done faster with a solver specifically designed for the Poisson equation.

6. Linear Stability Analysis

In this section we present the conditions for stability based on von Neumann analysis [10]. Consider the linearized equations (Cf.(2.6a)-(2.7))

$$u_t + q_1 u_x + q_2 u_y = -p_x + Pr.\nabla^2 u \tag{6.1a}$$

$$v_t + q_1 v_x + q_2 v_y = -p_y + Pr\nabla^2 v + RaPr(T - \tfrac{1}{2}) \tag{6.1b}$$

$$T_t + q_1 T_x + q_2 T_y = \nabla^2 T, \tag{6.2}$$

where q_1 and q_2 are properly chosen approximations to u and v. Note that the computation of T is now decoupled from the computation of u and v, and hence the term $RaPr(T - \frac{1}{2})$ in (6.1b) can be considered as a source term, which has no influence on stability. Therefore we can leave out this term in our analysis. For the sake of simplicity we also leave out the terms $-p_x$ and $-p_y$, and thus consider the equation

$$f_t+(\mathbf{q}\cdot\nabla)f = \epsilon\nabla^2 f, \quad t>0, \ \mathbf{x}\in\mathbb{R}^d, \tag{6.3}$$

where $f = u,v$ or T, $\mathbf{q} = (q_1,\ldots,q_d)^T$ is the constant velocity and $\epsilon>0$ the viscosity parameter. This equation models the convective and viscous effects in (6.1a)-(6.2).

Suppose that for space discretization of (6.3) we use standard central differences with mesh size h in all space directions, then von Neumann stability analysis applied to the OEH scheme for (6.3) yields the following time step restriction [14]

$$d.(\frac{\tau}{h})^2 \sum_{k=1}^{d} q_k^2 \leqslant 1. \tag{6.4}$$

Observe that the time step restriction is independent of the viscosity parameter ϵ. In our actual computations ($d=2$), the value of τ is based on the choice $q_1 = u_{\max}, q_2 = v_{\max}$, where $u_{\max}$ and $v_{\max}$ are the maximum values of u and v, respectively, computed by de Vahl Davis [15]. (see Table 1).

For the sake of comparison, we give the necessary conditions for von Neumann stability of the forward Euler central difference scheme for (6.3)[5]

$$\frac{2d\epsilon\tau}{h^2} \leqslant 1, \sum_{k=1}^{d} \frac{q_k^2\tau}{2\epsilon} \leqslant 1. \tag{6.5}$$

The second inequality (convextion-diffusion barrier) shows that the forward Euler central difference scheme becomes unconditionally unstable as $\epsilon \to 0$, whereas the OEH scheme is conditionally stable uniformly in ϵ, i.e. $\tau = O(h)$ independent of ϵ. The first inequality of (6.3) implies that $\tau = O(\epsilon^{-1}h^2)$ for stability, which is disadvantageous for large values of ϵ. From the above it is clear that the OEH scheme has much better stability properties than the forward Euler central difference scheme.

7. Computational Results

We have computed the solution of the free convection problem (2.5)-(2.9) for the Rayleigh numbers $Ra = 10^3, 10^4, 10^5$ and 10^6. The Prandtl number is set equal to 0.71 (air). Computations were performed on a 20×20 grid and a 40×40 grid, until the steady state was obtained. We assume that the solution has reached its steady state if $\|U^{n+1}-U^n\|_1 < \delta\tau, \|V^{n+1}-V^n\|_1 < \delta\tau$ and $\|T^{n+1}-T^n\|_1 < \delta\tau$, where δ is a sufficiently small number. In our computations we took $\delta = 10^{-2}$. Initial conditions will be specified later. Since the steady state solution is independent of the initial conditions, we can choose these arbitrarily, provided that the initial velocity field is divergence free.

In Fig.3,4 and 5 we present the steady state streamline patterns, velocity fields and isotherms, respectively, computed on the 40×40 grid. In the vectorplots of Fig.4 the velocity field is scaled to the maximum occurring velocity. From these figures we can draw the following conclusions.

For $Ra = 10^3$, the temperature field satisfies in a crude approximation the Laplace equation, and therefore heat is transferred between the vertical walls primarily by conduction. In the velocity field, a weak clockwise vortex is prevailing. For $Ra = 10^4, 10^5$ a vertical temperature gradient in the core of the flow is present. Now heat transfer between the vertical walls takes place through convection and conduction. A boundary layer for the temperature as well as for the velocity starts to develop near the vertical walls. Finally, for $Ra = 10^6$ the isotherms in the interior flow field are almost horizontal and the predominant mode of heat transfer is convection. Clearly a boundary layer for the temperature and the velocity near the vertical walls and a weak two eddy structure in the core of the flow can be observed.

At this point it is emphasized once more, that the OEH-PC scheme is a solution technique for the transient Navier-Stokes equations. However, we use the scheme to compute the steady state solution of the free convection problem, in order to be able to compare our solution with the benchmark solution of de Vahl Davis [15], which is one of the most accurate solutions available for this problem. For that purpose we have computed a few characteristic values of the flow, which can be readily compared.

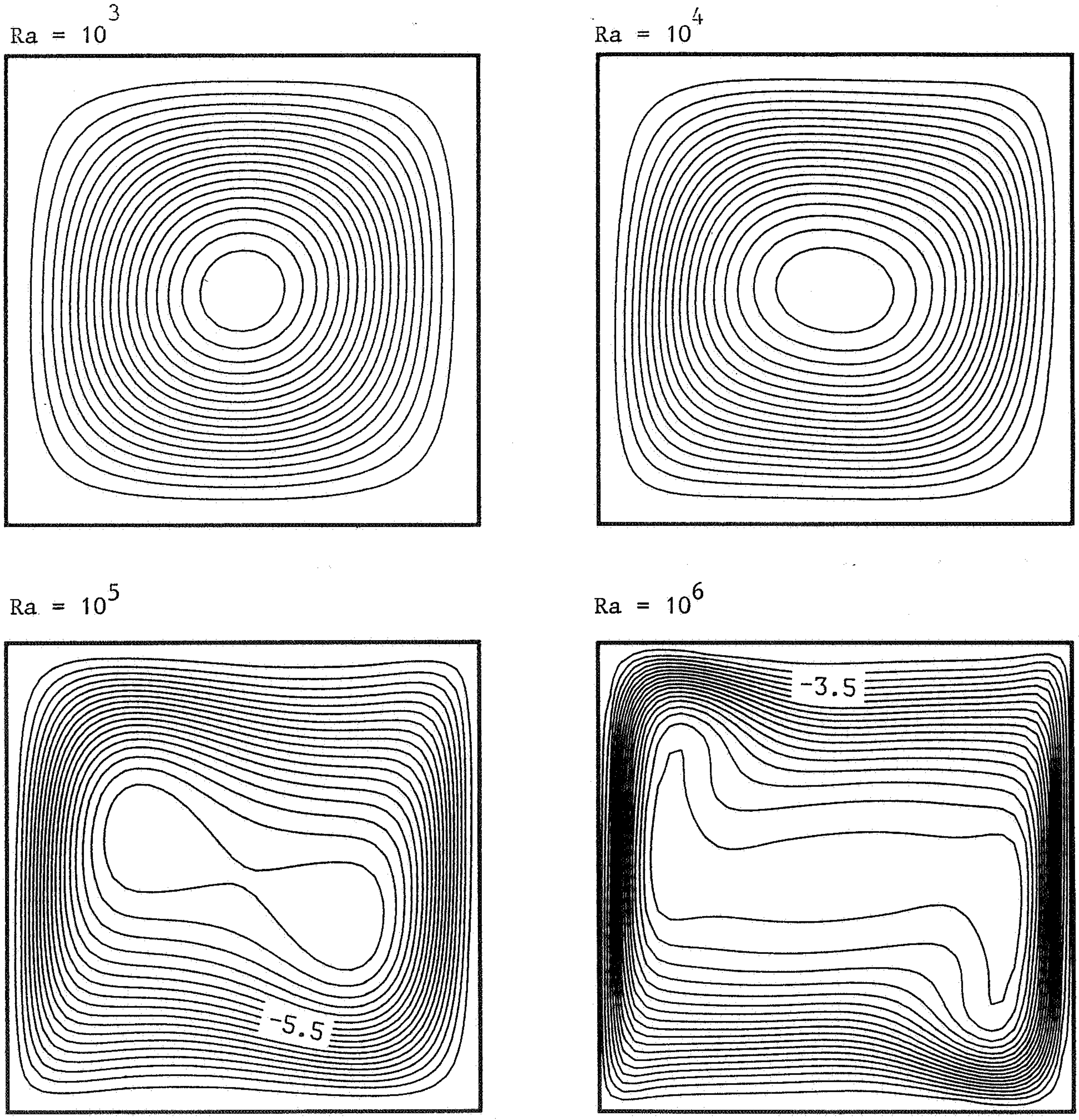

Fig.3. Streamline patterns for $Ra = 10^3, 10^4, 10^5$ and 10^6.

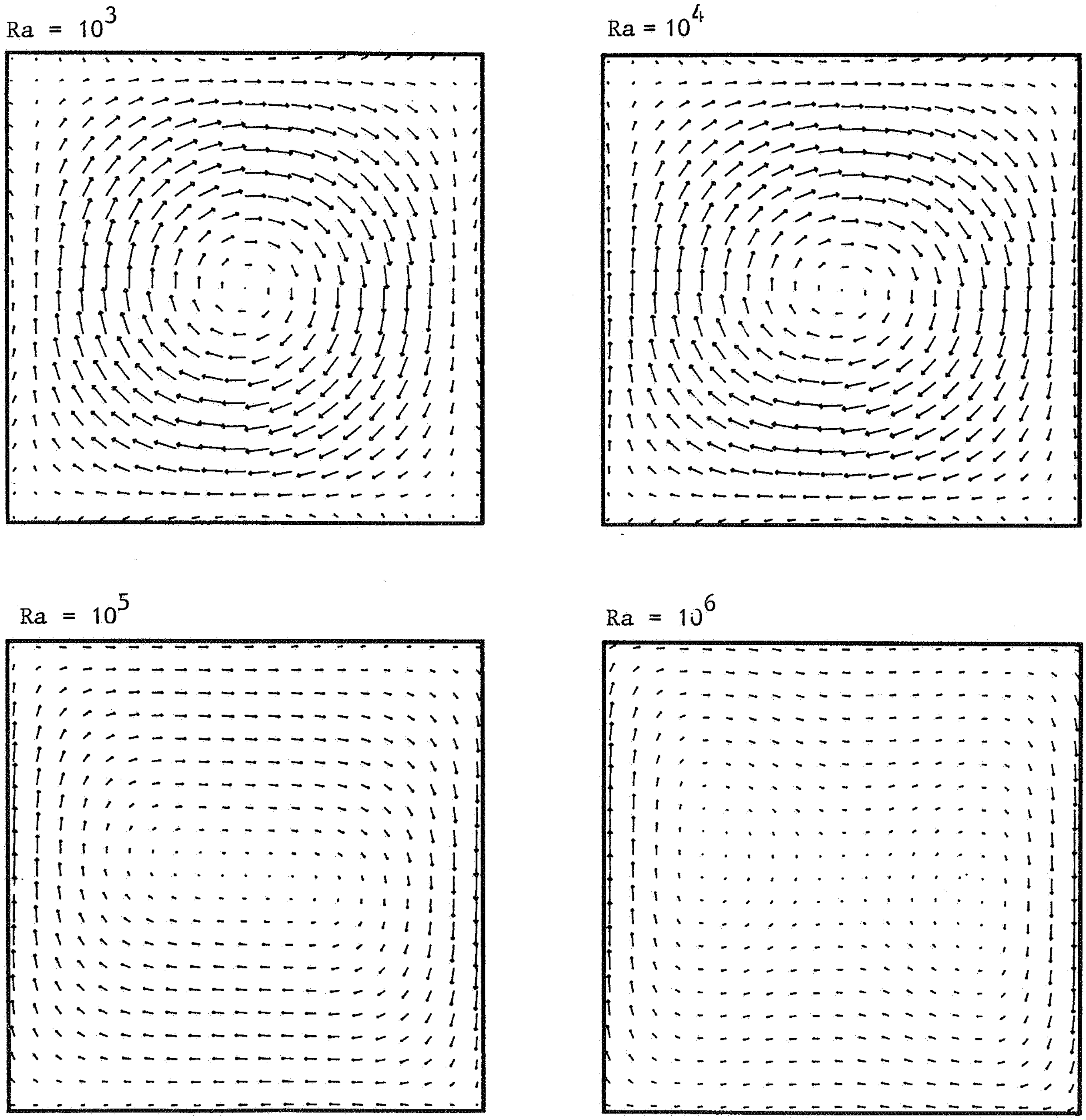

Fig.4. Velocity fields for $Ra = 10^3, 10^4, 10^5$ and 10^6.

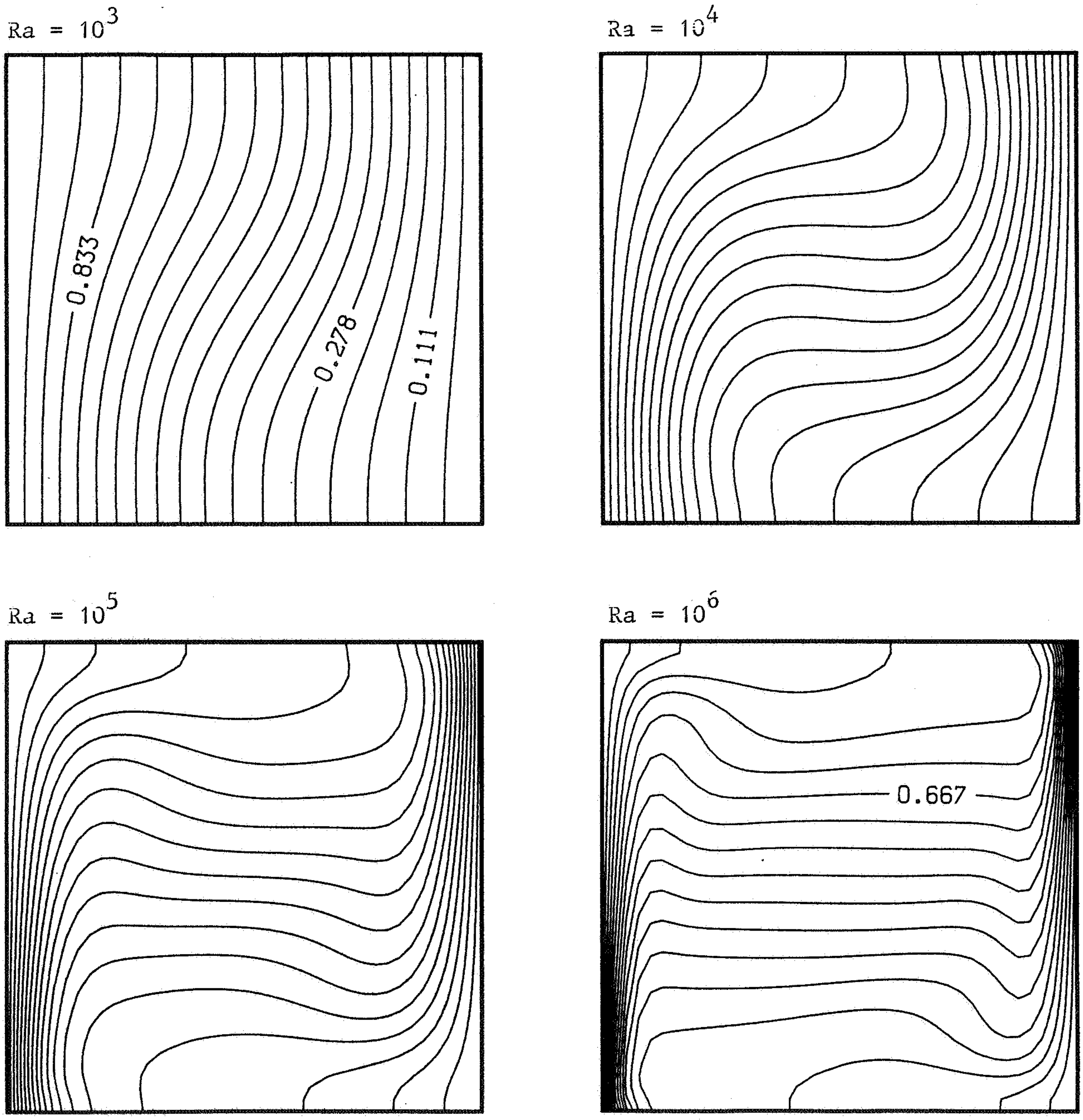

Fig.5. Isotherms for $Ra = 10^3, 10^4, 10^5$ and 10^6.

Table 1 shows, for the benchmark solution of de Vahl Davis and our solution, the following characteristic values

$U_{\max}$: the maximum horizontal velocity on the line $x=0.5$ (together with its location)

$V_{\max}$: the maximum vertical velocity on the line $y=0.5$ (together with its location)

ψ_{mid} : the absolute value of the stream function at the point (0.5,0.5)

$\psi_{\max}$: the maximum absolute value of the streamfunction (together with its location)

$\overline{Nu}$: the average Nusselt number throughout the cavity

$Nu_{\frac{1}{2}}$: the average Nusselt number on the line $x=0.5$

Nu_0 : the average Nusselt number on the line $x=0$

$Nu_{\max}$: the maximum absolute value of the local Nusselt number on the line $x=0$ (together with its location)

$Nu_{\min}$: the minimum absolute value of the local Nusselt number on the line $x=0$ (together with its location).

Concerning the characteristic values in Table 1 we note the following. The streamfunction $\psi(x,y)$ is computed from the Poisson equation

$$\nabla^2\psi = u_y - v_x, \tag{7.1}$$

subject to the boundary condition $\psi = const.\,(=0)$ on $\partial\Omega$ (the boundary is a streamline). The local Nusselt number is the local heat flux in the horizontal direction, and is given by [15]

$$Q(x,y) = uT - T_x. \tag{7.2}$$

Through any vertical line $x = x_0$, the total heat flux is given by

$$Nu_{x_0} = \int_0^1 Q(x_0,y)dy. \tag{7.3}$$

Nu_{x_0} is called the average Nusselt number on the line $x = x_0$. In a cavity with insulated horizontal boundaries, Nu_{x_0} must be independent of x_0. Finally, the average Nusselt number $\overline{Nu}$ throughout the cavity is given by

$$\overline{Nu} = \int_0^1 Nu_x dx. \tag{7.4}$$

The integrals in (7.3) and (7.4) have been computed using Simpson's rule, and the term T_x in (7.2) is evaluated using a second order finite difference approximation (also on the boundary $x=0$). de Vahl Davis computed the maximum (minimum) values from Table 1, and their locations, by numerical differentiation [15]. The maximum (minimum) values computed with the OEH-PC scheme are the largest (smallest) grid-point values, and their locations are the corresponding gridpoint coordinates.

From Table 1 we can draw the following conclusions. The 40×40 solution is an accurate solution, except for the characteristic values in the boundary layer $(v_{max}, Nu_0, Nu_{max}, Nu_{min})$ occurring for $Ra=10^6$. The 20×20 solution is still a fairly accurate solution for $Ra=10^3, 10^4$, however not for $Ra=10^5, 10^6$. We also note that the values in Table 1 become less accurate for increasing Ra. This is to be expected as the problem becomes increasingly difficult for increasing Ra due to the emerging of a boundary layer.

So far we did not specify the initial conditions. A possible choice is the zero-solution. In this case there exists a discontinuity between the initial- and boundary-conditions for the temperature (see (2.9)). This introduces high frequency components into the solution, and the exponential decay of these components is not properly simulated by many time-integration techniques, such as e.g. the

Solution after [15]

Ra	$U_{\max}$	$V_{\max}$	ψ_{mid}	$\psi_{\max}$	$\overline{Nu}$	$Nu_{\frac{1}{2}}$	Nu_0	$Nu_{\max}$	$Nu_{\min}$
	y	x		x/y				y	y
10^3	3.649 0.813	3.697 0.178	1.174		1.118	1.118	1.117	1.505 0.092	0.692 1.0
10^4	16.178 0.823	19.617 0.119	5.071		2.243	2.243	2.238	3.528 0.143	0.586 1.0
10^5	34.73 0.855	68.59 0.066	9.111	9.612 0.285/0.601	4.519	4.519	4.509	7.717 0.081	0.729 1.0
10^6	64.63 0.850	219.36 0.0379	16.32	16.750 0.151/0.547	8.800	8.799	8.817	17.925 0.0378	0.989 1.0

Solution on 40×40 grid

Ra	$U_{\max}$	$V_{\max}$	ψ_{mid}	$\psi_{\max}$	$\overline{Nu}$	$Nu_{\frac{1}{2}}$	Nu_0	$Nu_{\max}$	$Nu_{\min}$
	y	x		x/y				y	y
10^3	3.647 0.8125	3.691 0.1875	1.176	1.176 0.5/0.5	1.117	1.117	1.119	1.504 0.1	0.695 1.0
10^4	16.131 0.8125	19.601 0.1125	5.080	5.080 0.5/0.5	2.240	2.237	2.245	3.531 0.15	0.590 1.0
10^5	34.543 0.8625	68.130 0.0625	9.077	9.564 0.275/0.60	4.519	4.511	4.604	7.988 0.075	0.744 1.0
10^6	64.728 0.8625	214.643 0.0375	16.106	16.601 0.15/0.55	8.927	8.919	9.858	21.509 0.05	1.081 1.0

Solution on 20×20 grid

Ra	$U_{\max}$	$V_{\max}$	ψ_{mid}	$\psi_{\max}$	$\overline{Nu}$	$Nu_{\frac{1}{2}}$	Nu_0	$Nu_{\max}$	$Nu_{\min}$
	y	x		x/y				y	y
10^3	3.632 0.825	3.693 0.175	1.180	1.180 0.5/0.5	1.114	1.115	1.117	1.495 0.1	0.702 1.0
10^4	16.143 0.825	19.574 0.125	5.104	5.104 0.5/0.5	2.231	2.218	2.273	3.611 0.15	0.601 1.0
10^5	34.203 0.875	64.916 0.075	9.070	9.522 0.3/0.6	4.565	4.513	5.063	9.226 0.1	0.777 1.0
10^6	62.217 0.875	215.920 0.025	15.153	15.768 0.15/0.55	9.425	9.096	12.635	24.180 0.1	0.928 1.0

Table 1. Some characteristic values of the free convection flow.

Crank-Nicolson scheme. The OEH scheme also suffers from this drawback. Unless τ is very small, the OEH scheme produces false results for T (and hence for U and V), for small t-values. In [13] this is demonstrated for the linear, 1-dimensional convection-diffusion equation. One should be aware of this when simulating the transient behaviour of the free convection problem, but when computing the steady state solution this is of minor importance. In our numerical experiments, we found unreliable solutions for small t-values (especially for the computation of the 40×40 solution), unless we chose τ much smaller than needed for stability. Obviously, when dealing with steady state problems one can always adjust the initial condition to the boundary data. A more suitable initial condition is therefore the steady state solution for the next lower *Ra*-number. Then there exists no discontinuity in the free convection problem and the OEH scheme does not show the aforementioned pathological behaviour. The computational results presented in this section are based on this choice for the initial condition.

We conclude this section with a few remarks. The OEH-PC scheme has proven to be a feasible and accurate scheme for the free convection problem for *Ra*-numbers in the range $10^3 - 10^6$. However, the boundary layers occurring for $Ra = 10^6$ are not represented very accurately. In order to improve the accuracy in the boundary layer, the best thing to do is to refine the mesh near the vertical walls. We left out this possibility, because it would lead to a rather severe time step restriction for stability. In this connection we note that the OEH-PC scheme is probably not very suitable for the computation of the solution of the free convection problem for higher *Ra*-numbers ($Ra = 10^7, 10^8, ...$) because of this severe time step restriction (In (6.4) q_1 and q_2 become very large for large *Ra*). In that case it would be better to consider the application of unconditional stable methods (e.g. ADI). Of course, stepwise such a method will be more expensive than our OEH-PC scheme and it would also require more memory.

REFERENCES

[1] CHANDRASEKHAR, S., *Hydrodynamic and Hydromagnetic Stability,* The Clarendon Press, Oxford (1961).

[2] GOURLAY, A.R., Hopscotch: *a fast second-order partial differential equation solver,* J. Inst. Maths. Applics. 6, 375-390 (1970).

[3] GOURLAY, A.R. & G.R. MCGUIRE, *General hopscotch algorithm for the numerical solution of partial differential equations,* J. Inst. Maths. Applics. 7, 216-227 (1971).

[4] HEMKER, P.W. & P.M. DE ZEEUW, *Some implementations of multigrid linear system solvers.* In: *Multigrid methods for integral and differential equations,* The institute of mathematics and its applications conference series, D.J. Paddon & H. Holstein, eds. p.85-116, Oxford University Press, New York (1985).

[5] HINDMARSH, A.C., P.M. GRESHO & D.F. GRIFFITH, *The stability of explicit Euler time-integration for certain finite difference approximations of the multi-dimensional advection-diffusion equation,* Int. J. Num. Meth. Fluids 4, 853-897 (1984).

[6] VAN DER HOUWEN, P.J. & J.G. VERWER, *One-step splitting methods for semi-discrete parabolic equations,* Computing 22, 291-309 (1979).

[7] VAN KAN, J., *A second-order accurate pressure-correction scheme for viscous incompressible flow,* SIAM J. Sci. Stat. Comput. 7, 870-891 (1986).

[8] PEYRET, R. & T.D. TAYLOR, *Computational methods for fluid flow,* Springer-Verlag, New York (1983).

[9] LE QUERE, P. & T. ALZIARY DE ROQUEFORT, *Computation of natural convection in two-dimensional cavities with Chebyshev polynomials,* J. Comput. Phys. 57, 210-228 (1985).

[10] RICHTMEYER, R.D. & K.W. MORTON, *Difference methods for initial value problems,* Interscience Publishers, New York (1967).

[11] SONNEVELD, P., P. WESSELING & P.M. DE ZEEUW, *Multigrid and conjugate gradient methods as convergence acceleration techniques.* In: *Multigrid methods for integral and differential equations,* The institute of Mathematics and its applications conference series, D.J. Paddon & H. Holstein, eds. p.117-167, Oxford University Press, New York (1985).

[12] TEN THIJE BOONKKAMP, J.H.M., *The odd-even hopscotch pressure correction scheme for the incompressible Navier-Stokes equations,* report NM-R8615, Centre for Mathematics and Computer Science, Amsterdam, 1986 (submitted for publication elsewhere).

[13] TEN THIJE BOONKKAMP, J.H.M. *A note on the behaviour of the odd-even hopscotch scheme for a convection-diffusion problem with discontinuous initial data,* report NM-N86..., Centre for Mathematics and Computer Science, Amsterdam, 1986.

[14] TEN THIJE BOONKKAMP, J.H.M. & J.G. VERWER, *On the odd-even hopscotch scheme for the numerical integration of time-dependent partial differential equations,* report NM-R8513, Centre for Mathematics and Computer Science, Amsterdam, 1985. (to appear in Applied Numerical Mathematics, 1986).

[15] DE VAHL DAVIS, G., *Natural convection of air in a square cavity: A bench mark numerical solution,* Int. J. Num. Meth. Fluids 3, 249-264 (1983).