

Report Number 10/33

**Error bounds on block GaussSeidel solutions of coupled
multiphysics problems**

by

J.P. Whiteley, K. Gillow, S.J. Tavener, A.C. Walter



Oxford Centre for Collaborative Applied Mathematics
Mathematical Institute
24 - 29 St Giles'
Oxford
OX1 3LB
England

Error bounds on block Gauss–Seidel solutions of coupled multiphysics problems

J.P. Whiteley^{1,2,*}, K. Gillow², S.J. Tavener^{2,3}, A.C. Walter²

¹ Oxford University Computing Laboratory, Wolfson Building, Parks Road, Oxford OX1 3QD, UK ² Mathematical Institute, University of Oxford, 24–29 St Giles', Oxford OX1 3LB, UK ³ Department of Mathematics, Colorado State University, 105 Weber Building, Fort Collins, CO 80523, USA

SUMMARY

Mathematical models in many fields often consist of coupled sub-models, each of which describe a different physical process. For many applications, the quantity of interest from these models may be written as a linear functional of the solution to the governing equations. Mature numerical solution techniques for the individual sub-models often exist. Rather than derive a numerical solution technique for the full coupled model, it is therefore natural to investigate whether these techniques may be used by coupling in a block Gauss–Seidel fashion. In this study, we derive two *a posteriori* bounds for such linear functionals. These bounds may be used on each Gauss–Seidel iteration to estimate the error in the linear functional computed using the single physics solvers, without actually solving the full, coupled problem. We demonstrate the use of the bound first by using a model problem from linear algebra, and then a linear ordinary differential equation example. We then investigate the effectiveness of the bound using a non-linear coupled fluid–temperature problem. One of the bounds derived is very sharp for most linear functionals considered, allowing us to predict very accurately when to terminate our block Gauss–Seidel iteration.

Copyright © 2000 John Wiley & Sons, Ltd.

KEY WORDS: block Gauss–Seidel, error bound, multiphysics model

1. INTRODUCTION

Recent increases in computational power and availability of experimental data in a wide variety of application domains has spawned a new generation of more complex mathematical models. These new models often fall into the category of *coupled multiphysics models*, where two or more established models are combined, allowing the interaction between several physical processes to be modelled. An established coupled multiphysics problem is fluid–structure interaction, where a model of viscous fluid flow is coupled with an elasticity model [1]. This coupled multiphysics problem allows computation of, for example, the force exerted by a fluid on a

*Correspondence to: Oxford University Computing Laboratory, Wolfson Building, Parks Road, Oxford OX1 3QD, UK.

structure, with obvious practical engineering applications. Many more recent examples may be drawn from biology and physiology, for example heart modelling [2]. Force in cardiac fibres is generated by electrophysiological processes within cardiac tissue. This force causes deformation of cardiac tissue — i.e. a heart beat — which in turn causes blood to be discharged from the heart. Furthermore, the deformation of cardiac tissue also affects cardiac electrophysiology, and so there is coupling in both directions between cardiac electrophysiology and cardiac tissue deformation. A comprehensive model of the whole heart therefore requires a coupled multiphysics model of all of these physical processes.

To fully utilise these coupled multiphysics models we require efficient numerical techniques to compute a solution to the governing equations to within a user-prescribed accuracy. Two approaches may be taken here. The most obvious approach is to couple the models using a block Gauss–Seidel iteration: see [3] for how this may be implemented for fluid–structure interaction problems. Another approach is to develop a numerical technique for the whole coupled multiphysics problem. Although this has been shown to be very effective for fluid–structure interaction problems [4], this technique is likely to be difficult to implement for the whole heart problem described above, where efficient computation of the electrophysiology component of this model has only been achieved for detailed physiological models by uncoupling the equations [5, 6], using numerical techniques for the single physics problems.

The single physics components of the coupled model usually have two features: (i) established, reliable numerical techniques exist for each component; and (ii) the spatial mesh coarseness and timestep required to compute a numerical solution of each component to a given accuracy are well understood. As such, we may assume that we may neglect the discretisation error that is induced by approximating the differential equations by numerical approximations. Instead, as a first step towards developing a coupled solver, we focus on the error that is induced by neglecting some of the coupling terms. This complements the work [7] on mesh refinement for both components of the model that takes account of the error transferred between models.

In this study, we address two issues related to computing the solution of coupled multiphysics problems by using numerical solvers for the single physics problem as described above. The first of these is *weak coupling* between components of the model: it is often assumed that one component of the model is only very weakly dependent — in some sense — on another component. Under these conditions, the weak coupling is often neglected. This is often the case for a coupled model of cardiac electrophysiology and cardiac tissue deformation: the electrophysiology is often assumed to be independent of tissue deformation, with the attractive computational feature that the electrophysiology may be computed independently of the tissue deformation. However, there are situations where tissue deformation can dramatically alter the electrophysiology — possibly fatally altering the electrophysiology, or being used to re-set malfunctioning electrophysiology. We present a technique that may be used to validate neglecting coupling in terms of an *a posteriori* bound. This bound may be used for any quantity of interest that may be written as a linear functional of the solution. As such, our bound may be used to fully interrogate the effect of neglecting coupling, thus informing us of in what sense the coupling is “weak”. The second goal of the paper is to investigate solving the coupled equations by solving the components independently, and iterating in a Gauss–Seidel fashion. We show that the bound derived for weak coupling may be adapted for application in this case, and present results that demonstrate that this iteration converges much faster for some linear functionals of the solution than for others.

This paper is arranged as follows. In Section 2 we derive the *a posteriori* error bound,

first in the case where we neglect coupling and then for a block Gauss–Seidel iteration, for linear problems. This bound is then extended to non-linear problems in Section 3. We then demonstrate the features of this bound using examples from: (i) linear algebra in Section 4; (ii) linear ordinary differential equations in Section 5; and (iii) a coupled fluid–temperature problem in Section 6. Finally, we give some concluding remarks in Section 7.

2. DERIVATION OF THE ERROR BOUND FOR LINEAR PROBLEMS

2.1. The coupled and uncoupled problems

Let us suppose that two uncoupled mathematical models are described by linear differential equations. We assume that these equations may be discretised using some numerical technique by the linear systems

$$A_{11}u_1 = b_1, \quad A_{22}u_2 = b_2, \quad (1)$$

where the first equation corresponds to the first model and the second equation to the second model. Let us suppose further that established techniques exist for solving these equations, i.e. A_{11}^{-1} and A_{22}^{-1} both exist, and we can compute the action of A_{11}^{-1} and A_{22}^{-1} on any vector of the correct size, thus allowing us to solve both equations in (1) to a prescribed level of accuracy using an iterative linear solver such as GMRES or the conjugate gradient method: see, for example, [8]. Our final assumption on the uncoupled model is that the discretised equations (1) give a sufficiently accurate approximation to the differential equation from which they are derived that we may neglect discretisation error.

Now suppose we wish to couple the two mathematical models that have been discretised by (1). Providing this coupling is described by a linear algebraic or differential relationship in u_1 , u_2 we may discretise this coupled problem by the matrix equation

$$\begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}, \quad (2)$$

where A_{11} is a $M \times M$ matrix, A_{12} is a $M \times N$ matrix, A_{21} is a $N \times M$ matrix, A_{22} is a $N \times N$ matrix, u_1 is a vector of length M , u_2 is a vector of length N , b_1 is a vector of length M , b_2 is a vector of length N . A_{12} characterises the effect of u_2 on the first model, A_{21} characterises the effect of u_1 on the second model.

Let us now assume that we are interested in some linear functional of the coupled solution, denoted by $J(u_1, u_2)$, and defined by

$$J(u_1, u_2) = \langle \psi_1, u_1 \rangle + \langle \psi_2, u_2 \rangle, \quad (3)$$

for prescribed ψ_1, ψ_2 , where $\langle f, g \rangle$ is an inner product between two vectors f and g . We wish to bound the error in approximating $J(u_1, u_2)$ by an approximate solution for u_1, u_2 . We now derive a bound for this error.

2.2. One-way coupled problems

We first consider the situation where the first model is believed to be only weakly dependent on the second model. Assuming, therefore, that the effect of A_{12} is negligible, we may write

this problem as

$$\begin{pmatrix} A_{11} & 0 \\ A_{21} & A_{22} \end{pmatrix} \begin{pmatrix} \hat{u}_1 \\ \hat{u}_2 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}, \quad (4)$$

which we may uncouple and solve as follows:

$$\begin{aligned} \hat{u}_1 &= A_{11}^{-1} b_1, \\ \hat{u}_2 &= A_{22}^{-1} (b_2 - A_{21} \hat{u}_1). \end{aligned}$$

Note that neglecting the effect of A_{12} results in the two systems above that may be solved using the solvers for the single physics problem. We henceforth refer to (4) as the *one way coupled problem*.

Suppose \hat{u}_1, \hat{u}_2 are solutions of the one way coupled problem (4). The residuals obtained when substituting these solutions into the fully coupled problem (2) are given by

$$\begin{aligned} R_1 &= A_{11} \hat{u}_1 + A_{12} \hat{u}_2 - b_1, \\ R_2 &= A_{21} \hat{u}_1 + A_{22} \hat{u}_2 - b_2. \end{aligned}$$

Using (4), we see that we may write

$$R_1 = A_{12} \hat{u}_2, \quad R_2 = 0. \quad (5)$$

The second matrix equation from each of (2) and (4) is

$$\begin{aligned} A_{21} u_1 + A_{22} u_2 &= b_2, \\ A_{21} \hat{u}_1 + A_{22} \hat{u}_2 &= b_2, \end{aligned}$$

from which we may deduce that

$$u_2 - \hat{u}_2 = -A_{22}^{-1} A_{21} (u_1 - \hat{u}_1). \quad (6)$$

Noting that the first matrix equation from each of (2) and (4) may be written

$$\begin{aligned} u_1 &= A_{11}^{-1} (b_1 - A_{12} u_2), \\ \hat{u}_1 &= A_{11}^{-1} b_1, \end{aligned}$$

we may write

$$\begin{aligned} u_1 - \hat{u}_1 &= -A_{11}^{-1} A_{12} u_2 \\ &= -A_{11}^{-1} A_{12} (u_2 - \hat{u}_2) - A_{11}^{-1} R_1. \end{aligned} \quad (7)$$

Substituting (6) into (7) we obtain

$$u_1 - \hat{u}_1 = A_{11}^{-1} A_{12} A_{22}^{-1} A_{21} (u_1 - \hat{u}_1) - A_{11}^{-1} R_1,$$

which may be written

$$u_1 - \hat{u}_1 = -(I - C)^{-1} A_{11}^{-1} R_1, \quad (8)$$

$$\text{where } C = A_{11}^{-1} A_{12} A_{22}^{-1} A_{21}. \quad (9)$$

Substituting (8), (9) into (6) yields

$$u_2 - \hat{u}_2 = A_{22}^{-1} A_{21} (I - C)^{-1} A_{11}^{-1} R_1. \quad (10)$$

As we are assuming that the first model is only weakly dependent on the second model, it is reasonable to assume that $\|A_{12}\|$ is small. Noting that $\|C\| \leq \|A_{11}^{-1}\| \|A_{12}\| \|A_{22}^{-1}\| \|A_{21}\|$, for sufficiently weak coupling we may assume that $\|C\| < 1$. Under these conditions we may write

$$(I - C)^{-1} = I + C + C^2 + C^3 + \dots, \quad (11)$$

which may be used to bound the error induced in the linear functional (3) that is induced when using \hat{u}_1, \hat{u}_2 instead of u_1, u_2 :

$$\begin{aligned} J(u_1, u_2) - J(\hat{u}_1, \hat{u}_2) &= \langle \psi_1, u_1 - \hat{u}_1 \rangle + \langle \psi_2, u_2 - \hat{u}_2 \rangle, \\ &= -\langle \psi_1, (I - C)^{-1} A_{11}^{-1} R_1 \rangle + \langle \psi_2, A_{22}^{-1} A_{21} (I - C)^{-1} A_{11}^{-1} R_1 \rangle, \\ &= -\sum_{k=0}^{\infty} \langle \psi_1, C^k A_{11}^{-1} R_1 \rangle + \sum_{k=0}^{\infty} \langle \psi_2, A_{22}^{-1} A_{21} C^k A_{11}^{-1} R_1 \rangle. \end{aligned} \quad (12)$$

Note that we can compute all terms appearing in this bound, as the terms A_{11}^{-1}, A_{22}^{-1} correspond to solving the single physics problems.

We now demonstrate that the matrix C given by (9) is independent of the use of preconditioners that may be used for the single physics problems. Suppose P_{11} is a preconditioner for the first problem, and P_{22} is a preconditioner for the second problem. Applying these preconditioners to (2) yields

$$\begin{pmatrix} P_{11} A_{11} & P_{11} A_{12} \\ P_{22} A_{21} & P_{22} A_{22} \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \begin{pmatrix} P_{11} b_1 \\ P_{22} b_2 \end{pmatrix}.$$

The matrix C for this system is then given by

$$\begin{aligned} C &= (P_{11} A_{11})^{-1} P_{11} A_{12} (P_{22} A_{22})^{-1} P_{22} A_{21} \\ &= A_{11}^{-1} P_{11}^{-1} P_{11} A_{12} A_{22}^{-1} P_{22}^{-1} P_{22} A_{21} \\ &= A_{11}^{-1} A_{12} A_{22}^{-1} A_{21}, \end{aligned}$$

and so C is unchanged.

We note that the bound given by (12) is an infinite series. We will discuss later how to bound this series: we first explain how (12) may be generalised to the solution of (2) by block Gauss-Seidel iteration.

2.3. Extending the bound to block Gauss-Seidel iteration

Suppose we solve (2) using a block Gauss-Seidel iteration

$$\begin{pmatrix} A_{11} & 0 \\ A_{21} & A_{22} \end{pmatrix} \begin{pmatrix} \hat{u}_1^{(n)} \\ \hat{u}_2^{(n)} \end{pmatrix} = \begin{pmatrix} b_1 - A_{12} \hat{u}_2^{(n-1)} \\ b_2 \end{pmatrix}, \quad n = 1, 2, 3, \dots, \quad (13)$$

where $\hat{u}_2^{(0)} = 0$.

Explicit expressions for $\hat{u}_1^{(n)}, \hat{u}_2^{(n)}$ are given by

$$\hat{u}_1^{(n)} = A_{11}^{-1} (b_1 - A_{12} \hat{u}_2^{(n-1)}), \quad (14)$$

$$\hat{u}_2^{(n)} = A_{22}^{-1} (b_2 - A_{21} \hat{u}_1^{(n)}), \quad (15)$$

$$= A_{22}^{-1} (b_2 - A_{21} A_{11}^{-1} b_1 + A_{21} A_{11}^{-1} A_{12} \hat{u}_2^{(n-1)}). \quad (16)$$

Noting that (14) and (15) may be written as the following difference equation for $\hat{u}_1^{(n)}$:

$$\hat{u}_1^{(n)} - C\hat{u}_1^{(n-1)} = A_{11}^{-1} (b_1 - A_{12}A_{22}^{-1}b_2), \quad \hat{u}_1^{(1)} = A_{11}^{-1}b_1, \quad (17)$$

we may deduce that the block Gauss–Seidel iteration will converge provided $\|C\| < 1$, which is identical to our assumption in the previous section.

Defining $R_1^{(n)}$ by

$$R_1^{(n)} = A_{11}\hat{u}_1^{(n)} + A_{12}\hat{u}_2^{(n)} - b_1,$$

(14) and (16) may be used to deduce that

$$\begin{aligned} R_1^{(n)} &= A_{12}A_{22}^{-1}A_{21}A_{11}^{-1}R_1^{(n-1)}, \\ &= A_{11}CA_{11}^{-1}R_1^{(n-1)}, \\ &= A_{11}C^{(n-1)}A_{11}^{-1}R_1^{(1)}. \end{aligned}$$

Noting that (13) is of the same form as (4), this may be substituted into (12) to give

$$\begin{aligned} J(u_1, u_2) - J(\hat{u}_1^{(n)}, \hat{u}_2^{(n)}) \\ = - \sum_{k=0}^{\infty} \langle \psi_1, C^{k+n-1}A_{11}^{-1}R_1^{(1)} \rangle + \sum_{k=0}^{\infty} \langle \psi_2, A_{22}^{-1}A_{21}C^{k+n-1}A_{11}^{-1}R_1^{(1)} \rangle, \end{aligned} \quad (18)$$

for $n = 1, 2, \dots$. We see from (12) and (18) that each iteration of Gauss–Seidel is equivalent to removing the leading order term from both of the infinite sums that appear in the error bound given by (12).

2.4. Bounding the error as a finite sum

In this section we write down two finite bounds for the infinite sums given by (12) and (18). In both cases we work with (18), from which we may recover (12) by writing $n = 1$.

For our first bound, we write (18) as

$$\begin{aligned} J(u_1, u_2) - J(\hat{u}_1^{(n)}, \hat{u}_2^{(n)}) \\ = - \langle \psi_1, C^{n-1}A_{11}^{-1}R_1^{(1)} \rangle + \langle \psi_2, A_{22}^{-1}A_{21}C^{n-1}A_{11}^{-1}R_1^{(1)} \rangle - \\ \sum_{k=1}^{\infty} \langle \psi_1, C^{k+n-1}A_{11}^{-1}R_1^{(1)} \rangle + \sum_{k=1}^{\infty} \langle \psi_2, A_{22}^{-1}A_{21}C^{k+n-1}A_{11}^{-1}R_1^{(1)} \rangle, \end{aligned}$$

from which we may deduce that

$$\begin{aligned}
 |J(u_1, u_2) - J(\hat{u}_1^{(n)}, \hat{u}_2^{(n)})| &\leq \left| \langle \psi_1, C^{m-1} A_{11}^{-1} R_1^{(1)} \rangle \right| + \left| \langle \psi_2, A_{22}^{-1} A_{21} C^{m-1} A_{11}^{-1} R_1^{(1)} \rangle \right| + \\
 &\quad \sum_{k=1}^{\infty} \|\psi_1\| \left\| C^{k+n-1} A_{11}^{-1} R_1^{(1)} \right\| + \sum_{k=1}^{\infty} \|\psi_2\| \left\| A_{22}^{-1} A_{21} C^{k+n-1} A_{11}^{-1} R_1^{(1)} \right\|, \\
 &\leq \left| \langle \psi_1, C^{m-1} A_{11}^{-1} R_1^{(1)} \rangle \right| + \left| \langle \psi_2, A_{22}^{-1} A_{21} C^{m-1} A_{11}^{-1} R_1^{(1)} \rangle \right| + \\
 &\quad \sum_{k=1}^{\infty} \|\psi_1\| \left\| C^{k-1} \right\| \left\| C^m A_{11}^{-1} R_1^{(1)} \right\| + \\
 &\quad \sum_{k=1}^{\infty} \|\psi_2\| \left\| A_{22}^{-1} A_{21} \right\| \left\| C^{k-1} \right\| \left\| C^m A_{11}^{-1} R_1^{(1)} \right\|, \\
 &= \left| \langle \psi_1, C^{m-1} A_{11}^{-1} R_1^{(1)} \rangle \right| + \left| \langle \psi_2, A_{22}^{-1} A_{21} C^{m-1} A_{11}^{-1} R_1^{(1)} \rangle \right| + \\
 &\quad \left\| C^m A_{11}^{-1} R_1^{(1)} \right\| \left(\|\psi_1\| + \|\psi_2\| \left\| A_{22}^{-1} A_{21} \right\| \right) \frac{1}{1 - \|C\|}, \tag{19}
 \end{aligned}$$

where we have used our assumption that $\|C\| < 1$ in bounding the infinite sum.

For our second bound, we write (18) as

$$\begin{aligned}
 J(u_1, u_2) - J(\hat{u}_1^{(n)}, \hat{u}_2^{(n)}) &= -\langle \psi_1, C^{n-1} A_{11}^{-1} R_1^{(1)} \rangle + \langle \psi_2, A_{22}^{-1} A_{21} C^{n-1} A_{11}^{-1} R_1^{(1)} \rangle - \\
 &\quad \sum_{k=1}^{\infty} \langle (C^\top)^{k+n-1} \psi_1, A_{11}^{-1} R_1^{(1)} \rangle + \sum_{k=1}^{\infty} \langle (C^\top)^{k+n-1} A_{21}^\top A_{22}^{-\top} \psi_2, A_{11}^{-1} R_1^{(1)} \rangle,
 \end{aligned}$$

from which we may derive the following bound in exactly the same way we derived (19):

$$\begin{aligned}
 |J(u_1, u_2) - J(\hat{u}_1^{(n)}, \hat{u}_2^{(n)})| &\leq \left| \langle \psi_1, C^{n-1} A_{11}^{-1} R_1^{(1)} \rangle \right| + \left| \langle \psi_2, A_{22}^{-1} A_{21} C^{n-1} A_{11}^{-1} R_1^{(1)} \rangle \right| + \\
 &\quad \left\| A_{11}^{-1} R_1^{(1)} \right\| \left(\left\| (C^\top)^n \psi_1 \right\| + \left\| (C^\top)^n A_{21}^\top A_{22}^{-\top} \psi_2 \right\| \right) \frac{1}{1 - \|C\|}. \tag{20}
 \end{aligned}$$

Suppose we normalise the definition of the linear functional given by (3) so that $\|\psi_1\| = \|\psi_2\| = 1$. The bound given by (19) is then independent of the choice of ψ_1, ψ_2 . In contrast, the bound given by (20) depends on the action of a power of C on ψ_1, ψ_2 : this bound will predict different rates of convergence for different linear functionals, and is therefore likely to be a tighter bound.

Both error bounds derived above, (19) and (20), require comparable quantities of work. Both require calculation of the norm of the matrix C , and computation of the vector $C^{n-1} A_{11}^{-1} R_1^{(1)}$. In addition, (19) requires computation of the vector $C^n A_{11}^{-1} R_1^{(1)}$, whilst (20) requires computation of the vectors $(C^\top)^n \psi_1$ and $(C^\top)^n A_{21}^\top A_{22}^{-\top} \psi_2$. The norm of C may be calculated once before the first block Gauss-Seidel iteration. The other terms all require one multiplication by C or C^\top on each iteration. We see from (9) that all of these quantities may

be calculated using the single physics solvers, as these solvers allow us to compute the action of A_{11}^{-1} and A_{22}^{-1} on a vector of the correct dimensions.

3. EXTENDING THE ERROR BOUND FOR NON-LINEAR PROBLEMS

We now consider a coupled, non-linear, multiphysics problem which has been discretised as

$$N_1(u_1, u_2) = 0, \quad N_2(u_1, u_2) = 0, \quad (21)$$

where N_1 is a vector of length M , N_2 is a vector of length N , the discretisation of u_1 has M unknowns, and the discretisation of u_2 has N unknowns. We assume that these discretisations are based on single physics problems, i.e. for given u_2 we may solve the first of these equations to calculate u_1 , and for given u_1 we may solve the second of these equations for u_2 . We define the Jacobian matrices as follows:

$$\begin{aligned} A_{11_{i,j}} &= \frac{\partial N_{1_i}}{\partial u_{1_j}} & i = 1, 2, \dots, M, \quad j = 1, 2, \dots, M, \\ A_{12_{i,j}} &= \frac{\partial N_{1_i}}{\partial u_{2_j}} & i = 1, 2, \dots, M, \quad j = 1, 2, \dots, N, \\ A_{21_{i,j}} &= \frac{\partial N_{2_i}}{\partial u_{1_j}} & i = 1, 2, \dots, N, \quad j = 1, 2, \dots, M, \\ A_{22_{i,j}} &= \frac{\partial N_{2_i}}{\partial u_{2_j}} & i = 1, 2, \dots, N, \quad j = 1, 2, \dots, N. \end{aligned}$$

Let us assume that the first equation is only weakly dependent on u_2 , given a suitable initial approximation $u_2 = u_2^{(0)}$. We may then compute \hat{u}_1 , an approximate solution of the first component, by solving

$$N_1(\hat{u}_1, u_2^{(0)}) = 0, \quad (22)$$

and then compute \hat{u}_2 , an approximate solution to the second component, by solving

$$N_2(\hat{u}_1, \hat{u}_2) = 0. \quad (23)$$

The residuals associated with the solution \hat{u}_1, \hat{u}_2 are defined to be the values given when these quantities are substituted into the left-hand-side of both equations in (21):

$$R_1 = N_1(\hat{u}_1, \hat{u}_2), \quad (24)$$

$$R_2 = N_2(\hat{u}_1, \hat{u}_2), \quad (25)$$

from which we may use (23) to deduce that $R_2 = 0$. Linearising (21) about the solution \hat{u}_1, \hat{u}_2 , we compute the Jacobian matrices at this point and obtain

$$\begin{aligned} N_1(u_1, u_2) &= N_1(\hat{u}_1, \hat{u}_2) + A_{11}(u_1 - \hat{u}_1) + A_{12}(u_2 - \hat{u}_2), \\ N_2(u_1, u_2) &= N_2(\hat{u}_1, \hat{u}_2) + A_{21}(u_1 - \hat{u}_1) + A_{22}(u_2 - \hat{u}_2). \end{aligned}$$

Using (21), (23), (24) and (25) we may write these equations as

$$\begin{aligned} R_1 + A_{11}(u_1 - \hat{u}_1) + A_{12}(u_2 - \hat{u}_2) &= 0, \\ A_{21}(u_1 - \hat{u}_1) + A_{22}(u_2 - \hat{u}_2) &= 0, \end{aligned}$$

which allow us to deduce that

$$\begin{aligned} u_2 - \hat{u}_2 &= -A_{22}^{-1} A_{21} (u_1 - \hat{u}_1), \\ u_1 - \hat{u}_1 &= -A_{11} A_{12} (u_2 - \hat{u}_2) - A_{11}^{-1} R_1, \end{aligned}$$

which are identical to (6) and (7). Subject to our linearisation being appropriate we may then apply the error bounds given by (19) and (20) to non-linear problems, although we note that the Jacobian matrices — and the matrix C — will now change on each block Gauss-Seidel iteration.

We now demonstrate application of the bounds given by (19) and (20).

4. EXAMPLES FROM LINEAR ALGEBRA

For our first example we consider the following matrices:

$$A_{11} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}; \quad A_{12} = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}; \quad A_{21} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}; \quad A_{22} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix},$$

and right-hand-side $b = (1 \ 1 \ 0 \ 0)^\top$. Using these matrices, C is then given by

$$C = \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}.$$

C has eigenvalues λ_1, λ_2 , and so $\|C\| = \max(\lambda_1, \lambda_2)$. The residual calculated from the block Gauss-Seidel iteration will converge provided $\lambda_1, \lambda_2 < 1$. We consider two linear functionals, J_1, J_2 that are both of the form (3). These residuals are given by

$$\begin{aligned} J_1(u_1, u_2) &= \langle \psi_{1,1}, u_1 \rangle, & \text{where } \psi_{1,1} &= (1 \ 0)^\top, \\ J_2(u_1, u_2) &= \langle \psi_{1,2}, u_1 \rangle, & \text{where } \psi_{1,2} &= (0 \ 1)^\top. \end{aligned}$$

In this case, the true solution is

$$J_1(u_1, u_2) = \frac{1}{1 - \lambda_1}, \quad J_2(u_1, u_2) = \frac{1}{1 - \lambda_2}.$$

We may also deduce, using (17), that on Gauss-Seidel iteration n these functionals are approximated by

$$J_1(\hat{u}_1^{(n)}, \hat{u}_2^{(n)}) = \frac{1 - \lambda_1^n}{1 - \lambda_1}, \quad J_2(\hat{u}_1^{(n)}, \hat{u}_2^{(n)}) = \frac{1 - \lambda_2^n}{1 - \lambda_2},$$

allowing us to deduce that

$$J_1(\hat{u}_1^{(n)}, \hat{u}_2^{(n)}) - J_1(u_1, u_2) = \frac{\lambda_1^n}{1 - \lambda_1}, \quad (26)$$

$$J_2(\hat{u}_1^{(n)}, \hat{u}_2^{(n)}) - J_2(u_1, u_2) = \frac{\lambda_2^n}{1 - \lambda_2}, \quad (27)$$

and so our block Gauss-Seidel estimate for J_1 converges at rate λ_1 , whilst our estimate for J_2 converges at rate λ_2 . By explicitly evaluating the terms in (19) and (20) in this simple

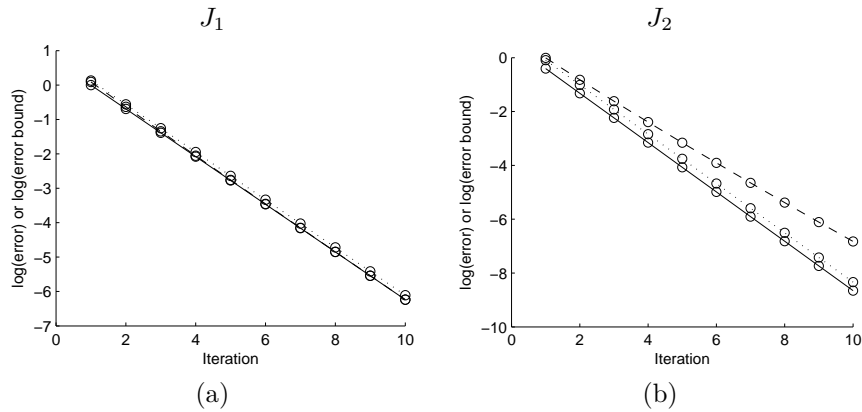


Figure 1. (a) The absolute error in J_1 , and (b) the absolute error in J_2 as a function of block Gauss–Seidel iteration number for $\lambda_1 = 0.5, \lambda_2 = 0.4$. In both plots, the solid line refers to the true value, the broken line to the value computed using (19), and the dotted line to the value computed using (20).

case we see that (19) always predicts convergence at rate λ_1 , regardless of the choice of linear functional. However, (20) predicts that J_1 will converge at rate λ_1 and J_2 will converge at rate λ_2 in agreement with (26) and (27). We now demonstrate this with the aid of numerical computations for this system.

For our first simulation, we use $\lambda_1 = 0.5, \lambda_2 = 0.4$. We plot the absolute error in J_1 as a function of iteration number as the solid line in Figure 1(a). In this figure, the broken line corresponds to the bound calculated by (19), and the dotted line corresponds to the bound calculated by (20). We see that both bounds are very sharp — this is because both bounds predicts that the error is decreasing by the true factor of λ_1 on each iteration. In Figure 1(b) we give the corresponding plot for the absolute error in J_2 . We see that in this case, the bound computed by (20) is sharp, but the bound computed by (19) is not as sharp, particularly as the number of iterations increases. This is because, as discussed above, the bound given by (19) predicts that the error is decreasing at rate λ_1 on each iteration, whilst (20) predicts that the error is decreasing at the true rate of λ_2 on each iteration.

We now repeat the simulations above for $\lambda_1 = 0.99, \lambda_2 = 0.01$. In Figure 2(a) we plot the absolute error in J_1 , and in Figure 2(b) we plot the absolute error in J_2 : again, the solid line refers to the true value, the broken line to the value computed using (19), and the dotted line to the value computed using (20). As for the computation shown in Figure 1, both error bounds are sharp for calculation of J_1 . However, for calculation of J_2 , the error bound given by (19) is not much use. The error bound given by (20) performs better, but is less sharp than in Figure 1. This is in agreement with the remarks made on Figure 1 on the rate of convergence of (19) and (20) as a function of λ_1, λ_2 . However, in this case the differences are more marked as the ratio λ_1/λ_2 is much larger.

In both Figures 1 and 2 we have seen that the calculation of the linear functional corresponding to the largest eigenvalue leads to a very sharp bound, whilst the calculation of the linear functional corresponding to the smallest eigenvalue yields a less sharp bound, especially in the limit $\lambda_1/\lambda_2 \rightarrow \infty$, although the bound given by (20) performs better than

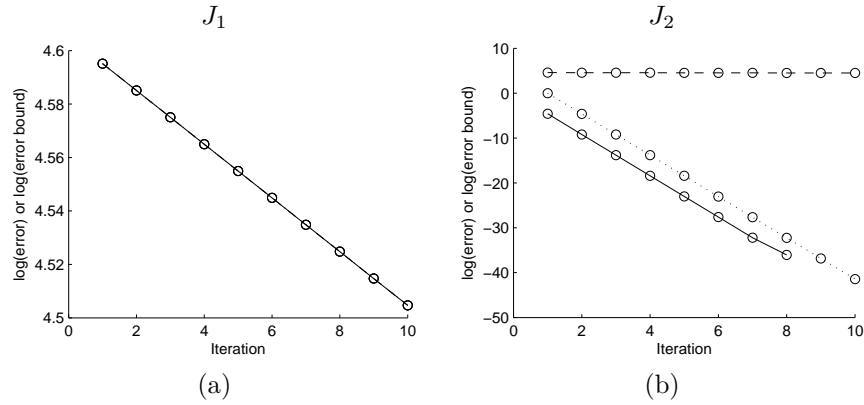


Figure 2. (a) The absolute error in J_1 , and (b) the absolute error in J_2 as a function of block Gauss–Seidel iteration number for $\lambda_1 = 0.99, \lambda_2 = 0.01$. In both plots, the solid line refers to the true value, the broken line to the value computed using (19), and the dotted line to the value computed using (20).

the bound given by (19).

5. A LINEAR DIFFERENTIAL EQUATION EXAMPLE

To investigate the bounds given by (19) and (20) for the numerical discretisation of a differential equation we use the model problem given by, for $-1 < x < 1$,

$$-\frac{d^2 u_1}{dx^2} + \alpha \frac{du_2}{dx} = f_1, \quad (28)$$

$$-\frac{du_1}{dx} - \frac{d^2 u_2}{dx^2} = f_2, \quad (29)$$

for specified α , where

$$\begin{aligned} f_1 &= \pi^2 \sin \pi x + 2\alpha + \frac{\alpha}{\delta} \left(\frac{4(x+1)^2}{\delta} - 2 \right) e^{-(x+1)^2/\delta}, \\ f_2 &= -\pi \cos \pi x + 2\alpha x - \frac{4(x+1)}{\delta^2} \left(3 - \frac{2(x+1)^2}{\delta} \right) e^{-(x+1)^2/\delta}, \end{aligned}$$

for specified δ , and with boundary conditions

$$u_1 = u_2 = 0, \quad x = -1, 1.$$

This system of differential equations has, for sufficiently small δ , approximate solution

$$\begin{aligned} u_1 &= \sin \pi x + \alpha (1 - x^2), \\ u_2 &= -\frac{2}{\delta} (x+1) e^{-(x+1)^2/\delta}. \end{aligned}$$

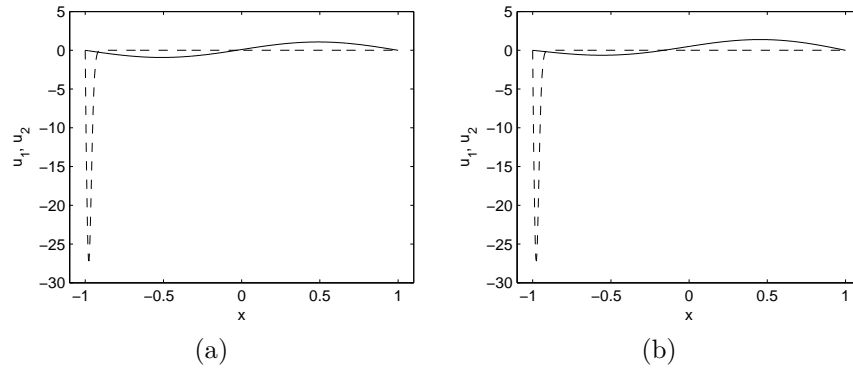


Figure 3. The solution of the differential equations given by (28), (29). In (a) $\alpha = 0.1, \delta = 0.001$, in (b) $\alpha = 0.5, \delta = 0.001$. In both cases u_1 is given by the solid line, and u_2 is given by the broken line.

We plot the solution for $\alpha = 0.1, \delta = 0.001$ in Figure 3(a), and the solution for $\alpha = 0.5, \delta = 0.001$ in Figure 3(b).

Clearly in the limit $\alpha \rightarrow 0$ the solution of (28) and (29) for u_1 becomes less dependent on u_2 , and so we may apply the bounds given by (19) and (20) to discretisations of these equations. We consider two linear functionals:

$$J_1 = u'_1(-1), \quad J_2 = u'_2(-1).$$

We investigate three different finite element discretisations of the differential equations: (i) a linear approximation on each element that is continuous across element boundaries; (ii) a quadratic approximation on each element that is continuous across element boundaries; and (iii) a cubic approximation on each element that is continuous across element boundaries. In the graphs shown in Figure 4 we investigate the error bounds for $\alpha = 0.1, \delta = 0.001$. In these graphs, the absolute error in J_1, J_2 as a function of block Gauss–Seidel iteration number is represented by the solid line, the error bound given by (19) by the broken line, and the error bound given by (20) by the dotted line. Figure 4(a),(b) show the absolute error in J_1, J_2 when linear elements are used, Figure 4(c),(d) show the absolute error in J_1, J_2 when quadratic elements are used, and Figure 4(e),(f) show the absolute error in J_1, J_2 when cubic elements are used. We emphasise that when we use the term “absolute error” we are referring to the error between the block Gauss–Seidel solution and the fully coupled discretised solution, and are assuming that discretisation error may be neglected. In these graphs, we see that: (i) the error bound given by (20) is much sharper than the error bound given by (19); and (ii) the sharpness of both error bounds seems to be unaffected by the choice of finite element discretisation. Note that, in all plots in Figure 4, the absolute error in J_1, J_2 tails off at a constant value after a certain number of block Jacobi iterations. This is due to the precision of the linear solver reaching machine precision.

Figure 5 shows the corresponding plots to Figure 4 for $\alpha = 0.5, \delta = 0.001$. As in Figure 4, we see that the error bound given by (20) is much sharper than the error bound given by (19), and also that the sharpness of both error bounds seems to be unaffected by the choice of finite element discretisation. In this case, however, both bounds are less sharp: this is because as

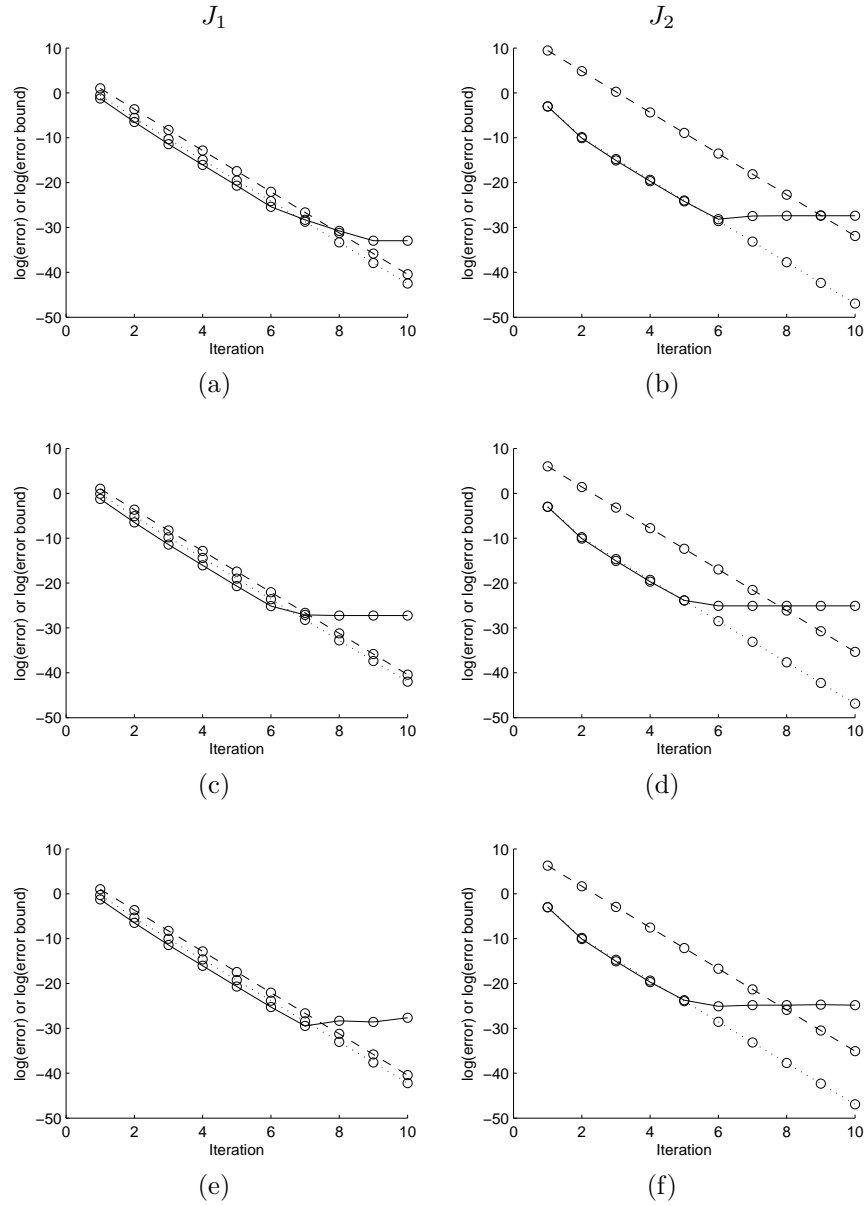


Figure 4. (a), (c), (e) The absolute error in J_1 , and (b), (d), (f) the absolute error in J_2 as a function of block Gauss-Seidel iteration number for $\alpha = 0.1, \delta = 0.001$. In (a), (b) linear finite elements were used to discretise (28), (29), in (c), (d) quadratic finite elements were used, and in (e), (f) cubic finite elements were used. In all plots, the solid line refers to the true value, the broken line to the value computed using (19), and the dotted line to the value computed using (20).

$\alpha = 0.5$, we can no longer claim that the two differential equations (28) and (29) are weakly coupled.

6. A NON-LINEAR FLUID-TEMPERATURE EXAMPLE

Our final example is a non-linear coupled fluid-temperature model, where the fluid flow is described by Stokes Flow, with viscosity dependent on temperature. We model flow through a channel occupying the region $0 < x, y < 1$, with the ends $x = 0, 1$ open, and with fixed walls at $y = 0, 1$. The governing equations are then given by, for $0 < x, y < 1$,

$$\nabla \cdot (\mu(T) \nabla u) - \frac{\partial p}{\partial x} = 0, \quad (30)$$

$$\nabla \cdot (\mu(T) \nabla v) - \frac{\partial p}{\partial y} = 0, \quad (31)$$

$$\nabla \cdot \mathbf{u} = 0, \quad (32)$$

$$\mathbf{u} \cdot \nabla T = D \nabla^2 T, \quad (33)$$

where $\mathbf{u} = (u, v)^\top$ is the velocity, p the pressure, T the temperature, μ the coefficient of viscosity, and D is a constant. We impose no-slip boundary conditions and a specified temperature on the fixed walls $y = 0, 1$. At the inflow boundary $x = 0$ we impose traction boundary conditions in the direction of the axis of the channel, no flow perpendicular to the axis of the channel, and no flux of heat out of the channel. At the outflow boundary $x = 1$ we impose traction boundary conditions in the direction of the axis of the channel, zero traction perpendicular to the axis of the channel, and no flux of heat out of the channel. These conditions may be written

$$\mathbf{u} = \mathbf{0}, \quad y = 0, 1, \quad 0 < x < 1, \quad (34)$$

$$T = T_0(x, y), \quad y = 0, 1, \quad 0 < x < 1, \quad (35)$$

$$2\mu \frac{\partial u}{\partial x} - p = -p_0, \quad x = 0, \quad 0 < y < 1, \quad (36)$$

$$v = 0, \quad x = 0, \quad 0 < y < 1, \quad (37)$$

$$2\mu \frac{\partial u}{\partial x} - p = -p_1, \quad x = 1, \quad 0 < y < 1, \quad (38)$$

$$\mu \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) = 0, \quad x = 1, \quad 0 < y < 1, \quad (39)$$

$$\frac{\partial T}{\partial x} = 0, \quad x = 0, 1, \quad 0 < y < 1. \quad (40)$$

for specified tractions p_0, p_1 , and temperature on the boundary walls $T_0(x, y)$. We require $p_0 > p_1$ to ensure that the end $x = 0$ is the inflow boundary of the channel. To close this system we must specify a relationship between the temperature and coefficient of viscosity. Several relationships exist: for illustration purposes we use the exponential relationship [9] given by

$$\mu(T) = \mu_0 + \exp(-\mu_b T), \quad (41)$$

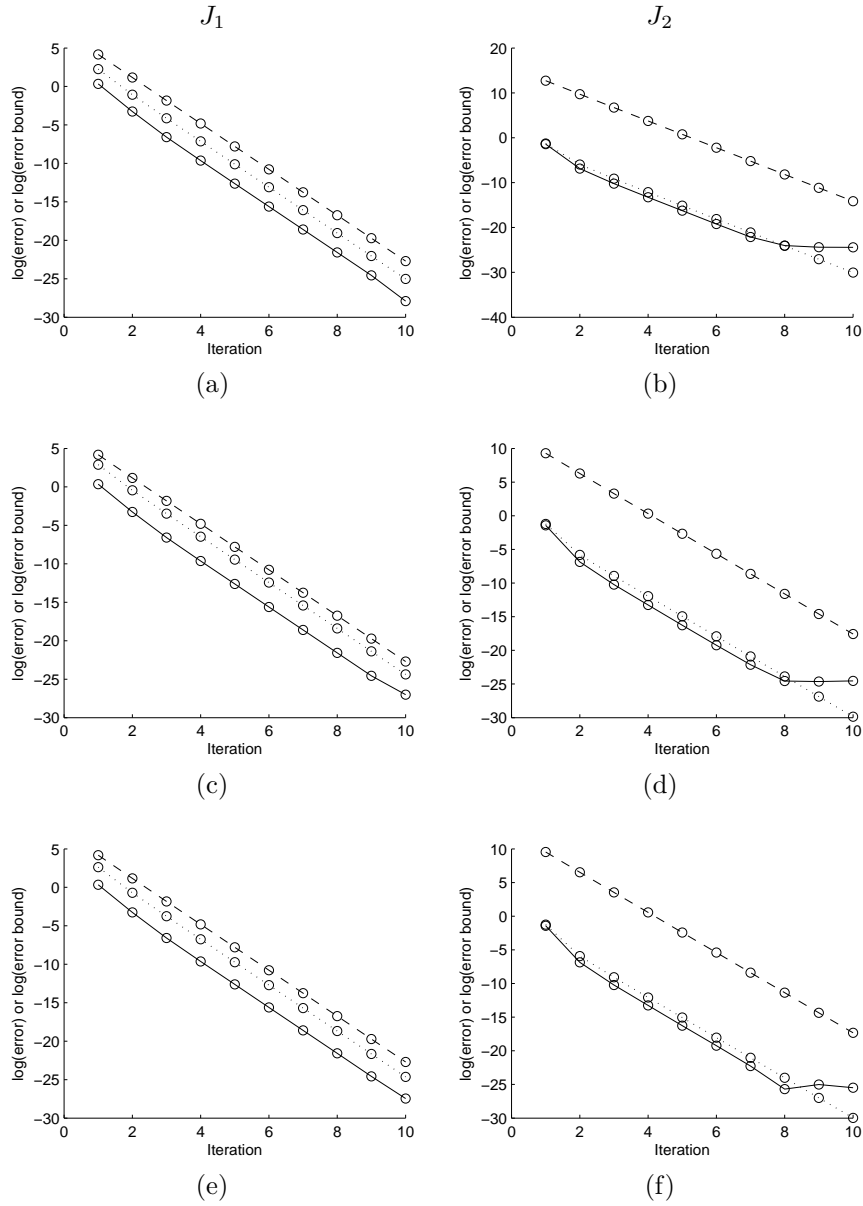


Figure 5. (a), (c), (e) The absolute error in J_1 , and (b), (d), (f) the absolute error in J_2 as a function of block Gauss-Seidel iteration number for $\alpha = 0.5, \delta = 0.001$. In (a), (b) linear finite elements were used to discretise (28), (29), in (c), (d) quadratic finite elements were used, and in (e), (f) cubic finite elements were used. In all plots, the solid line refers to the true value, the broken line to the value computed using (19), and the dotted line to the value computed using (20).

for prescribed constants μ_0, μ_b . We will consider the case where the viscosity is only weakly dependent on the temperature. This allows us to consider (30)–(33) as two coupled multiphysics models. Under the assumption of weak dependence of viscosity on temperature we may first solve the fluids component, (30)–(32), of this model using an approximate guess for the viscosity of the form (41), and then solve (33) to compute the temperature field. Assuming that the block Gauss–Seidel iteration converges, we may repeat this procedure until we are satisfied that our solution is sufficiently accurate. This allows us to investigate the validity of (19) and (20) for non-linear problems.

6.1. Details of computations

We compute a numerical solution of the fully coupled system described by (30)–(41) using the finite element method. The computational domain is divided into a uniform grid of 40×40 elements. On each element we make a quadratic approximation to both components of the velocity, and a linear approximation to the pressure and temperature. These approximations are continuous across boundaries between neighbouring elements. The resulting system of non-linear equations are solved using Newton’s method. See [10] for more details on this numerical method.

To pose this problem as a coupled multiphysics problem, we first make an estimate for the temperature field. We then use the finite element method described above for computing a numerical approximation to \mathbf{u} and p . The numerical approximation to T is then computed by using both the approximation to \mathbf{u} already calculated, and the finite element method described above.

In our simulations we set

$$T_0(x, y) = x^3,$$

and use the following parameters: $\mu_0 = 1$; $\mu_b = 0.1$; $p_0 = 20$; $p_1 = 0$; $D = 1$. As $T_0(x, y)$ varies from 0 to 1, our initial guess for T is $T(x, y) = 0.5$. The numerical solutions of the fully coupled problem for u, v, p and T are shown in Figure 6.

In our computations we investigate the efficacy of the error bounds (19) and (20) when computing six linear functionals of \mathbf{u}, p, T :

$$J_1 = \int_0^1 u(x, 1) \, dx; \tag{42}$$

$$J_2 = u(1, 0.5); \tag{43}$$

$$J_3 = u_y(0.5, 0); \tag{44}$$

$$J_4 = v(1, 0.75); \tag{45}$$

$$J_5 = \int_{x,y=0}^1 T \, dx \, dy; \tag{46}$$

$$J_6 = T_y(1, 0). \tag{47}$$

These functionals may be interpreted physically as follows: J_1 is the flux out of the outflow boundary; J_2 corresponds to the value of u at the centre of the outflow boundary; J_3 is the partial derivative of u with respect to y on the channel wall, and would be required if the stress on the channel wall were being calculated; J_4 is the value of v at a point on the outflow boundary; J_5 is the average temperature within the channel; and J_6 is the partial derivative of

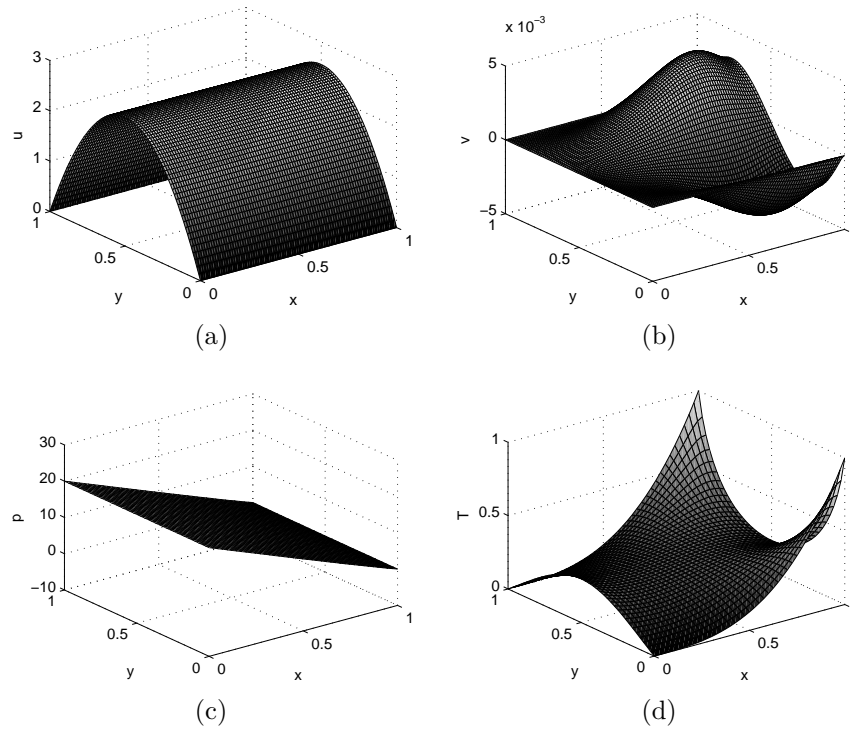


Figure 6. The solution of equations (30)–(41) using the parameters given in Section 6.1. (a) shows u ; (b) shows v ; (c) shows p ; and (d) shows T .

T on the channel wall, where the true solution takes its steepest value. The partial derivatives occurring in these functionals are approximated by differentiating the finite element solution.

6.2. Computational results

The error in the linear functionals defined by (42)–(47), and the errors predicted by the bounds (19) and (20) are given in Figure 7. In all plots the solid line refers to the true magnitude of the absolute error, the broken line to the value computed using (19), and the dotted line to the value computed using (20). For all linear functionals considered here, in common with the previous examples, we see that the bound (20) is much sharper than the bound predicted by (19).

There are two differences between the performance of our error bounds for the non-linear problem, and the performance of these bounds for the linear problems considered earlier. The first is that the logarithm of the true error does not decrease as uniformly. The second difference is that in some cases the bound given by (20) is not strict: this is due to the assumption that the problem may be linearised at each iteration. However, the violation of the bound never affects accuracy significantly for the simulations presented here.

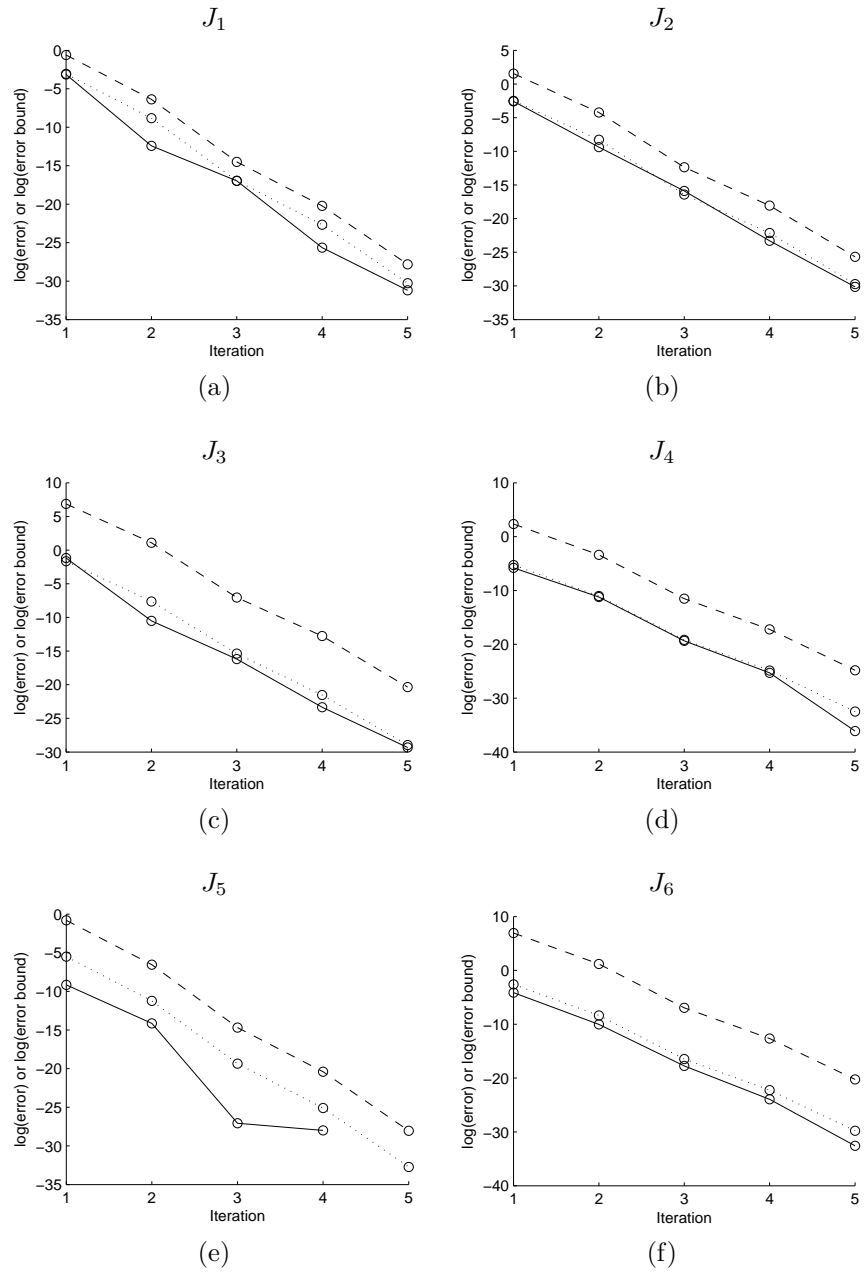


Figure 7. (a) The absolute error in J_1 (b) the absolute error in J_2 , (c) the absolute error in J_3 , (d) the absolute error in J_4 (e) the absolute error in J_5 , and (f) the absolute error in J_6 as a function of block Gauss-Seidel iteration number. $J_1, J_2, J_3, J_4, J_5, J_6$ are defined by (42)–(47). In all plots, the solid line refers to the true value, the broken line to the value computed using (19), and the dotted line to the value computed using (20).

7. DISCUSSION

As explained in Section 1, the aim of many simulations is to compute a linear functional of the solution of a coupled multiphysics problem that is modelled by a system of partial differential equations. We assume that appropriate computational grids for all components of the model exist. This is usually true for coupled multiphysics models: if grids do not exist then methods exist for generating them using *a posteriori* error analysis [7]. A numerical solution to the whole coupled problem is often computed using a block Gauss–Seidel iteration. Even if the computational grid is sufficiently fine that error due to numerical discretisation may be neglected, it is still possible that significant error may exist through the passing of the solution of one component to another model.

In this study we have derived two bounds on this linear functional that may be used when the coupled problems is solved using solvers for the individual physics problems using a block Gauss–Seidel iteration. These bounds were dependent on the linear functional of interest. We first investigated these bounds using model problems drawn from linear algebra and differential equations. Using these model problems we found that one of our bounds — that given by (20) — gave a much sharper bound than that given by (19). Furthermore, we tested these bounds for a selection of different numerical methods and found that both bounds were not significantly affected by choice of method. We then investigated the performance of the bounds using a coupled, non-linear, fluid–temperature problem. As with the model problems, the bound given by (20) has been found to be reasonably sharp, and can therefore be used to predict whether or not a given iteration of block Gauss–Seidel iteration is sufficiently accurate that the iteration may be terminated.

ACKNOWLEDGEMENT

This publication is based on work supported by Award No. KUK-C1-013-04, made by King Abdullah University of Science and Technology (KAUST).

REFERENCES

1. Bungartz H-J, Schäfer M. *Fluid–Structure Interaction*. Springer: Berlin, 2007.
2. Hunter PJ, Pullan AJ, Smaill BH. Modelling total heart function. *Annual Reviews in Biomedical Engineering* 2003; **5**: 147–177.
3. Joosten MM, Dettmer WG, Perić D. Analysis of the block Gauss–Seidel solution procedure for a strongly coupled model problem with reference to fluid–structure interaction. *International Journal for Numerical Methods in Engineering* 2009; **78**: 757–778.
4. Heil M. An efficient solver for the fully-coupled solution of large-displacement fluid–structure interaction problems. *Computer Methods in Applied Mechanics and Engineering* 2004; **193**: 1–23.
5. Sundnes J, Nielsen BF, Mardal KA, Cai X, Lines GT, Tveito A. On the computational complexity of the bidomain and monodomain models of electrophysiology. *Annals of Biomedical Engineering* 2006; **34**: 1088–1097.
6. Southern JA, Plank G, Vigmond EJ, Whiteley JP. Solving the coupled system improves computational efficiency of the bidomain equations. *IEEE Transactions on Biomedical Engineering* 2009; **56**: 2404–2009.
7. Carey V, Estep, D, Tavenor S. A posteriori analysis and adaptive error control for multiscale operator decomposition solution of elliptic systems I: triangular systems. *SIAM Journal on Numerical Analysis* 2009; **47**: 740–761.
8. Trefethen LN, Bau D. *Numerical Linear ALgebra*. SIAM: Philadelphia, 1997.

9. Capone F, Gentile M. Nonlinear stability analysis of convection for fluids with exponentially temperature-dependent viscosity. *Acta Mechanica* 1994; **107**: 53–64.
10. Elman HC, Silvester DJ, Wathen AJ. *Finite Elements and Fast Iterative Solvers*. Oxford University Press: Oxford UK, 2005.

RECENT REPORTS

09/10	Anticavitation and differential growth in elastic shells	Moulton Goriely
10/10	On the mechanical stability of growing arteries	Goriely Vandiver
11/10	Nonlinear Correction to the Euler Buckling Formula for Compressible Cylinders	De Pascalis Destrade Goriely
12/10	Nonlinear Morphoelastic Plates I: Genesis of Residual Stress	McMahon Goriely Tabor
13/10	Nonlinear Morphoelastic Plates II: Exodus to Buckled States	McMahon Goriely Tabor
14/10	Analysis of Brownian dynamics simulations of reversible biomolecular reactions	Lipkova Zygalakis Chapman Erban
15/10	Travelling waves in hyperbolic chemotaxis equations	Xue Hwang Painter Erban
16/10	The Physics and Mechanics of Biological Systems	Goriely Moulton
17/10	Crust formation in drying colloidal suspensions	Style Peppin
18/10	A Mathematical Model of Tumor-Immune Interactions	Robertson-Tessi El-Kareh Goriely
19/10	Elastic cavitation, tube hollowing, and differential growth in plants and biological tissues	Goriely Moulton Vandiver
20/10	Asymptotic expressions for the nearest and furthest dislocations in a pile-up against a grain boundary	Hall
21/10	Cardiac electromechanics: the effect of contraction model on the mathematical problem and accuracy of the numerical scheme	Pathmanathan Chapman Gavaghan Whiteley
22/10	Fat vs. thin threading approach on GPUs: application to stochastic simulation of chemical reactions	Klingbeil Erban Giles Maini
23/10	Asymptotic analysis of a system of algebraic equations arising in dislocation theory	Hall Chapman Ockendon

26/10	On an evolution equation for sand dunes	Ellis Fowler
27/10	On Liquid Films on an Inclined Plate	Benilov Chapman McLoed Ockendon Zubkov
28/10	An a posteriori error analysis of a mixed finite element Galerkin approximation to second order linear parabolic problems	Memon Nataraj Pani
29/10	A Priori Error Estimates for Semidiscrete Finite Element Approximations to Equations of Motion Arising in Oldroyd Fluids of Order One	Goswami Pani
30/10	The Landau-de Gennes theory of nematic liquid crystals: Uniaxiality versus Biaxiality	Majumdar
31/10	The Radial-Hedgehog Solution in Landau-de Gennes' theory	Majumdar
32/10	Nonlinear instability in flagellar dynamics: a novel modulation mechanism in sperm migration?	Gadelha Gaffney Smith Kirkman-Brown

Copies of these, and any other OCCAM reports can be obtained from:

**Oxford Centre for Collaborative Applied Mathematics
Mathematical Institute
24 - 29 St Giles'
Oxford
OX1 3LB
England**

www.maths.ox.ac.uk/occam