ESAIM: M2AN
Vol. 39, N° 5, 2005, pp. 909–929
DOI: 10.1051/m2an:2005040

ESAIM: Mathematical Modelling and Numerical Analysis

SIMPLIFYING NUMERICAL SOLUTION OF CONSTRAINED PDE SYSTEMS THROUGH INVOLUTIVE COMPLETION

Bijan Mohammadi[1] and Jukka Tuomela[1]

Abstract. When analysing general systems of PDEs, it is important first to find the involutive form of the initial system. This is because the properties of the system cannot in general be determined if the system is not involutive. We show that the notion of involutivity is also interesting from the numerical point of view. The use of the involutive form of the system allows one to consider quite general situations in a unified way. We illustrate our approach on the numerical solution of several flow equations with the aim of showing the impact of the involutive form of the systems in simplifying numerical schemes.

Mathematics Subject Classification. 35G15, 35N10, 65M60, 65N30.

Received: September 28, 2004.

1. Introduction

Since various systems of PDEs can have very different properties, it may seem hopeless to try to treat arbitrary systems of PDEs in a unified and meaningful way. However, the so called *formal theory of PDEs* [22] provides an interesting approach to the analysis of the given arbitrary system of PDEs. The history of the formal theory is more than hundred years old and mathematicians like Riquier, Élie Cartan, Spencer, Kuranishi and many others have contributed to it. The intuitive idea is to try determine as many qualitative properties as possible of the system before fixing some function spaces where the solutions and data are required to be. One consequence of this point of view is the emergence of an important concept, the *involutive form* (also called completed form) of the given system. The technical definition of this notion is quite complicated, but essentially this means that one has to find all the integrability conditions of the given system. This may be easy in some cases, but in others it may be require the use of symbolic computation.

Anyway it turns out that determining the properties of the system is in general possible only if the system is involutive. For example some systems may not be elliptic initially, but their involutive forms are elliptic (we give examples below). Now if one wants to compute a numerical solution to a given problem, it is of course essential to know if the system is elliptic or not, and consequently some structural analysis of the system is necessary before choosing any functional analytic framework for the problem. Moreover this preliminary analysis may also suggest what is in fact the proper framework.

The framework of formal theory has been used in [23–25] in the numerical solution of ODE (or DAE) systems. In this article we show how the formal theory and in particular the notion of involutivity is useful in the numerical

Keywords and phrases. Overdetermined PDEs, involution, discretization.

[1] Mathematics and Modeling Institute, Montpellier University, France and Department of Mathematics, University of Joensuu, PO Box 111, 80101 Joensuu, Finland. bijan.mohammadi@math.univ-montp2.fr; jukka.tuomela@joensuu.fi

solution of PDEs. Some preliminary results of this approach were presented in [15]. Here we consider only linear PDEs, but the formal theory works also in the nonlinear case.

The approach by formal theory is helpful especially in situations where the physical models have constraints or conserved quantities which make the system essentially overdetermined. This is a very frequent situation. However, usually in numerical computations one uses square models (as many equations as unknowns). But then if one "forces" the system to be square by dropping some relevant equations/constraints one may encounter great difficulties in designing appropriate numerical methods because the methods should then take into account the properties of the system which are only implicitly represented in the system. We propose to use the involutive form, *i.e.* all relevant information is explicitly available. Thus it is necessary to consider also nonsquare systems when analysing the properties of the system. For numerical compuations, however, we propose to "return" to square systems, not by dropping some equations, but by introducing some auxiliary variables. The resulting system is called *augmented system*. This may sound like an ad hoc trick but in fact this can be done in a canonical way using the compatibility operator which is naturally associated to the involutive system.

The contents of our article is as follows. In Section 2 we give some necessary background to our approach. Then in Section 3 we consider two simple test problems where one can clearly see essential features of our approach. In Section 4 we analyse the properties of the augmented systems. In particular it is seen that the auxiliary variables can in fact be useful from numerical point of view. In Section 5 we take up the Stokes problem. One surprising consequence is that inf-sup or LBB condition for the Stokes problem disappears: in the augmented system we can use finite elements of the same order for the velocity and pressure. In Section 6 we study a compressible flow problem. The numerical results show that quite straightforward numerical schemes work when using the involutive form while classical approach needs quite elaborate specific schemes. Finally in Section 7 we conclude with some perspectives for future research.

One of the main issues we would like to insist on is that the completed/augmented form of the system will be solvable with simpler and more generic numerical methods than the original system. Indeed, complex numerical methods are often required to recover what is missed by not considering the right continuous system. The completed system allows the use of generic solvers such as `Femlab` [7] and `FreeFem` [8] which are now commonly available. In this paper we have therefore used only such generic solvers to show the interest of switching to the completed form of the PDE system before discretization.

2. Preliminaries

2.1. Some notation

Let $\Omega \subset \mathbb{R}^n$ be a domain with coordinates $x = (x_1, \dots, x_n)$ and let $y = (y^1, \dots, y^m)$ be a map $y : \Omega \to \mathbb{R}^m$. The boundary of Ω is denoted by Γ. We suppose for simplicity that Ω is open and bounded, and that Γ is sufficiently smooth. Let $\mu \in \mathbb{N}^n$ be a multi-index and let $|\mu| = \mu_1 + \cdots + \mu_n$. $\mathbf{1}_j$ is a multi index whose jth component is one and other components are zero. The derivatives of y are denoted by

$$\frac{\partial^{|\mu|} y}{\partial x_1^{\mu_1} \cdots \partial x_n^{\mu_n}} = \frac{\partial^{|\mu|} y}{\partial x^\mu} = \partial^\mu y = y_\mu.$$

We order the multi-indices and derivatives with the degree reverse lexicographic order:

$$\alpha < \mu \quad \text{if} \quad \begin{cases} |\alpha| < |\mu| \ , \ \text{or} \\ |\alpha| = |\mu| \ , \ \alpha_i = \mu_i \ \text{ for } \ 1 \le i < j \text{ and } \alpha_j > \mu_j \end{cases} \tag{2.1}$$

$$y^i_\alpha < y^j_\mu \quad \text{if} \quad \alpha < \mu \quad \text{and} \quad y^i_\mu < y^j_\mu \quad \text{if} \quad i > j.$$

The *class* of the multi-index μ and derivative y^j_μ is k, if $\mu_1 = \cdots = \mu_{k-1} = 0$ and $\mu_k \neq 0$.

Let $E = \Omega \times \mathbb{R}^m$. We denote the usual Sobolev spaces of maps $y : \Omega \to \mathbb{R}^m$ of order α by $H_\alpha(E)$. The corresponding norm is denoted by $||y||_\alpha$. Similar notation is used for maps defined on the boundary Γ.

2.2. **Elliptic systems**

A general linear qth order PDE can be written as

$$Ay = \sum_{|\mu| \le q} a_\mu(x) \partial^\mu y = f \tag{2.2}$$

where $a_\mu(x)$ are matrices of size $k \times m$ and the component functions of a_μ are sufficiently smooth. The *principal symbol* of A is

$$\sigma A = \sum_{|\mu| = q} a_\mu(x) \xi^\mu.$$

Recall that the principal symbol is coordinate independent. This is because we may view ξ as a one form which is defined in a coordinate free way. Now fixing x and ξ, σA can be interpreted as a linear map $\mathbb{R}^m \to \mathbb{R}^k$.

Definition 2.1. A is elliptic in Ω, if σA is injective for all real $\xi \neq 0$ and for all $x \in \Omega$.

The notion of ellipticity has been generalized by Douglis and Nirenberg [1,5]. They introduced two sets of weights (integers) as follows: let s_i be the weights for the equations, $1 \le i \le k$, and t_j the weights for the unknowns, $1 \le j \le m$. They must be chosen such that

$$s_i + t_j \ge q_{ij}$$

where q_{ij} is the maximal order of a derivative of the jth unknown function in the ith equation.

Definition 2.2. The *weighted (principal) symbol* of the differential operator A is

$$\big(\sigma_w A\big)_{i,j} = \sum_{|\mu| = s_i + t_j} \big(a_\mu(x)\big)_{i,j} \xi^\mu.$$

A is DN-elliptic (elliptic in the sense of Douglis and Nirenberg), if $\sigma_w A$ is injective for all real $\xi \neq 0$.

We get ordinary ellipticity as a special case, if we choose

$$s_1 = \cdots = s_k = 0 \quad \text{and} \quad t_1 = \cdots = t_m = q.$$

Note that without loss of generality we may always suppose, if convenient, that

$$s_1 \le s_2 \le \cdots \le s_k = 0 \quad \text{and} \quad t_1 \ge t_2 \ge \cdots \ge t_m \ge 1.$$

In fact the class of DN-elliptic systems is not really bigger than the class of elliptic systems: in [13] it is shown that any DN-elliptic system becomes elliptic when completed to the involutive form. The apparent generality is just the result of restricting the attention to square systems.

To get a well-posed problem we need to impose correct boundary conditions. In the present paper we need to formulate this only for square systems. In this case we can define

$$p_A(\xi) = \det\big(\sigma_w A\big).$$

It follows that p_A (which will be called the *characteristic polynomial* of A) is a homogeneous polynomial of degree $\sum_{i=1}^m s_i + t_i$.

Now let us choose the coordinates x such that the boundary is given in these coordinates by the equation $x_n = 0$ in the neighborhood of some boundary point $(x', 0)$. Similarly let us denote the dual variables by $\xi = (\xi', \zeta)$. So fixing some point $(x', 0)$ and some $\xi' \neq 0$ we may view p_A as a polynomial in ζ.

Definition 2.3. The operator A is *properly DN-elliptic*, if it is DN-elliptic and if its characteristic polynomial p_A at all boundary points and for all $\xi' \neq 0$ has equally many roots in the upper and lower half of the complex plane.

It follows that if A is properly DN-elliptic, the degree of p_A must be even and in this case we denote $\sum_{i=1}^{m} s_i + t_i = 2\nu$. Note that in fact this definition is needed only in case $n = 2$ because it can be shown that if $n > 2$, ellipticity implies proper ellipticity. Then we have the following fact:

Theorem 2.1. *Let A be a square properly DN-elliptic operator whose characteristic polynomial has degree 2ν. Then the correct number of boundary conditions is ν.*

So consider the boundary operator $By = g$ where B is of size $\nu \times m$. To formulate the criterion for the correct kind of boundary conditions we need a third set of integers: $r_1, \ldots, r_\nu$. Then we define the *weighted (principal) symbol* of the boundary operator B as follows:

$$\big(\sigma_w B\big)_{i,j} = \sum_{|\mu| = r_i + t_j} \big(b_\mu(x)\big)_{i,j} \xi^\mu$$

where $b_\mu(x)$ are matrices of size $\nu \times m$ and the weights are chosen such that

$$r_i + t_j \geq q_{ij}^b$$

where q_{ij}^b is the maximal order of a derivative of the jth unknown function in the ith boundary condition.

The relevant condition which guarantees that the problem is well-posed is called *Shapiro-Lopatinskij condition*. To state it let us again use coordinates $x = (x', x_n)$ and the corresponding dual variables $\xi = (\xi', \zeta)$ such that the boundary is given by the equation $x_n = 0$. Now fixing some boundary point $(x', 0)$ let us consider the following ordinary differential operators with constant coefficients:

$$\begin{cases} \sigma_w A(x', 0, i\,\xi', \partial^{\mathbf{1}_n}) u(x_n) = 0 & ,\ x_n > 0 \\ \sigma_w B(x', 0, i\,\xi', \partial^{\mathbf{1}_n}) u(x_n) = 0 & ,\ x_n = 0. \end{cases} \tag{2.3}$$

Definition 2.4. The boundary operator satisfies the Shapiro–Lopatinskij condition, if the initial value problem (2.3) has only the trivial solution for all $\xi' \neq 0$ in the space of functions which tend to zero as x_n tends to infinity.

So let us finally consider the boundary problem

$$\begin{cases} Ay = f & x \in \Omega \\ By = g & x \in \Gamma. \end{cases} \tag{2.4}$$

Definition 2.5. The square boundary problem (2.4) is *DN-elliptic*, if the operator A is properly DN-elliptic and the operator B satisfies the Shapiro-Lopatinskij condition.

After these preliminaries we can now state

Theorem 2.2. *If the boundary problem* (2.4) *is DN-elliptic, then the following a priori estimate holds*

$$\sum_j ||y^j||_{\alpha + t_j} \leq C \left(\sum_i ||f^i||_{\alpha - s_i} + \sum_i ||g^i||_{\alpha - r_i - 1/2} + \sum_j ||y^j||_0 \right).$$

If the solution is unique, the last sum on the right hand side can be omitted.

So we see that DN-elliptic systems have quite similar *a priori* estimates as standard elliptic systems, and that the weights have a natural interpretation in terms of Sobolev norms.

2.3. **Involutive systems**

Unfortunately the rigorous definition of involutivity is quite complicated. The precise formulation can be found for example in [6, 18, 21, 22]. Here we will only explain how the involutive form can be recognized and computed in a constructive way.

Let us start with an example. Consider the system $\nabla \times y + y = 0$. Taking the divergence we see that if y is a solution, then it must also satisfy $\nabla \cdot y = 0$. This new equation is called a *differential consequence or integrability condition* of the initial system. Hence we have two systems:

$$\mathcal{S} \quad : \quad \nabla \times y + y = 0 \qquad\qquad \mathcal{S}' \quad : \quad \begin{cases} \nabla \times y + y = 0 \\ \nabla \cdot y = 0. \end{cases} \tag{2.5}$$

We say that $\mathcal{S}'$ is the involutive form of $\mathcal{S}$ because no more new first order differential consequences can be found. So informally we may define involutivity as follows:

A system is involutive, if it contains all its differential consequences (up to given order).

Note that the systems $\mathcal{S}$ and $\mathcal{S}'$ have the same set of smooth and distribution solutions. However, $\mathcal{S}$ is not elliptic (not even DN-elliptic) while $\mathcal{S}'$ is elliptic.

Then let us try to formulate involutivity in a more general fashion. We saw above that one obstruction to involution is that by differentiating the equations and then eliminating the highest order derivatives we got a new equation which was algebraically independent of the original equations. However, there is another obstruction which is more subtle. To understand it we need to introduce another symbol of the system, sometimes called the *geometric* or *Spencer symbol*.

Definition 2.6. Consider the system in (2.2) and let M_q be the following matrix

$$M_q = \left(a_{\mu^1}, a_{\mu^2}, \ldots, a_{\mu^{n_q}} \right)$$

where $\mu^1 > \mu^2 > \cdots > \mu^{n_q}$ and $|\mu^i| = q$. The (geometric) symbol $\mathcal{M}_q$ is a family of vector spaces defined by the kernel of M_q.

One may also call the matrix M_q the symbol of the system (2.2). The above construction attaches to each point $p \in \Omega$ a certain vector space. We will suppose that the dimension of this vector space does not depend on p; hence the symbol is in fact a vector bundle. In [18,21,22] one can find a detailed description of this bundle.

Note that the principal symbol uses the same "information" as geometric symbol, namely the matrices a_μ. Hence there must be a close connection between these two symbols. To describe the connection let us set

$$\Xi^q = \left(\xi^{\mu^1}, \xi^{\mu^2}, \ldots, \xi^{\mu^{n_q}} \right)$$

where $\mu^1 > \mu^2 > \cdots > \mu^{n_q}$ and $|\mu^i| = q$. Then

$$\sigma A = M_q \big(\Xi^q \otimes I \big)$$

where I is the identity matrix of size $m \times m$ and $\otimes$ is the Kronecker or tensor product. In [22] one can find a more geometric formulation of this connection.

Anyway, it turns out that the symbol contains information which is needed to recognize if the system is involutive or not. The problem is that we do not know *a priori* how many times one should differentiate the given system in order to find all the integrability conditions. The relevant property is called the *involutivity* of the symbol. However, we refer to [6, 14, 18, 21, 22] for the actual definition and give only a criterion which can be used to check involutivity. Another kind of constructive test can be found in [14]. To formulate the criterion we will need the indices of the symbol.

Definition 2.7. Let us suppose that the symbol matrix M_q is in the row echelon form. A derivative y^i_μ is a *leader*, if there is a row whose first nonzero element is in the column which correspondes to y^i_μ. Let $\beta^{(k)}_q$ be the number of leaders of class k. These are the *indices* of M_q.

Strictly speaking in the above definition one must suppose that the coordinate system is δ-regular. Without going into details let us simply observe that coordinate systems are generically δ-regular, so this is not a big problem in practice; see [21] for the precise definition and discussion of this condition. Let us denote by $\mathcal{M}_{q+1}$ the prolonged symbol, *i.e.* the symbol of the system obtained by differentiating the initial system once with respect to all variables. Now we can state

Theorem 2.3. *The symbol $\mathcal{M}_q$ is involutive if and only if*

$$\operatorname{rank}(M_{q+1}) = \sum_{k=1}^{n} k\beta^{(k)}_q. \tag{2.6}$$

It is seen that the criterion is quite reasonable: *i.e.* it can effectively be tested using standard operations of algebra. The next result shows why the involutivity of the symbol is important.

Theorem 2.4. *Let us suppose that the symbol $\mathcal{M}_q$ is involutive. If no new integrability conditions are obtained by differentiating the system once, then there are no integrability conditions at all.*

Now we say that the system (2.2) is *involutive*, if its symbol is involutive and there are no integrability conditions.

The above discussion suggests the following algorithm to compute the involutive form of the system:

(1) The system is differentiated until its symbol becomes involutive.
(2) The system is differentiated once more to check if there are integrability conditions.
(3) If there are no new equations in the previous step, the system is now involutive. Otherwise go back to step one.

This is often called the *Cartan-Kuranishi completion algorithm*. One can show that under appropriate hypothesis this algorithm terminates. The analysis and description of these hypothesis is, however, beyond the scope of the present article and we refer to [6, 18, 21, 22] for more information.

Note finally that there exist other approaches to completion than the one outlined above. However, the differences in these approaches are irrelevant from the point of view of the present article. Hence we will often use the term *completed system* to mean the system obtained by adding some or all integrability conditions to the initial system using any convenient method.

3. Model problems

Let us consider the following system

$$A_I y = \begin{cases} -u_{20} - v = 0 \\ -u_{02} + v = 0 \end{cases} \qquad \sigma_w A_I = \begin{pmatrix} -\xi_1^2 & -1 \\ -\xi_2^2 & 1 \end{pmatrix} \tag{3.1}$$

where $y = (u, v)$. This is DN–elliptic with $s_1 = s_2 = t_2 = 0$ and $t_1 = 2$. By cross differentiating the equations we can eliminate u and obtain $\Delta v = 0$. Hence the completed system is

$$A_0 y = \begin{cases} -u_{20} - v = 0 \\ -u_{02} + v = 0 \\ -\Delta v = 0 \end{cases} \qquad \sigma A_0 = \begin{pmatrix} -\xi_1^2 & 0 \\ -\xi_2^2 & 0 \\ 0 & -|\xi|^2 \end{pmatrix}. \tag{3.2}$$

Clearly this system is elliptic in the ordinary sense.

Let us then look for the compatibility operator for A_0, *i.e.* an operator A_1 such that $A_1 A_0 = 0$ and that A_1 is the "biggest" operator with this property[1]. In the present case it is rather immediate that

$$A_1 = \big(-\partial^{02}, \partial^{20}, 1\big).$$

Let us introduce a new variable z, and denote $\tilde{y} = (y, z) = (u, v, z)$ and define

$$A\tilde{y} = \big(A_0 \ , \ A_1^T\big)\tilde{y} = A_0 y + A_1^T z$$

where A_1^T is the formal transpose or adjoint of A_1. The system we are going to solve numerically can then be written as

$$A\tilde{y} = \begin{cases} -u_{20} - v - z_{02} = 0 \\ -u_{02} + v + z_{20} = 0 \\ -\Delta v + z = 0. \end{cases} \tag{3.3}$$

We call this system the *augmented system*. Note that the augmented system is also elliptic in the standard sense. In the next section we will discuss some advantages of this formulation. Now operating by A_1 to (3.3) we get

$$A_1 A\tilde{y} = z_{40} + z_{04} + z = 0.$$

Hence by itself z is a solution of an elliptic equation.

Now according to Theorem 2.1 we need one boundary condition for system (3.1). For example a standard Dirichlet or Neumann condition on u satisfies the Shapiro-Lopatinskij condition. Then if the boundary is given by $x_2 = 0$ and we choose Dirichlet condition the relevant initial value problem (2.3) becomes

$$\begin{cases} \xi_1^2 u - v = 0 \\ -u'' + v = 0 \\ u(0) = 0. \end{cases}$$

A simple computation gives the solution

$$\begin{aligned} u(x_2) &= c \sinh(\xi_1 x_2) \\ v(x_2) &= c\, \xi_1^2 \sinh(\xi_1 x_2). \end{aligned}$$

Obviously only the zero solution goes to zero at infinity. For (3.3) we need 3 boundary conditions, hence in addition to the condition on u we need two more conditions. Of course z should be identically zero for the exact solution, so it is reasonable to set z to zero on the boundary. Note that setting z and its normal derivative to zero on the boundary would of course force z to be identically zero, but this choice violates the Shapiro-Lopatinskij condition. Hence we need a condition which involves also v. Choosing some Dirichlet condition also for v we obtain

$$\begin{cases} \xi_1^2 u - z'' = 0 \\ -u'' - \xi_1^2 z = 0 \\ \xi_1^2 v - v'' = 0 \\ u(0) = v(0) = z(0) = 0 \end{cases}$$

[1] The rigorous definition of the compatibility operator is beyond scope of the present article and we refer to [6] for a precise formulation.

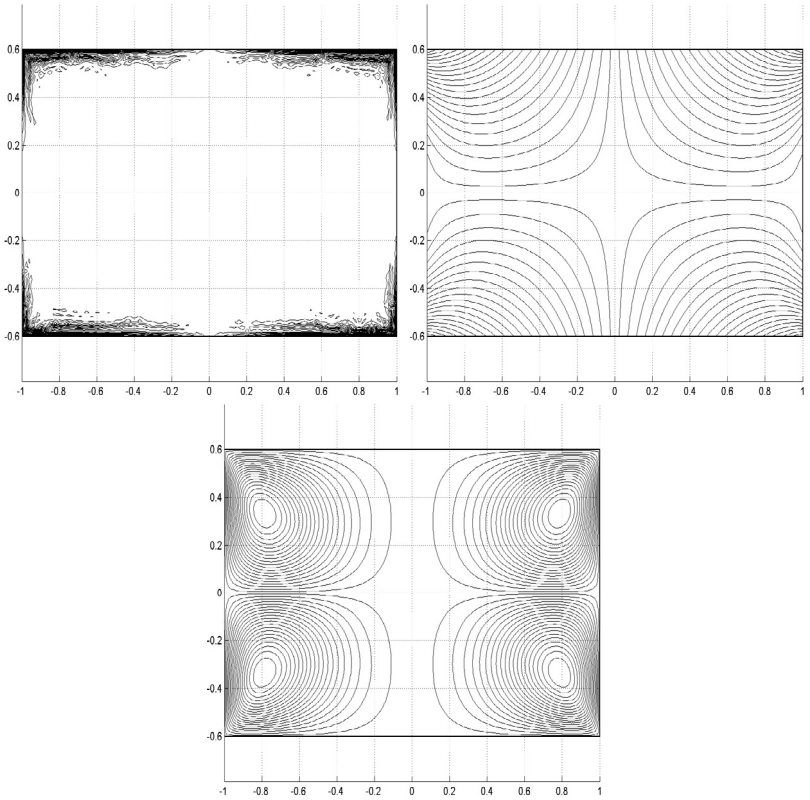

FIGURE 3.1. v computed using the system (3.1) (top left) and the system (3.3) (top right). Bottom: level contours of z; $\max|z| \sim 5 \times 10^{-3}$.

whose solution is

$$\begin{aligned} u(x_2) &= c_1\big(\mathrm{e}^{a\xi_1 x_2} - \mathrm{e}^{-a\xi_1 x_2}\big) + c_2\big(\mathrm{e}^{\bar{a}\xi_1 x_2} - \mathrm{e}^{-\bar{a}\xi_1 x_2}\big) \\ v(x_2) &= c_3 \sinh(\xi_1 x_2) \\ z(x_2) &= -i\, c_1\big(\mathrm{e}^{a\xi_1 x_2} - \mathrm{e}^{-a\xi_1 x_2}\big) + i\, c_2\big(\mathrm{e}^{\bar{a}\xi_1 x_2} - \mathrm{e}^{-\bar{a}\xi_1 x_2}\big) \end{aligned}$$

where $a = (1+i)/\sqrt{2}$. Hence the Shapiro-Lopatinskij condition is satisfied. Also one immediately sees that if there is no condition for v then the Shapiro-Lopatinskij condition is *not* satisfied.

Now let us consider a test case where the boundary conditions for v are readily available. We choose a rectangular domain and impose the following conditions:

$$\begin{cases} u = \sin(x_1 x_2), & \text{on the whole boundary} \\ v = x_2^2 \sin(x_1 x_2), & \text{on the horizontal and} \\ v = -x_1^2 \sin(x_1 x_2), & \text{on the vertical part of the boundary} \\ z = 0, & \text{on the whole boundary.} \end{cases} \tag{3.4}$$

Note that the condition on v comes from extending the system from the domain to its boundary.

Results show how solving the initial model (3.1) leads to unrealistic results for v while with the same numerical implementation the augmented model (3.3) leads to the correct solution, see Figure 3.1. We show

only the computed v's because the difference of computed u's was negligible. This is analogous to the well-known property of Stokes system: the velocity is easier to compute than the pressure. We will discuss this point in more detail below. These solutions have been computed with classical linear finite elements on triangular meshes with a direct LU solver for the solution of the linear system with no preconditioning.

3.1. **Another problem hiding advection**

Consider the following problem also containing first order derivatives:

$$A_I y = \begin{cases} -y^1_{20} + y^2 = 0 \\ -y^1_{02} + y^3 = 0 \\ -y^2_{20} + a_1 y^2_{01} - y^3_{02} + a_2 y^3_{10} = 0 \end{cases} \qquad \sigma_w A_I = \begin{pmatrix} -\xi_1^2 & 1 & 0 \\ -\xi_2^2 & 0 & 1 \\ 0 & -\xi_1^2 & -\xi_2^2 \end{pmatrix}. \tag{3.5}$$

The system is DN-elliptic. We consider again a rectangular domain. The correct boundary conditions (*i.e.* which satisfy the Shapiro–Lopatinskij condition) can be specified as follows:

> y^1 is given on the whole boundary while y^2 is given on the vertical and y^3 on the horizontal parts of the boundary.

Adding the integrability condition gives the following elliptic system:

$$A_0 y = \begin{cases} -y^1_{20} + y^2 = 0 \\ -y^1_{02} + y^3 = 0 \\ -y^2_{20} + a_1 y^2_{01} - y^3_{02} + a_2 y^3_{10} = 0 \\ -y^2_{02} + y^3_{20} = 0 \end{cases} \qquad \sigma A_0 = \begin{pmatrix} -\xi_1^2 & 0 & 0 \\ -\xi_2^2 & 0 & 0 \\ 0 & -\xi_1^2 & -\xi_2^2 \\ 0 & -\xi_2^2 & \xi_1^2 \end{pmatrix}. \tag{3.6}$$

Then computing the augmented system gives

$$A\tilde{y} = \begin{cases} -y^1_{20} + y^2 - z_{02} = 0 \\ -y^1_{02} + y^3 + z_{20} = 0 \\ -y^2_{20} + a_1 y^2_{01} - y^3_{02} + a_2 y^3_{10} = 0 \\ -y^2_{02} + y^3_{20} - z = 0. \end{cases} \tag{3.7}$$

For this system we need the values of y^2 and y^3 on the whole boundary. For our rectangular domain the "missing" boundary conditions can be deduced from the operator itself. We used the values $a_1 = a_2 = 1$ in the computations. Numerical results are similar to the previous case: for the y^1 the results obtained using the systems (3.5) and (3.7) were quite close to each other, but for example the computed y^2 was very different in the two cases, see Figure 3.2.

4. Properties of involutive and augmented systems

4.1. **Motivation**

We introduced above the augmented system using the compatibility operator. But why should this method lead to a reasonable system of equations? Let us consider our problem in a general form

$$A_0 y = f \tag{4.1}$$

and let us suppose that A_0 is elliptic and involutive. Further let A_1 be the compatibility operator for A_0. Hence we have the complex

$$0 \longrightarrow V_0 \xrightarrow{A_0} V_1 \xrightarrow{A_1} V_2 \longrightarrow 0$$

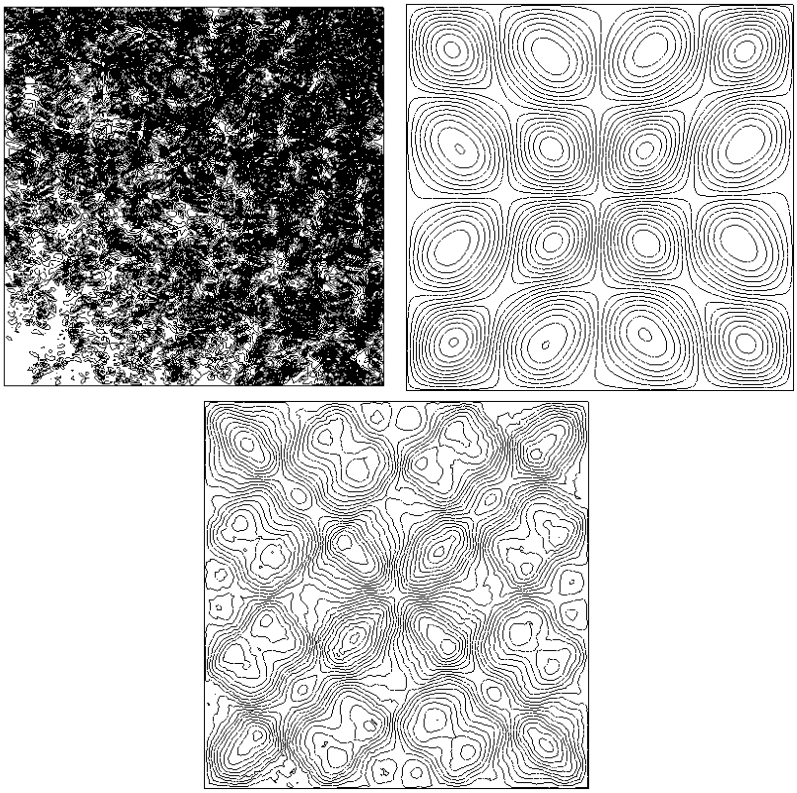

FIGURE 3.2. On the left y^2 computed with the system (3.5) and on the right y^2 computed with the system (3.7). Lower: iso-z for (3.7) with a maximum of about 10^{-3} in absolute value.

where V_i are some convenient vector spaces. Now let us suppose that the above complex is exact. This suggests that we can decompose V_1 as follows:

$$\text{image}\big(A_0\big) \oplus \text{image}\big(A_1^T\big) \simeq V_1$$

where A_1^T is the formal transpose of A_1. Of course to be able to write equality instead of $\simeq$ we should specify carefully the relevant vector spaces. However, proceeding formally, this decomposition suggests that it is indeed possible to find some functional framework such that the combined operator (A_0, A_1^T) would be bijective. Hence reasonable discretizations of these operators should yield a well-posed numerical problem.

So instead of trying to solve the original system (4.1) in some least square sense, we introduce an auxiliary variable z and solve

$$A_0 y + A_1^T z = f. \tag{4.2}$$

The idea of introducing artificial variables to the system is not entirely new. For example in [12] stationary Maxwell's equation and Stokes equation in vorticity formulation is analysed in this way. However, the operator introduced is not the formal transpose of the compatibility operator. It is merely an operator such that the resulting square system is elliptic. Moreover this artificial variable is not used in numerical computations.

Anyway our formulation (4.2) has some good properties precisely because A_1 is the compatibility operator of A_0 and not just random operator which would yield a square system of equations. Indeed, suppose that $\tilde{y} = (y, z)$ is a solution of the system (4.2). Then it immediately follows that y is (formally) a least squares solution of $A_0 y = f$ and z is a least squares solution of $A_1^T z = f$.

So in some sense the components y and z do not interact. Of course this would not be exactly true at discrete level without constructing some special schemes. However, if the discretization is consistent then this is approximately true, and probably no more is really needed. Since consistent schemes are usually used anyway to keep the approximation error small the above remarks seem to imply that the resulting numerical augmented system would in general have the expected good properties.

4.2. **Conditioning**

We have seen that involutive systems are not usually square. One possibility to solve such systems directly is to compute a least squares solution to it. However, we expect that the conditioning of the augmented system is in general better in the following sense. Let us suppose that the operators A_0 and A_1^T (with appropriate boundary conditions) have been discretised, and let us denote the corresponding discrete operators by A_0^h and $(A_1^h)^T$. Further let $A^h = \big(A_0^h, (A_1^h)^T\big)$.

Now recall that a convenient way to measure the conditioning of the system is by

$$\kappa(A_i^h) = \sigma^i_{\max}/\sigma^i_{\min}$$

where $\sigma^i_{\max}$ (resp. $\sigma^i_{\min}$) is the biggest (resp. smallest) singular value of A_i^h.

But solving directly (4.1) leads to an overdetermined system $A_0^h y = f$. If the matrix A_0^h is of moderate size one could use QR or singular value decomposition to solve the system in the least squares sense in a stable way. But if one uses iterative methods, it is known that conditioning of the system is $\sim \kappa(A_0^h)^2$ [2]. This is of same order of magnitude as if one were solving the least squares problem using the normal equations.

But let us now suppose that the matrix A_1^h is the compatibility operator for A_0^h; *i.e.* $\operatorname{image}\big(A_0^h\big) = \ker\big(A_1^h\big)$ and A_1^h is surjective. Then if $\tilde{y} = (y, z)$ is a solution to $A^h\tilde{y} = f$, then y (resp. z) is automatically a *least squares* solution to $A_0^h y = f$ (resp. $(A_1^h)^T z = f$). Moreover we have

Lemma 4.1.

$$\kappa(A^h) = \frac{\max\{\sigma^0_{\max}, \sigma^1_{\max}\}}{\min\{\sigma^0_{\min}, \sigma^1_{\min}\}}.$$

Proof. Let $A_0^h = U_0\Sigma_0 V_0^T$ and $(A_1^h)^T = U_1\Sigma_1 V_1^T$ be the reduced singular value decompositions of A_0^h and $(A_1^h)^T$. Then the SVD of A^h is (up to ordering of the relevant columns)

$$A^h = U\Sigma V^T = \big(U_0, U_1\big)\begin{pmatrix}\Sigma_0 & 0\\ 0 & \Sigma_1\end{pmatrix}\begin{pmatrix}V_0 & 0\\ 0 & V_1\end{pmatrix}^T. \qquad \square$$

As a consequence, if the conditioning of A_0^h and $(A_1^h)^T$ is reasonable, then the same holds also for the matrix A^h. But A^h is a square matrix and hence we may use any convenient iterative method to solve this system.

Of course in practice A_1^h is almost surely not exactly the compatibility operator. However, we expect that $A_1^h A_0^h \approx 0$ when using consistent discretisation, and hence we expect that the whole operator A^h still has the nice properties as in Lemma 4.1.

The idea of completing a rectangular matrix to a square matrix is called *bordering* in the literature. There are numerous articles discussing this topic, but as far as we know there seem to be no unified approach to this problem, perhaps because the reasons for using bordering and contexts where bordering has been found useful are so different. In our context the bordering matrix A_1^h comes naturally from the problem itself, so there is no need for any ad hoc constructions.

4.3. **Spurious solutions**

One of the main motivations of using involutive form in ODE/DAE context in [23–25] was to avoid the *drift-off*. Geometrically this means that while the exact solution stays in some submanifold of some euclidean space, the numerical solution obtained by using noninvolutive form tends to leave (quite rapidly) this manifold.

When the involutive form is used this difficulty does not occur. In PDE context a similar phenomenon goes by the name *spurious solutions*. This term is mostly used in connection with Maxwell's equations, see [12] for a discussion. Note that Maxwell's equations are naturally overdetermined (8 equations and 6 unknowns), so some tricks must be used if the solution of the full system is desired. Of course there are also numerous physically relevant situations in electromagnetics where the full system is not needed.

Anyway a simplified example of the problem is provided by the systems $\mathcal{S}$ and $\mathcal{S}'$ in (2.5). If one discretizes $\mathcal{S}$ the principal part of the operator has an infinite dimensional kernel. Hence in the numerical solution there may appear components which are approximately in this kernel; these are called spurious solutions. Working with the involutive form $\mathcal{S}'$ one does not encounter spurious solutions, because the infinite dimensional kernel disappears.

As far as we know spurious solutions have not been defined or discussed in general terms. However, apparently the term is always used in situations where the numerical solution of noninvolutive system is attempted.

4.4. **Mesh refinement**

The auxiliary variable z should of course be zero if we have the exact solution to our original problem. So monitoring the values of z in the numerical solution gives us information about the errors in the solution. Hence we can in fact use this artificial variable in a useful way in different mesh adaptation approaches already available. For instance in a mesh refinement technique we can subdivide the elements where z is large (see Fig. 5.2) while with Delaunay mesh regeneration by metric control [3, 4, 9, 11], the metric can be modified to reduce the size of the elements where z is large. In the latter approach consider a metric defined through a positive definite modification of the Hessian of a variable available through its piecewise linear discretisation. The interpolation error is bounded by:

$$\mathbf{E} = |u - \Pi_h u|_0 \leq ch^2 |D^2 u|_0, \tag{4.3}$$

where h is the element size, $\Pi_h u$ the P^1 interpolation of u and D^2u its Hessian matrix which is symmetric.

$$D^2 u = \mathbf{R} \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix} \mathbf{R}^{-1},$$

where R is the eigenvectors matrix of D^2u and λ_i its eigenvalues (always real). Using this information, we introduce the following metric tensor $\mathbf{M}$:

$$\mathbf{M} = \mathbf{R} \begin{pmatrix} \tilde{\lambda}_1 & 0 \\ 0 & \tilde{\lambda}_2 \end{pmatrix} \mathbf{R}^{-1}, \tag{4.4}$$

where

$$\tilde{\lambda_i} = \min \left(\max \left(|\lambda_i|, \frac{1}{h_{\max}^2} \right), \frac{1}{h_{\min}^2} \right),$$

with $h_{\min}$ and $h_{\max}$ being the minimal and maximal edge lengths allowed in the mesh.

Now, if we generate a Delaunay equilateral mesh with edges of length of 1 in the metric $\mathbf{M}/(c\mathbf{E})$, the interpolation error $\mathbf{E}$ is equi-distributed over the edges of length a_i if

$$\frac{1}{c\mathbf{E}} a_i^T M a_i = 1. \tag{4.5}$$

This leads to the definition of a metric for each variable of a system and in practice we consider a metric defined by the intersection of all these metrics: for an Euclidean metric the unit ball is a sphere while in these metrics we have ellipsoids to intersect. In fact, an approximate intersection is suffisent. For instance, for two metrics let λ_i^j and v_i^j, $i, j = 1, 2$ the eigenvalues and eigenvectors of $\mathbf{M}_j$, $j = 1, 2$. The intersection metric ($\hat{\mathbf{M}}$) is defined by

$$\hat{\mathbf{M}} = \frac{\hat{\mathbf{M}}_1 + \hat{\mathbf{M}}_2}{2} \tag{4.6}$$

where $\hat{\mathbf{M}}_1$ (resp. $\hat{\mathbf{M}}_2$) has the same eigenvectors as $\mathbf{M}_1$, (resp. $\mathbf{M}_2$) but with eigenvalues defined by:

$$\tilde{\lambda_i^1} = \max\left(\lambda_i^1, {v_i^1}^T \mathbf{M}_2 v_i^1\right), \quad i = 1, 2. \tag{4.7}$$

This algorithm is easy to extend to the case of several variables. One difficulty comes from the fact that in systems we work with variables with different physical meaning and scale. The following relative error estimation avoids this problem and also permits the introduction of z:

$$\tilde{\mathbf{E}} = \left|\frac{u - \Pi_h u}{\max(|\Pi_h u|, \epsilon)}\right|_0 \leq ch^2 \left|\frac{D^2 u}{\max(|\Pi_h u|, \epsilon)}\right|_0. \tag{4.8}$$

We have introduced the local value of the variable in the norm. ϵ is a cut-off to avoid numerical difficulties and also to define the difference between the orders of magnitude of the smallest and largest scales we try to capture. Indeed, when a phenomena falls below ϵ, it will not be captured. This is similar to looking for a more precise estimation in regions where the variable is small. Another important consequence of this estimation is that it removes the dimensionality problems when intersecting metrics coming from different quantities. In addition to the metric intersection procedure, an admissible level of z can be introduced here. We can also reduce ϵ for all variables where z is large.

5. Stokes system of fluid mechanics

Numerical solution of Stokes (and Navier-Stokes) problem has received a great deal of interest for many years now and we refer to [10,17,20] for details and further references to (very large) literature on the subject. What will be said here also applies not only to classical fluid mechanics but also to the solution of flows in porous regions, several biological fluids and microfluids which are not yet fully understood [19]. These situations involve large coupled systems including the calculation of flow polarization and the behavior of various species in the flow. The Stokes system is therefore only a small part of the whole ensemble. Any simplification leading to a global understanding and unification of the numerical methods needed to solve these systems is of great importance.

We consider the following familiar Stokes problem in 2 dimensions:

$$\begin{cases} -\Delta u + \nabla p = f \\ \nabla \cdot u = g \end{cases} \tag{5.1}$$

where u is the velocity field and p is the pressure. This is a DN-elliptic system, and by Theorem 2.1 we need 2 boundary conditions; normally this means Dirichlet conditions for u and no conditions for p.

By taking the divergence of the first equation we obtain

$$-\Delta p = -\nabla \cdot f - \Delta g. \tag{5.2}$$

Putting $y = (u, p)$ we can write the whole system as

$$A_0 y = \begin{cases} -\Delta u + \nabla p = f \\ -\Delta p = -\nabla \cdot f - \Delta g \\ -\nabla \cdot u = -g. \end{cases}$$

This is clearly elliptic. The compatibility operator is in the present case given by

$$A_1 = \left(\nabla\cdot, 1, -\Delta\right).$$

Let us again introduce a new variable z, and denote $\tilde{y} = (y, z) = (u, p, z)$ and define

$$A\tilde{y} = \begin{pmatrix} A_0 \ , \ A_1^T \end{pmatrix} \tilde{y} = A_0 y + A_1^T z.$$

The augmented system can then be written as

$$A\tilde{y} = \begin{cases} -\Delta u + \nabla p - \nabla z = f \\ -\Delta p + z = -\nabla \cdot f - \Delta g \\ -\nabla \cdot u - \Delta z = -g. \end{cases} \tag{5.3}$$

This is elliptic in the standard sense and according to Theorem 2.1 we need 4 boundary conditions. For u we use of course the original boundary conditions, and for z a natural choice is $z = 0$ on the boundary. However, we still need something for p. This is precisely the same situation as in the problem (3.1): the initial problem is well-posed without the boundary condition for p, but for the augmented system we need the boundary condition. So it appears that our formulation is very different from the standard Stokes problem.

The situation is a bit different, however, when the solution of actual discrete equations is computed. In fact many methods for the solution of the discretised problem use the equation (5.2) in some way and/or make some hypothesis about the normal derivative of p on the boundary; see the discussion in [20], pp. 325–337. For example one way to solve the Stokes system is to consider the "stabilized" problem:

$$\begin{cases} -\Delta u + \nabla p = f \\ -\varepsilon \Delta p + \nabla \cdot u = g \end{cases} \tag{5.4}$$

where in practice the parameter ε is chosen such that $\varepsilon \sim h^2$. But of course this means that implicitly some (Neumann) boundary condition is assumed for p. Moreover usually there is no clear method for tuning such a coefficient, especially in highly non-isotropic configurations.

In this approach we have made the choice of reducing the impact of this boundary condition by solving the following modified system:

$$A\tilde{y} = \begin{cases} -\Delta u + \nabla p - \nabla z = f \\ -\varepsilon \Delta p = (1 - \varepsilon)\Delta p - z - \nabla \cdot f - \Delta g, \quad \epsilon \sim 10^{-10} \\ -\nabla \cdot u - \Delta z = -g \end{cases} \tag{5.5}$$

with the following boundary conditions

$$\begin{cases} u \quad \text{as in the original system (5.1)} \\ \dfrac{\partial p}{\partial n} = 0 \\ z = 0. \end{cases} \tag{5.6}$$

In the decomposition above the boundary condition on p only affects regions close to the boundary and the right-hand-side for p equation is only assembled inside the domain where Δ operator can be defined. In case of Navier-Stokes equations when the Reynolds number is large the normal derivative of p is small and (5.3) can be solved directly with (5.6).

We solved the system (5.3) with equal order linear finite elements for all variables on triangular meshes with a direct LU solver for the solution of the linear system with no preconditioning. Now one of the issues in the numerical solution of the Stokes problem is that the relevant finite element spaces should satisfy the inf-sup or LBB condition [17,20]. In particular it is not possible to use equal order discretisation for u and p when solving the system (5.1). However, for our system (5.5) this problem does not arise. Looking at the operator A in

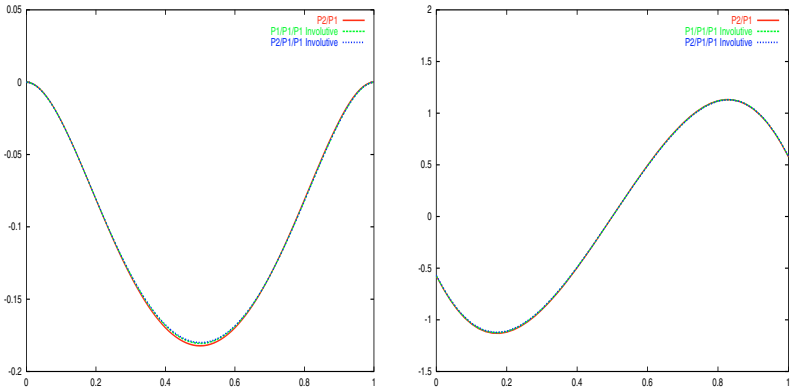

FIGURE 5.1. Stokes flow in a square cavity: first velocity component (left) and pressure (right) along $y = 0.5$. The augmented Stokes system (5.3) has been solved using $p2/p1/p1$ and $p1/p1/p1$ discretizations. The results compare to the $p2/p1$ solution of the Stokes system. No stability term has been added for the equal order discretization in the involutive case.

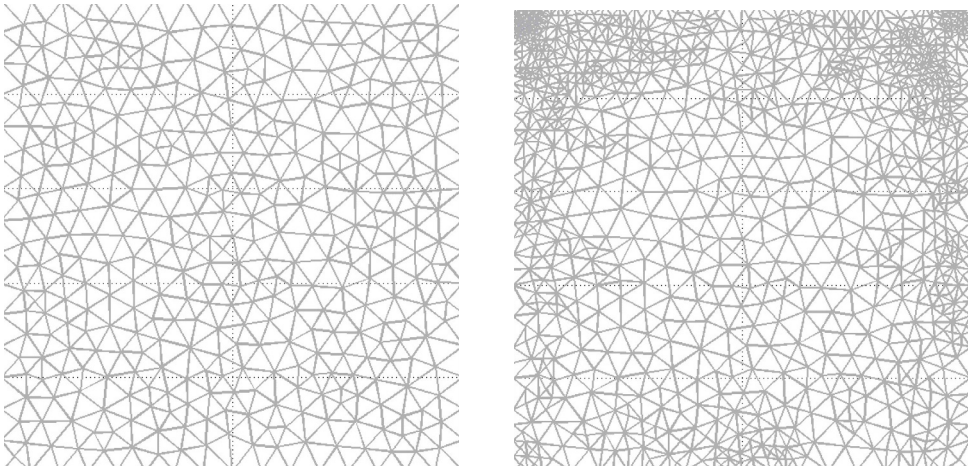

FIGURE 5.2. On the left the original mesh for the cavity. On the right the mesh has been refined where z was big.

(5.3) it is seen that it has Laplacian on the diagonal and some lower order terms outside the diagonal. Hence basically it is as easy or difficult to discretise than the standard scalar Laplacian.

To illustrate the behavior of the new system, we consider the well-known flow in a $(0,1) \times (0,1)$ square cavity. The boundary condition for the velocity $u = (u_1, u_2)$ is given by $u_1 = 4x(1-x)$ on the upper boundary and u_1 and u_2 are set to zero on all other boundaries. The numerical results in Figure 5.1 show that using our formulation (5.3) we get essentially same results with equal order P1 discretisation as with classical P2/P1 computation. The maximum value of z is about 10^{-3} and it decreases with the mesh size. Figure 5.2 illustrates the use of z in mesh adaptation described above. On this mesh z is reduced by almost one order of magnitude.

6. Compressible flow model

6.1. **Notation**

Let Ω be some domain in $\mathbb{R}^n$ as usual. In case $n = 2$ we set

$$\mathbf{rot}(v) = v^2_{10} - v^1_{01} \qquad \text{and} \qquad \mathbf{Rot}(u) = \big(u_{01}, -u_{10}\big)$$

where $u : \Omega \to \mathbb{R}$ and $v : \Omega \to \mathbb{R}^2$. In this section the jacobian (or first differential) of $y : \Omega \to \mathbb{R}^m$ is denoted for traditional reasons by ∇y. The second differential is denoted by d^2y as usual. Let A and B be matrices of the same size. Then their double contraction is

$$A \diamond B = \sum_{i,j} a_{ij} b_{ij}.$$

If $n = m$, $y\nabla y$ is a vector whose i'the component is $\langle y, \nabla y^i \rangle$ (as a matrix vector product $y\nabla y$ should thus be written as $(\nabla y)y$). It will be convenient at times to interpret a vector field as a differential operator, as is customary in differential geometry. In this case we will use a capital letter for the vector field. So if Y is a vector field and f is some smooth function defined on Ω, then

$$Y(f) = \sum_{i=1}^{n} y^i \frac{\partial f}{\partial x_i}.$$

6.2. **The model**

We consider the problem of low-Mach number hydro-dynamical flow simulation. These equations are of importance as direct numerical simulation of low speed flows with large density variations based on full Navier-Stokes system is not suitable because of the limitations due to acoustic time scales in the flow which brings very penalizing CFL limitations for time stepping. We would like to avoid this when considering accurate unsteady calculations dedicated to capture dynamic flow features which are essentially unaffected by its acoustic characteristics. This is the case for instance in some engine calculations involving combustion. Here we are interested in particular in the solution of linearized systems necessary in the study of the stability of engineering devices. In addition, nonlinear systems are usually solved by fixed point iterations involving the solution of the corresponding linearized systems or sub-systems. This again makes important any simplification of the required numerical methods.

The equations are derived from the Navier-Stokes system where assumptions are made on the state equation linking the density ρ and temperature T [16]:

$$\begin{cases} \rho_t + \nabla \cdot (\rho u) = 0 \\ \rho u_t + \rho u \nabla u + \nabla p - \frac{1}{\mathsf{Re}} \nabla \cdot \tau = 0 \\ \rho C_p T_t + \rho C_p \langle u, \nabla T \rangle - \frac{1}{\mathsf{RePr}} \nabla \cdot (k \nabla T) - \frac{\gamma - 1}{\gamma} P^0_t = 0 \\ \nabla \cdot u - \frac{1}{P^0 C_p \mathsf{RePr}} \nabla \cdot (k \nabla T) - \frac{1}{P^0 C_p} \Big(\frac{\gamma - 1}{\gamma} - C_p \Big) P^0_t = 0. \end{cases} \tag{6.1}$$

Here the tensor τ is given by

$$\tau = \mu \bigg(\nabla u + (\nabla u)^T - \frac{2}{3} \nabla \cdot u \, I \bigg).$$

A small calculation gives

$$\nabla \cdot \tau = \mu \bigg(\Delta u + \frac{1}{3} \nabla (\nabla \cdot u) \bigg) = \mu \bigg(\frac{4}{3} \Delta u + \frac{1}{3} \nabla \times \nabla \times u) \bigg).$$

The equation of state is $P^0 = \rho T$, and it is supposed that P^0 (the ambient pressure) depends only on time. To simplify the discussion we will actually suppose that it is constant: $P^0 = \rho T = 1$. Let us set other constants to one except the Reynolds number Re and the Prandtl number Pr. This yields

$$\begin{cases} \rho_t + \nabla \cdot (\rho u) = 0 \\ \rho u_t + \rho u \nabla u - \frac{4}{3\mathsf{Re}} \Delta u - \frac{1}{3\mathsf{Re}} \nabla \times \nabla \times u + \nabla p = 0 \\ \rho T_t + \rho \langle u, \nabla T \rangle - \frac{1}{\mathsf{RePr}} \Delta T = 0 \\ \nabla \cdot u - \frac{1}{\mathsf{RePr}} \Delta T = 0 \\ \rho T - 1 = 0. \end{cases}$$

Let us now eliminate ρ using the last equation. We obtain

$$\begin{cases} u_t - \frac{4}{3\mathsf{Re}} T \Delta u - \frac{1}{3\mathsf{Re}} T \nabla \times \nabla \times u + u \nabla u + T \nabla p = 0 \\ \nabla \cdot u - \frac{1}{\mathsf{RePr}} \Delta T = 0 \\ T_t - T \nabla \cdot u + \langle u, \nabla T \rangle = 0. \end{cases} \tag{6.2}$$

Denoting $y = (u, T, p)$ we can write the whole system as $\mathcal{N}_0 y = 0$. Linearizing around the reference solution $\bar{y} = (\bar{u}, \bar{T}, \bar{p})$ yields

$$\begin{cases} u_t - \frac{4}{3\mathsf{Re}} \bar{T} \Delta u - \frac{4}{3\mathsf{Re}} T \Delta \bar{u} - \frac{1}{3\mathsf{Re}} \bar{T} \nabla \times \nabla \times u - \frac{1}{3\mathsf{Re}} T \nabla \times \nabla \times \bar{u} \\ + \bar{u} \nabla u + u \nabla \bar{u} + \bar{T} \nabla p + T \nabla \bar{p} = 0 \\ \nabla \cdot u - \frac{1}{\mathsf{RePr}} \Delta T = 0 \\ T_t - \bar{T} \nabla \cdot u - T \nabla \cdot \bar{u} + \langle \bar{u}, \nabla T \rangle + \langle u, \nabla \bar{T} \rangle = 0. \end{cases} \tag{6.3}$$

This system is denoted by $\mathcal{L}_0 y = 0$. Now these are both second order systems which are not involutive: by differentiating and eliminating we can find an *integrability condition* which is analogous to Poisson equation for the pressure in the classical Stokes system. In order to compute the new equations let us define the operator

$$\mathcal{N}_1 = \left(\nabla \cdot, \frac{4\mathsf{Pr}}{3} \left(\partial_t + U \right), -\frac{4}{3\mathsf{Re}} \Delta \right).$$

Here U is the vector field u interpreted as a differential operator. The corresponding linear operator is simply obtained by replacing U by $\bar{U}$. Then we compute

$$\begin{aligned} -\mathcal{N}_1 \mathcal{N}_0 y = & - T \Delta p - \langle \nabla T, \nabla p \rangle - \frac{7}{3\mathsf{Re}} \langle \nabla T, \nabla \times \nabla \times u \rangle \\ & + \frac{8}{3\mathsf{Re}} \nabla u \diamond d^2 T - (\nabla u)^t \diamond \nabla u \\ & + \left(\frac{4\mathsf{Pr}}{3} - 1 \right) \left(\nabla \cdot u_t + \langle u, \nabla \nabla \cdot u \rangle \right) - \frac{4\mathsf{Pr}}{3} (\nabla \cdot u)^2. \end{aligned}$$

In the linear case we obtain

$$\begin{aligned}-\mathcal{L}_1\mathcal{L}_0 y = &-\bar{T}\Delta p - T\Delta\bar{p} - \langle\nabla\bar{T},\nabla p\rangle - \langle\nabla T,\nabla\bar{p}\rangle\\ &-\frac{7}{3\mathsf{Re}}\langle\nabla\bar{T},\nabla\times\nabla\times u\rangle - \frac{7}{3\mathsf{Re}}\langle\nabla T,\nabla\times\nabla\times\bar{u}\rangle\\ &+\frac{8}{3\mathsf{Re}}\nabla\bar{u}\diamond d^2T + \frac{8}{3\mathsf{Re}}\nabla u\diamond d^2\bar{T}\\ &+\left(\frac{4\mathsf{Pr}}{3}-1\right)\left(\nabla\cdot u_t + \langle\bar{u},\nabla\nabla\cdot u\rangle + \langle u,\nabla\nabla\cdot\bar{u}\rangle\right)\\ &-\frac{8\mathsf{Pr}}{3}(\nabla\cdot\bar{u})(\nabla\cdot u) - 2(\nabla\bar{u})^t\diamond\nabla u.\end{aligned}$$

We wrote the above formulas in 3D case. The corresponding 2D case is obtained by replacing $\nabla\times\nabla\times u$ by $\mathbf{Rot}(\mathbf{rot}(u))$.

6.3. **1D case**

Models (6.2) and (6.3) are already nontrivial in 1D case, unlike the case of incompressible Navier-Stokes system. In fact the impact of our approach is already visible in the case of the linearised model, so we give numerical results only in this case. So we consider the following linear model with constant reference solution $(\bar{u},\bar{T},\bar{p})$:

$$\mathcal{L}_0 y = \begin{cases} u_t - \frac{4}{3\mathsf{Re}}\bar{T}u_{xx} + \bar{u}u_x + \bar{T}p_x = 0\\ -u_x + \frac{1}{\mathsf{RePr}}T_{xx} = 0\\ T_t - \bar{T}u_x + \bar{u}T_x = 0\\ -\bar{T}p_{xx} + (\frac{4\mathsf{Pr}}{3}-1)\big(u_{xt}+\bar{u}u_{xx}\big) = 0.\end{cases} \tag{6.4}$$

The operator $\mathcal{L}_1$ is in this case simply

$$\mathcal{L}_1 = \left(\partial_x, \frac{4\mathsf{Pr}}{3}\Big(\partial_t + \bar{u}\partial_x\Big), -\frac{4}{3\mathsf{Re}}\partial_{xx}, 1\right).$$

Since this is a time dependent problem the augmented system is not quite obtained from the formal transpose of $\mathcal{L}_1$: the sign of the time derivative is not changed. Otherwise proceeding as before gives the following augmented system.

$$\mathcal{L}\tilde{y} = \begin{cases} u_t - \frac{4}{3\mathsf{Re}}\bar{T}u_{xx} + \bar{u}u_x + \bar{T}p_x - z_x = 0\\ \frac{4\mathsf{Pr}}{3}z_t - u_x + \frac{1}{\mathsf{RePr}}T_{xx} - \frac{4\mathsf{Pr}}{3}\bar{u}z_x = 0\\ T_t - \bar{T}u_x + \bar{u}T_x - \frac{4}{3\mathsf{Re}}z_{xx} = 0\\ -\bar{T}p_{xx} + \quad (\frac{4\mathsf{Pr}}{3}-1)\big(u_{xt}+\bar{u}u_{xx}\big) + z = 0.\end{cases} \tag{6.5}$$

Then we compute:

$$\mathcal{L}_1\mathcal{L}\,\tilde{y} = \frac{16\mathsf{Pr}^2}{9}z_{tt} + \frac{16}{9\mathsf{Re}^2}z_{xxxx} - \left(1+\frac{16\bar{u}^2\mathsf{Pr}^2}{9}\right)z_{xx} + z = 0.$$

Hence z by itself is a solution of a well-posed (hyperbolic) problem.

6.4. **Results**

We considered periodic boundary conditions for all variables. Since z is supposed to be zero it is natural to take zero initial conditions for it. We took zero initial conditions for u and p, and Gaussian "perturbation" for T. In this case it is known that the solution tends to an equilibrium (not stationary) solution, and moreover the L^2–norm of the pressure depends linearly on Pr [16].

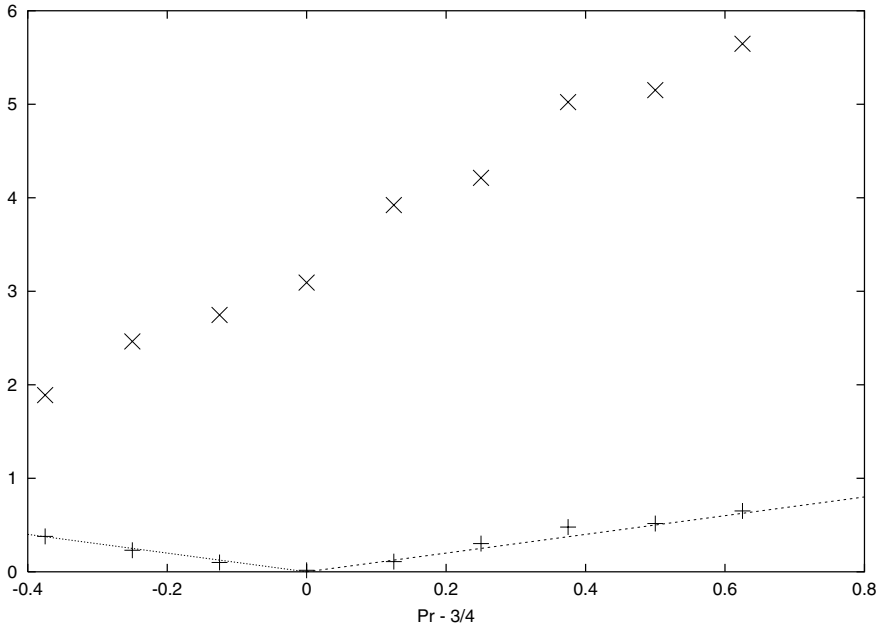

FIGURE 6.1. The L^2–norm of the pressure in the equilibrium solution as a function of Prandtl number. $\times$ denotes the results obtained using the initial system (6.2) and $+$ denotes the results obtained with augmented system (6.5). The line gives the value given by theory [16].

As seen in Figure 6.1 our formulation produces the desired result while a discretization of the original system with the same numerical schemes fails. Figure 6.4 gives the time evolution of L^2–norm of z for various values of Pr. It is seen that the norm is quite big in the beginning, but is very small when the equilibrium solution is found. Note that taking an equilibrium solution as initial value would keep z small even in the beginning.

7. Concluding remarks

We have presented a general framework on how to use the involutive form of the given PDE system in the numerical computations. The approach has been applied to several examples where the results show that the involutive form is more suitable for discretization than the original system and reduces the need for numerical recipes which are specific for one particular system.

Our approach leads also to some unexpected consequences: for example in the Stokes problem one does not need to take into account the familiar inf-sup condition. This suggests that these kind of extra conditions for the numerical schemes are at least partly due to noninvolutiveness of the system.

One difficulty with involutive systems is that usually they are not square which makes their numerical treatment difficult. We proposed above one possible way to deal with this issue: we introduced an auxiliary variable which restored the squareness of the system. Of course it may be possible to solve the resulting system efficiently by some other method. However, the use of the compatibility operator appears rather natural, and this operator might in any case be useful in the analysis and numerical treatment of the system. In addition, we have seen above that this auxiliary variable is in fact useful in adaptive error control and mesh adaptation.

Another difficulty is the problem of boundary conditions: the involutive form often needs boundary conditions that are apparently not naturally given by the original problem. In the test problems above we could easily determine the "extra" conditions; however, in more general situations it is not clear how this should be done. On the other hand in the Stokes problem it was seen that the problem of the boundary condition exists in a hidden way also in classical treatment of the problem.

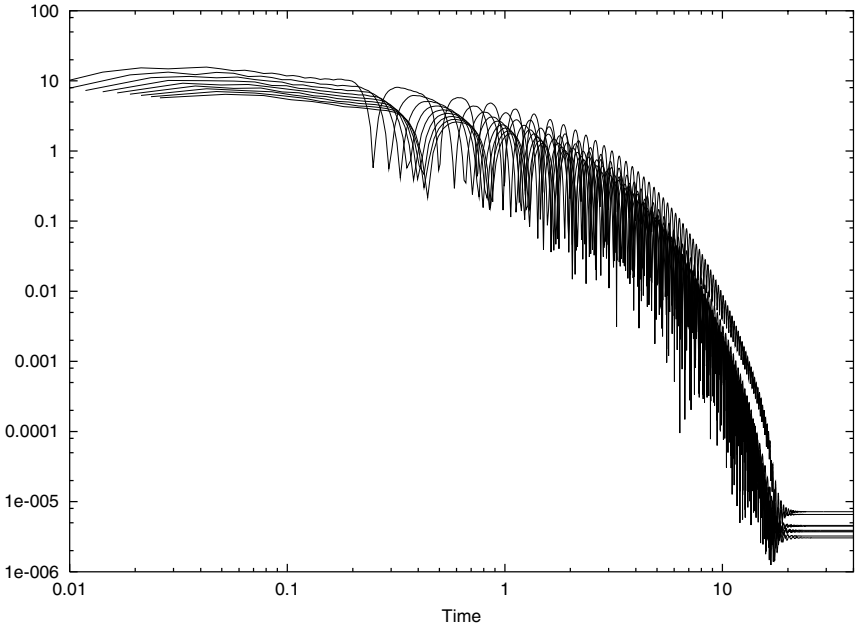

FIGURE 6.2. The L^2–norm of z as a function of time for various values of Pr.

In the present paper we have only given numerical results for the linear systems. However, our methodology can be applied also in the nonlinear case.

All in all our point of view raises many questions which have only been partially answered in the present article. However, we feel that the results presented are quite promising and we hope to clarify some of the open problems in future works.

Acknowledgements. The authors would like to thank Professors F. Hecht, F. Nicoud, O. Pironneau and J.H. Saiac for their deep interest and helpful comments during the realization of this work.

REFERENCES

[1] M.S. Agranovich, *Elliptic boundary problems, Partial differential equations IX*. M.S. Agranovich, Yu.V. Egorov and M.A. Shubin, Eds., Springer. *Encyclopaedia Math. Sci.* **79** (1997) 1–144.

[2] Å. Björck, *Numerical methods for least squares problems*, SIAM (1996).

[3] H. Borouchaki, P.L. George and B. Mohammadi, Delaunay mesh generation governed by metric specifications. Parts i & ii. Finite Elem. Anal. Des., Special Issue on Mesh Adaptation (1996) 345–420.

[4] M. Castro-Diaz, F. Hecht and B. Mohammadi, Anisotropic grid adaptation for inviscid and viscous flows simulations. *Int. J. Numer. Meth. Fl.* **25** (1995) 475–491.

[5] A. Douglis and L. Nirenberg, Interior estimates for elliptic systems of partial differential equations. *Comm. Pure Appl. Math.* **8** (1955) 503–538.

[6] P.I. Dudnikov and S.N. Samborski, Linear overdetermined systems of partial differential equations. Initial and initial-boundary value problems, Partial Differential Equations VIII, M.A. Shubin, Ed., Springer-Verlag, Berlin/Heidelberg. *Encyclopaedia Math. Sci.* **65** (1996) 1–86.

[7] FEMLAB 3.0, `http://www.comsol.com/products/femlab/`

[8] FREEFEM, `http://www.freefem.org/`

[9] P.L. George, *Automatic mesh generation. Applications to finite element method*, Wiley (1991).

[10] R. Glowinski, Finite element methods for incompressible viscous flow. *Handb. Numer. Anal.* Vol. IX, North-Holland, Amsterdam (2003) 3–1176.

[11] F. Hecht and B. Mohammadi, *Mesh adaptation by metric control for multi-scale phenomena and turbulence*. American Institute of Aeronautics and Astronautics **97-0859** (1997).

[12] B. Jiang, J. Wu and L. Povinelli, The origin of spurious solutions in computational electromagnetics. *J. Comput. Phys.* **7** (1996) 104–123.
[13] K. Krupchyk, W. Seiler and J. Tuomela, Overdetermined elliptic PDEs. *J. Found. Comp. Math.*, submitted.
[14] E.L. Mansfield, A simple criterion for involutivity. *J. London Math. Soc. (2)* **54** (1996) 323–345.
[15] B. Mohammadi and J. Tuomela, Involutivity and numerical solution of PDE systems, in *Proc. of ECCOMAS 2004, Vol. 1*, Jyväskylä, Finland. P. Neittaanmäki, T. Rossi, K. Majava and O. Pironneau, Eds., University of Jyväskylä (2004) 1–10.
[16] F. Nicoud, Conservative high-order finite-difference schemes for low-Mach number flows. *J. Comput. Phys.* **158** (2000) 71–97.
[17] O. Pironneau, *Finite element methods for fluids*, Wiley (1989).
[18] J.F. Pommaret, Systems of partial differential equations and Lie pseudogroups. *Math. Appl.*, Gordon and Breach Science Publishers **14** (1978).
[19] R.F. Probstein, *Physicochemical hydrodynamics*, Wiley (1995).
[20] A. Quarteroni and A. Valli, Numerical approximation of partial differential equations. *Springer Ser. Comput. Math.* **23** (1994).
[21] W.M. Seiler, *Involution — the formal theory of differential equations and its applications in computer algebra and numerical analysis*, Habilitation thesis, Dept. of Mathematics, Universität Mannheim (2001) (manuscript accepted for publication by Springer-Verlag).
[22] D. Spencer, Overdetermined systems of linear partial differential equations. *Bull. Am. Math. Soc.* **75** (1969) 179–239.
[23] J. Tuomela and T. Arponen, On the numerical solution of involutive ordinary differential systems. *IMA J. Numer. Anal.* **20** (2000) 561–599.
[24] J. Tuomela and T. Arponen, On the numerical solution of involutive ordinary differential systems: Higher order methods. *BIT* **41** (2001) 599–628.
[25] J. Tuomela, T. Arponen and V. Normi, On the numerical solution of involutive ordinary differential systems: Enhanced linear algebra. *IMA J. Numer. Anal.*, submitted.