

## COMPUTATIONAL ANALYSIS OF A CONTRACTION RHEOMETER FOR THE GRADE-TWO FLUID MODEL

SARA POLLOCK<sup>✉1</sup> AND L. RIDGWAY SCOTT<sup>✉2</sup>

<sup>1</sup>Department of Mathematics, University of Florida, USA

<sup>2</sup>University of Chicago, USA

(Communicated by Annalisa Quaini)

**ABSTRACT.** We explore the possibility of simulating the grade-two fluid model in a geometry related to a contraction rheometer, and we provide details on several key aspects of the computation. We show how the results can be used to determine the viscosity  $\nu$  from experimental data. We also explore the identifiability of the grade-two parameters  $\alpha_1$  and  $\alpha_2$  from experimental data. In particular, as the flow rate varies, force data appears to be nearly the same for certain distinct pairs of values  $\alpha_1$  and  $\alpha_2$ ; however, we determine a regime for  $\alpha_1$  and  $\alpha_2$  for which the parameters may be identifiable with a contraction rheometer.

**1. Introduction.** A rheometer is a device that is used to determine physical properties of fluids such as viscosity, but also other properties for non-Newtonian fluids, via controlled experiments. To use a rheometer to characterize parameters used in computational models of fluids, it is necessary to have a way to convert data from the form produced by the rheometer into an estimate of model parameters. In ideal cases of hypothetical rheometers [20], this can be done with analytical solutions of the model equations. But in more realistic cases, it is necessary to solve the model equations numerically in the rheometer geometry, creating a mapping from model parameters to approximations of the rheometer data. Then one can attempt to invert this mapping to generate model parameters from experimental data. Here we explore the first step (the forward problem) for a particular model and a particular rheometer geometry. In a subsequent paper [18], we will examine the corresponding inverse problem.

The grade-two fluid model is the lowest-order member of a family of models proposed by Rivlin and Ericksen [7, 9]. In these models, the stress-strain relationship involves derivatives of the fluid velocity. The grade-two model involves two parameters in addition to the fluid viscosity and has been widely studied [6]. On the other hand, computational models have been limited so far by restrictions on the two parameters, the dimension, or boundary conditions [3, 4, 8, 9].

Recently, an algorithm was proposed [16] for both two and three dimensions that supports the use of general parameters and inflow boundary conditions. We use

---

2020 *Mathematics Subject Classification.* Primary: 76A05, 65N30; Secondary: 65N50.

*Key words and phrases.* Non-Newtonian fluid model, grade-two fluid, rheology, contraction rheometer, finite element approximation, Anderson acceleration.

The first author is supported by NSF grant DMS-2011519.

\*Corresponding author: Sara Pollock.

this algorithm to compute a force integral over the contraction region for the flow in a contracting duct. We compute the force integral over a range of parameters and carefully analyze the structure of the resulting data to determine a regime in which we may be able to identify parameters in the grade-two model.

We study the proposed computational method in detail. We point out another algorithm designed for more restrictive settings. The latter algorithm is able to find solutions for much larger parameter values, for constrained parameter values ( $\alpha_1 + \alpha_2 = 0$ ). To improve our solves and to extend the parameter regime where we are able to solve with the more general method, we consider an implementation of the extrapolation technique known as Anderson Acceleration (AA) [2], similar to that given in [14, 15].

The remainder of this paper is organized as follows. Sections 1.1-1.7 summarize the main results from [16], namely the new algorithm for the grade-two model and the theory describing its convergence; additionally, here we provide details on an Anderson accelerated version of the nonlinear iteration. Section 2 describes the contracting duct domain over which we will perform the force integral computations, and describes the main characteristics of the flow in this duct. Section 3 contains the main results of the paper. In this section we define the force integral and provide a study of the structure of the computed data with respect to each of its parameters in order to determine a regime in which this rheometer may be used to identify parameters in the grade-two model. Section 4 contains the computational details of the calculations performed in the preceding section focusing on the computational mesh and its local refinement. The included appendix contains further technical details on determining appropriate inflow boundary conditions for grade-two flow in both a channel and a pipe.

**1.1. Grade-two fluid model.** The lowest-order, grade-two model of Rivlin and Ericksen [7, 9] can be expressed as

$$\begin{aligned} -\nu \Delta \mathbf{u} + \mathbf{u} \cdot \nabla \mathbf{u} + \nabla p &= \nabla \cdot \hat{\boldsymbol{\tau}}, \\ \nabla \cdot \mathbf{u} &= 0 \quad \text{in } \Omega, \quad \mathbf{u} = \mathbf{g} \quad \text{on } \partial\Omega, \end{aligned} \quad (1)$$

where

$$\hat{\boldsymbol{\tau}} = \alpha_1 (\mathbf{u} \cdot \nabla \mathbf{A} - \mathbf{A} \circ (\nabla \mathbf{u})^t - (\nabla \mathbf{u}) \circ \mathbf{A}) + (2\alpha_1 + \alpha_2) \mathbf{A} \circ \mathbf{A}, \quad (2)$$

and  $\mathbf{A} = \nabla \mathbf{u} + (\nabla \mathbf{u})^t$ . We use the operator  $\circ$  to explicitly denote tensor multiplication, which in this case is simply matrix multiplication. We assume that the boundary data  $\mathbf{g}$  is defined on all  $\Omega$ , is divergence free, and sufficiently smooth, to be specified subsequently.

The equations (1-2) can be viewed as a perturbation of the Navier–Stokes system

$$\begin{aligned} -\Delta \mathbf{u} + R \mathbf{u} \cdot \nabla \mathbf{u} + \nabla p &= \mathbf{0}, \\ \nabla \cdot \mathbf{u} &= 0 \quad \text{in } \Omega, \quad \mathbf{u} = \mathbf{g} \quad \text{on } \partial\Omega, \end{aligned} \quad (3)$$

where  $R = UL/\nu$  is the Reynolds number ( $U$  is a velocity scale and  $L$  is a length scale used to nondimensionalize the equations). When  $R = 0$ , the system (3) is called the Stokes equations. The pressure  $p$  has been rescaled as well.

**1.2. Special case in dimension two.** When  $\alpha_1 + \alpha_2 = 0$ , there is a simplification [8, 9] that can be made in two dimensions that reduces the system (1-2) to

$$-\nu \Delta \mathbf{u} + z(u_2, -u_1)^t + \nabla q = \mathbf{0}, \quad \nabla \cdot \mathbf{u} = 0, \quad \nu z + \alpha_1 \mathbf{u} \cdot \nabla z = \nu \operatorname{curl} \mathbf{u}, \quad \text{in } \Omega, \quad (4)$$

where  $\operatorname{curl} \mathbf{u} = (u_{2,1} - u_{1,2})$  and  $p = q + \frac{1}{2}|\mathbf{u}|^2$ . Here,  $z = \operatorname{curl}(\mathbf{u} - \alpha_1 \Delta \mathbf{u})$  [9]. Unfortunately, this simplification does not generalize to  $\alpha_2 \neq -\alpha_1$  or to three dimensions. Thus we consider the following.

**1.3. Alternate formulation of the grade-two model equations.** Define the tensor  $\mathbf{u} \otimes \mathbf{u}$  by  $(\mathbf{u} \otimes \mathbf{u})_{ij} = u_i u_j$ . Then

$$\nabla \cdot (\mathbf{u} \otimes \mathbf{u}) = \mathbf{u} \cdot \nabla \mathbf{u}.$$

Let  $\pi$  be related to  $p$  by

$$\nu \pi + \alpha_1 \mathbf{u} \cdot \nabla \pi = p. \quad (5)$$

Define

$$\boldsymbol{\tau} = \alpha_1 (\nabla \mathbf{u})^t \circ \mathbf{A} + (\alpha_1 + \alpha_2) \mathbf{A} \circ \mathbf{A} - \mathbf{u} \otimes \mathbf{u}, \quad (6)$$

and

$$N(\mathbf{u}, \pi) = -\alpha_1 \pi \nabla \mathbf{u}^t + \boldsymbol{\tau}.$$

Note that  $N$  is not a symmetric tensor due to the term  $\pi \nabla \mathbf{u}^t$ . The incompressibility condition  $\nabla \cdot \mathbf{u} = 0$  implies that

$$\nabla \cdot (\pi \nabla \mathbf{u}^t) = \nabla \mathbf{u}^t \nabla \pi, \quad \nabla \cdot N(\mathbf{u}, \pi) = -\alpha_1 \nabla \mathbf{u}^t \nabla \pi + \nabla \cdot \boldsymbol{\tau}. \quad (7)$$

Therefore

$$\nabla \cdot N(\mathbf{u}, \pi) = -\alpha_1 \nabla \mathbf{u}^t \nabla \pi + \nabla \cdot (\alpha_1 \nabla \mathbf{u}^t \circ \mathbf{A} + (\alpha_1 + \alpha_2) \mathbf{A} \circ \mathbf{A} - \mathbf{u} \otimes \mathbf{u}). \quad (8)$$

Now consider the problem proposed in [16]:

$$\begin{aligned} -\Delta \mathbf{u} + \nabla \pi &= \mathbf{w} & \text{in } \Omega, \quad \nabla \cdot \mathbf{u} = 0 & \text{in } \Omega, \quad \mathbf{u} = \mathbf{g} & \text{on } \partial \Omega, \\ (\nu I + \alpha_1 \mathbf{u} \cdot \nabla) \mathbf{w} &= \nabla \cdot N(\mathbf{u}, \pi) & \text{in } \Omega, \quad \mathbf{w} = \mathbf{w}_b & \text{on } \Gamma_-, \end{aligned} \quad (9)$$

where

$$\Gamma_- = \{ \mathbf{x} \in \partial \Omega \mid \alpha_1 \mathbf{g}(\mathbf{x}) \cdot \mathbf{n} < 0 \}. \quad (10)$$

The following is proved in [16].

**Theorem 1.1.** Suppose that  $(\mathbf{u}, \pi)$  solves (9) and  $p$  is given by (5). Then  $(\mathbf{u}, p)$  satisfies (1) with  $\hat{\boldsymbol{\tau}}$  defined by (2). The vector function  $\mathbf{w}$  satisfies

$$\mathbf{w} = \frac{1}{\nu} (\nabla \cdot \hat{\boldsymbol{\tau}} - \mathbf{u} \cdot \nabla \mathbf{u} - \nabla p) + \nabla \pi.$$

One modeling challenge arises because (1) is a third-order PDE due to the presence of the term  $\mathbf{u} \cdot \nabla(\Delta \mathbf{u})$ . Therefore it is necessary to specify an additional boundary condition beyond what would be done for the Navier–Stokes equations to have a unique solution. The quantity  $\mathbf{w}$  on which we pose a boundary condition is the divergence of the stress.

**1.4. An algorithm for the transformed equations.** The system (9) is analogous to the reduced system in [9], and that paper suggested the algorithm used in [16] for solving (9): start with some  $\mathbf{w}^0$ , then solve for  $n \geq 1$

$$\begin{aligned} -\Delta \mathbf{u}^n + \nabla \pi^n &= \mathbf{w}^{n-1} & \text{in } \Omega, \quad \nabla \cdot \mathbf{u}^n = 0 & \text{in } \Omega, \quad \mathbf{u}^n = \mathbf{g} & \text{on } \partial \Omega, \\ (\nu I + \alpha_1 \mathbf{u}^n \cdot \nabla) \mathbf{w}^n &= \nabla \cdot N(\mathbf{u}^n, \pi^n) & \text{in } \Omega, \quad \mathbf{w}^n = \mathbf{w}_b & \text{on } \Gamma_-. \end{aligned} \quad (11)$$

For definiteness, we will take  $\mathbf{w}^0 = \mathbf{w}_b$ . In [9], convergence of this iteration is proved for small data  $\mathbf{g}$  and  $\mathbf{w}_b$ .

The following discrete variational form of (11) which is suitable for finite element approximation, and accompanying numerical algorithm, is given in [16]. Let  $W_h$  be the space of continuous, vector-valued, piecewise polynomials of degree  $k$ , let

$V_h = \{\mathbf{v} \in W_h \mid \mathbf{v} = \mathbf{0} \text{ on } \partial\Omega\}$ , and let  $\Pi_h$  be continuous, scalar-valued, piecewise polynomials of degree  $k-1$ . For the computations shown throughout this paper, we use  $k=4$ .

First, using the iterated penalty method: find  $\mathbf{u}^{n,\ell} \in V_h + \mathbf{g}$  such that

$$\int_{\Omega} \nabla \mathbf{u}^{n,\ell} : \nabla \mathbf{v} \, dx + \rho \int_{\Omega} \nabla \cdot \mathbf{u}^{n,\ell} \nabla \cdot \mathbf{v} \, dx = \int_{\Omega} \mathbf{w}^{n-1} \cdot \mathbf{v} \, dx - \int_{\Omega} \nabla \cdot \mathbf{z}^{\ell} \nabla \cdot \mathbf{v} \, dx, \forall \mathbf{v} \in V_h, \quad (12)$$

$$\mathbf{z}^{\ell+1} = \mathbf{z}^{\ell} + \rho \mathbf{u}^{n,\ell}.$$

Once this is converged, we set  $\mathbf{u}^n = \mathbf{u}^{n,\ell}$  and define the pressure via [12]

$$\int_{\Omega} \pi^n q \, dx = \int_{\Omega} -\nabla \cdot \mathbf{z}^{\ell+1} q \, dx \quad \forall q \in \Pi_h. \quad (13)$$

We can pose the transport equation (11) via: find  $\mathbf{w}^n \in \tilde{V}_h + \mathbf{w}_b$  such that

$$\nu \int_{\Omega} \mathbf{w}^n \cdot \mathbf{v} \, dx + \alpha_1 \int_{\Omega} (\mathbf{u}^n \cdot \nabla \mathbf{w}^n) \cdot \mathbf{v} \, dx - \int_{\Omega} (\nabla \cdot N(\mathbf{u}^n, \pi^n)) \cdot \mathbf{v} \, dx = 0 \quad \forall \mathbf{v} \in \tilde{V}_h, \quad (14)$$

where  $\mathbf{w}_b$  is posed only on the inflow boundary, that is,

$$\tilde{V}_h = \{\mathbf{v} \in W_h \mid \mathbf{v} = \mathbf{0} \text{ on } \Gamma_- \}, \quad \Gamma_- = \{\mathbf{x} \in \partial\Omega \mid \mathbf{n} \cdot \mathbf{g} < 0\}.$$

**1.5. Anderson accelerating the solution sequence.** We consider augmenting the solver for nonlinear system (11), which is implemented via (12)-(14), by applying a filtered version of AA as in [14, 15] to the approximation sequences  $\{\mathbf{u}^n\}$  and  $\{\mathbf{z}^n\}$ .

The first equation in (11) is solved by the iterated penalty method (12), which also generates  $\mathbf{z}^n$ . The  $L_2$  projection of the divergence of  $\mathbf{z}^n$  is computed via (13) and used to solve for the auxiliary variable  $\mathbf{w}^n$  in (14). For consistency, it makes sense to perform the extrapolation on  $\mathbf{z}^n$  along with  $\mathbf{u}^n$ , with the extrapolation parameter determined entirely by  $\mathbf{u}^n$ .

Specifically, we consider the following modification to (12)-(14). Starting with an initial iterate  $\mathbf{u}^0$ , let  $\hat{\mathbf{u}}^n = \mathbf{u}^{n,l}$  upon convergence of (12). The algorithm recombines up to  $m_{max}$  previous iterates  $\hat{\mathbf{u}}^j$  and update steps  $\delta \mathbf{u}^j = \hat{\mathbf{u}}^j - \mathbf{u}^{j-1}$ , for  $n-m \leq j \leq n$ , to form the next iterate  $\mathbf{u}^n$ . It also recombines the corresponding iterates and updates steps  $\mathbf{z}^j$  and  $\delta \mathbf{z}^j = \hat{\mathbf{z}}^j - \mathbf{z}^{j-1}$  to form the next iterate  $\mathbf{z}^n$ . The algorithm with depth  $m=0$  reduces to the original fixed-point iteration without acceleration.

The filtered version of AA described below was introduced in [14] and built upon in [15] to better control the accumulation of higher-order terms in the residual expansion by enforcing a sufficient linear independence condition (or, if the parameter  $\sigma$  is chosen close enough to 1, a near-orthogonality condition), between the columns of the coefficient matrix of the underlying least-squares problem. The acceleration becomes an enabling technology allowing the solution for larger parameter pairs  $(\alpha_1, \alpha_2)$  than can be solved without the acceleration. This is demonstrated and further discussed in appendix C.

Filtered AA is implemented and described naturally as a linear algebra routine, operating on the coefficients  $U^n$  of the basis expansion  $\mathbf{u}^n = \sum U_i^n \varphi_i$ , where the  $\{\varphi_i\}$  span the discrete space  $V_h$ . In agreement with standard practice, the inner optimization for this problem is performed with respect to the  $l_2$  norm.

**Algorithm 1.2. (Filtered AA.)** Set depth  $m_{max}$ . Compute  $\hat{U}^1$  and  $\delta U^n = \hat{U}^1 - U^0$ .

Set  $m_0 = 0$ ,  $F_0 = ((\delta U^{n+1} - \delta U^n))$  and  $E_0 = ((U^n - U^{n-1}))$ .

For  $n = 1, 2, \dots$ , set  $m_n = \min\{m_{n-1} + 1, m_{max}\}$

Compute  $\hat{X}^{n+1}$  and  $\delta X^{n+1} = \hat{X}^{n+1} - X^n$ , for  $X = \{U, Z\}$   
Set  $FX_n = ((\delta X^{n+1} - \delta X^n) \quad FX_{n-1})$  and  $EX_n = ((X^n - X^{n-1}) \quad EX_{n-1})$ ,  
for  $X = \{U, Z\}$ .  
Set  $(EU_n, FU_n, EZ_n, FZ_n, m_n, \gamma_n) = \text{Filter}(EU_n, FU_n, \delta U^{n+1}, EZ_n, FZ_n, m_n)$ .  
Set damping factor  $0 < \beta_n \leq 1$ .  
Set  $X^{n+1} = X^n + \beta_n \delta X^{n+1} - (EX_n + \beta_n FX_n) \gamma_n$ , for  $X = \{U, Z\}$ .

The filtering algorithm computes the solution to a least-squares problem of the form  $\gamma_n = \operatorname{argmin}_{\gamma \in \mathbb{R}^{m_n}} \|\delta U^{n+1} - F_n \gamma\|_{l_2}$ , such that columns of  $F_n$  are filtered out if the direction sine between any column of  $F_n$  and the subspace spanned by the columns to its left are less in magnitude than parameter  $\sigma$ . In [15] this is referred to as *angle filtering*. Setting  $\sigma = 0$  means no filtering is performed, and setting  $\sigma = 1$  filters out any column of  $F_n$  that is not orthogonal to the columns to its left. Here we use a dynamic filtering strategy as was shown effective in [14, 15]. This method starts with a higher filtering tolerance  $\sigma_{max}$  which filters out more columns in the preasymptotic regime and relaxes to a lower tolerance  $\sigma_{min}$  which uses more columns for a better optimization in the asymptotic regime. Here we use  $\sigma_{min} = 0.1$ ,  $\sigma_{max} = 2^{-1/2}$ , and  $\sigma = \max\{\min\{\sigma_{max}, \|\delta U^{n+1}\|_{l_2}^{1/2}\}, \sigma_{min}\}$ .

**Algorithm 1.3.**  $((E, F, EZ, FZ, m, \gamma) = \text{Filter}(E, F, \delta U, EZ, FZ, m))$  Given minimum and maximum filtering thresholds  $0 \leq \sigma_{min} < \sigma_{max} < 1$

Compute  $F = QR$ , the thin QR decomposition of  $F$

Set  $\sigma = \max\{\min\{\sigma_{max}, \|\delta U^{n+1}\|_{l_2}^{1/2}\}, \sigma_{min}\}$

For  $i = 2, \dots, m_n$

Compute  $\sigma_i = |r_{ii}|/\|f_i\|_{l_2}$ , where  $r_{ii}$  is the diagonal entry of  $R$ , and  $f_i$  is column  $i$  of  $F$

If  $\sigma_i < \sigma$ , remove column  $i$  from  $F$ ,  $E$ ,  $FZ$  and  $EZ$ , and set  $m = m - 1$

If any columns were removed, recompute  $F = QR$

Solve  $R\gamma = Q^T \delta U$  for  $\gamma$

In the examples that follow we perform the iterations without damping ( $\beta_n = 1$  for all  $n$ ). In practice,  $0 < \beta_n < 1$  can often be used to solve problems for a wider range of parameters.

**1.6. Required inflow boundary conditions.** One feature of the proposed method (11) is that it clarifies the required additional boundary condition, namely for  $\mathbf{w} = -\Delta \mathbf{u} + \nabla \pi$ . Although we cannot say how to pick this in general, Appendix B computes  $\mathbf{w}$  for typical flow geometries. We can extend this using Amick's theorem [1] as described in [17].

**1.7. Theoretical details.** We collect in Appendix A details on the Lebesgue and Sobolev spaces and norms used. Let  $d$  be the dimension of  $\Omega$ . Assume that the domain regularity assumption [16, (4.2)] holds for  $Q_0 > d$  and  $Q_1 > d/2$ , as follows. Suppose that the solution of

$$-\Delta \mathbf{u} + \nabla p = \mathbf{w} \quad \text{and} \quad \nabla \cdot \mathbf{u} = 0 \quad \text{in } \Omega, \quad \mathbf{u} = \mathbf{g} \quad \text{on } \partial\Omega, \quad (15)$$

satisfies, for  $1 \leq q \leq Q_s$ ,  $s = 0, 1$ , and any  $\mathbf{g} \in W_q^{s+2}(\Omega)$  and  $\mathbf{w} \in W_q^s(\Omega)$ , the following estimate:

$$\|\mathbf{u}\|_{W_q^{s+2}(\Omega)} + \|\pi\|_{W_q^{s+1}(\Omega)} \leq c_{q,s} (\|\mathbf{w}\|_{W_q^s(\Omega)} + \|\mathbf{g}\|_{W_q^{s+2}(\Omega)}), \quad (16)$$

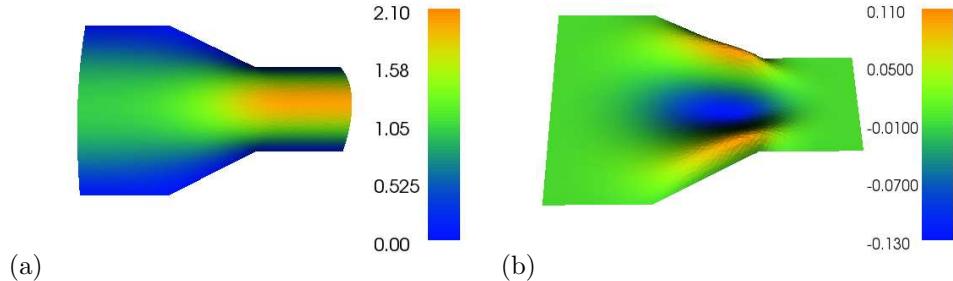


FIGURE 1. Flow in a contracting duct for Stokes flow  $\mathbf{u}_S$  and Navier-Stokes flow  $\mathbf{u}_N$ . (a) horizontal flow component of  $\mathbf{u}_S$ , and (b) horizontal flow component of  $\mathbf{u}_N - \mathbf{u}_S$ , for  $R = 10$ , both with mesh parameter 64. The computational domain  $\Omega$  is as specified in (20), with  $b_i = 1$ ,  $b_o = 1$ ,  $L = 1$ ,  $H = 0.5$ . Computed using (3).

for a constant  $c_{q,s}$  that depends only on  $q$  and  $s$ . This assumption holds if we round off the corners of the contracting duct. The following is proved in [16, Theorem 4.1].

**Theorem 1.4.** *Suppose that  $d < q < Q_0$ . If the boundary data and initial iterates are sufficiently small, the iterates (11) are bounded for all  $n > 0$ :*

$$\|\mathbf{w}^n\|_{L^q(\Omega)} \leq \mathcal{K}, \quad \|\mathbf{u}^n\|_{W_q^2(\Omega)} + \|\pi^n\|_{W_q^1(\Omega)} \leq c_q(\|\mathbf{g}\|_{W_q^2(\Omega)} + \mathcal{K}), \quad (17)$$

where  $\mathcal{K}$  is a finite positive constant. Suppose further that  $r \leq Q_1$  satisfies

$$\frac{2}{d} > \frac{1}{r} > \frac{1}{q} + \frac{1}{2}. \quad (18)$$

Then

$$\|\mathbf{w}^n\|_{W_r^1(\Omega)} \leq \mathcal{K}, \quad \|\mathbf{u}^n\|_{W_r^3(\Omega)} + \|\pi^n\|_{W_r^2(\Omega)} \leq c_q(\|\mathbf{g}\|_{W_r^3(\Omega)} + \mathcal{K}). \quad (19)$$

Moreover,  $(\mathbf{u}^n, \pi^n, \mathbf{w}^n)$  converge geometrically in  $W_r^2(\Omega)^d \times W_r^1(\Omega) \times L^r(\Omega)^d$  to the solution  $(\mathbf{u}, \pi, \mathbf{w})$  of (9). In view of Theorem 1.1,  $(\mathbf{u}, p)$  is the solution of the grade-two model (1), where  $p$  is related to  $\pi$  by (5).

Note that there is a typo in [16, Theorem 4.1], where the estimates for  $s = 1$  should have  $q$  replaced by  $r$ .

The constraint (18) implies  $q > 2$  for  $d = 2$  and  $q > 6$  for  $d = 3$ , and thus the constraint  $q > d$  is satisfied implicitly.

**2. Grade-two flow in a contracting duct.** We begin by describing a typical flow problem involving a contracting duct. We pose a Poiseuille flow profile at the inlet and exit of the channel. We allow “buffers” at each end of the contraction for the flow to regain the Poiseuille flow profile. Thus the domain consists of three parts, first the inlet buffer

$$\Omega_i = \{(x, y) \mid -b_i \leq x \leq 0, |y| \leq 1\}.$$

The contraction zone has length  $L$  and height  $H$  and is defined by

$$\Omega_e = \{(x, y) \mid 0 \leq x \leq L, |y| \leq 1 + ((H-1)/L)x\}.$$

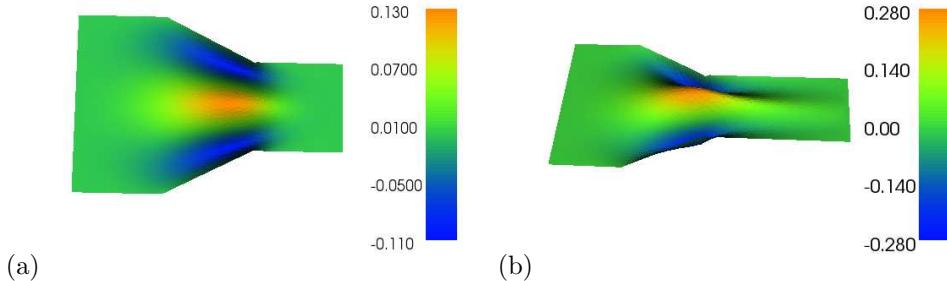


FIGURE 2. Horizontal flow of the difference  $\mathbf{u}_G - \mathbf{u}_N$  for  $\mathbf{u}_G$  being the solution of the grade-two model (4) with (a)  $R = 10$ ,  $\alpha_1 = 10$ ,  $\alpha_2 = -10$ , and (b)  $R = 40$ ,  $\alpha_1 = 1$ ,  $\alpha_2 = -1$ , both with mesh parameter 64. The computational domain  $\Omega$  is as specified in (20), with  $b_i = 1$ ,  $L = 1$ ,  $H = 0.5$ , and with (a)  $b_o = 1$ , (b)  $b_o = 2$ . Computed using (4).

Finally, the outlet buffer is

$$\Omega_o = \{(x, y) \mid L \leq x \leq L + b_o, |y| \leq H\}.$$

Then the computational domain is

$$\Omega = \Omega_i \cup \Omega_e \cup \Omega_o. \quad (20)$$

We will see that the lengths of these buffer zones influence the results substantially in some cases.

The Poiseuille-like boundary conditions we choose are as follows. At the inlet, we choose

$$\mathbf{u}(-b_i, y) = (1 - y^2, 0)^t, \quad y \in [-1, 1].$$

At the outlet, we choose

$$\mathbf{u}(L + b_o, y) = (H^{-1}(1 - (y/H)^2), 0)^t, \quad y \in [-H, H].$$

The choice of Poiseuille-like boundary conditions for the flow variable  $\mathbf{u}$  is based on Amick's theorem [1] as described in [17].

The corresponding inflow boundary conditions for  $\mathbf{w}$  [16] are

$$\mathbf{w}(-b_i, y) = \left(0, \frac{4U^2}{\nu} y(3\alpha_1 + 2\alpha_2)\right)^t, \quad y \in [-1, 1]. \quad (21)$$

**2.1. Stokes versus Navier–Stokes.** In figure 1, we see the horizontal flow component in the domain  $\Omega$  for Stokes flow  $\mathbf{u}_S$ , shown in panel (a), and the horizontal flow component of the difference  $\mathbf{u}_N - \mathbf{u}_S$ , shown in panel (b), between the horizontal flow component of the Navier-Stokes solution ( $R = 10$ ) and the Stokes solution. Several features are of interest. First of all, the Navier-Stokes solution returns to the parabolic profile quickly both before and after the contraction. Secondly, there is a significant boundary layer for the Navier-Stokes solution in the contraction zone, and the flow there is more plug-like (flatter in the middle), with the Stokes solution being larger in the middle of the contraction zone. This may be counter-intuitive, in that the Stokes flow is faster in the middle of the contraction zone, but this is consistent with what is known for Jeffrey-Hamel flow [10].

Re	$\alpha_1$	$\mathbf{u}_N - \mathbf{u}_G$	$\mathbf{u}_S - \mathbf{u}_G$	$\mathbf{u}_N - \mathbf{u}_S$	$\ z\ _{L^2}$	$\Delta p_N$	$\Delta p_G$
0.1	0.01	2.26e-06	8.17e-04	8.18e-04	8.7577	575.2	575.3
1.0	0.01	2.25e-04	7.98e-03	8.09e-03	8.7497	58.45	58.55
1.0	0.1	1.87e-03	7.00e-03	8.09e-03	8.6597	58.45	59.45
1.0	1.0	6.07e-03	3.34e-03	8.09e-03	7.6517	58.45	68.49
10.0	1.0	6.64e-02	6.61e-03	6.97e-02	4.7639	6.801	16.83
50.0	1.0	2.09e-01	7.31e-03	2.14e-01	3.5618	2.232	12.24
10.0	10.0	6.93e-02	7.48e-04	6.97e-02	3.2948	6.801	107.2
50.0	10.0	2.14e-01	8.12e-04	2.14e-01	3.0423	2.232	102.6

TABLE 1. Relative differences  $\|\mathbf{u}_a - \mathbf{u}_b\|_{H^1(\Omega)} / \|\mathbf{u}_S\|_{H^1(\Omega)}$ , indicated in columns 3–5 by  $\mathbf{u}_a - \mathbf{u}_b$ , between solutions to Grade-Two  $\mathbf{u}_G$ , Navier-Stokes  $\mathbf{u}_N$ , and Stokes  $\mathbf{u}_S$ .  $\|\mathbf{u}_S\|_{H^1(\Omega)} = 9.2616$  in all cases. The parameters defining the computational domain (20) are, in all cases,  $H = 0.5$ ,  $L = 1$ ,  $b_o = 3$ , and  $b_i = 1$ , and the meshsize is 64. In all cases,  $\alpha_2 = -\alpha_1$ . Computed using (4).

**2.2. Grade-two with  $\alpha_1 + \alpha_2 = 0$ .** In figure 2(a), we depict the grade-two flow with  $\alpha_2 = -\alpha_1$  computed via (4) in the domain  $\Omega$  defined in (20), showing the horizontal flow component for the difference  $\mathbf{u}_G - \mathbf{u}_N$  for  $R = 10$  and  $\alpha_1 = 10$ . For that domain, we have  $b_i = 1$ ,  $b_o = 1$ ,  $L = 1$ ,  $H = 0.5$ . Here we use the shorthand Re for the Reynolds number, and for the numerical value we write  $R$ . In figure 2(b), we depict the grade-two flow in the domain  $\Omega$  defined in (20), showing the horizontal flow component for the difference  $\mathbf{u}_G - \mathbf{u}_N$  for  $R = 40$  and  $\alpha_1 = 1$ . For that domain, we have changed to  $b_o = 2$  to allow the flow to return to a parabolic form in the outflow buffer.

Table 1 gives data for other values of  $\alpha_1$ , but still with  $\alpha_2 = -\alpha_1$ . Several things emerge from this table. First of all, it is evident that the discrepancy between the Stokes and Navier-Stokes equations is close to linear for small Re, with the coefficient in this case being on the order of 0.008. But for larger Re, the relationship is sublinear. Similarly, the discrepancy between Navier-Stokes and grade-two is close to linear for small  $\alpha_1$ , with the coefficient in this case being on the order of 0.02 for  $R = 1$ . However, for  $R = 10$ , the difference between  $\alpha_1 = 1$  and  $\alpha_1 = 10$  is minimal. The same thing is true for  $R = 50$ . Rather, as  $\alpha_1$  increases,  $\mathbf{u}_G$  tends to  $\mathbf{u}_S$ . We would describe this behavior as shear thickening.

**2.3. Grade-two with  $\alpha_1 + \alpha_2 \neq 0$ .** In figure 3 we contrast the difference in results between the Stokes solution and the grade-two model when parameters  $\alpha_1$  and  $\alpha_2$  are chosen independently, with  $\alpha_1 + \alpha_2 \neq 0$ . We observe that we are no longer able to obtain solutions for such large parameter values, and more care is required with defining the mesh. In the plot on the right, with  $\alpha_1 = \alpha_2 = 0.2$ , we see some nonsmoothness in  $\mathbf{u}$  arising in the vicinity of the reentrant corners. The plot on the left with  $\alpha_1 = \alpha_2 = 0.02$  remains smooth. While we have  $\|\mathbf{u}^n\|_{H^2(\Omega)}$  bounded in terms of  $\|\mathbf{w}^{n-1}\|_{L^2(\Omega)}$ , from (8)–(9), the norm of the auxiliary variable  $\|\mathbf{w}^n\|_{L^2(\Omega)}$  is bounded by  $\|\nabla \cdot N(\mathbf{u}^n, \pi^n)\|_{L^2(\Omega)}$ , which requires  $\|\mathbf{u}^n\|_{H^2(\Omega)}$  to be bounded. More details on these bounds may be found in [16]. Thus  $\mathbf{w}^n$  is sensitive to higher derivatives of  $\mathbf{u}^n$ , and as illustrated in figure 4, these are not bounded for domains with nonconvex corners. These corner singularities motivate the localized

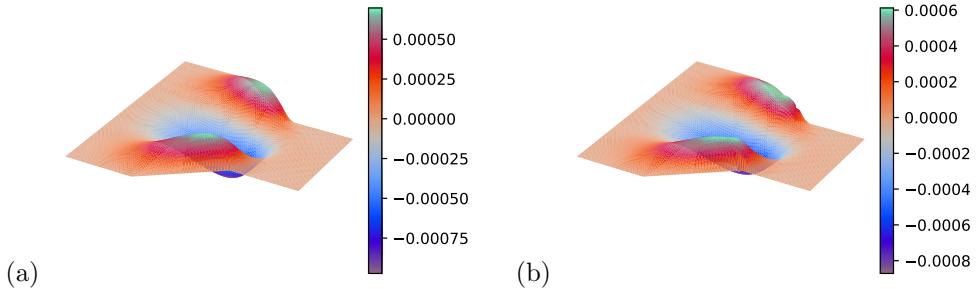


FIGURE 3. Horizontal flow component of the difference  $\mathbf{u}_G - \mathbf{u}_S$  for  $\nu = 1$ ,  $U = 2^{-2}$  and (a)  $\alpha_1 = \alpha_2 = 0.02$ , (b)  $\alpha_1 = \alpha_2 = 0.2$ . The computational domain  $\Omega$  is as specified in (20), with  $b_o = 1$ ,  $b_i = 1$ ,  $L = 1$ , and  $H = 0.5$ . The computational mesh was generated by four uniform refinements of the left-most mesh of figure 10.

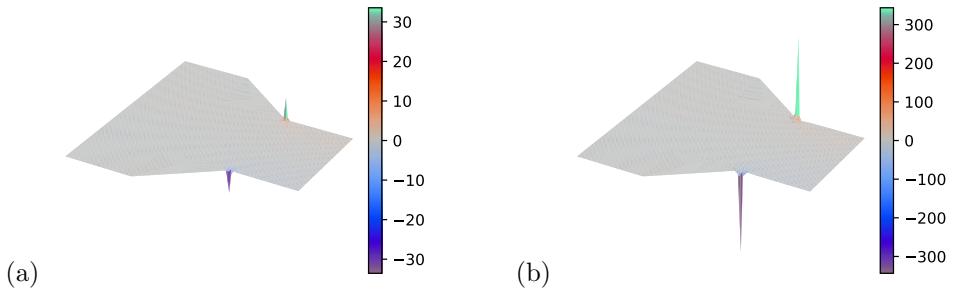


FIGURE 4. Vertical component of the vector-valued auxiliary variable  $\mathbf{w}$  for  $\nu = 1$ ,  $U = 2^{-2}$  and (a)  $\alpha_1 = \alpha_2 = 0.02$ , (b)  $\alpha_1 = \alpha_2 = 0.2$ . The computational domain  $\Omega$  is as specified in (20), with  $b_o = 1$ ,  $b_i = 1$ ,  $L = 1$ , and  $H = 0.5$ . The computational mesh was generated by four uniform refinements of the left-most mesh of figure 10.

refinement of the mesh for the accurate computation of the integral over the contraction boundary described in the next section. The localized refinement is further discussed in section 4.

**3. Contraction rheometer.** Contraction rheometers have been constructed e.g. by Stading [13]. The contraction zone generates a complex flow pattern that can be used to measure nonlinear relationships between the stress and rate of strain. A recent paper [20] examines the concept of identifiability for a rheometer for a given fluid model. The typical experiment with a rheometer involves varying the flow rate and measuring a force as a function of flow rate. In [20], it is shown how to extract model parameters from such a function for certain models.

For some models and rheometers, it is not possible to distinguish certain parameters in a model, so these models are not identifiable by that rheometer. For example [20], an extensional flow rheometer can determine the sum  $\alpha_1 + \alpha_2$  for the grade-two model, but it does not distinguish the individual values  $\alpha_i$ . And a simple

viscosity	$\nu = 1$			$\nu = 2$		
$(\alpha_1, \alpha_2)$	(0.1, 0.1)	(0.1, 0.2)	(0.2, 0.1)	(0.1, 0.1)	(0.1, 0.2)	(0.2, 0.1)
$U = 2^{-6}$	-11.00226	-11.01962	-11.02835	-21.95676	-21.97412	-21.98282
$U = 2^{-7}$	-10.97838	-10.98706	-10.99141	-21.93292	-21.94159	-21.94593
$U = 2^{-8}$	-10.96646	-10.97080	-10.97297	-21.92101	-21.92534	-21.92751
$\Delta^2$	-10.95458	-10.95458	-10.95458	-21.90911	-21.90912	-21.90913

TABLE 2.  $F(U)/U$  with force  $F$  defined in (23), over the computational domain  $\Omega$  as specified in (20), with  $b_i = 1$ ,  $b_o = 1$ ,  $L = 1$ ,  $H = 0.5$ . Results are shown for small values of  $U$  and varying values of  $\alpha_1$  and  $\alpha_2$  with  $\nu = 1$  and  $\nu = 2$ . The bottom row shows the extrapolated values as  $U \rightarrow 0$  computed by the Aitken  $\Delta^2$  process using the three rows above. Computations were performed on a local refinement of the mesh shown in figure 10, as described in subsection 4.1

shear rheometer is insensitive completely to both parameters  $\alpha_i$ . Thus a natural question arises: can a contraction rheometer identify the grade-two model?

**3.1. Force measurements.** The force  $F$  that is measured by a contraction rheometer is defined as:

$$F = \int_{\partial\Omega} \nu \psi \mathbf{n}^t (\nabla \mathbf{u} + \nabla \mathbf{u}^t) \hat{\mathbf{x}} \, ds - \int_{\partial\Omega} \psi p \mathbf{n}^t \hat{\mathbf{x}} \, ds, \quad (22)$$

where  $\hat{\mathbf{x}}$  is the first Euclidean basis vector and  $\psi = 1$  for  $x \in (0, L)$  and zero elsewhere (that is before and after the contraction, see (20) and preceding definitions). Recall that  $p$  is given by (5). Since  $\mathbf{u}$  is zero on the boundary, the formula simplifies to  $p = \nu\pi$  on the support of  $\psi$  on  $\partial\Omega$ . Thus (22) simplifies to

$$F = \nu \left( \int_{\partial\Omega} \nu \psi \mathbf{n}^t (\nabla \mathbf{u} + \nabla \mathbf{u}^t) \hat{\mathbf{x}} \, ds - \int_{\partial\Omega} \psi \pi \mathbf{n}^t \hat{\mathbf{x}} \, ds \right). \quad (23)$$

Experimentally, this force is measured by a null balance device that keeps the contraction portion of the device from moving. The required force is thus proportional to  $F$ . Computing  $F(U)$  for various flow rates allows us to attempt to identify the parameters of a model. Typically, the flow rate  $U$  is increased steadily from zero, possibly plateauing at given values of  $U$  temporarily to allow a steady state to re-establish. The limiting value of  $F(U)/U$  for small values of  $U$  is typically proportional to the viscosity  $\nu$ . Other features of  $F$  can be used to identify other parameters.

In particular, the bottom row of table 2, which was computed by applying Aitken's  $\Delta^2$  process to the three rows above to compute a value of  $F(U)/U$  as  $U \rightarrow 0$ , shows the computed force is approximately  $-10.9546\nu$ . Table 3 shows the result of applying the same process across a wider range of  $(\alpha_1, \alpha_2)$  parameters. We see the same results as in table 2 for small values of  $\alpha_1$  and  $\alpha_2$ , with a difference in the fourth decimal place when these parameters are an order of magnitude larger. We next turn our attention to the identifiability of  $\alpha_1$  and  $\alpha_2$ .

**3.2. Visualizing the data.** In this section we consider the computed data from our force measurement simulations  $f(U, \nu, \alpha) = F(U)/U$  for  $\alpha = (\alpha_1, \alpha_2)$ . Writing  $\alpha$  in polar coordinates as  $\alpha = \alpha(\cos(\theta), \sin(\theta))$ , we can visualize the dependence of

$\alpha_1/\alpha_2$	-0.64	-0.16	-0.04	-0.01
0.01	-10.9545	-10.9545	-10.9545	-10.9546
0.04	-10.9545	-10.9545	-10.9546	-10.9546
0.16	-10.9545	-10.9546	-10.9546	-10.9546
0.64	-10.9546	-10.9546	-10.9546	-10.9547
$\alpha_1/\alpha_2$	0.01	0.04	0.16	0.64
0.01	-10.9546	-10.9546	-10.9546	-10.9546
0.04	-10.9546	-10.9546	-10.9546	-10.9546
0.16	-10.9546	-10.9546	-10.9546	-10.9546
0.64	-10.9547	-10.9547	-10.9547	-10.9548

TABLE 3.  $\lim_{U \rightarrow 0} F(U)/U$  with force  $F$  defined in (23), over the computational domain  $\Omega$  as specified in (20), with  $b_i = 1$ ,  $b_o = 1$ ,  $L = 1$ ,  $H = 0.5$ . Results shown are determined by applying Aitken's  $\Delta^2$  process as in table 2 to the computed  $F(U)/U$  with  $U = 2^{-6}, 2^{-7}, 2^{-8}$ , for different  $(\alpha_1, \alpha_2)$  pairs with  $\nu = 1$ . Computations were performed on a local refinement of the mesh shown in figure 10, as described in subsection 4.1

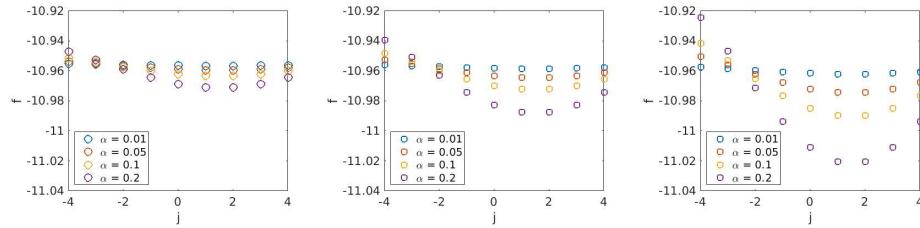


FIGURE 5. Generated data  $f(U, \nu, \alpha) = F(U)/U$  with the same computational domain and mesh as in table 2 for  $\alpha = \alpha(\cos(j\pi/8), \sin(j\pi/8))$  with  $\nu = 1$  and varying  $\alpha$  and  $j$ . Left:  $U = 2^{-8}$ ; center:  $U = 2^{-7}$ ; right:  $U = 2^{-6}$ .

$f$  on  $U$ ,  $\alpha$  and  $\theta$  as follows. This allows us to better understand regimes where the parameters  $\alpha_1, \alpha_2$  could be identifiable using this rheometer.

First in figure 5 we consider snapshots in  $U$  while  $\theta$  is varied for different values of  $\alpha$ . We make three important observations from these three snapshots. First,  $f$  has a nonlinear dependence on  $\theta$  for fixed  $U$ . In contrast, as shown in figure 6 and table 4 which display the difference between the computed data and the best fit linear, quadratic and cubic polynomials for a range of  $U$  values with  $\alpha$  and  $\theta$  fixed,  $f$  is close to linear in  $U$ ; this is further illustrated in figures 7 and 8. Second, it appears that each of the fixed- $\alpha$  trajectories of figure 5 cross at  $\theta = j\pi/8$  for  $j \approx -2.5$ , which we will look at more closely in table 5 and see further illustrated in figure 7. Third, it appears  $f$  is symmetric over  $\theta \approx j\pi/8$  for  $j = 1.5$ , which we will look at more closely in figure 8.

To visualize the near linearity of  $f$  as a function of  $U$ , we computed the best fit regression line  $\mathcal{I}_1 f$ , quadratic fit  $\mathcal{I}_2 f$ , and cubic fit  $\mathcal{I}_3 f$  to the 10 data points for  $U = 0.01, 0.02, \dots, 0.1$ , with fixed  $\theta \in \{\pm 3\pi/16, \pm \pi/16\}$  and  $\alpha \in \{0.03, 0.07, 0.14\}$ . The results of the errors  $f - \mathcal{I}_i f$  normalized by its maximum value for each line are shown in figure 6. The normalization factors are shown in table 4. We see that

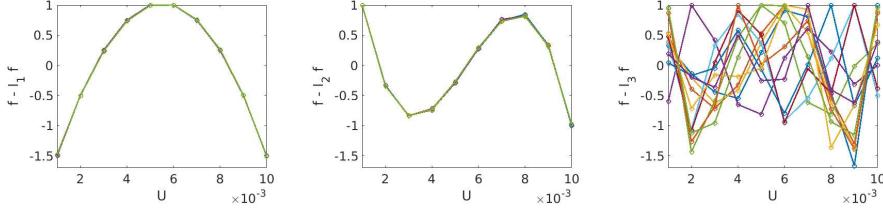


FIGURE 6. Normalized difference of the error in the least squares regression line (left), quadratic fit (center) and cubic fit (right) and the generated data  $f(U, \nu, \alpha) = F(U)/U$ . The normalization factors (maximum errors) are shown in table 4. Each plot shows an overlay of 12  $(\alpha, \theta)$  pairs for  $\alpha = \alpha(\cos(\theta), \sin(\theta))$  with  $\theta \in \{\pm\pi/16, \pm 3\pi/16\}$ ,  $\alpha \in \{0.03, 0.07, 0.14\}$  and  $U$  increasing from 0.01 to 0.1. The data  $f(U, \nu, \alpha)$  were generated with the same computational domain and mesh as in table 2.

max $(f - \mathcal{I}_1 f)$				
$\alpha/\theta$	$-3\pi/16$	$-\pi/16$	$\pi/16$	$3\pi/16$
0.03	3.810e-6	0.419e-6	0.533e-6	0.563e-6
0.07	0.576e-6	1.052e-6	1.360e-6	1.420e-6
0.14	1.306e-6	2.532e-6	3.339e-6	3.423e-6
max $(f - \mathcal{I}_2 f)$				
$\alpha/\theta$	$-3\pi/16$	$-\pi/16$	$\pi/16$	$3\pi/16$
0.03	5.760e-8	0.020e-8	0.040e-8	0.043e-8
0.07	0.054e-8	0.305e-8	0.562e-8	0.579e-8
0.14	0.554e-8	2.622e-8	4.677e-8	4.769e-8
max $(f - \mathcal{I}_3 f)$				
$\alpha/\theta$	$-3\pi/16$	$-\pi/16$	$\pi/16$	$3\pi/16$
0.03	6.385e-10	0.052e-10	0.049e-10	0.038e-10
0.07	0.062e-10	0.057e-10	0.057e-10	0.097e-10
0.14	0.124e-10	0.694e-10	1.283e-10	0.564e-10

TABLE 4. Maximum error between the computed data points  $f(U, \nu, \alpha) = F(U)/U$  and the regression line  $\mathcal{I}_1 f$ , quadratic fit  $\mathcal{I}_2 f$ , and cubic fit  $\mathcal{I}_3 f$ . Each line fits the data for the flow rate  $U$  increasing linearly from 0.01 to 0.1 with fixed values of  $\alpha = \{0.03, 0.07, 0.14\}$  and  $\theta = \{\pm\pi/16, \pm 3\pi/16\}$ .

- the error in each linear regression line is approximately quadratic, with its maximum magnitude on the order of  $10^{-6}$ ;
- the error in each best-fit quadratic is approximately cubic, with its maximum magnitude on the order of  $10^{-8}$ ;
- and the error in the best-fit cubic is not clearly structured but has maximum magnitude on the order of  $10^{-10}$ .

To investigate the crossing of trajectories of each fixed value of  $\alpha$  near  $\theta = -5\pi/16$ , for each  $U_k = 2^{-k}$ ,  $k = 6, 7, 8$ , we computed the intersection of the linear interpolants connecting  $f$  at  $\theta_L = -2.65\pi/8$  (left of the crossing) to  $\theta_R = -2.5\pi/8$

$U = 2^{-8}, \theta_8 = -2.50266827$			$U = 2^{-7}, \theta_7 = -2.50268795$		
$\alpha$	$\theta = \theta_8 \cdot \pi/8$	$\theta = -2.5\pi/8$	$\alpha$	$\theta = \theta_7 \cdot \pi/8$	$\theta = -2.5\pi/8$
0.01	-10.9556149	-10.9556158	0.01	-10.9566795	-10.9566812
0.05	-10.9556149	-10.9556190	0.05	-10.9566795	-10.9566878
0.1	-10.9556149	-10.9556231	0.1	-10.9566795	-10.9566961
0.2	-10.9556149	-10.9556313	0.2	-10.9566795	-10.9567126

$U = 2^{-6}, \theta_6 = -2.50272560$		
$\alpha$	$\theta = \theta_6 \cdot \pi/8$	$\theta = -2.5\pi/8$
0.01	-10.9588091	-10.9588125
0.05	-10.9588091	-10.9588259
0.1	-10.9588091	-10.9588427
0.2	-10.9588091	-10.9588762

TABLE 5. Generated data  $f(U, \nu, \alpha) = F(U)/U$  with the same computational domain and mesh as in table 2. Results are shown for values of  $\alpha = \alpha(\cos(\theta), \sin(\theta))$  near the crossing of trajectories of constant  $\alpha$  values shown in figure 5. The crossing for each value of  $U$  occurs at approximately  $\theta = \theta_k \cdot \pi/8$ ,  $k = 6, 7, 8$ . Left:  $U = 2^{-8}$ ; center:  $U = 2^{-7}$ ; right:  $U = 2^{-6}$ .

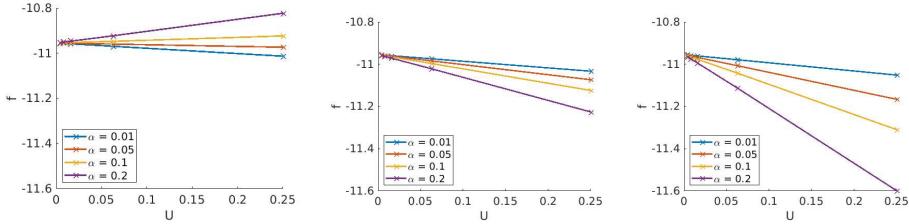


FIGURE 7. Generated data  $f(U, \nu, \alpha) = F(U)/U$  with the same computational domain and mesh as in table 2 showing  $f$  changes with  $U$  for  $U = 2^{-j}, j = \{8, 7, 6, 4, 2\}$  and different values of  $\alpha$ , the magnitude of  $\alpha = \alpha(\cos(\theta), \sin(\theta))$ . Left:  $\theta = -3\pi/8$ ; center:  $\theta = -2\pi/8$ ; right:  $\theta = -\pi/8$ .

(right of the crossing) for  $\alpha = 0.01$  and  $\alpha = 0.2$ . The computed crossing points  $\theta = \theta_k \pi/8$  are shown in table 5 along with the value of  $f(U, 1, \alpha)$  at each of the given coordinates. As shown in the table,  $f$  is constant through 7 decimal places at each  $\theta = \theta_k \pi/8, U = 2^{-k}$  pair of coordinates (not shown: the values are decreasing with  $\alpha$  for each at the 9th digit). At  $\theta = -2.5\pi/8$ , about 0.001 to the right in  $\theta$ , the values of  $f$  are increasing with  $\alpha$ , where the differences are seen in the fifth decimal place, which is on the order of the tolerance of our solve. This shows that  $\alpha$ , the magnitude of  $\alpha$ , is not identifiable using the contraction rheometer if the argument  $\theta$  of  $\alpha$  is close to  $-2.5\pi/8$ .

Figure 7 shows snapshots of  $f$  as a function of  $U$  for different values of  $\alpha$  with  $\theta = -3\pi/8$  (left) and  $\theta = -2\pi/8$  (center), which lie on either side of the crossing. We observe that the constant- $\alpha$  trajectories are in opposite order on these two plots. Comparing the center plot for  $\theta = -2\pi/8$  and the right plot for  $\theta = -\pi/8$ , we can see how the angle between fixed- $\alpha$  trajectories changes for different values of  $\theta$ .

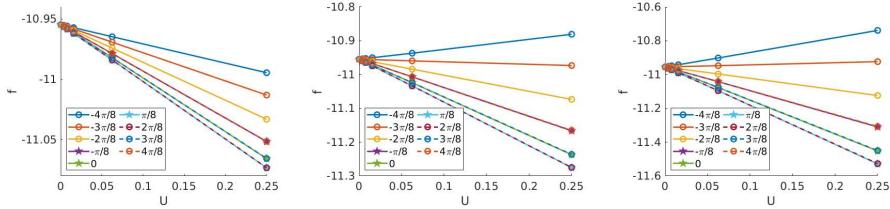


FIGURE 8. Generated data  $f(U, \nu, \alpha) = F(U)/U$  with the same computational domain and mesh as in table 2 showing how  $f$  changes with  $U$  for  $U = 2^{-j}, j = \{8, 7, 6, 4, 2\}$  and different values of  $\theta$ , the argument of  $\alpha = \alpha(\cos(\theta), \sin(\theta))$ . Left:  $\alpha = 0.01$ ; center:  $\alpha = 0.05$ ; right:  $\alpha = 0.1$ .

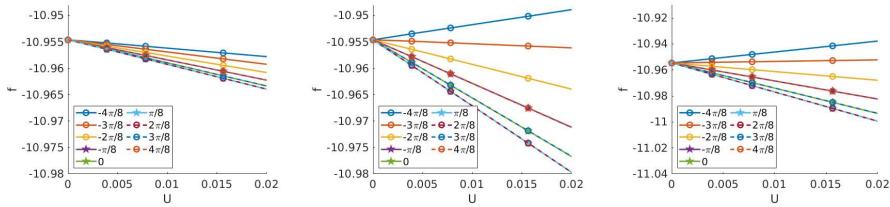


FIGURE 9. Zoomed in view of figure 8 showing generated data  $f(U, \nu, \alpha) = F(U)/U$  with the same computational domain and mesh as in table 2 showing how  $f$  changes with  $U$  for  $U = 2^{-j}, j = \{8, 7, 6, 4, 2\}$  and different values of  $\theta$ , the argument of  $\alpha = \alpha(\cos(\theta), \sin(\theta))$ . Left:  $\alpha = 0.01$ ; center:  $\alpha = 0.05$ ; right:  $\alpha = 0.1$ . The first two plots use the same scaling.

Our third observation from figure 5 is that  $f$  appears symmetric around  $\theta_S \approx 3\pi/16$ , that is,

$$f(U, \nu, \alpha(\cos(\theta_S + t), \sin(\theta_S + t))) = f(U, \nu, \alpha(\cos(\theta_S - t), \sin(\theta_S - t))), \quad (24)$$

where the equality holds up to the tolerance of our solve. This matching of trajectories of  $f$  as  $U$  is increased is further illustrated in figure 8 which shows  $f$  for four different values of  $\alpha$  and varying  $U$ . The fixed- $\theta$  trajectories are nearly linear in  $U$ ; and,

- the data for  $\theta = \pi/2 = 8\pi/16$  overlays the data for  $\theta = -\pi/8 = -2\pi/16$  ( $t = 5\pi/16$  in (24)),
- the data for  $\theta = 3\pi/8 = 6\pi/16$  overlays the data for  $\theta = 0$  ( $t = 3\pi/16$  in (24)), and
- the data for  $\theta = \pi/4 = 2\pi/8 = 4\pi/16$  overlays the data for  $\theta = \pi/8 = 2\pi/16$  ( $t = \pi/16$  in (24)).

Figure 9 shows a zoomed-in view of figure 8. The point at  $(0,0)$  was added to each of these plots to illustrate again the near-linearity in  $U$  of data  $f$  for small flow rates  $U$ . In terms of identifying  $\alpha$  given a sequence of measurements of  $f(U, \nu, \alpha)$  for varying flow rates  $U$ , we conclude that we can only expect to identify  $\alpha = \alpha(\cos(\theta), \sin(\theta))$  in a limited range in arguments  $\theta$  of  $\alpha$ , namely  $\theta \in (-5\pi/16, 3\pi/16]$ .

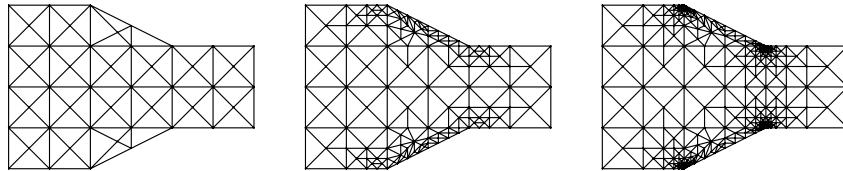


FIGURE 10. Boundary refinements ( $r_b$ ) and point refinements ( $r_p$ ) of the computational mesh for the contracting duct. Left:  $r_b = 0, r_p = 0$ ; center:  $r_b = 2, r_p = 0$ ; right:  $r_b = 2, r_p = 4$ .

**4. Computational details.** For our simulations of flow in a contracting duct, as described in section 2, we used the following parameters. We solved to a default relative tolerance of  $\|\mathbf{u}^n - \mathbf{u}^{n-1}\|_{H^1} / \|\mathbf{u}^n\|_{H^1} < 10^{-5}$ , and an iterated-penalty-method (IPM) [5, 19] tolerance of  $\|\nabla \cdot (\mathbf{u}^{n,l})\| < 10^{-10}$ , with IPM parameter  $\rho = 10^4$ . We used Lagrange degree  $k = 4$  elements for the space  $V_h$  from subsection 1.4. All computations were performed using FEniCS [11, 19].

**4.1. Localized mesh refinement.** For the contraction rheometer computations described in Section 3, we performed a local refinement of the left-hand mesh in figure 10. The localized refinements consisted of boundary refinements (denoted  $r_b$ ) and point refinements (denoted  $r_p$ ). The boundary refinements first mark any element sharing an edge with the contraction boundary. These elements were identified by (a) having a maximum distance of 0.65 to the points  $(0.4, \pm 0.5)$ , and (b) sharing an edge with the boundary. The refinement includes subdivision of the marked elements, followed by a completion step refining certain neighboring elements to ensure the mesh is conforming. The point refinements (denoted  $r_p$ ) first subdivide any element that contains one of the four endpoints of the contraction boundary, namely  $(1, \pm 0.5)$  and  $(0, \pm 1)$ ; followed by a completion step to ensure the mesh is conforming. Meshes with  $r_b = 2, r_p = 0$  and  $r_b = 2, r_p = 4$  are illustrated in the center and right of figure 10.

To determine a computationally efficient but accurate mesh, we first compared uniform refinements (denoted  $r_u$ ) to refinements focused on the contraction boundary, as described above. As shown in table 6, without any point refinements, boundary refinements consistently gave the same result as uniform refinements to two decimal places in the computation of the normalized force integral  $F/U$ . Comparing entries with like errors, we see with boundary refinements alone, computing with 75,274 degrees of freedom (dof) gives similar accuracy to 367,178 dof with uniform refinements alone.

As shown in table 7, with 12 point refinements, uniform and boundary refinement gave the same results to three significant digits on the least refined and four on the most refined meshes. Again comparing entries with similar accuracy we see seven boundary refinements with 99,850 dof gives similar accuracy to four uniform refinements with 389,770 dof. By comparison with table 8, we see that after 7 refinements of either type plus the 12 point refinements,  $F/U$  converged to two decimal places. To maintain efficiency, we eliminated the uniform refinements, and attained significantly more accuracy using a combination of point refinements and boundary refinements, as shown in table 8. In particular, 9 boundary refinements followed by 12 point refinements resolves  $F/U$  to at least  $5 \times 10^{-4}$ . Increasing  $r_p$

$F(U)/U$					
$r_p = 0$	$r_b = 0$	$r_b = 1$	$r_b = 2$	$r_b = 3$	
$r_u = 0$	-0.00000	-5.08710	-7.80120	-9.30473	
$r_u = 1$	-5.08429	-7.80074	-9.30503	-10.15792	
$r_u = 2$	-7.80075	-9.30300	-10.15784	-10.64975	
$r_u = 3$	-9.30527	-10.15532	-10.64951	-10.93667	
$r_u = 4$	-10.15816	-10.64720	-10.93639	-11.10560	
$F(U)/U$					
$r_p = 0$	$r_b = 4$	$r_b = 5$	$r_b = 6$	$r_b = 7$	
$r_u = 0$	-10.15756	-10.64941	-10.93652	-11.10571	
$r_u = 1$	-10.64980	-10.93692	-11.10612		
$r_u = 2$	-10.93688	-11.10607			
$r_u = 3$	-11.10587				

$\dim(V_h)$								
$r_p = 0$	$r_b = 0$	$r_b = 1$	$r_b = 2$	$r_b = 3$	$r_b = 4$	$r_b = 5$	$r_b = 6$	$r_b = 7$
$r_u = 0$	1258	2186	3834	6474	11226	20538	39098	75274
$r_u = 1$	4882	6338	8978	13730	23042	41602	77778	
$r_u = 2$	19234	21618	26370	35682	54242	90418		
$r_u = 3$	76354	80594	89906	108466	144642			
$r_u = 4$	304258	312442	331002	367178				

TABLE 6.  $F/U$  with force  $F$  defined in (23), and  $\dim(V_h)$  using degree  $k = 4$  elements over the computational domain  $\Omega$  is as specified in (20), with  $b_i = 1$ ,  $b_o = 1$ ,  $L = 1$ ,  $H = 0.5$  and parameters  $\alpha_1 = 0.3$ ,  $\alpha_2 = 0.1$ ,  $\nu = 1$  and  $U = 2^{-4}$ . The results shown for  $r_u$  uniform refinements and  $r_b$  boundary refinements of the mesh shown in figure 10 demonstrate that without point refinements, uniform refinements can be exchanged for refinements over the contraction boundary to increase the accuracy of the computation with fewer degrees of freedom.

from 12 to 16 however did not appear to increase the accuracy of the computation. The results of subsection 3.2 were computed with the most accurate combination found here:  $r_b = 9$  and  $r_p = 12$ , for a total of 315,922 degrees of freedom in the discrete space  $V_h$ , using quartic (degree 4) vector-valued Lagrange basis functions. As seen in the analysis above, this combination appears to balance the resolution of the boundaries and the reentrant corners with the interior of the mesh.

**4.2. Computational mesh.** In this section we summarize our findings on defining an appropriate computational mesh for this problem. First, the discrete inf-sup constant should not degenerate as the mesh size is decreased. The convergence of the IPM algorithm slows with diminishing returns if the inf-sup constant is sufficiently small. This slow convergence can be mitigated to some extent by an early exit strategy for the IPM iteration defined in (12), allowing the iterations to terminate if  $\|\nabla \cdot \mathbf{u}^{n,l}\|/\|\nabla \cdot \mathbf{u}^{n,l-1}\| > 1/2$ , if this occurs before a tolerance or a maximum number of iterations is reached. Of course this only improves the slow convergence of IPM with a small inf-sup constant and relaxes the requirement of choosing an IPM tolerance that is not too small; the entire algorithm will still ultimately diverge

$F(U)/U$					
$r_p = 12$	$r_b = 0$	$r_b = 1$	$r_b = 2$	$r_b = 3$	
$r_u = 0$	-11.33672	-11.34446	-11.34942	-11.35244	
$r_u = 1$	-11.34479	-11.34947	-11.35249	-11.35439	
$r_u = 2$	-11.34960	-11.35246	-11.35435	-11.35554	
$r_u = 3$	-11.35258	-11.35434	-11.35553	-11.35626	
$r_u = 4$	-11.35440	-11.35552	-11.35625	-11.35668	
$F(U)/U$					
$r_p = 12$	$r_b = 4$	$r_b = 5$	$r_b = 6$	$r_b = 7$	
$r_u = 0$	-11.35434	-11.35553	-11.35626	-11.35670	
$r_u = 1$	-11.35558	-11.35631	-11.35674		
$r_u = 2$	-11.35627	-11.35671			
$r_u = 3$	-11.35669				

$\dim(V_h)$								
$r_p = 12$	$r_b = 0$	$r_b = 1$	$r_b = 2$	$r_b = 3$	$r_b = 4$	$r_b = 5$	$r_b = 6$	$r_b = 7$
$r_u = 0$	22186	23754	25850	29002	34266	44090	63162	99850
$r_u = 1$	26002	27906	31058	36322	46146	65218	101906	
$r_u = 2$	40354	43186	48450	58274	77346	114034		
$r_u = 3$	97474	102162	111986	131058	167746			
$r_u = 4$	325378	334010	353082	389770				

TABLE 7.  $F/U$  with force  $F$  defined in (23), and  $\dim(V_h)$  using degree  $k = 4$  elements over the computational domain  $\Omega$  is as specified in (20), with  $b_i = 1$ ,  $b_o = 1$ ,  $L = 1$ ,  $H = 0.5$  and parameters  $\alpha_1 = 0.3$ ,  $\alpha_2 = 0.1$ ,  $\nu = 1$  and  $U = 2^{-4}$ . The results shown for  $r_u$  uniform refinements,  $r_b$  boundary refinements and 12 point refinements of the mesh shown in figure 10 demonstrate that used together with point refinements, uniform refinements can be exchanged for refinements over the contraction boundary to increase the accuracy of the computation with fewer degrees of freedom.

if the IPM does not converge to  $\mathbf{u}^{n,l}$  with a sufficiently small divergence. We include a further discussion of the convergence of IPM for Stokes in subsection 4.3.

The problem of non-convex corners causes an issue for solving the transport problem. Figure 11 shows the second component of the terminal approximation to  $\mathbf{w}$  via (14) on three meshes with comparable mesh sizes. The figure on the left is computed on a mesh generated by `mshr` with parameter  $M = 16$ . In the center is the same computation performed on the computational mesh shown in figure 10 with two uniform refinements. On the right of figure 11 is a variant in which the re-entrant corner is smoothed into two corners with smaller internal angles. The height of the spike is ten times higher on the the mesh generated by `mshr` than either of the other two. Simulations run on either uniform or boundary refinements of this mesh realized poor convergence of the IPM iteration, indicating degeneration of the inf-sup constant. The smoothed domain on the right has multiple spikes which are moderately shorter than on the mesh shown in 10. The singular behavior of  $\mathbf{w}$  at nonconvex corners is a feature of the problem formulation given by (9). For this problem, a good computational mesh should allow for local refinement and resolution of the singularity without causing the algorithm to diverge.

$F(U)/U$					
$r_u = 0$	$r_p = 0$	$r_p = 4$	$r_p = 8$	$r_p = 12$	$r_p = 16$
$r_b = 4$	-11.35705	-11.35434	-11.33619	-11.20465	-10.15756
$r_b = 5$	-10.64941	-11.26481	-11.34434	-11.35553	-11.35706
$r_b = 6$	-10.93652	-11.30100	-11.34936	-11.35626	-11.35716
$r_b = 7$	-11.10571	-11.32291	-11.35247	-11.35670	-11.35709
$r_b = 8$	-11.20614	-11.33624	-11.35439	-11.35738	-11.35547
$r_b = 9$	-11.26616	-11.34440	-11.35559	-11.35767	-11.35522

$\dim(V_h)$					
$r_u = 0$	$r_p = 0$	$r_p = 4$	$r_p = 8$	$r_p = 12$	$r_p = 16$
$r_b = 4$	11226	20186	27226	34266	41306
$r_b = 5$	20538	30010	37050	44090	51130
$r_b = 6$	39098	49082	56122	63162	70202
$r_b = 7$	75274	85770	92810	99850	106890
$r_b = 8$	147098	158106	165146	172186	179226
$r_b = 9$	290322	301842	308882	315922	322962

TABLE 8.  $F/U$  with force  $F$  defined in (23), and  $\dim(V_h)$  using degree  $k = 4$  elements over the computational domain  $\Omega$  is as specified in (20), with  $b_i = 1$ ,  $b_o = 1$ ,  $L = 1$ ,  $H = 0.5$  and parameters  $\alpha_1 = 0.3$ ,  $\alpha_2 = 0.1$ ,  $\nu = 1$  and  $U = 2^{-4}$ . The results shown for no uniform refinements,  $r_b$  boundary refinements and  $r_p$  point refinements of the mesh shown in figure 10 demonstrate that sufficient refinement at the four endpoints of the contraction boundary together with boundary refinements provides an increase in accuracy. As seen with  $r_p = 16$ , too many refinements at those points does not further increase the accuracy.

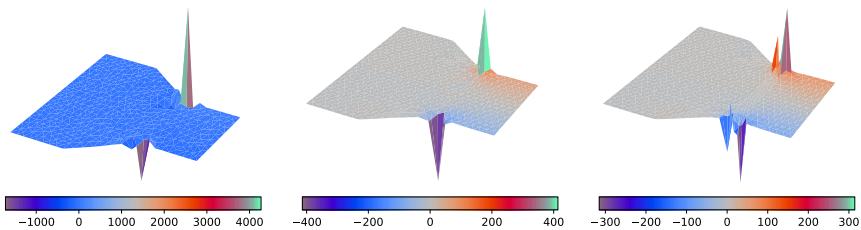


FIGURE 11. Spikes in  $w$  at the re-entrant corners of the domain on three different meshes with  $U = 1$ ,  $\nu = 1$ , and  $\alpha_1 = \alpha_2 = 0.1$ .

**4.3. IPM and Stokes.** We noted in the previous section that the grade-two simulations for Poiseuille flow may degrade slightly as the mesh is refined and as the iterated-penalty method (IPM) tolerance is decreased. This is unfortunately a feature of the IPM which may be mitigated by defining a mesh on which the inf-sup

$M$	$\ \mathbf{u} - \mathbf{u}_h\ _{H^1}$	$\ \nabla \cdot \mathbf{u}_h\ _{L^2}$	IPM iters	$\rho$	split type
32	1.67e-08	1.02e-11	3	1.00e+04	crossed-triangle
32	2.34e-09	4.90e-11	4	1.00e+03	crossed-triangle
32	3.68e-10	3.91e-11	7	1.00e+02	crossed-triangle
32	4.27e-09	1.21e-11	3	1.00e+04	right-triangle
32	5.76e-10	5.52e-11	4	1.00e+03	right-triangle
32	3.22e-10	5.63e-11	9	1.00e+02	right-triangle

TABLE 9. Stokes errors for Poiseuille flow in the domain (25) with  $L = 1$  for  $\nu = 1$ , using the iterated penalty method [19] (IPM) with quartics on an  $M \times M$  array of squares split in two ways. The Malkus crossed-triangle split consists of squares divided into four triangles by the bisectors. The right-triangle split consists of squares divided into two right triangles.

constant is robust. Table 9 indicates the effect of the penalty parameter  $\rho$  on the convergence of IPM. Larger  $\rho$  gives faster convergence for the divergence, but at the expense of less accuracy for the velocity. Table 9 shows the errors after the optimal number of IPM iterations in terms of minimizing the velocity error,  $\mathbf{u} - \mathbf{u}_h$ . We see that a smaller error can be achieved at the cost of doing more iterations with a smaller  $\rho$ . As  $\rho$  is decreased further, the number of IPM iterations becomes prohibitively large.

**5. Conclusion.** We have demonstrated that it is possible, with suitable care, to simulate the grade-two model in a geometry related to a contraction rheometer. We have indicated how the results can be used to determine the viscosity  $\nu$  from experimental data. We have also explored issues related to identifying the grade-two parameters  $\alpha_1$  and  $\alpha_2$  with certain caveats. In particular, the force data appears to be the same for distinct values of  $\alpha$ , however, we identify a regime in which  $\alpha$  may be identifiable with a contraction rheometer.

**6. Data availability.** Our codes and mesh data will be available on the corresponding author's webpage.

**7. Acknowledgments.** We thank Mats Stading for valuable discussions. SP was supported in part by the National Science Foundation NSF DMS-2011519. We also thank the anonymous referees for thoughtful suggestions that improved this paper.

**Appendix A. Spaces.** Here we collect the notation used for various Sobolev spaces and norms. We denote by  $L^p(\Omega)$  the Lebesgue spaces [5] of  $p$ -th power integrable functions, with norm

$$\|f\|_{L^p(\Omega)} = \left( \int_{\Omega} |f(\mathbf{x})|^p dx \right)^{1/p}.$$

Note that we can easily apply the same notation to vector or tensor valued  $f$ . We think of tensors of any arity as vectors of the appropriate length, and we think of  $|f(\mathbf{x})|$  as the Euclidean length of this vector. For tensors of arity 2 (i.e., matrices) this is the same as the Frobenius norm. We will write the spaces for such tensor-valued functions as  $L^p(\Omega)^m$  for the appropriate  $m$  (e.g.,  $m = d^2$  for arity 2).

Similarly, we denote by  $L^\infty(\Omega)$  the Lebesgue space of essentially bounded functions, with

$$\|f\|_{L^\infty(\Omega)} = \sup \{|f(\mathbf{x})| \mid \text{a.e. } \mathbf{x} \in \Omega\}.$$

Correspondingly, we define Sobolev spaces and norms of order  $m$  by

$$\|f\|_{W_p^m(\Omega)} = \left( \sum_{|\alpha| \leq m} \|D^\alpha f\|_{L^p(\Omega)}^p \right)^{1/p},$$

where  $D^\alpha$  is the weak derivative  $\partial^\alpha / \partial \mathbf{x}^{|\alpha|}$  [5]. More precisely, the spaces  $W_p^m(\Omega)$  are defined as the subspaces of  $L^p(\Omega)$  for which the corresponding norm is finite. The case  $p = 2$  is denoted by  $H$ :

$$H^m(\Omega) = W_2^m(\Omega).$$

**Appendix B. Determining inflow boundary conditions.** The proposed method (11) requires specification of boundary conditions for  $\mathbf{w} = -\Delta \mathbf{u} + \nabla \pi$ . Here we compute  $\mathbf{w}$  for typical flow geometries. It corresponds to the divergence of the stress.

**B.1. Grade-two channel flow.** In [16], simple two-dimensional grade-two flows (Couette and Poiseuille) are presented for the domain  $\Omega$  defined by

$$\Omega = \{\mathbf{x} \in \mathbb{R}^2 \mid 0 < x_1 < L, 0 < x_2 < 1\}. \quad (25)$$

Suppose that  $u_2 \equiv 0$  and  $u_1$  depends only on  $x_2$ . This is true for shear flow (Couette flow) and pressure-driven flow (Poiseuille flow). For the remainder of this subsection, we refer to  $u_1$  as just  $u$  to simplify notation. For shear (Couette) flow,  $\mathbf{w} = \mathbf{0}$ . For Poiseuille flow, in the channel (25),

$$\mathbf{g} = \mathbf{u} = U \begin{pmatrix} x_2(L-x_2) \\ 0 \end{pmatrix}, \quad \mathbf{w} = -\frac{2U^2}{\nu}(L-2x_2) \begin{pmatrix} 0 \\ 2\alpha_2 + 3\alpha_1 \end{pmatrix}.$$

Furthermore,

$$p(\mathbf{x}) = -2U\nu x_1 + (2\alpha_1 + \alpha_2)U^2(L-2x_2)^2 + c_p. \quad (26)$$

**B.2. Grade-two pipe flow.** Consider Poiseuille flow in a circular pipe. To be specific, we define the domain  $\Omega$  to be

$$\Omega = \{\mathbf{x} \in \mathbb{R}^3 \mid x_1^2 + x_2^2 < 1, 0 < x_3 < L\}. \quad (27)$$

Suppose that  $u_1 = u_2 \equiv 0$  and

$$u_3(\mathbf{x}) = U(1 - (x_1^2 + x_2^2)). \quad (28)$$

This is true for pressure-driven flow (Poiseuille flow). For such flows,  $\mathbf{u} \cdot \nabla \mathbf{u} = \mathbf{0}$ , and the strain rate  $\nabla \mathbf{u}$  is given by

$$\nabla \mathbf{u} = -2U \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ x_1 & x_2 & 0 \end{pmatrix}, \quad \nabla \mathbf{u}^t = -2U \begin{pmatrix} 0 & 0 & x_1 \\ 0 & 0 & x_2 \\ 0 & 0 & 0 \end{pmatrix}.$$

Thus

$$\mathbf{A} = -2U \begin{pmatrix} 0 & 0 & x_1 \\ 0 & 0 & x_2 \\ x_1 & x_2 & 0 \end{pmatrix}, \quad \mathbf{u} \cdot \nabla \mathbf{A} = \mathbf{0}, \quad \mathbf{A} \circ \mathbf{A} = 4U^2 \begin{pmatrix} x_1^2 & x_1 x_2 & 0 \\ x_1 x_2 & x_2^2 & 0 \\ 0 & 0 & x_1^2 + x_2^2 \end{pmatrix},$$

$$\mathbf{A} \circ (\nabla \mathbf{u}) = 4U^2 \begin{pmatrix} x_1^2 & x_1 x_2 & 0 \\ x_1 x_2 & x_2^2 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad \mathbf{A} \circ (\nabla \mathbf{u}^t) = 4U^2 \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & x_1^2 + x_2^2 \end{pmatrix},$$

$$\begin{aligned}
(\nabla \mathbf{u})^t \circ \mathbf{A} &= (\mathbf{A}^t \circ (\nabla \mathbf{u}))^t = (\mathbf{A} \circ (\nabla \mathbf{u}))^t = \mathbf{A} \circ (\nabla \mathbf{u}) = 4U^2 \begin{pmatrix} x_1^2 & x_1 x_2 & 0 \\ x_1 x_2 & x_2^2 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \\
\mathbf{A} \circ (\nabla \mathbf{u}) + (\nabla \mathbf{u})^t \circ \mathbf{A} &= 8U^2 \begin{pmatrix} x_1^2 & x_1 x_2 & 0 \\ x_1 x_2 & x_2^2 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \\
(\nabla \mathbf{u}) \circ \mathbf{A} &= (\mathbf{A}^t \circ (\nabla \mathbf{u})^t)^t = (\mathbf{A} \circ (\nabla \mathbf{u})^t)^t = \mathbf{A} \circ (\nabla \mathbf{u})^t = 4U^2 \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & x_1^2 + x_2^2 \end{pmatrix}.
\end{aligned}$$

We can simplify this by introducing two matrices

$$\mathbf{J} = 4U^2 \begin{pmatrix} x_1^2 & x_1 x_2 & 0 \\ x_1 x_2 & x_2^2 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad \mathbf{K} = 4U^2 \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & x_1^2 + x_2^2 \end{pmatrix}.$$

For example,

$$\mathbf{A} \circ \mathbf{A} = \mathbf{J} + \mathbf{K}, \quad \mathbf{A} \circ (\nabla \mathbf{u}) + (\nabla \mathbf{u})^t \circ \mathbf{A} = 2\mathbf{J}, \quad (\nabla \mathbf{u})^t \circ \mathbf{A} = \mathbf{J}.$$

For the steady-state, grade-two fluid model, the stress tensor simplifies [16] to

$$\begin{aligned}
\mathbf{T}_G &= \nu \mathbf{A} + \alpha_1 (\mathbf{u} \cdot \nabla \mathbf{A} + \mathbf{A} \circ (\nabla \mathbf{u}) + (\nabla \mathbf{u})^t \circ \mathbf{A}) + \alpha_2 \mathbf{A} \circ \mathbf{A} \\
&= \mathbf{T}_N + 2\alpha_1 \mathbf{J} + \alpha_2 (\mathbf{J} + \mathbf{K}).
\end{aligned} \tag{29}$$

The tensor  $\boldsymbol{\tau}$  is given by (6):

$$\begin{aligned}
\boldsymbol{\tau} &= \alpha_1 (\nabla \mathbf{u})^t \circ \mathbf{A} + (\alpha_1 + \alpha_2) \mathbf{A} \circ \mathbf{A} - \mathbf{u} \otimes \mathbf{u} \\
&= \alpha_1 \mathbf{J} + (\alpha_1 + \alpha_2) (\mathbf{J} + \mathbf{K}) - \mathbf{u} \otimes \mathbf{u}.
\end{aligned} \tag{30}$$

We can compute  $\nabla \cdot \boldsymbol{\tau}$  as follows. By definition,

$$(\nabla \cdot \mathbf{J})_i = \sum_j J_{ij,j} = J_{i1,1} + J_{i2,2}, \quad \nabla \cdot \mathbf{J} = \begin{pmatrix} J_{11,1} + J_{12,2} \\ J_{21,1} + J_{22,2} \\ 0 \end{pmatrix} = 4U^2 \begin{pmatrix} 3x_1 \\ 3x_2 \\ 0 \end{pmatrix}$$

since  $\mathbf{J}$  is constant in  $x_3$  and thus  $\mathbf{J}_{i3,3} = 0$ . Similarly,  $\nabla \cdot \mathbf{K} = \mathbf{0}$ . Therefore

$$\nabla \cdot \boldsymbol{\tau} = (2\alpha_1 + \alpha_2) \nabla \cdot \mathbf{J} = 12U^2 (2\alpha_1 + \alpha_2) (x_1 \ x_2 \ 0)^t.$$

From (7) we find

$$\nabla \cdot N(\mathbf{u}, \pi) = -\alpha_1 \nabla \mathbf{u}^t \nabla \pi + \nabla \cdot \boldsymbol{\tau} = (2\alpha_1 U \pi_{x_3} + 12U^2 (2\alpha_1 + \alpha_2)) (x_1 \ x_2 \ 0)^t. \tag{31}$$

Also  $-\nu \Delta \mathbf{u} = 4\nu U (0 \ 0 \ 1)^t$ . Thus

$$4\nu U \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} + \nabla p = \nabla \cdot (\mathbf{T}_G - \mathbf{T}_N) = \nabla \cdot ((2\alpha_1 + \alpha_2) \mathbf{J}) = 12U^2 (2\alpha_1 + \alpha_2) \begin{pmatrix} x_1 \\ x_2 \\ 0 \end{pmatrix}.$$

Therefore

$$p_{x_3} = -4\nu U, \quad \begin{pmatrix} p_{x_1} \\ p_{x_2} \end{pmatrix} = 12U^2 (2\alpha_1 + \alpha_2) \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}.$$

These equations are solved by

$$p(\mathbf{x}) = -4\nu U x_3 + 6U^2 (2\alpha_1 + \alpha_2) (x_1^2 + x_2^2).$$

Let us make the ansatz that  $\pi(\mathbf{x}) = -4Ux_3 + f(x_1, x_2)$ . We have

$$p = \nu\pi + \alpha_1 u\pi_{x_3} = p - 6U^2(2\alpha_1 + \alpha_2)(x_1^2 + x_2^2) + \nu f - 4U\alpha_1 u, \quad (32)$$

which suggests that

$$\begin{aligned} \nu f &= 6U^2(2\alpha_1 + \alpha_2)(x_1^2 + x_2^2) + 4U^2\alpha_1(1 - (x_1^2 + x_2^2)) \\ &= U^2(8\alpha_1 + 6\alpha_2)(x_1^2 + x_2^2) + 4U^2\alpha_1. \end{aligned} \quad (33)$$

Therefore

$$\pi(\mathbf{x}) = -4Ux_3 + \nu^{-1}U^2((8\alpha_1 + 6\alpha_2)(x_1^2 + x_2^2) + 4\alpha_1).$$

In particular, (31) implies

$$\nabla \cdot \mathbf{N}(\mathbf{u}, \pi) = U^2(16\alpha_1 + 12\alpha_2) (x_1 \quad x_2 \quad 0)^t. \quad (34)$$

Similarly,

$$\mathbf{w} = -\Delta \mathbf{u} + \nabla \pi = \nu^{-1}U^2(16\alpha_1 + 12\alpha_2) \begin{pmatrix} x_1 \\ x_2 \\ 0 \end{pmatrix}.$$

This provides an appropriate initial condition for solving the third (transport) equation in (11) for pipe flow with boundary conditions given by (28).

Note that  $\mathbf{w}_{,x} \equiv 0$ . Thus (9) implies that

$$\nabla \cdot \mathbf{N} = \nu \mathbf{w},$$

which is consistent with (34).

**Appendix C. Benefits of Anderson acceleration.** In subsection 1.5 it was remarked that AA can be used as an enabling technology to allow the use of larger parameter pairs  $(\alpha_1, \alpha_2)$ . A demonstration and discussion of this observation follows.

As can be seen from the theoretical foundations in [16] referenced in Theorem 1.4, how small the boundary data and solution has to be in order to ensure convergence depends on parameters  $\alpha_1$  and  $\alpha_2$ . In table 10 we consider the flow rate  $U$  and parameter pairs  $(\alpha_1, \alpha_2)$  on the boundary of the convergent regime with fixed  $\nu = 1$ . The computations are performed on the same fixed mesh used throughout the numerical experiments in this paper, namely over the computational domain  $\Omega$  as specified in (20), with  $b_i = 1$ ,  $b_o = 1$ ,  $L = 1$ ,  $H = 0.5$ , with local refinement of the mesh shown in figure 10, as described in subsection 4.1. Filtered AA was implemented as described in subsection 1.5 with algorithmic depth  $m_{max} = 10$  in algorithm 1.2 and without relaxation:  $\beta_n = 1$  for all  $n$ . The filtering parameters in algorithm 1.3 were set as  $\sigma_{max} = 1/\sqrt{2}$  and  $\sigma_{min} = 0.1$ , as used in [15]. The initial iterate in each case was taken as the solution to the corresponding Stokes problem.

The results of table 10 show that when the iteration without AA converges, the iteration with AA converges as well, and the iteration counts differ in this example by at most one. However, in seven of the ten cases in which the iteration without AA diverged, as indicated by “D” in the table, the iteration with AA still converged. The heuristic explanation for this can be found in [14], which shows that the effect of AA on each iteration is to reduce the first order term in the residual proportional to the gain factor from the optimization problem, at the cost of introducing higher-order terms. In the notation of algorithm 1.2, the gain factor is given by  $\|\delta U^{n+1} - FU_{n+1}\gamma_n\| / \|\delta U^{n+1}\|$ . By its definition the gain factor is no greater than one, as  $\gamma$  is the solution to the least-squares problem that minimizes

$U = 2^{-3}$		Iteration (11)					Algorithm 1.2 and iteration (11)				
$\theta$		$-\pi/4$	$-\pi/8$	0	$\pi/8$	$\pi/4$	$-\pi/4$	$-\pi/8$	0	$\pi/8$	$\pi/4$
$\alpha = 0.1$		4	4	4	4	4	4	5	5	5	5
$\alpha = 0.2$		4	6	D	D	D	5	6	9	6	6
$\alpha = 0.3$		4	D	D	D	D	5	15	D	D	D
$U = 2^{-4}$		Iteration (11)					Algorithm 1.2 and iteration (11)				
$\theta$		$-\pi/4$	$-\pi/8$	0	$\pi/8$	$\pi/4$	$-\pi/4$	$-\pi/8$	0	$\pi/8$	$\pi/4$
$\alpha = 0.2$		4	4	4	4	4	4	5	5	5	5
$\alpha = 0.3$		4	5	6	7	5	4	5	6	6	5
$\alpha = 0.4$		4	6	D	D	D	5	6	9	9	7

TABLE 10. Iteration count to relative tolerance of  $\|\mathbf{u}^n - \mathbf{u}^{n-1}\|_{H^1} / \|\mathbf{u}^n\|_{H^1} < 10^{-5}$  for iteration 11 with and without AA algorithm 1.2, for varying  $(\alpha_1, \alpha_2)$  parametrized by  $\alpha(\cos(\theta), \sin(\theta))$  and fixed  $\nu = 1$ . The letter “D” indicates divergence. The computations are performed over the domain  $\Omega$  as specified in (20), with  $b_i = 1$ ,  $b_o = 1$ ,  $L = 1$ ,  $H = 0.5$ , with local refinement of the mesh shown in figure 10, as described in subsection 4.1.

the numerator of this fraction. With a successful optimization, which occurs when  $\delta U^{n+1}$  lies close to the span of  $FU_{n+1}$ , the gain factor can be close to zero. When the residual is small enough for the higher-order terms not to interfere, AA can by this mechanism effectively enable convergence.

## REFERENCES

- [1] C. J. Amick, Steady solutions of the Navier–Stokes equations in unbounded channels and pipes, *Annali della Scuola Normale Superiore di Pisa–Classe di Scienze*, **4** (1977), 473–513.
- [2] D. G. Anderson, Iterative procedures for nonlinear integral equations, *J. Assoc. Comput. Mach.*, **12** (1965), 547–560.
- [3] N. Arada, P. Correia and A. Sequeira, Analysis and finite element simulations of a second-order fluid model in a bounded domain, *Numerical Methods for Partial Differential Equations: An International Journal*, **23** (2007), 1468–1500.
- [4] J.-M. Bernard, Fully nonhomogeneous problem of two-dimensional second grade fluids, *Mathematical Methods in the Applied Sciences*, **41** (2018), 6772–6792.
- [5] S. C. Brenner and L. R. Scott, *The Mathematical Theory of Finite Element Methods*, Third edition, Texts in Applied Mathematics, 15. Springer, New York, 2008.
- [6] D. Cioranescu, V. Girault and K. R. Rajagopal, *Mechanics and Mathematics of Fluids of the Differential Type*, Advances in Mechanics and Mathematics, 35. Springer, 2016.
- [7] J. L. Ericksen and R. S. Rivlin, Stress-deformation relations for isotropic materials, *Archive for Rational Mechanics and Analysis*, **4** (1955), 323–425.
- [8] V. Girault and L. R. Scott, Finite element discretizations of a two-dimensional grade-two fluid model, *M2AN Math. Model. Numer. Anal.*, **35** (2001), 1007–1053.
- [9] V. Girault and L. R. Scott, Analysis of a two-dimensional grade-two fluid model with a tangential boundary condition, *J. Math. Pures Appl.*, **78** (1999), 981–1011.
- [10] L. D. Landau and E. M. Lifshitz, *Fluid Mechanics*, Pergamon Press, 1959.
- [11] A. Logg, K. A. Mardal and G. Wells, *Automated Solution of Differential Equations by the Finite Element Method: The FEniCS Book*, Springer-Verlag New York Inc, 2012.
- [12] H. Morgan and L. R. Scott, Towards a unified finite element method for the Stokes equations, *SIAM Journal on Scientific Computing*, **40** (2018), A130–A141.
- [13] M. Nyström, H. R. T. Jahromi, M. Stading and M. F. Webster, Hyperbolic contraction measuring systems for extensional flow, *Mechanics of Time-Dependent Materials*, **21** (2017), 455–479.

- [14] S. Pollock and L. G. Rebholz, [Anderson acceleration for contractive and noncontractive operators](#), *IMA Journal of Numerical Analysis*, **41** (2021), 2841-2872.
- [15] S. Pollock and L. G. Rebholz, [Filtering for Anderson acceleration](#), *SIAM Journal on Scientific Computing*, **45** (2023), A1571-A1590.
- [16] S. Pollock and L. R. Scott, [An algorithm for the grade-two rheological model](#), *ESAIM Math. Model. Numer. Anal.*, **56** (2021), 1007-1025.
- [17] S. Pollock and L. R. Scott, [Transport equations with inflow boundary conditions](#), *Partial Differential Equations and Applications*, **3** (2021), Paper No. 35, 20 pp.
- [18] S. Pollock and L. R. Scott, Identifying parameters in the grade-two fluid model, work in progress.
- [19] L. R. Scott, *Introduction to Automated Modeling with FEniCS*, Computational Modeling Initiative, 2018.
- [20] L. R. Scott, [Rheology, rheometers, and matching models to experiments](#), *Fluid Dynamics Research*, **55** (2023), 015501, 22 pp.

Received April 2024; revised July 2024; early access August 2024.