

Time-dependent simulation of viscoelastic flows at high Weissenberg number using the log-conformation representation

Raanan Fattal^a, Raz Kupferman^{b,*}

^a School of Computer Science and Engineering, The Hebrew University, Jerusalem 91904, Israel

^b Institute of Mathematics, The Hebrew University, Jerusalem 91904, Israel

Received 18 August 2004; received in revised form 12 December 2004; accepted 12 December 2004

Abstract

We present a second-order finite-difference scheme for viscoelastic flows based on a recent reformulation of the constitutive laws as equations for the matrix logarithm of the conformation tensor. We present a simple analysis that clarifies how the passage to logarithmic variables remedies the high-Weissenberg numerical instability. As a stringent test, we simulate an Oldroyd-B fluid in a lid-driven cavity. The scheme is found to be stable at large values of the Weissenberg number. These results support our claim that the high Weissenberg numerical instability may be overcome by the use of logarithmic variables. Remaining issues are rather concerned with accuracy, which degrades with insufficient resolution.

© 2005 Elsevier B.V. All rights reserved.

Keywords: High Weissenberg number problem; Matrix logarithm; Finite-differences; Oldroyd-B fluid; Lid-driven cavity

1. Introduction

The high-Weissenberg number problem (HWNP) has been the major obstacle in computational rheology since the early 1970s (see [1,2] for recent reviews on current challenges in computational rheology). It was diagnosed as a numerical phenomenon that causes all computations to break down at frustratingly low values of the Weissenberg number. Most of the work in computational rheology has focused on steady two-dimensional creeping flows using finite element methods (FEM) (see [3] for an early reference and [4] for a recent review). Then, the HWNP usually manifests as a lack of convergence of an iterative system. It is remarkable that although the HWNP has played such a central role for three decades, its origin has remained somewhat of a mystery. Even the fundamental question whether the HWNP is a purely numerical phenomenon, or rather a breakdown of the constitutive laws has remained, to some extent, under debate.

In this article, we identify the mechanism responsible for the HWNP. In summary, the stress experiences a combination of deformation and convection, which gives rise to steep exponential profiles. Even for moderate Weissenberg numbers, these spatial profiles are poorly approximated by numerical schemes, which are based (either explicitly as in FEM or implicitly as in finite-differences) on polynomial interpolation. The failure to properly balance the deformation with the convection yields a numerical instability. This is a fundamental instability present in all constitutive models that satisfy Oldroyd's frame invariance principle, and shared by all standard numerical methods.

Since the HWNP is due to the inadequacy of polynomial interpolation to approximate exponential profiles, two possible remedies come to mind: either to use exponential basis functions for the stress variables, or to make a change of variables into new variables that scale logarithmically with the stress. In either case, this requires the stress field $\tau(\mathbf{x}, t)$ to remain strictly positive, which cannot be guaranteed. A physical quantity, directly related to the stress, that preserves positivity is the conformation tensor, $\sigma(\mathbf{x}, t)$, which is symmetric positive-definite (SPD). As such, the

* Corresponding author.

E-mail addresses: raananf@cs.huji.ac.il (R. Fattal);
raz@math.huji.ac.il (R. Kupferman).

conformation tensor has a well-defined matrix-logarithm, which we denote by $\boldsymbol{\psi}(\mathbf{x}, t) = \log \boldsymbol{\sigma}(\mathbf{x}, t)$. We claim that the HWNP can be remedied if $\boldsymbol{\psi}(\mathbf{x}, t)$ is approximated, rather than $\boldsymbol{\tau}(\mathbf{x}, t)$ or $\boldsymbol{\sigma}(\mathbf{x}, t)$. To be more precise, the logarithmic transformation removes the instability that has caused numerical computations to blowup, or linear solvers not to converge; less catastrophic effects, like the spurious instabilities observed in the simulation of viscometric flows at high Weissenberg numbers [5,6] seem to be caused by a different mechanism.

In [7] we reformulated a class of differential constitutive models as equations for $\boldsymbol{\psi}(\mathbf{x}, t)$. We call the transformed system the log-conformation representation (LCR). The work [7] was supplemented with preliminary numerical results exhibiting stability at parameter regimes far beyond the breakdown threshold reported in the literature. The fact that the HWNP is only due to a poor treatment of the convection is further supported in [8], where we develop a second-order finite-difference scheme for $\boldsymbol{\sigma}(\mathbf{x}, t)$, but use a logarithmic change of variables back-and-forth to implement the convection. This alternative method was tested for planar contraction flow, and was found stable for very high Weissenberg numbers. Preliminary results indicate that the same stabilizing effect occurs with FEM as well [9].

In this paper, we present a second-order finite-difference scheme based on the log-conformation representation of the constitutive laws. The scheme is applicable to a large variety of differential constitutive laws in two and three dimensions. We implement our method for creeping flow of an Oldroyd-B fluid in a lid-driven cavity. While this system is not normally used as a benchmark, it is known as a very stringent test problem [10].

We next introduce notations. We have already defined the fields $\boldsymbol{\tau}(\mathbf{x}, t)$, $\boldsymbol{\sigma}(\mathbf{x}, t)$, and $\boldsymbol{\psi}(\mathbf{x}, t)$. The Eulerian velocity field is denoted by $\mathbf{u}(\mathbf{x}, t)$ and the pressure field by $p(\mathbf{x}, t)$. For creeping flows, the momentum balance equation is

$$-\nabla p + \nu_s \nabla^2 \mathbf{u} + \nu_p \nabla \cdot \boldsymbol{\tau} = 0, \quad (1.1)$$

supplemented with the continuity equation,

$$\nabla \cdot \mathbf{u} = 0.$$

Here ν_s and ν_p are the respective solvent and polymer viscosities, and the stress tensor $\boldsymbol{\tau}(\mathbf{x}, t)$ is assumed of the form

$$\boldsymbol{\tau} = \frac{g(\boldsymbol{\sigma})}{We} (\boldsymbol{\sigma} - \mathbf{I}),$$

where We is the Weissenberg number, and $g(\boldsymbol{\sigma})$ is a scalar-valued function, which only depends on the invariants of $\boldsymbol{\sigma}$. The constitutive equation, written as an evolution equation for the conformation tensor, is

$$\frac{\partial \boldsymbol{\sigma}}{\partial t} + (\mathbf{u} \cdot \nabla) \boldsymbol{\sigma} - (\nabla \mathbf{u}) \boldsymbol{\sigma} - \boldsymbol{\sigma} (\nabla \mathbf{u})^T = \frac{g(\boldsymbol{\sigma})}{We} P(\boldsymbol{\sigma}), \quad (1.2)$$

where $P(z)$ is a polynomial. For an Oldroyd-B fluid $g(\boldsymbol{\sigma}) = 1$ and $P(z) = 1 - z$.

The structure of this paper is as follows: In Section 2 we analyze the high-Weissenberg number instability, which can be mimicked by a simple one-dimensional linear toy model. In Section 3 we review the log-conformation representation of the constitutive laws. In Section 4 we present our numerical scheme, which is based on a two-step backward differentiation formula (BDF) for time stepping, and second-order spatial discretization. Numerical results are presented in Section 5. We present results for intermediate and high Weissenberg numbers, supported by numerical convergence analyses. Our scheme is found to be immune to high-Weissenberg numerical instabilities, although, as expected, accuracy problems emerge at insufficient resolution. A discussion follows in Section 6.

2. The high-Weissenberg instability

When one tries to solve, say, the upper-convected Maxwell (UCM) equations at moderately large values of We , using any standard provably-stable method, the numerical solution diverges in time. A more detailed examination of this blowup reveals that the conformation tensor grows unbounded exponentially fast. The only term in the UCM equation that can lead to exponential growth of the conformation tensor is the deformation term in the upper-convected derivative. For a sufficiently large deformation rate, this term cannot be balanced by the relaxation, and thus, the only term that can balance this growth is the convection; there is no blowup because the stretched fluid elements are eventually convected away from the stretching region.

Much insight into the origin of the high-Weissenberg instability can be gained by performing a sequence of simplifications of the original model. First, it can be verified that the instability persists even if the constitutive equation is decoupled from the momentum equation. Thus, the emergence of the instability can be studied in a much simpler setting, where the velocity field is kept fixed. Second, we observed that the instability is still present if the constitutive tensor equation is replaced by a scalar equation of similar type in one space dimension. This sequence of simplifications leads us to the following ‘‘cartoon model’’: a one-dimensional linear equation for $\phi = \phi(x, t)$, $x \in [0, 1]$,

$$\frac{\partial \phi}{\partial t} + a(x) \frac{\partial \phi}{\partial x} - b(x) \phi = -\frac{1}{We} \phi, \quad (2.1)$$

with $a(x), b(x) > 0$ and boundary condition $\phi(0, t) = 1$. This equation represents a field $\phi(x, t)$ that is convected to the right with velocity $a(x)$ and grows exponentially at a rate $b(x) - We^{-1}$. With reference to the UCM equation, $a(x)$ plays the role of the flow field $\mathbf{u}(\mathbf{x}, t)$, and $b(x)$ plays the role of the deformation rate $\nabla \mathbf{u}(\mathbf{x}, t)$.

The solution to (2.1) reaches a steady state,

$$\phi(x) = \int_0^x \exp\left(\frac{b(x') - We^{-1}}{a(x')}\right) dx'.$$

Suppose we solve (2.1) numerically using, for example, a first-order upwind scheme [11],

$$\phi_j^{n+1} = \phi_j^n - \frac{a_j \Delta t}{\Delta x} (\phi_j^n - \phi_{j-1}^n) + \Delta t \left(b_j - \frac{1}{We} \right) \phi_j^n,$$

where $a_j = a(x_j)$, $b_j = b(x_j)$. Rewriting this scheme as

$$\phi_j^{n+1} = \left[1 - \frac{a_j \Delta t}{\Delta x} + \Delta t \left(b_j - \frac{1}{We} \right) \right] \phi_j^n + \left[\frac{a_j \Delta t}{\Delta x} \right] \phi_{j-1}^n,$$

it is easily seen that the numerical solution diverges in time unless

$$1 - \frac{a_j \Delta t}{\Delta x} + \Delta t \left(b_j - \frac{1}{We} \right) \leq 1,$$

which implies that either $We < 1/b_j$ or

$$\Delta x \leq \frac{a_j}{b_j - We^{-1}}. \quad (2.2)$$

This condition has to hold at all mesh points j . This is a restriction on the *spatial mesh size*; it is not a CFL restriction on the time step. This stability criterion has the following interpretation: the spatial profile of $\phi(x, t)$ is exponential, therefore every convection scheme that is based on a polynomial reconstruction of fluxes underestimates the flux at the right edge of every computational cell. Thus, the rate at which the field ϕ is removed from the computational cell fails to balance its multiplicative growth rate, resulting in numerical blowup. This scenario remains unchanged if the first-order upwind scheme is replaced by a higher-order method; a higher-order scheme increases the critical mesh size by, at most, an order one factor. The use of implicit schemes does not help either.

To generalize the above analysis to viscoelastic flows, assume a fixed velocity field $\mathbf{u}(\mathbf{x})$ (as would be attained by a stable steady state), and consider the UCM equation,

$$\frac{\partial \sigma}{\partial t} + (\mathbf{u} \cdot \nabla) \sigma - (\nabla \mathbf{u}) \sigma - \sigma (\nabla \mathbf{u})^T = \frac{1}{We} (\mathbf{I} - \sigma).$$

This is a linear tensor-valued hyperbolic equation, which can be solved by the method of characteristics. Doing so, one obtains a tensor-valued equation in one space dimension of the form

$$\frac{\partial \sigma}{\partial t} + u(x) \frac{\partial \sigma}{\partial x} - (\nabla \mathbf{u}) \sigma - \sigma (\nabla \mathbf{u})^T = \frac{1}{We} (\mathbf{I} - \sigma), \quad (2.3)$$

where x is the arclength along the characteristic and $u(x) = |\mathbf{u}(\mathbf{x})|$. In the absence of stagnation points, $u(x)$ has fixed sign, which without loss of generality is assumed to be positive.

Consider a first order upwind scheme for solving (2.3):

$$\begin{aligned} \sigma_j^{n+1} &= \frac{1}{2} \left[\mathbf{I} - \frac{u_j \Delta t}{\Delta x} \mathbf{I} + \Delta t \left(2 \nabla \mathbf{u}_j - \frac{\mathbf{I}}{We} \right) \right] \sigma_j^n \\ &+ \frac{1}{2} \sigma_j^n \left[\mathbf{I} - \frac{u_j \Delta t}{\Delta x} \mathbf{I} + \Delta t \left(2 \nabla \mathbf{u}_j - \frac{\mathbf{I}}{We} \right) \right]^T \\ &+ \left[\frac{u_j \Delta t}{\Delta x} \right] \sigma_{j-1}^n + \frac{\Delta t}{We} \mathbf{I}. \end{aligned}$$

This is a linear difference equation with a block lower-diagonal transition matrix. The solution remains bounded in time if the three eigenvalues of this matrix, which are

$$1 - \Delta t \left(\frac{u_j}{\Delta x} + \frac{1}{We} \right),$$

$$1 - \Delta t \left(\frac{u_j}{\Delta x} + \frac{1}{We} \right) \pm 2 \Delta t \sqrt{-\det(\nabla \mathbf{u}_j)},$$

$j = 1, 2, \dots$, are all inside the unit disc. Note that $\det(\nabla \mathbf{u})$ is negative unless the flow is strongly rotational. For $\det \nabla \mathbf{u} < 0$, we obtain a stability condition that generalizes (2.2):

$$\Delta x \leq \frac{|\mathbf{u}|}{2\sqrt{-\det(\nabla \mathbf{u})} - We^{-1}} \quad (2.4)$$

(as in the scalar model, there is no restriction on Δx if the denominator is negative). The stability condition (2.4) may be very restrictive when convection is weak (small numerator) and in the presence of large deformation rates (large denominator). Regions near stagnation points and strong deformation rates (e.g., near geometric singularities) are prone to such numerical instability. We emphasize that the issue is not a lack of convergence in the strict sense—even the first order upwind scheme does converge as $\Delta x, \Delta t \rightarrow 0$; it is rather an issue of stiffness due to sharp spatial gradients.

We revert our attention to the scalar Eq. (2.1). The restriction on the mesh size is removed at once by a change of variables $\psi = \log \phi$, in which case $\psi(x, t)$ satisfies the equation

$$\frac{\partial \psi}{\partial t} + a(x) \frac{\partial \psi}{\partial x} - b(x) = -\frac{1}{We}, \quad (2.5)$$

with boundary condition $\psi(0, t) = 0$. Now, even a first-order upwind scheme,

$$\psi_j^{n+1} = \psi_j^n - \frac{a_j \Delta t}{\Delta x} (\psi_j^n - \psi_{j-1}^n) + \Delta t \left(b_j - \frac{1}{We} \right), \quad (2.6)$$

no longer imposes practical restrictions on the size of Δx . While this stable behaviour may be attributed to the transformation of multiplicative growth into additive growth, the reason for stability should rather be attributed to the improved treatment of convection. To see this, exponentiate (2.6) to regain an equation for ϕ_j^n ,

$$\phi_j^{n+1} = (\phi_j^n)^{(1-a_j \Delta t / \Delta x)} (\phi_{j-1}^n)^{a_j \Delta t / \Delta x} e^{\Delta t (b_j - We^{-1})},$$

and expand the multiplicative source, $\exp(b \Delta t) \sim 1 + b \Delta t$, to get, to first order in space and time,

$$\begin{aligned} \phi_j^{n+1} &= (\phi_j^n)^{(1-a_j \Delta t / \Delta x)} (\phi_{j-1}^n)^{a_j \Delta t / \Delta x} \\ &+ \Delta t \left(b_j - \frac{1}{We} \right) \phi_j^n. \end{aligned} \quad (2.7)$$

The transformation back and forth to a logarithmic scale results in a convection scheme that uses geometrical weights rather than algebraic weights, as all standard schemes do. Although (2.7) involves multiplicative growth, it is subject to the much weaker stability constraint: $\Delta x \leq a / \log(b - We^{-1})$.

3. The log-conformation representation

In order for the logarithm of a second-rank tensor to exist, the tensor needs to be positive definite. This is the reason we choose to formulate the constitutive law (1.2) in terms of the conformation tensor, although the more popular form is in terms of the stress tensor. LCR replaces (1.2) by an equivalent equation for $\psi(\mathbf{x}, t) = \log \sigma(\mathbf{x}, t)$. (Recall that an SPD matrix A can always be diagonalized, $A = R \Lambda R^T$, and that $\log A = R \log \Lambda R^T$.)

Differential constitutive laws dictate the dynamics of the conformation tensor as a composition of convection, deformation and relaxation. When a tensor field is convected by an incompressible flow, any continuous function of this tensor satisfies the same convection equation. It is also a relatively easy task to rewrite the relaxation equation for $\sigma(\mathbf{x}, t)$ as a relaxation equation for its matrix-logarithm. The main effort is to transform the deformation terms. This is easy to do once the deformation has been decomposed as a composition of pure extension and pure rotation. For that we need the following decomposition rule, proven in [7]:

Proposition. *Let $\mathbf{u}(\mathbf{x})$ be a divergence-free velocity field and let $\sigma(\mathbf{x})$ be a symmetric positive-definite tensor field. Then the velocity gradient $\nabla \mathbf{u}$ can be decomposed as*

$$\nabla \mathbf{u} = \boldsymbol{\Omega} + \mathbf{B} + N \boldsymbol{\sigma}^{-1}, \quad (3.1)$$

where $\boldsymbol{\Omega}$ and N are anti-symmetric, and \mathbf{B} is symmetric, traceless, and commutes with σ .

Substituting the decomposition (3.1) into the constitutive law (1.2), the $N \boldsymbol{\sigma}^{-1}$ -term cancels by anti-symmetry, and thus

$$\frac{\partial \sigma}{\partial t} + (\mathbf{u} \cdot \nabla) \sigma - (\boldsymbol{\Omega} \sigma - \sigma \boldsymbol{\Omega}) - 2 \mathbf{B} \sigma = \frac{g(\sigma)}{We} P(\sigma). \quad (3.2)$$

The tensor \mathbf{B} generates a pure (area preserving) extension, as it commutes with σ , and $\boldsymbol{\Omega}$ generates a pure rotation. The passage to an equation for ψ is now straightforward (see [7] for details),

$$\begin{aligned} \frac{\partial \psi}{\partial t} + (\mathbf{u} \cdot \nabla) \psi - (\boldsymbol{\Omega} \psi - \psi \boldsymbol{\Omega}) - 2 \mathbf{B} \\ = \frac{g(e^\psi)}{We} e^{-\psi} P(e^\psi). \end{aligned} \quad (3.3)$$

Like in (2.5), the extensional component becomes additive as a result of the logarithmic transformation. Eq. (3.3) is the equation we are going to approximate.

Two-dimensional case: The decomposition (3.1) is particularly simple in two dimensions. If σ is proportional to the unit tensor then simply set $\mathbf{B} = \frac{1}{2}[(\nabla \mathbf{u}) + (\nabla \mathbf{u})^T]$ and $\boldsymbol{\Omega} = 0$. Otherwise, calculate the diagonalizing transformation:

$$\sigma = R \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix} R^T,$$

and set

$$\begin{pmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{pmatrix} = R^T (\nabla \mathbf{u}) R.$$

It is easily verified that

$$\begin{aligned} N &= R \begin{pmatrix} 0 & n \\ -n & 0 \end{pmatrix} R^T, & \mathbf{B} &= R \begin{pmatrix} m_{11} & 0 \\ 0 & m_{22} \end{pmatrix} R^T, \\ \boldsymbol{\Omega} &= R \begin{pmatrix} 0 & \omega \\ -\omega & 0 \end{pmatrix} R^T, \end{aligned}$$

with $n = (m_{12} + m_{21})/(\lambda_2^{-1} - \lambda_1^{-1})$, and $\omega = (\lambda_2 m_{12} + \lambda_1 m_{21})/(\lambda_2 - \lambda_1)$, satisfy the required properties.

4. The numerical scheme

4.1. Temporal discretization

Let $\mathbf{u}^n = (u^n, v^n)$ and ψ^n denote the numerical approximations to the fields \mathbf{u} , ψ at the discrete time t_n , with $\Delta t_n = t_{n+1} - t_n$. The temporal discretization of the system (3.3) is based on a two-step backward differentiation formula (BDF) with variable time step [12]. For a differential equation $y' = f(t, y)$, the two-step BDF is

$$\alpha_0 y_{n+1} + \alpha_1 y_n + \alpha_2 y_{n-1} = f(t_{n+1}, y_{n+1}),$$

where

$$\begin{aligned} \alpha_0 &= \frac{2 \Delta t_n + \Delta t_{n-1}}{\Delta t_n (\Delta t_n + \Delta t_{n-1})}, & \alpha_1 &= -\frac{\Delta t_n + \Delta t_{n-1}}{\Delta t_n \Delta t_{n-1}}, \\ \alpha_2 &= \frac{\Delta t_n}{\Delta t_{n-1} (\Delta t_n + \Delta t_{n-1})}. \end{aligned}$$

For fixed time steps these reduce to $\alpha_0 = 3/2 \Delta t$, $\alpha_1 = -2/\Delta t$, and $\alpha_2 = 1/2 \Delta t$. While this scheme has good stability properties, it is time consuming due to its implicitness. A workaround is to use an implicit setting only for the linear terms on the right-hand side, whereas non-linear terms are

discretized by a two-step interpolation formula:

$$f(t_{n+1}, y_{n+1}) \approx \beta_1 f(t_n, y_n) + \beta_2 f(t_{n-1}, y_{n-1}),$$

with

$$\beta_1 = \frac{\Delta t_n + \Delta t_{n-1}}{\Delta t_{n-1}}, \quad \beta_2 = -\frac{\Delta t_n}{\Delta t_{n-1}}.$$

For fixed time steps, $\beta_1 = 2$ and $\beta_2 = -1$. (A similar temporal discretization for Newtonian flows is used in [13].)

Specifically, the evolution of ψ^n is governed by the two-step difference equation:

$$\begin{aligned} \alpha_0 \psi^{n+1} + \alpha_1 \psi^n + \alpha_2 \psi^{n-1} &= \beta_1 N_c(\mathbf{u}^n, \psi^n) + \beta_2 N_c(\mathbf{u}^{n-1}, \psi^{n-1}) \\ &+ 2[\beta_1 \mathbf{B}(\nabla \mathbf{u}^n, \psi^n) + \beta_2 \mathbf{B}(\nabla \mathbf{u}^{n-1}, \psi^{n-1})] \\ &+ [\beta_1 \boldsymbol{\Omega}(\nabla \mathbf{u}^n, \psi^n) + \beta_2 \boldsymbol{\Omega}(\nabla \mathbf{u}^{n-1}, \psi^{n-1})] \psi^{n+1} \\ &+ \psi^{n+1} [\beta_1 \boldsymbol{\Omega}(\nabla \mathbf{u}^n, \psi^n) + \beta_2 \boldsymbol{\Omega}(\nabla \mathbf{u}^{n-1}, \psi^{n-1})]^T \\ &+ \beta_1 N_r(\psi^n) + \beta_2 N_r(\psi^{n-1}), \end{aligned} \quad (4.1)$$

where $\mathbf{B}(\nabla \mathbf{u}, \psi)$ and $\boldsymbol{\Omega}(\nabla \mathbf{u}, \psi)$ are the tensor fields that compose $\nabla \mathbf{u}$ in (3.1), and

$$N_c(\mathbf{u}, \psi) = -(\mathbf{u} \cdot \nabla) \psi, \quad N_r(\psi) = \frac{1}{We} [\exp(-\psi) - \mathbf{I}],$$

are the nonlinear convection and relaxation terms.

For creeping flow, only $\psi(\mathbf{x}, t)$ satisfies a dynamical evolution equation, whereas the velocity and pressure are determined, given $\psi(\mathbf{x}, t)$, by the solution of an elliptic system: at every time step t_n the velocity \mathbf{u}^n and pressure p^n are obtained by the solution of the Stokes system:

$$\begin{aligned} -\nabla p^n + \nu_s \nabla^2 \mathbf{u}^n + \frac{\nu_p}{We} \nabla \cdot [\exp(\psi^n) - \mathbf{I}] &= 0 \\ \nabla \cdot \mathbf{u}^n &= 0, \end{aligned} \quad (4.2)$$

subject to the impermeability and no-slip boundary conditions at solid walls. Inflow and outflow boundary conditions are treated as easily.

4.2. Spatial discretization

We divide the domain into $n_x \times n_y$ rectangular cells of size $\Delta x \times \Delta y$; for a unit square $n_x \Delta x = n_y \Delta y = 1$. The cell labelled (i, j) is centered at the point

$$(x_i, y_j) = \left((i + \frac{1}{2}) \Delta x, (j + \frac{1}{2}) \Delta y \right),$$

with $i = 0, 1, \dots, n_x - 1, j = 0, 1, \dots, n_y - 1$.

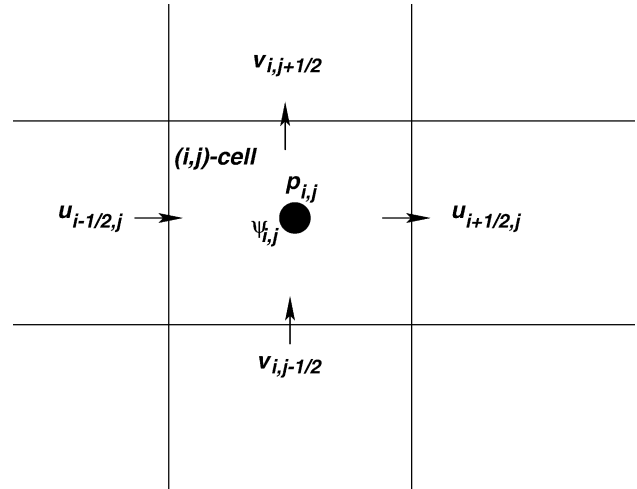


Fig. 1. The position of the variables in a computational cell.

The values of the log-conformation ψ and the pressure p are stored at cell centers, and are denoted by

$$\psi_{i,j} = \begin{pmatrix} (\psi_{xx})_{i,j} & (\psi_{xy})_{i,j} \\ (\psi_{xy})_{i,j} & (\psi_{yy})_{i,j} \end{pmatrix} \quad (4.3)$$

and $p_{i,j}$, respectively. For the velocity we use the so-called staggered, or Marker-and-Cell (MAC) discretization [14]; the discrete velocity variables are stored at the centers of cell edges, such that only the normal component is defined at each edge. Thus, the velocity component u is defined at the centers of left and right edges and the velocity component v is defined at the centers of top and bottom edges. The velocity variables associated with the cell (i, j) are denoted by $u_{i\pm 1/2, j}$ and $v_{i, j\pm 1/2}$, the indexing being self-explanatory. Note that the only variables defined at the domain boundaries are normal velocities, which are dictated by the boundary conditions. The geometry of the discrete variables is illustrated in Fig. 1.

Eq. (4.1) is a difference equation for the log-conformation ψ , and therefore needs to be prescribed at cell centers, i.e., at points (i, j) ; the same is also true for the incompressibility constraint in (4.2), which is a “pressure-like” equation. The momentum equations in (4.2) are equations for the velocity components u, v , and therefore need to be prescribed at the points $(i \pm 1/2, j)$ and $(i, j \pm 1/2)$, respectively. Values of u, v at the domain boundaries are determined by the boundary conditions, and no equation needs to be solved there.

We now specify one-by-one the spatial discretizations of the various terms in (4.1) and (4.2).

4.2.1. Velocity gradient

The velocity gradient tensor $\nabla \mathbf{u}$ needs to be computed at cell centers. The diagonal components $\partial u / \partial x$ and $\partial v / \partial y$ are readily obtained by compact stencils, which take advantage

of the staggering,

$$\left(\frac{\partial u}{\partial x}\right)_{i,j} = \frac{u_{i+1/2,j} - u_{i-1/2,j}}{\Delta x},$$

$$\left(\frac{\partial v}{\partial y}\right)_{i,j} = \frac{v_{i,j+1/2} - v_{i,j-1/2}}{\Delta y}.$$

These expressions hold everywhere in the domain, up to the boundary. Compact stencils cannot be constructed for the off-diagonal components $\partial u/\partial y$ and $\partial v/\partial x$, which are discretized by central differences,

$$\left(\frac{\partial u}{\partial y}\right)_{i,j} = \frac{\bar{u}_{i,j+1} - \bar{u}_{i,j-1}}{2\Delta y} \quad \left(\frac{\partial v}{\partial x}\right)_{i,j} = \frac{\bar{v}_{i+1,j} - \bar{v}_{i-1,j}}{2\Delta x},$$

where

$$\bar{u}_{i,j} = \frac{1}{2}(u_{i+1/2,j} + u_{i-1/2,j}) \quad \bar{v}_{i,j} = \frac{1}{2}(v_{i,j+1/2} + v_{i,j-1/2}).$$

In boundary cells, these stencils have to be modified to account for the boundary conditions; we evaluate these derivatives with one-sided stencils. With $\nabla \mathbf{u}$ at hand, the tensors $\mathbf{B}(\nabla \mathbf{u}, \boldsymbol{\psi})$ and $\boldsymbol{\Omega}(\nabla \mathbf{u}, \boldsymbol{\psi})$ can be computed.

4.2.2. Convection

We next turn to the convection term $(\mathbf{u} \cdot \nabla)\boldsymbol{\psi}$. We discretize it with the Kurganov–Tadmor (KT) scheme for hyperbolic conservation laws [15,16]. The KT scheme is a high-resolution central scheme, which assumes a semi-discrete limit (i.e., reduces to a “method of lines” as $\Delta t \rightarrow 0$ [11]). In a semi-discrete setting, the temporal discretization is independent of the spatial discretization; this is particularly convenient in conjunction with our two-step temporal discretization.

The KT scheme can be written in conservation form,

$$[(\mathbf{u} \cdot \nabla)\boldsymbol{\psi}]_{i,j} = \frac{H_{i+1/2,j}^x - H_{i-1/2,j}^x}{\Delta x} + \frac{H_{i,j+1/2}^y - H_{i,j-1/2}^y}{\Delta y},$$

where $H_{i\pm 1/2,j}^x$ and $H_{i,j\pm 1/2}^y$ are numerical fluxes of the form,

$$H_{i+1/2,j}^x = u_{i+1/2,j} \frac{\boldsymbol{\psi}_{i+1/2,j}^+ + \boldsymbol{\psi}_{i+1/2,j}^-}{2} - c |u_{i+1/2,j}| (\boldsymbol{\psi}_{i+1/2,j}^+ - \boldsymbol{\psi}_{i+1/2,j}^-)$$

$$H_{i+1/2,j}^y = v_{i,j+1/2} \frac{\boldsymbol{\psi}_{i,j+1/2}^+ + \boldsymbol{\psi}_{i,j+1/2}^-}{2} - c |v_{i,j+1/2}| (\boldsymbol{\psi}_{i,j+1/2}^+ - \boldsymbol{\psi}_{i,j+1/2}^-),$$

and c is a smoothing factor. The tensors $\boldsymbol{\psi}_{i+1/2,j}^+$, $\boldsymbol{\psi}_{i+1/2,j}^-$ are (discontinuous) evaluations of the log-conformation at cell edges, on the right and on the left, respectively. Their construction, in a manner that does not introduce spurious oscillations, can be done in several ways. We adopt the second-

order piecewise-linear reconstruction

$$\boldsymbol{\psi}_{i+1/2,j}^- = \boldsymbol{\psi}_{i,j} + \frac{1}{2} \min\text{mod}(\boldsymbol{\psi}_{i+1,j} - \boldsymbol{\psi}_{i,j}, \boldsymbol{\psi}_{i,j} - \boldsymbol{\psi}_{i-1,j})$$

$$\boldsymbol{\psi}_{i+1/2,j}^+ = \boldsymbol{\psi}_{i+1,j} - \frac{1}{2} \min\text{mod}(\boldsymbol{\psi}_{i+2,j} - \boldsymbol{\psi}_{i+1,j}, \boldsymbol{\psi}_{i+1,j} - \boldsymbol{\psi}_{i,j}),$$

where $\min\text{mod}(a, b) = \frac{1}{2}[\text{sgn}(a) + \text{sgn}(b)] \cdot \min(|a|, |b|)$ is the min-mod limiter (see e.g. [11] for an extensive reference on slope limiters and flux limiters). In principle, the optimal choice for the smoothing factor is $c = 1/2$; our calculations were found much better behaved for a value of c twice as large. The need for a larger amount of smoothing is discussed further below.

4.2.3. Newtonian viscosity

The Laplacian operator in (4.2) is discretized with the standard five-point stencil. For example,

$$(\nabla^2 u)_{i+1/2,j} = \frac{u_{i+3/2,j} - 2u_{i+1/2,j} + u_{i-1/2,j}}{\Delta x^2}$$

$$+ \frac{u_{i+1/2,j+1} - 2u_{i+1/2,j} + u_{i+1/2,j-1}}{\Delta y^2}.$$

This expression is valid at all points, except for those at a distance $\Delta y/2$ from the top or bottom lids, that is, except for when $j = 0$ and $j = n_y - 1$. There, boundary conditions are imposed by using “ghost cells”, and reflecting the interior values with respect to the boundary values (i.e., the value at the boundary is equal to the average of the values at the adjacent interior cell and the ghost cell). A similar modification is needed for the Laplacian of the vertical velocity components $(\nabla^2 v)_{i,j+1/2}$ for $i = 0$ and $i = n_x - 1$.

4.2.4. Pressure gradient

The pressure gradient is evaluated at cell edges. It is given by

$$(\nabla p)_{i+1/2,j} = \frac{p_{i+1,j} - p_{i,j}}{\Delta x}$$

$$(\nabla p)_{i,j+1/2} = \frac{p_{i,j+1} - p_{i,j}}{\Delta y}.$$

The pressure gradient does not need to be evaluated at the boundary because no equation is being solved there.

4.2.5. Velocity divergence

The velocity divergence associated with the cell (i, j) is

$$(\nabla \cdot \mathbf{u})_{i,j} = \frac{u_{i+1/2,j} - u_{i-1/2,j}}{\Delta x} + \frac{v_{i,j+1/2} - v_{i,j-1/2}}{\Delta y}.$$

Since the normal velocity is prescribed at the boundary, the discrete divergence is well defined at all cells.

4.2.6. Stress divergence

The divergence of the stress $\nabla \cdot \boldsymbol{\tau}$ is a “velocity-like” field, i.e., needs to be evaluated at cell edges. Take for example $(\nabla \cdot \boldsymbol{\tau})_{i+1/2,j}$: the contribution from τ_{xx} exploits the staggering,

$$\frac{(\tau_{xx})_{i+1,j} - (\tau_{xx})_{i,j}}{\Delta x}.$$

The contribution from τ_{xy} cannot be obtained by a compact stencil. As for the off-diagonal terms of the velocity gradient we use a wide-stencil central difference:

$$\frac{(\bar{\tau}_{xy})_{i+1/2,j+1} - (\bar{\tau}_{xy})_{i+1/2,j-1}}{2 \Delta y},$$

where

$$(\bar{\tau}_{xy})_{i+1/2,j} = \frac{1}{2}[(\tau_{xy})_{i,j} + (\tau_{xy})_{i+1,j}].$$

In the vicinity of boundaries, one-sided modifications are used.

4.3. Other technical issues

4.3.1. Choice of spatial discretization

The use of a staggered discretization for the velocity variables is very convenient for a stable treatment of the Stokes operator. In particular, it is immune to “checkerboard” pressure modes that infect wide-stencil discretizations. The staggered setting also provides, *in part*, a compact discretization of the velocity-stress interaction: the diagonal elements of the velocity gradient and the stress divergence exploit the staggering. This is not the case, however, with the off-diagonal elements, which are given by wide-stencil central differences. This implies that the rotational components of the system may be sensitive to numerical instabilities. A natural remedy would have been to store the off-diagonal elements of the stress tensor at cell corners. This cannot be done in our framework as the log-conformation tensor is an entity whose tensorial nature is essential. Other choices of spatial discretizations and their effect on stability will be studied elsewhere.

4.3.2. Linear solver

The Stokes Eq. (4.2) is a linear system for the velocity and pressure, given the stress, or the log-conformation. It can be solved by various methods. We used a multigrid solver. Multigrid solvers are iterative methods. Their central pillar is a standard iterative scheme, such as Jacobi or Gauss–Seidel. Iterative schemes are known to rapidly reduce high frequency modes of the error, but perform poorly on the lower frequency modes; that is, they rapidly smooth the error, which is why they are often called *smoothers*. The multigrid method reduces the lower frequency modes of the error by transferring

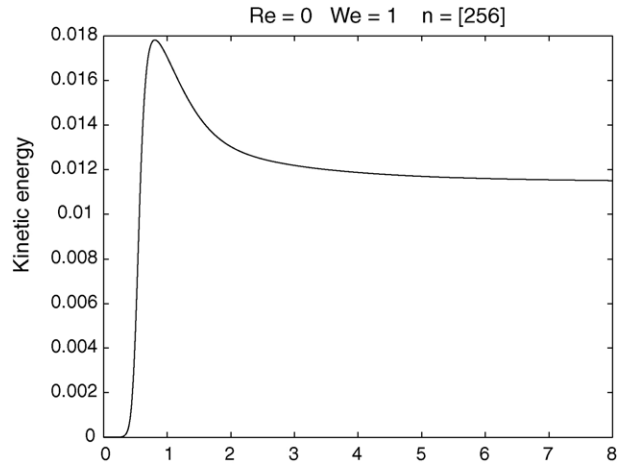


Fig. 2. The L^2 norm of the velocity (the “kinetic energy”) as function of time for $We = 1.0$.

the residual equation to a lower resolution grid and interpolating the obtained low-resolution solution back onto the original grid; this process is repeated recursively. The transfer back and forth between two levels of discretizations requires the definition of *restriction* (fine-to-coarse) and *prolongation* (coarse-to-fine) operators. The solution of the Stokes equations by a multigrid method has received much attention in the literature. A comprehensive description and numerous references may be found in [17].

4.3.3. Time step selection

The constraint on the time step comes from the hyperbolic CFL condition associated with the convection. For cavity flow the maximal velocity is always $u = 1$, thus, the CFL condition is

$$\Delta t \leq C \Delta x,$$

where C is the CFL constant. The KT scheme is stable for $C \leq 0.5$ [15].

5. Numerical results

We implemented the above scheme for an Oldroyd-B fluid in a lid-driven cavity. The fluid is confined in a unit square, $(x, y) \in [0, 1]^2$, bounded by solid walls, with the top boundary moving to the right. For Newtonian fluids, the discontinuity of the flow field at the upper corners causes the pressure to diverge, without affecting the well-posedness of the system. A viscoelastic fluid cannot sustain deformations at a stagnation point, therefore the motion of the lid needs to be regularized such that $\nabla \mathbf{u}$ vanishes at the corners. Also, to avoid errors resulting from an impulsive start, the motion of the lid was started smoothly. Specifically, the velocity profile

of the lid was taken to be

$$u_{\text{lid}}(x, t) = 8 \left[1 + \tanh 8 \left(t - \frac{1}{2} \right) \right] x^2(1 - x)^2.$$

For $t \gg \frac{1}{2}$, the lid velocity attains its maximum, $u = 1$, at the center, $x = 1/2$. In all our calculations we took $\nu_p = \nu_s$, i.e., equal contributions of solvent and polymeric viscosities. Thus, the only remaining parameter is the Weissenberg number.

5.1. $We = 1.0$

In Fig. 2 we plot the L^2 -norm of the velocity field—the “kinetic energy”—as function of time for $We = 1.0$ (properly speaking, a fluid without inertia does not have kinetic energy). The kinetic energy grows as the upper lid accelerates, reaches a maximum at the end of the acceleration, and then decreases toward a steady value as elastic energy builds up. At time $t = 8$ the solution seems to have approached a steady state.

The fields at the steady state are displayed in Fig. 3. The log-conformation exhibits steep gradients only in the vicinity of the upper lid; ψ_{xx} has a thin boundary layer along the lid, and all three components have large gradients near the upper corners. Note the asymmetry of the stream function, which would have had left–right symmetry for an inertia-less Newtonian fluid.

In Table 1 we display a mesh refinement analysis for the velocity component u and the log-conformation component ψ_{xx} . We ran simulations for 64×64 , 128×128 , and 256×256 point grids. If $\phi^{(N)}$ denotes the field ϕ computed with an $N \times N$ point grid, its relative error is estimated by comparison to the most refined computation,

$$e(\phi^{(N)}) = \frac{\|\phi^{(N)} - \phi^{(256)}\|_2}{\|\phi^{(256)}\|_2},$$

where $\|\cdot\|_2$ is the L^2 norm. The table shows second-order accuracy for short times, but the estimated convergence

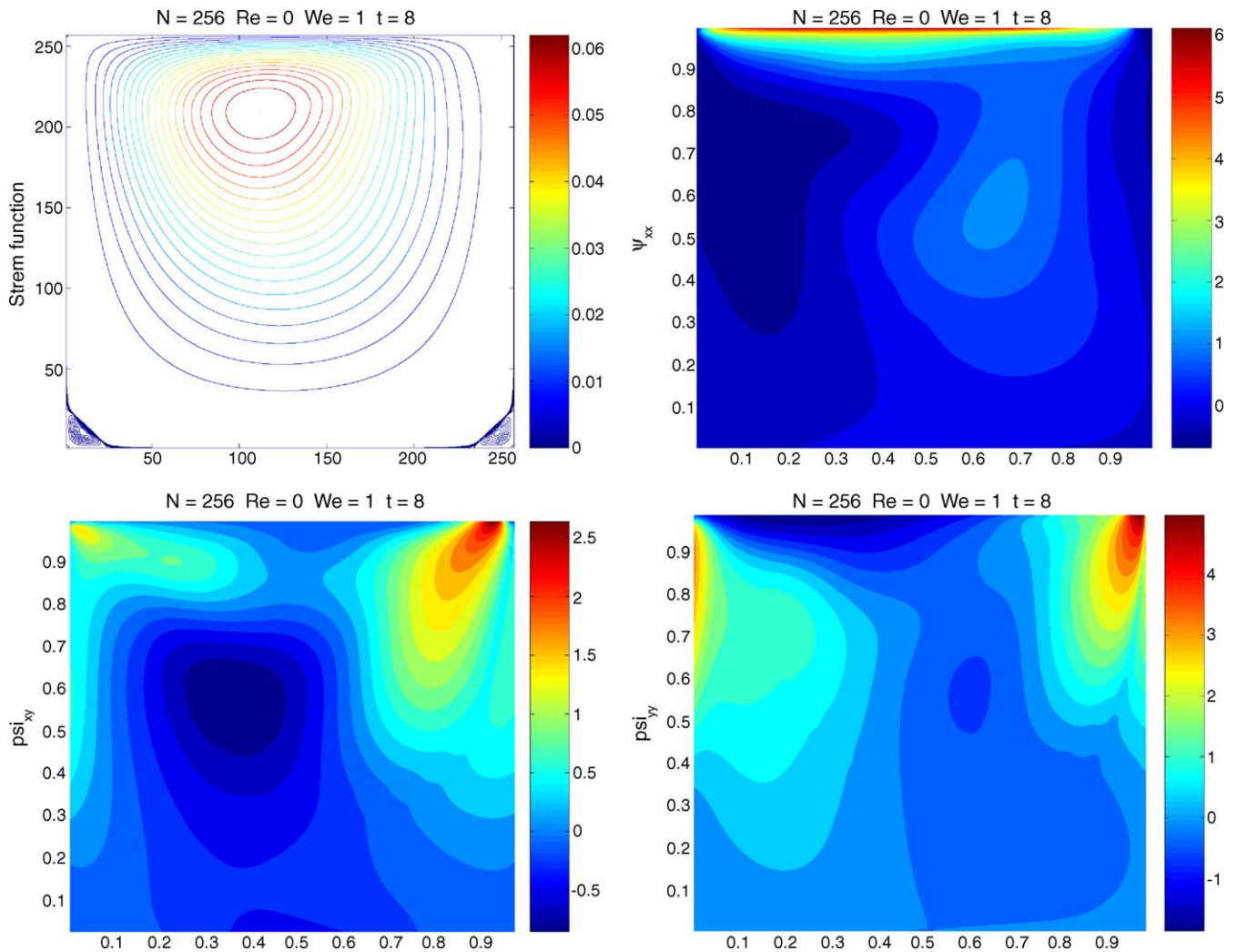


Fig. 3. (a) The stream function, (b)–(d) the three components of the log-conformation at time $t = 8$ for $We = 1.0$.

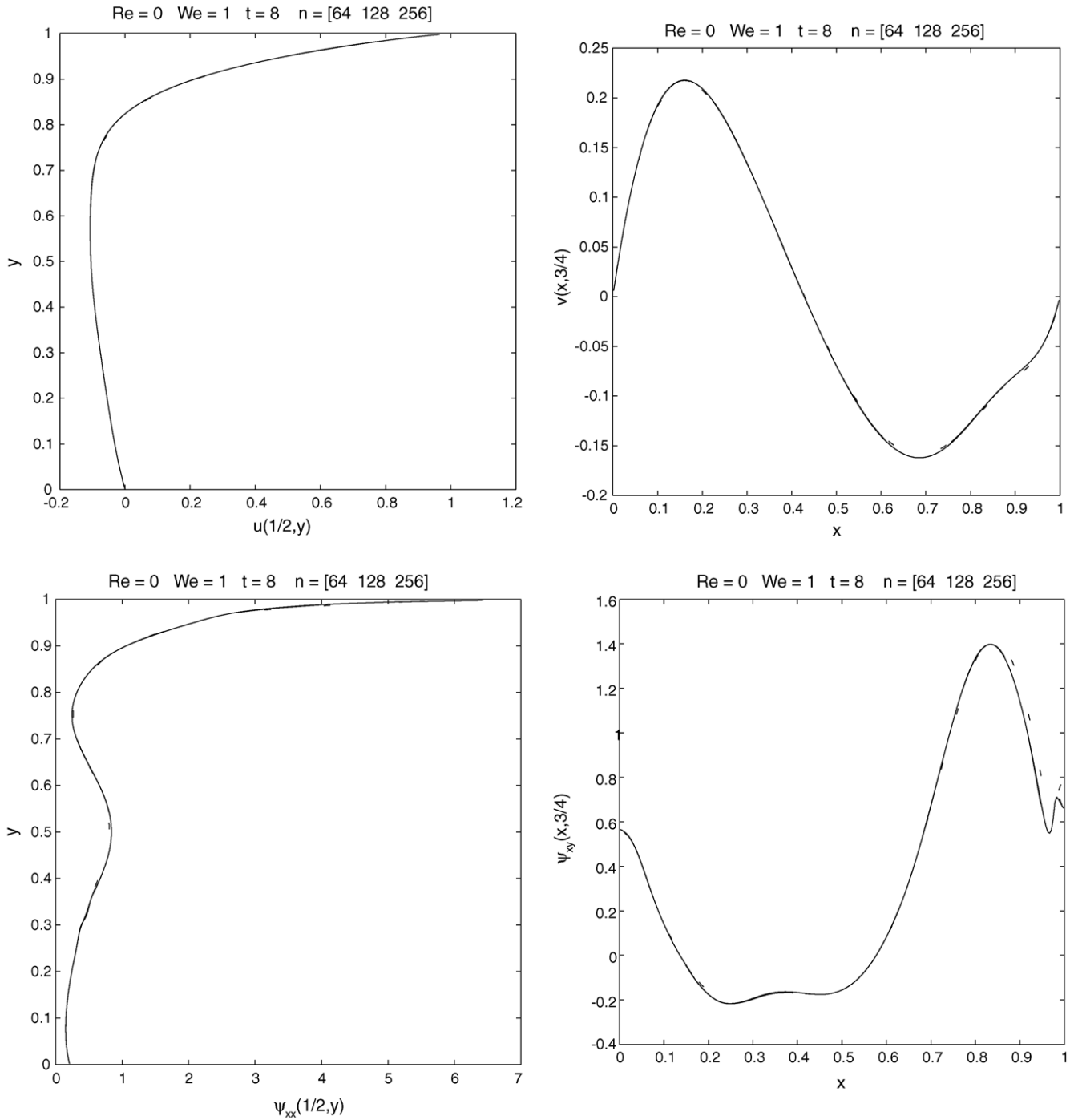


Fig. 4. Comparison between simulation results for a 64×64 point grid (dotted lines), a 128×128 point grid (dashed lines), and 256×256 point grid (solid lines) at time $t = 8$ for $We = 1.0$. The four graphs represent $u(1/2, y)$, $v(x, 3/4)$, $\psi_{xx}(1/2, y)$, and $\psi_{xy}(x, 3/4)$.

rate deteriorates with time until it reaches a value around 1.4 at the steady state. Note that both the velocity and the log-conformation attain a similar rate of convergence. An examination of the error reveals that it is mostly concentrated near the upper-right corner of the cavity. The fluid enters this region after having experienced contraction along the x -direction, and exits after having been rotated clockwise by 90° .

Table 1
Mesh refinement analysis for the fields u and ψ_{xx} at various times for $We = 1.0$

t	$e(u^{(64)})$	$e(u^{(128)})$	Rate	$e(\psi_{xx}^{(64)})$	$e(\psi_{xx}^{(128)})$	Rate
1.0	2.2×10^{-3}	4.0×10^{-4}	2.35	7.8×10^{-3}	1.6×10^{-3}	2.28
2.0	1.2×10^{-2}	2.8×10^{-3}	2.08	3.8×10^{-2}	1.0×10^{-2}	1.85
4.0	1.8×10^{-2}	6.0×10^{-3}	1.60	8.8×10^{-2}	2.9×10^{-2}	1.60
8.0	1.5×10^{-2}	5.4×10^{-3}	1.48	9.9×10^{-2}	3.8×10^{-2}	1.38

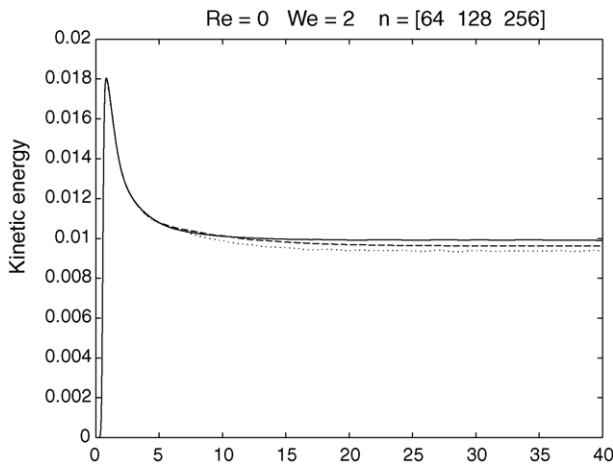


Fig. 5. The L^2 norm of the velocity as function of time for $We = 2.0$. The dotted line corresponds to a 64×64 point grid, the dashed line corresponds to a 128×128 point grid, and the solid lines corresponds to a 256×256 point grid.

In Fig. 4 we show selected profiles of the fields u , v , ψ_{xx} , and ψ_{xy} at the steady state. We compare results obtained from calculations at three different resolutions; differences for u , v , and ψ_{xx} are hardly seen at the scale of the graphs. The profiles of ψ_{xy} along the horizontal line $y = 3/4$ show that close to the right wall the lowest resolution graph has somewhat larger errors than the two higher resolution graphs.

To summarize, the calculations for $We = 1.0$ are convergent but the rate of convergence is somewhat lower than the expected second order. Errors are found to accumulate mostly in the upper-right part of the system.

5.2. $We = 2.0$

In Figs. 5 and 6 we show the evolution of the kinetic energy for $We = 2.0$. The three curves represent three mesh sizes. The numerical convergence analysis in Table 2 indicates that the results are still convergent, but errors are larger than for $We = 1.0$. In particular, there is a drastic drop in convergence rate at intermediate times. The fields' profiles displayed in

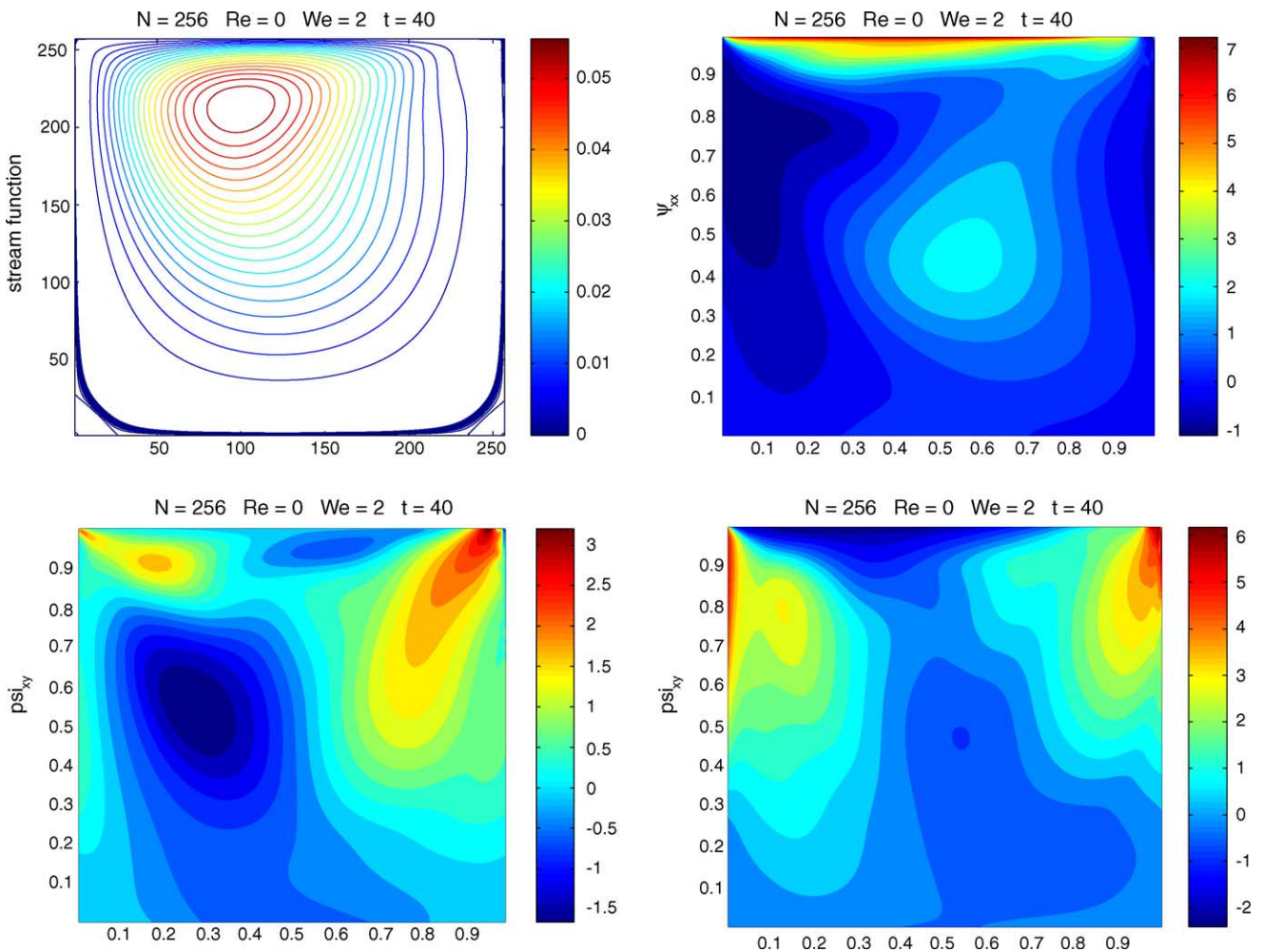


Fig. 6. (a) The stream function, (b)–(d) the three components of the log-conformation at time $t = 40$ for $We = 2.0$.

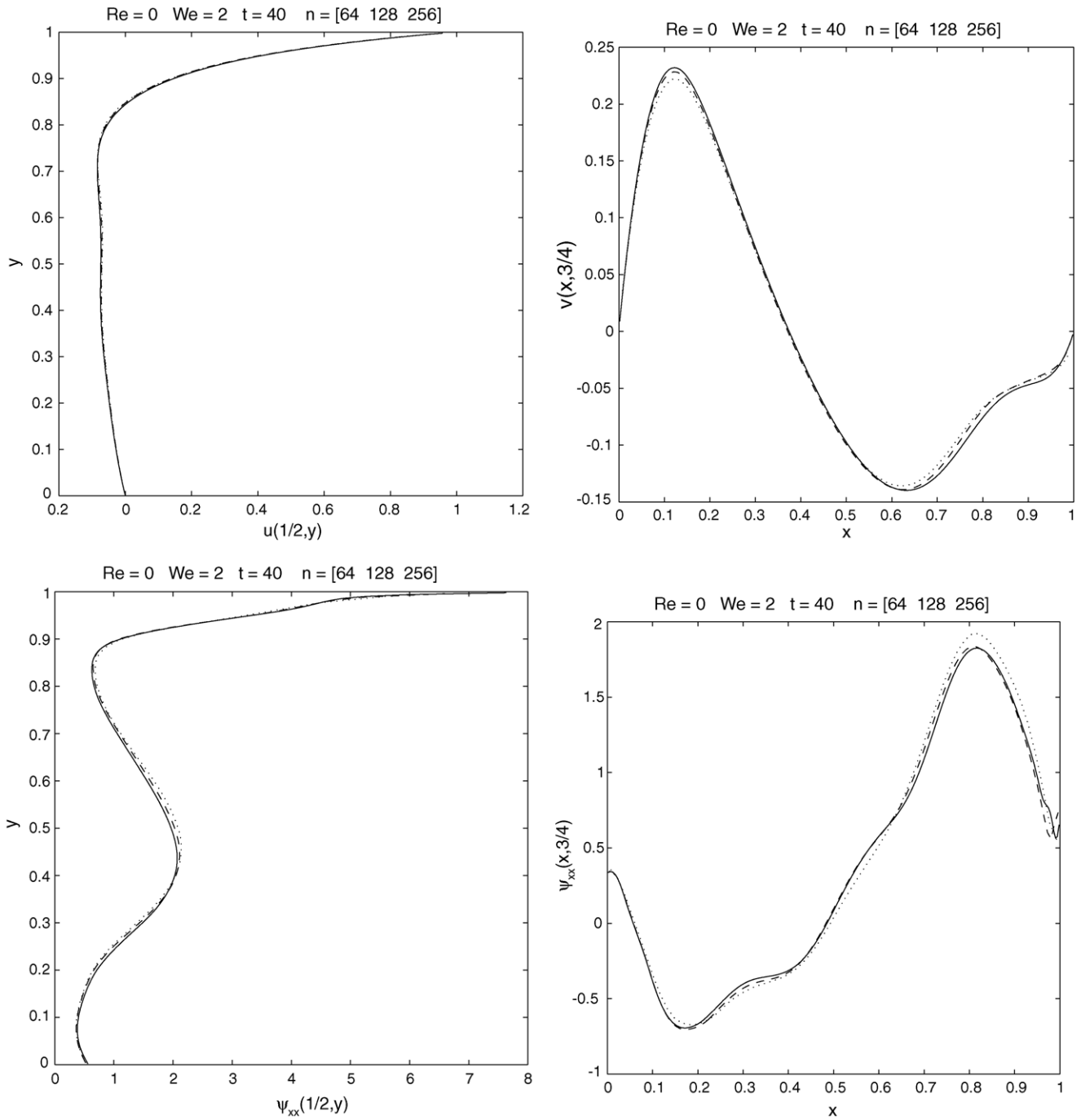


Fig. 7. Comparison between simulation results for a 64×64 point grid (dotted lines), a 128×128 point grid (dashed lines), and 256×256 point grid (solid lines) at time $t = 40$ for $We = 2.0$. The four graphs represent $u(1/2, y)$, $v(x, 3/4)$, $\psi_{xx}(1/2, y)$, and $\psi_{xy}(x, 3/4)$.

Fig. 7 confirm that the errors are indeed larger than for $We = 1.0$, but yet convergent.

5.3. $We = 3.0$

For $We = 3.0$ the tendencies observed in the passage from $We = 1.0$ to $We = 2.0$ are further amplified. In Fig. 8 we show the evolution of the kinetic energy for three mesh sizes.

The kinetic energy is found to be oscillatory. An examination of the flow field reveals that these oscillations are caused by vortices that are repeatedly created in the vicinity of the upper-right corner, and propagate downwards until being eventually damped out. For the 64×64 point grid (dotted line) the oscillations occur at intermediate times, but the system tends eventually to a steady state. The 128×128 and 256×256 point computations exhibit persistent oscillations,

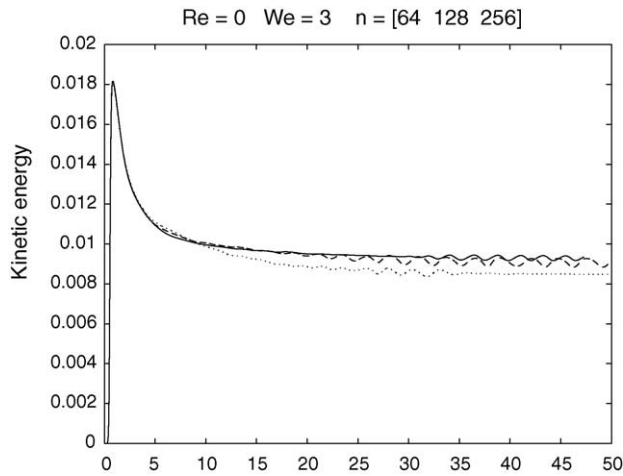


Fig. 8. The L^2 norm of the velocity as function of time for $We = 3.0$. The dotted line corresponds to a 64×64 point grid, the dashed line corresponds to a 128×128 point grid, and the solid lines corresponds to a 256×256 point grid.

Table 2

Mesh refinement analysis for the fields u and ψ_{xx} at various times for $We = 2.0$

t	$e(u^{(64)})$	$e(u^{(128)})$	Rate	$e(\psi_{xx}^{(64)})$	$e(\psi_{xx}^{(128)})$	Rate
4.0	2.5×10^{-2}	8.6×10^{-3}	1.55	1.0×10^{-1}	3.2×10^{-2}	1.69
8.0	1.8×10^{-2}	1.7×10^{-2}	0.10	1.2×10^{-1}	6.0×10^{-2}	1.05
16.0	3.3×10^{-2}	1.5×10^{-2}	1.11	1.0×10^{-1}	4.3×10^{-2}	1.22
32.0	3.7×10^{-2}	2.5×10^{-2}	0.56	1.1×10^{-1}	4.4×10^{-2}	1.27

Table 3

Mesh refinement analysis for the fields u and ψ_{xx} at various times for $We = 3.0$

t	$e(u^{(64)})$	$e(u^{(128)})$	Rate	$e(\psi_{xx}^{(64)})$	$e(\psi_{xx}^{(128)})$	Rate
5.0	4.7×10^{-2}	2.2×10^{-2}	1.08	1.3×10^{-1}	5.0×10^{-2}	1.38
10.0	3.6×10^{-2}	1.9×10^{-2}	0.87	1.4×10^{-1}	6.3×10^{-2}	1.14
20.0	5.3×10^{-2}	1.3×10^{-2}	1.94	1.0×10^{-1}	4.2×10^{-2}	1.33
40.0	5.5×10^{-2}	2.6×10^{-2}	1.50	1.3×10^{-1}	4.4×10^{-2}	1.56

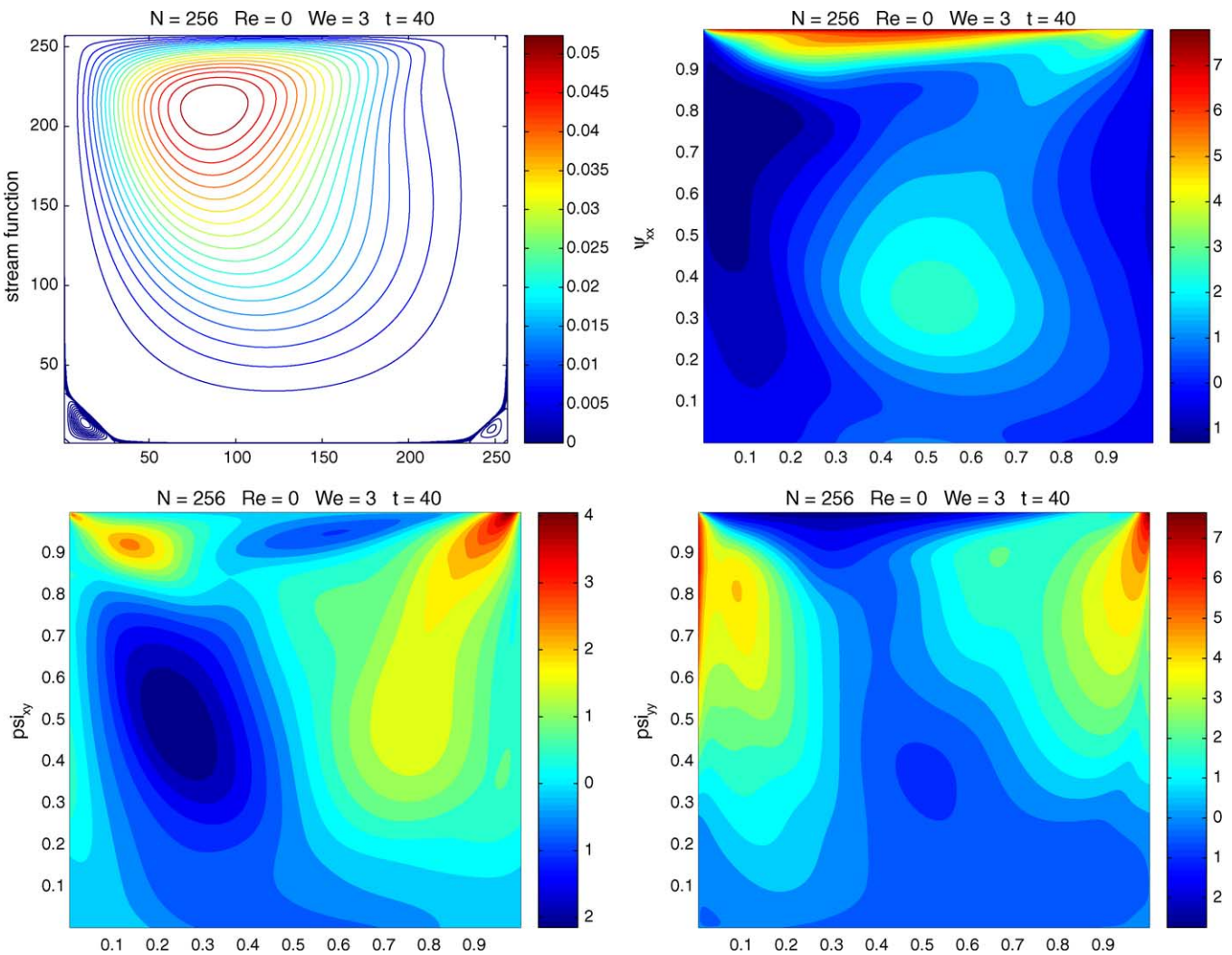


Fig. 9. (a) The stream function, (b)–(d) the three components of the log-conformation at time $t = 40$ for $We = 3.0$.

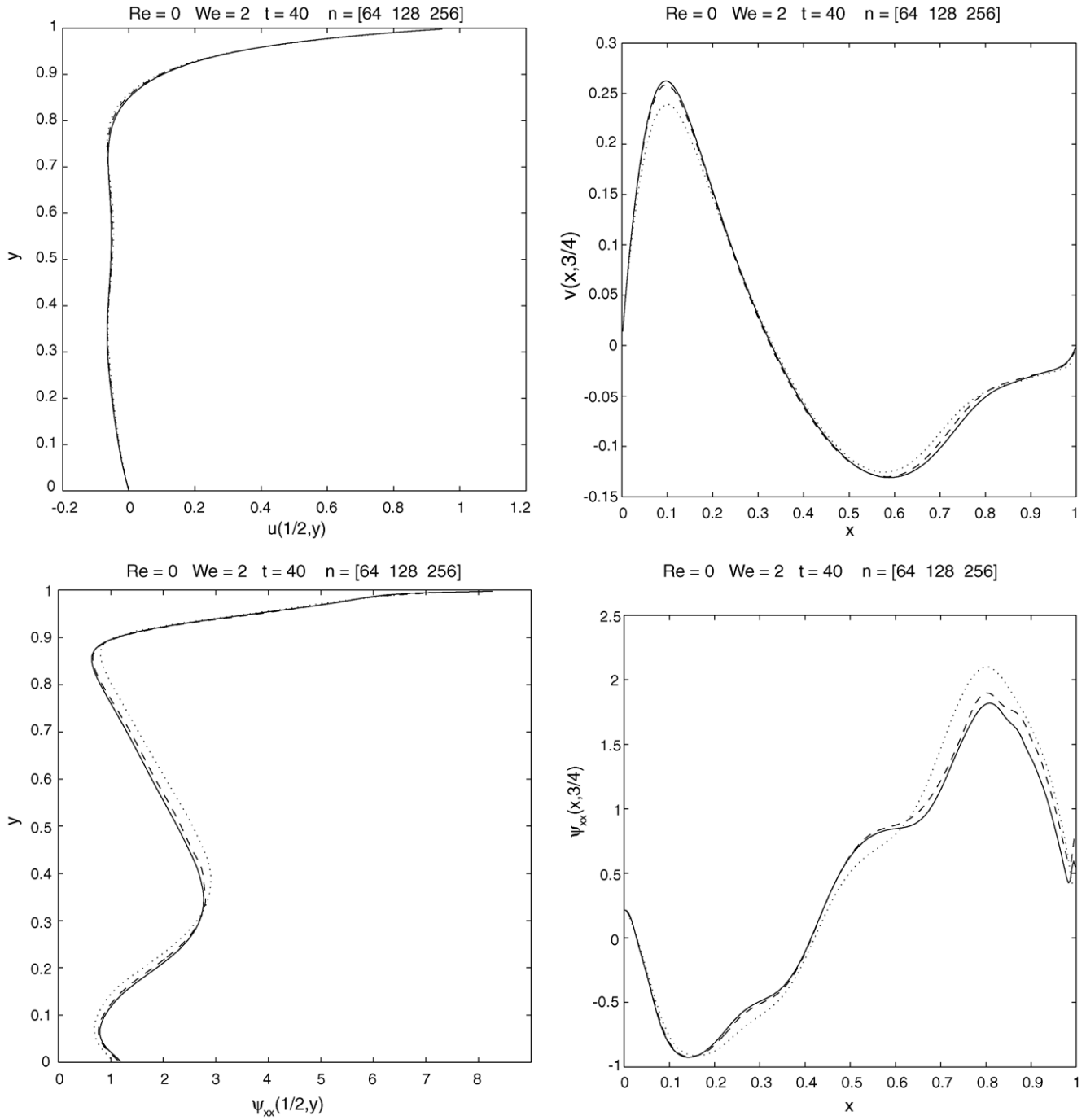


Fig. 10. Comparison between simulation results for a 64×64 point grid (dotted lines), a 128×128 point grid (dashed lines), and 256×256 point grid (solid lines) at time $t = 40$ for $We = 3.0$. The four graphs represent $u(1/2, y)$, $v(x, 3/4)$, $\psi_{xx}(1/2, y)$, and $\psi_{xy}(x, 3/4)$.

although the oscillations obtained at higher resolution have somewhat lower amplitude. The state of the system at time $t = 40$ is displayed in Fig. 9. The mesh refinement analysis in Table 3 shows convergence; errors are larger although the estimated convergence rate is higher than for $We = 2.0$. Larger errors are also apparent in the profiles shown in Fig. 10, notably in the upper-right part of the system.

5.4. Higher We

At even higher values of the Weissenberg number, the numerical solution exhibits stronger oscillations, and we can no longer claim for convergence (for example, we show in Fig. 11 the evolution of the kinetic energy for $We = 5.0$). On the other hand, the calculations are perfectly stable, showing

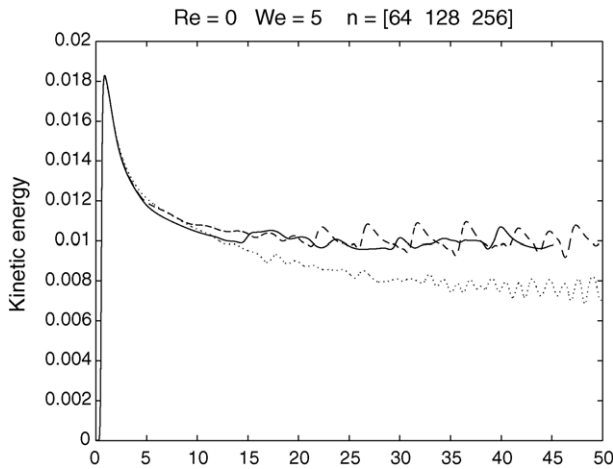


Fig. 11. The L^2 norm of the velocity as function of time for $We = 5.0$. The dotted line corresponds to a 64×64 point grid, the dashed line corresponds to a 128×128 point grid, and the solid lines corresponds to a 256×256 point grid.

that the passage to logarithmic variables does indeed remedy the HWNP instability.

6. Discussion

- (1) Is the HWNP solved? Our claim is that we have elucidated the high Weissenberg number *instability*. Our numerical experiments indicate that it is now possible to perform stable simulations at very large values of the Weissenberg number. (For comparison, computations with a standard finite-difference scheme that does not use logarithmic transformations reach a steady state only for $We < 0.5$.) Yet, as one should expect, the change of variables does not guarantee that *accurate* computations can be performed at arbitrarily high We . The situation can be compared with classical CFD, where one can perform stable calculations at arbitrarily large Reynolds numbers, but accuracy is lost when the resolution becomes insufficient. The analogy is in fact most appropriate given the recent identification of so-called elastic turbulence [18].
- (2) While the present paper describes an implementation of the LCR approach for an Oldroyd-B fluid, the LCR approach is applicable to a large class of differential constitutive models, and so is our numerical scheme. The method can be generalized to three dimensions, nonlinear constitutive models, and different systems of coordinates. Our scheme is also easily augmented to fourth-order accuracy (a fourth-order scheme for Newtonian fluids that uses compact stencils is developed in [13]). The LCR approach is also easily implemented within the FEM framework [9], particle tracking methods, and various hybrid methods (e.g., Brownian configuration fields [19,20]). Note that while Lagrangian-based methods are different in the sense that advection is not represented by a partial differential operator, and therefore its imple-

mentation does not involve the approximation of spatial derivatives. On the other hand, Lagrangian-based methods require either the insertion and removal of stress elements [21], or, a re-meshing stage in hybrid methods [22]. It is at this stage that interpolation is introduced, and like for Eulerian-based methods, the use of polynomial interpolation causes stress amplification in areas of high deformation.

- (3) Our computational method can also be applied to inertial flows. There, the main computational difficulty is the need to use small time steps to satisfy the CFL condition imposed by the elastic shear waves. This is a serious limitation at low Reynolds numbers, $Re \ll 1$, as the characteristic speeds scale like $Re^{-1/2}$.
- (4) Eq. (2.4) provides a quantitative criterion for when a method, that does not use matrix logarithms, is expected to lead to numerical blowup. Different schemes alter the criterion (2.4) by at most an order one prefactor. It would be of interest to re-examine past results, and verify whether the limiting Weissenberg number can indeed be related to such a stability criterion. Moreover, past ambiguities for whether increased resolution increases or reduces the maximum attainable Weissenberg number can be understood in light of (2.4). While increased resolution seems to be a stabilizing factor, it may cause the numerical estimate of the velocity gradient to increase, thus being destabilizing.
- (5) Having elucidated the fundamental HWNP instability, there remain problems of accuracy. As our results show, large errors and (possibly) spurious oscillations are generated in regions of large stress and strong rotations. The instability that results from under-resolution may be understood as a “checkerboard instability”, caused by the wide stencils employed in the calculation of the off-diagonal elements of the velocity gradient, and the stress divergence. As pointed out above, it would have been more natural to associate the off-diagonal elements of tensor-valued fields with cell corners, rather than cell centers (i.e., use a staggered setting for tensors as well). Such a splitting between diagonal and off-diagonal elements is problematic in a method that is heavily based on tensorial operations, such as matrix exponentiation and diagonalization.

We believe that the loss of accuracy and the generation of spurious oscillations has the same origins as the (spurious) unstable modes observed in numerical solutions of plane Couette flow [5]. In [6] various numerical schemes were benchmarked for Couette flow. In particular, the authors report that “a factor that influences the behavior of the DEVSS-G/SUPG method seems a proper choice for the adaptive viscosity function”. In our case too, the formation of vortices was found very sensitive to the amount of smoothing in the KT convection scheme. An understanding of this numerical artifact is of fundamental importance, and is left for future work.

(6) A similar passage to logarithmic variables may be of use in situations other than computational rheology. Generally, such an approach could be useful in any situation where a physical quantity that preserves positivity is simultaneously convected and amplified (e.g., reactive flows). The use of a logarithmic transformation is, by itself, not a novel idea in mathematics. A classical example is the wkb expansion [23], where a power series expansion is constructed for the logarithm of the sought solution. Another example is the Cole-Hopf transformation that turns the nonlinear viscous Burgers equation into a linear heat equation [24].

Acknowledgements

We are grateful to Frank Baaijens, Alexandre Chorin, Martien Hulsen, Roland Keunings, Randy LeVeque, and Eitan Tadmor for precious advice and to Irad Yavneh for guiding us with multigrid methods. We are also grateful to Morton Denn for continual support. Part of this research was carried out while the authors were visiting the Department of Applied Mathematics at Lawrence Berkeley National Laboratory. This research was funded in part by the Applied Mathematical Sciences subprogram of the Office of Energy Research of the US Department of Energy under Contract DE-AC03-76-SF00098.

References

- [1] R. Keunings, A survey of computational rheology, in: D.B. et al. (Ed.), *Proceedings of the 13th International Congress on Rheology*, vol. 1, British Society of Rheology, 2000.
- [2] K. Walters, M. Webster, The distinctive CFD challenges of computational rheology, *Int. J. Numer. Methods Fluids* 43 (2003) 577–596.
- [3] M. Crochet, A. Davies, K. Walters, *Numerical Simulation of Non-Newtonian Flow*, Elsevier, Amsterdam, 1984.
- [4] F. Baaijens, Mixed finite element methods for viscoelastic flow analysis: a review, *J. Non-Newton. Fluid Mech.* 79 (1998) 361–385.
- [5] R. Keiller, Numerical instability of time-dependent flows, *J. Non-Newton. Fluid Mech.* 43 (1992) 229–246.
- [6] A. Bogaerds, G. Peters, F. Baaijens, Temporal stability of low-order continuous and discontinuous mixed finite element techniques for viscoelastic fluid mechanics, *Proceedings of XIIIth International Congress on Rheology*, Cambridge, UK, 2000.
- [7] R. Fattal, R. Kupferman, Constitutive laws for the matrix-logarithm of the conformation tensor, *J. Non-Newton. Fluid Mech.* 123 (2004) 281–285.
- [8] R. Fattal, R. Kupferman, Time-dependent simulation of an Oldroyd-B fluid at high Reynolds and Weissenberg numbers: planar contraction flow, *J. Non-Newton. Fluid Mech.*, 2004, in preparation.
- [9] M. Hulsen, R. Fattal, R. Kupferman, Flow of viscoelastic fluids past a cylinder at high Weissenberg number: stabilized simulations using matrix logarithms, *J. Non-Newton. Fluid Mech.*, 2004, in press.
- [10] A. Grillet, B. Yang, B. Khomami, E. Shaqfeh, Modeling of viscoelastic lid driven cavity flow using finite element simulations, *J. Non-Newton. Fluid Mech.* 88 (1999) 99–131.
- [11] R. LeVeque, *Numerical Methods for Conservation Laws*, Birkhauser Verlag, 1992.
- [12] J. Lambert, *Numerical Methods for Ordinary Differential Equations*, John Wiley and Sons, 1991.
- [13] A. Brüger, J. Nilsson, W. Kress, A compact higher order finite difference method for the incompressible Navier-Stokes equations, *J. Sci. Comp.* 17 (2002) 551–560.
- [14] F. Harlow, J. Welch, Numerical calculation of time-dependent viscous incompressible flow of fluid with a free surface, *Phys. Fluids* 8 (1965) 2182.
- [15] A. Kurganov, E. Tadmor, New high-resolution central schemes for nonlinear conservation laws and convection–diffusion equations, *J. Comp. Phys.* 160 (2000) 214–282.
- [16] A. Kurganov, D. Levy, A third-order semi-discrete central scheme for conservation laws and convection–diffusion equations, *SIAM J. Sci. Comp.* 22 (2000) 1461–1488.
- [17] U. Trottenberg, C. Oosterlee, A. Schüller, *Multigrid Solution of Automatically Generated High Order Discretizations for the Biharmonic Equation*, Academic Press, 2001.
- [18] A. Groisman, V. Steinberg, Elastic turbulence in a polymer solution flow, *Nature* 405 (2000) 53.
- [19] M. Hulsen, A. van Heel, B. van den Brule, Simulation of viscoelastic flows using brownian configuration fields, *J. Non-Newton. Fluid Mech.* 70 (1997) 79–101.
- [20] R. Fattal, R. Kupferman, Logarithmic variable transformations for the Brownian configuration fields method, 2004, in preparation.
- [21] M. Laso, H. Ottinger, Calculation of viscoelastic flow using molecular models: the CONNFFESSIT approach, *J. Non-Newton. Fluid Mech.* 47 (1993) 1–20.
- [22] H. Rasmussen, O. Hassager, Simulation of transient viscoelastic flow, *J. Non-Newton. Fluid Mech.* 64 (1993) 289–305.
- [23] C. Bender, S. Orszag, *Advanced Mathematical Methods for Scientists and Engineers*, Springer-Verlag, New York, 1999.
- [24] G. Whitham, *Linear and Nonlinear Waves*, Wiley and Sons, 1974.