Journal of Engineering Mathematics 18 (1984) 219–233.

Forms of the generalised Navier-Stokes equations

P.R. VOKE and M.W. COLLINS

The City University, Department of Mechanical Engineering, Northampton Square, London EC1V 0HB, England

(Received January 3, 1984)

Summary

We review the governing equations of fluid dynamics in general co-ordinates, and present forms particularly suitable for efficient numerical solution. Methods are given, both for compressible and incompressible flow, of eliminating explicit use of the connection coefficients and transformation matrix elements from the computations. Special problems associated with spacially varying viscosity and with eddy-viscosity turbulence models are also discussed.

1. Introduction

Co-ordinate transformation methods have a long and honourable history of service to computational fluid dynamics. Their use has usually been restricted to two-dimensional calculations for a number of sound reasons.

First, a three-dimensional geometry will give rise to a three-dimensional flow, which is expensive to compute compared to a flow with one translationally symmetric dimension. If co-ordinate transformations are also introduced, the computational expense frequently becomes prohibitive. Second, an orthogonal transformation may always be found to map a two-dimensional geometry onto a simple region such as a rectangle, but in general orthogonal surface-following co-ordinate systems cannot be found for three-dimensional geometries; a general co-ordinate transformation must be used instead.

It is the purpose of this paper to consider the use of general co-ordinate transformations, and ways of reducing the penalties incurred by attempting to solve the generalised Navier-Stokes equations in two or three dimensions. We shall not consider here the question of how the general co-ordinate transformation is initially defined [1]; only how the resulting equations may be efficiently solved numerically. The equations are well known [2], but have not found extensive use. An alternative form of the momentum equation suggested by Peyret and Viviand [3] is sometimes more computationally convenient, and is discussed in Section 4.

The difficulties of numerical calculation, particularly in the three-dimensional case, are formidable. Co-ordinate transformations are used in order to simplify the form of boundary condition: in a complex geometry the boundary conditions have a correspondingly complex form, but by introducing a surface-following co-ordinate system the boundary conditions become extremely simple [4] (for instance some velocity or potential may be fixed on a surface on which one of the transformed co-ordinates is also fixed). The

price paid for this simplification is that the equations become more complicated, since they now involve geometry-dependent terms arising from the transformation. In the case of a general co-ordinate transformation, the increase in the complexity of the equation is so great that the price has often been considered too high in comparison with the benefits. It is for this reason that general co-ordinate transformation methods have been neglected in favour of regular meshes, finite elements, or orthogonal co-ordinate systems.

The additional complexity of the generalised equations arises from two easily identified sources. The first is the metric tensor of the transformation; the second is the affine connection coefficients. Both are defined in Appendix B. For a three-dimensional transformation, the metric has six independent components at each point in space, while the connection (which is used for calculating derivatives) has eighteen independent components. The alternative form of the momentum equations [3] does not contain the connection coefficients but instead involves the transformation matrices, with nine independent components.

It is easy to see that the amount of computation can become prohibitive if an attempt is made to perform a practical calculation on a three-dimensional mesh of significant size. The geometric quantities (the metric and the connection coefficients or transformation matrices) must either be recomputed each time they are needed (each iteration, or each time step), or be computed once and stored. The storage of fifteen or twenty-four independent components for each mesh point is likely to be too high a price to pay for the benefits a co-ordinate transformation yields. Even if this problem can be solved satisfactorily, it is evident that the geometric objects are used at several stages in any solution, and impose a significant extra computational load. Storage may be saved by recomputing the metric components from the transformation matrices, or the connection coefficients from the metric, but this requires additional processor time.

In Sections 5 to 8 we attempt to mitigate some of these penalties by developing forms of the generalised Navier-Stokes equations in which neither the connection coefficients nor the transformation matrix elements appear explicitly. This simultaneously simplifies the computations needed to solve the equations and reduces by a factor of three or four the storage (or recomputation) needed for the geometric objects, since only the components of the metric are used. In Sections 10 and 11 attention is given to cases in which the connection coefficients cannot be completely eliminated.

2. The Cartesian equations

The reader is assumed to be familiar with three-dimensional Cartesian tensors and the notations associated with them [2,5]. The Einstein summation convention, stating that indices repeated in any term are summed from 1 to 3, is used throughout.

The equations that form the basis for our discussions are as follows:

$$\partial_t \rho + \partial_i(\rho v_i) = 0, \tag{1}$$

$$\rho \mathrm{D}v_i/\mathrm{D}t = F_i + \partial_j \tau_{ij}. \tag{2}$$

The first of these is the equation of continuity, representing conservation of mass. The second gives the conservation of momentum. $\mathrm{D}/\mathrm{D}t$ is the so-called material derivative,

related to the partial derivative at a fixed point by

$$D/Dt = \partial_t + v_i \partial_i. \tag{3}$$

The symbols are defined in Appendix A. These equations are for a Newtonian fluid. The total stress is split into a pressure and a residual stress term:

$$\tau_{ij} = \sigma_{ij} - p\delta_{ij} \tag{4}$$

with the pressure usually assumed to be given by

$$p = -\tau_{ii}/3. \tag{5}$$

A Newtonian fluid supports a simple relationship between σ_{ij}, the strain rate

$$s_{ij} = \tfrac{1}{2}(\partial_i v_j + \partial_j v_i) \tag{6}$$

and the dilatation

$$\theta = \partial_i v_i. \tag{7}$$

This relationship is

$$\sigma_{ij} = 2\mu s_{ij} + \lambda\theta\delta_{ij}. \tag{8}$$

If the definition (5) of pressure is assumed, (4), (6), (7) and (8) together imply that

$$\lambda = -2\mu/3. \tag{9a}$$

This relation is closely followed by real fluids, though in general

$$\lambda < -2\mu/3. \tag{9b}$$

The momentum equation may therefore be written:

$$\rho(\partial_t v_i + v_j \partial_j v_i) = F_i + \partial_j(-p\delta_{ij} + 2\mu s_{ij} + \lambda\theta\delta_{ij}). \tag{10}$$

With these equations we take a typical form for a temperature equation:

$$\rho c_p DT/Dt = k\partial^2 T - p\theta + 2\mu s_{ij} s_{ij} + \lambda\theta^2. \tag{11}$$

This will serve as a paradigm of all equations governing the transport of scalar quantities in the flow, and is not intended to be the most general or the most useful form of the energy conservation equation. The set of equations is completed by a constitutive equation relating density, pressure and temperature:

$$\rho = \rho(p, T). \tag{12}$$

3. The general equations

Properly formulated tensor equations are valid in any co-ordinate system in a Riemannian space. The conservation equations (9), (10), (11) and (12) given above must therefore be special (Cartesian) cases of generalised conservation laws.

There are two major differences. First, the partial derivatives used in the Cartesian equations are a special case of the covariant derivative, which is written using a comma notation thus:

$$s_{ij} = \tfrac{1}{2}(u_{i,j} + u_{j,i}). \tag{13}$$

Covariant differentation is defined in Appendix B.

Second, co-ordinate differentials $\mathrm{d}x^i$ and scalar gradients such as $\partial_i p$ transform differently under co-ordinate transformations. The two types of description are distinguished by the terms "contravariant" and "covariant", and by super- and subscript indices respectively, as shown above. For further details of the notation and methods of generalised tensors, the reader is referred to Appendix B, to a previous publication [6] or to any standard text [2,5]. The above comments, however, are sufficient to justify the following form of the generalised equations:

$$\partial_t \rho + (\rho u^i)_{,i} = 0, \tag{14}$$

$$\rho(\partial_t u^i + u^j u^i_{,j})$$

$$= F^i - (g^{ij}p)_{,j} + (2\mu s^{ij})_{,j} + (\lambda\theta g^{ij})_{,j}, \tag{15a}$$

$$\theta = u^i_{,i}, \tag{15b}$$

$$s^{ij} = \tfrac{1}{2}(g^{ik}u^j_{,k} + g^{jk}u^i_{,k}), \tag{15c}$$

$$\rho c_p(\partial_t T + u^j T_{,j}) = k g^{ij} T_{,ij} - p u^i_{,i} + 2\mu s^{ij} s_{ij} + \lambda\theta^2, \tag{16}$$

$$\rho = \rho(p, T). \tag{17}$$

The tensor g^{ij} appearing in Eqns. (15) and (16) is the metric, defined in Appendix B. The reader should note that the comma notation for the covariant derivative makes the equations appear deceptively clear and simple. It is important to remember that the covariant derivative of any vector or tensor involves terms containing the connection coefficients (Appendix B). To prevent any misunderstanding, the full form of the momentum equation (15) is given below, with all the covariant derivatives expanded:

$$\rho\partial_t u^i + \rho u^j(\partial_j u^i + \Gamma^i_{jk} u^k)$$

$$= F^i - g^{ij}p_{,j} + \partial_j(2\mu s^{ij}) + 2\mu\Gamma^i_{jk}s^{kj} + 2\mu\Gamma^j_{jk}s^{ik} + g^{ij}\partial_j(\lambda\theta), \tag{18a}$$

$$\theta = \partial_i u^i + \Gamma^i_{ik} u^k, \tag{18b}$$

$$s^{ij} = \tfrac{1}{2}(g^{ik}\partial_k u^j + g^{ik}\Gamma^j_{kl}u^l + g^{jk}\partial_k u^i + g^{jk}\Gamma^i_{kl}u^l). \tag{18c}$$

These equations should be sufficient to convince the reader that computations using general co-ordinates can be onerous if approached in the wrong way.

In Sections 5 to 8 we shall consider each of the equations (14) to (17) in turn, in order to determine the best form for its use in computation.

4. The semi-Cartesian form

The form of the Navier-Stokes equations given in the previous section is one that comes naturally from any analysis imposing the conservation of scalar mass and energy, and the contravariant components of the vector momentum. It is equivalent to that given by Aris [2], and rederived by Gal-Chen and Somerville [7]. The latter authors found a computationally more efficient form of the equations since they went on to use them in a practical calculation [8].

In 1975 Peyret and Viviand [3] suggested that "a general conservative form of the Navier-Stokes equations can be obained by making an arbitrary time-dependent transformation of co-ordinates on the scalar equations based on the cartesian components of velocity and momentum". They gave a two-dimensional version of such equations.

Roscoe [15] independently derived one form of these equations, and gave a three-dimensional version in a particularly clear and concise notation. The equations were used for a practical calculation, though Roscoe chose to restrict the study to orthogonal co-ordinates, constant viscosity, and steady flow.

This form of the equations arises from the observation that the components of momentum are physically conserved, regardless of which co-ordinate basis is used to project the components. Since the Cartesian components are conserved and are related in a known way through the transformation matrices to the conyariant components that occur in the continuity equation, the conservation equation for a scalar quantity may be applied to each Cartesian component separately. The resulting form of the momentum equation is

$$\rho \mathrm{D}v_i/\mathrm{D}t = F_i - \partial p/\partial y^i + \left(g^{jl}\mu v_{i,j}\right)_{,l} + (\lambda\theta)_{,i}. \tag{19}$$

Using Eqn. (B5) (Appendix B) to write the Cartesian components in terms of the covariant components, we obtain

$$\rho \mathrm{D}\left(c_j^i u^j\right)/\mathrm{D}t = F_i - \bar{c}_i^j \partial p/\partial x^j + \left[g^{jl}\mu\left(c_k^i u^k\right)_{,j}\right]_{,l} + (\lambda\theta)_{,i}. \tag{20}$$

The matrix elements c_j^i are given by (B2). Since the viscous term has become a simple divergence (the Cartesian velocity component is essentially a scalar in this context), Eqn. (B18) may be used to replace the covariant derivative by an appropriate partial derivative:

$$\begin{aligned}
&\rho\partial_t\left(c_j^i u^j\right) + \rho u^k \partial\left(c_j^i u^j\right)/\partial x^k \\
&\quad = F_i - \bar{c}_i^j \partial p/\partial x^j + J^{-1}\frac{\partial}{\partial x^j}\left[J\mu g^{jl}\partial\left(c_k^i u^k\right)/\partial x^l\right] \\
&\quad = \partial(\lambda\theta)/\partial x^i.
\end{aligned} \tag{21}$$

The equation used by Roscoe was a particular case of this equation, in conjunction with the form of the continuity equation given in the next section.

The semi-Cartesian form (our term) for the momentum equation has the advantage of eliminating the connection coefficients, at the expense of introducing the matrix elements c_k^i. This represents a significant reduction in the computational complexity of the equations. The form is convenient for finite difference or finite element computations, though not for spectral methods. It is not clear whether a semi-Cartesian form of the energy transport equation can be formulated if the viscous dissipation term is included.

The following four sections are an attempt to eliminate the use of both the connection coefficients and the transformation elements from the full set of equations (14) to (17).

5. Continuity

Equation (14) may be expanded as follows:

$$\partial_t \rho + u^i \partial_i \rho + \rho u^i{}_{,i} = 0. \tag{22}$$

We use the relationship (B18) to rewrite:

$$u^i{}_{,i} = \partial_i u^i + \Gamma^j_{ji} u^i = \partial_i u^i + \left(J^{-1} \partial_i J \right) u^i = J^{-1} \partial_i (J u^i). \tag{23}$$

This allows the continuity equation to be written without involving the connection coefficients:

$$\partial_t \rho + u^i \partial_i \rho + \rho J^{-1} \partial_i (J u^i) = 0. \tag{24}$$

The dilatation may also be computed without involving the connection, by using the original equation:

$$\theta = u^i{}_{,i} = -\rho^{-1} \partial_i \rho - \rho^{-1} u^i \partial_i \rho. \tag{25}$$

In incompressible flow, Eqn. (24) reduces to a particularly simple and well known form:

$$\partial_i (J u^i) = 0. \tag{26}$$

6. Conservation of momentum

As we have seen, the fully expanded form of the momentum equations is not simple. In order to make computations practical, we first note that the vorticity may be computed without involving the connection coefficients,

$$2\omega_{ij} = u_{j,i} - u_{i,j} = \partial_i u_j - \partial_j u_i - \Gamma^k_{ij} u_k + \Gamma^k_{ij} u_k = \partial_i u_j - \partial_j u_i \tag{27}$$

which is clearly true of any skew-symmetric tensor that is the "curl" of a vector.

This convenient property of the vorticity suggests that the rotation form of the general Navier-Stokes equation may be simpler than other forms. Defining a total pressure by

$$P = p/\rho + u^i u_i/2 \tag{28}$$

and noting that

$$u^j u^i_{,j} = u^j g^{ik}\left(u_{k,j} - u_{j,k}\right) + \tfrac{1}{2} g^{ik}\left(u^j u_j\right)_{,k} \tag{29}$$

we can rewrite (15a) as

$$\partial_t u^i = F^i/\rho - g^{ij}\partial_j P + 2u^j\omega^i_j + 2\rho^{-1}\left(\mu s^{ij}\right)_{,j} + \rho^{-1}\left(\lambda\theta g^{ij}\right)_{,j} \tag{30}$$

where the vorticity components ω^i_j are defined by (27) and (B11).

The major source of complexity in the equations remains in the viscous terms. We first note that

$$\left(\mu s^{ij}\right)_{,j} = s^{ij}\partial_j\mu + \mu s^{ij}_{,j} \tag{31}$$

and then use the continuity equation (25) to rewrite the second term:

$$\mu s^{ij}_{,j} = \tfrac{1}{2}\mu\left(g^{jk}u^i_{,jk} + g^{ik}u^j_{,jk}\right) = \tfrac{1}{2}\mu\left(g^{jk}u^i_{,jk} - g^{ik}u^j_{,jk}\right) + \mu g^{ik}\theta_{,k} = -\mu\omega^{ij}_{,j} + \mu g^{ik}\theta_{,k}. \tag{32}$$

Since the first term in (32) is a simple divergence, we may again use Eqn. (B18) to eliminate one occurrence of the connection coefficients:

$$\omega^{ij}_{,j} = \partial_j\omega^{ij} + \Gamma^j_{jk}\omega^{ik} + \Gamma^i_{jk}\omega^{kj} = J^{-1}\partial_j\left(J\omega^{ij}\right) + \Gamma^i_{jk}\omega^{kj}. \tag{33}$$

However, since the vorticity is a skew-symmetric tensor, while the connection Γ^i_{jk} is symmetric in its lower two indices, the term on the right of (33) also vanishes:

$$\omega^{ij}_{,j} = J^{-1}\partial_j\left(J\omega^{ij}\right). \tag{34}$$

Substituting (34), (32) and (31) into (30), the momentum equations may be written in a form which only involves the connection in the computation of s in the last term:

$$\partial_t u^i = \rho^{-1}F^i - g^{ij}\partial_j P + 2u^j\omega^i_j + \\ -2\nu J^{-1}\partial_j\left(J\omega^{ij}\right) + \rho^{-1}g^{ij}\left[\left(\lambda + 2\mu\right)\partial_j\theta + \theta\partial_j\lambda\right] + 2\rho^{-1}s^{ij}\partial_j\mu. \tag{35}$$

The terms involving θ would be computed using (7), so that the connection is not involved.

In incompressible flow, the above equations reduce further to

$$\partial_t u^i = \rho^{-1}F^i - g^{ij}\partial_j P + 2u^j\omega^i_j - 2\nu J^{-1}\partial_j\left(J\omega^{ij}\right) + 2\rho^{-1}s^{ij}\partial_j\mu. \tag{36}$$

In particular cases in which the viscosity may be assumed to be constant, the last term on the right-hand side of (35) or (36) is absent. These are forms of the momentum

equations in which the connection coefficients do not appear at all. In conjunction with the continuity equation (24) or (26), and the temperature and constitutive equations to be discussed below, they represent a very efficient approach to the solution of the generalised equations, in the case where the spacial variation of the viscosity is negligible. The best approach in cases where there are significant viscosity gradients will be discussed in Section 10.

7. Driving forces

The driving force F^i on the right-hand side of the momentum equation may take a number of forms. The forces most commonly encountered are imposed pressure gradients or buoyancy forces. In many types of calculation where inflow and outflow boundary conditions are imposed, a mean pressure gradient will not be explicitly inserted into the calculation in this fashion, but rather will be a calculated result.

If explicit vector driving forces are to be included in a computation, their contravariant components in the curvilinear co-ordinate system may be computed in advance, using the equation

$$F^i = g^{ij} c_j^k \partial p / \partial y^k \tag{37}$$

for a typical pressure gradient, and

$$F^i = (\rho - \rho_0) c_3^i g \tag{38}$$

for a buoyancy force acting in the y^3 direction.

This represents a more efficient approach than storing the elements of c_i^j, but still requires the storage of two or three additional real arrays, the contravariant components of the driving force at each mesh point. We consider this to be a significant penalty associated with explicit vector driving force terms, since otherwise only nineteen such arrays are needed even for a full three-dimensional direct simulation in a three-dimensional geometry [9]. To reduce the storage required, we note that a mean scalar pressure may be stored for each mesh point, and its covariant derivative recomputed quite simply as required; this requires only one array instead of two or three. A similar trick can be used for buoyancy forces. These considerations are of particular importance for three-dimensional transformations.

8. Scalar equations

We now turn to the energy conservation equation (11), and the constitutive equation (12). Since the latter is a pure scalar relationship, it has the same form in any co-ordinate system, and we need consider it no further.

The energy equation is also scalar, but involves vector or tensor quantities in the computation of several terms. The Laplacian term is a divergence and may be treated in a now familiar manner:

$$g^{ij} T_{,ij} = g^{ij} (\partial_i T)_{,j} = J^{-1} \partial_j \left(J g^{ij} \partial_i T \right). \tag{39}$$

The terms involving θ may also be calculated using (7), as in the momentum equations.

The viscous dissipation $2\mu s^{ij}s_{ij}$ can only be found by computing the strain rate using Eqn. (6), and involves the connection coefficients. In a wide class of flows this term may be neglected, however; it is only important in rather low Reynolds-number flows. For an incompressible flow in which the dissipation makes a negligible contribution to the total energy budget, the transport equation reduces to

$$\rho c_p \partial_t T = -\rho c_p u^j \partial_j T + kJ^{-1}\partial_j\left(Jg^{ij}\partial_i T\right). \tag{40}$$

This equation is typical of scalar transport equations in general, which can usually be reduced to some form which excludes the explicit use of the connection coefficients.

9. Summary of compact forms

The foregoing sections lead us to sets of equations for both compressible and incompressible flow which do not involve the connection coefficients in any explicit computations. The conditions under which these compact forms exist are as follows:

(a) the viscosity (or effective viscosity) present in the momentum equations must be spacially unvarying;
(b) if an energy equation is part of the system, the viscous dissipation term must be negligible.

Provided these two conditions hold, considerable savings may be effected, either of computation time or of storage space, in iterative solutions or time-stepping simulations of the generalised Navier-Stokes equations using primitive variables. For a three-dimensional geometry, the six independent covariant components of the metric g_{ij} are computed initially, and stored for each point of the mesh. For a two-dimensional geometry there are three independent components of the metric. The components of the affine connection are not needed; neither are the transformation matrices or the Cartesian positions of the mesh points, except for transforming information back to Cartesian co-ordinates for output.

The other quantities that are needed are the contravariant components g^{ij} of the metric, and the Jacobian of the transformation J. These may be computed from g_{ij} using Eqns. (B8) and (B4) as needed, or stored, according to the relative cost of space and time for the computation. We adopt the latter philosophy for two-dimensional transformations, the former for three-dimensional.

These methods have been implemented for Navier-Stokes simulations of incompressible flow in distorted two-dimensional geometries. Typically the simultions of the generalised equations take 50 to 90% longer than comparable simultions in Cartesian co-ordinates, and require 50% more storage space. The results of these simulations and of simulations in three-dimensional geometries will be reported elsewhere. Our intention is to extend the code to three dimensions when this is required. The three-dimensional calculations require more than twice as much computer storage space as the Cartesian case, and we expect a 100 to 150% increase in the time required. Nevertheless, these increases are within reasonable bounds; in all cases the connection coefficients are not computed or employed, and the transformation matrix elements are retained in backing storage for graphical output.

The equations used are summarised here, first for compressible flow:

$$\partial_t \rho + u^i \partial_i \rho + \rho J^{-1} \partial_i (J u^i) = 0, \tag{41}$$

$$\partial_t u^i = \rho^{-1} F^i - g^{ij} \partial_j P + 2 u^j \omega_j^i - 2 \nu J^{-1} \partial_j (J \omega^{ij}) + \rho^{-1} g^{ij} \left[(\lambda + 2\mu) \partial_j \theta + \theta \partial_j \lambda \right], \tag{42}$$

$$\rho c_p \partial_t T = -\rho c_p u^j \partial_j T + k J^{-1} \partial_j \left(J g^{ij} \partial_i T \right) - p\theta + \lambda \theta^2, \tag{43}$$

$$\rho = \rho(p, T). \tag{44}$$

For incompressible flow:

$$\partial_i (J u^i) = 0, \tag{45}$$

$$\partial_t u^i = \rho^{-1} F^i - g^{ij} \partial_j P + 2 u^j \omega_j^i - 2 \nu J^{-1} \partial_j (J \omega^{ij}), \tag{46}$$

$$\rho c_p \partial_t T = -\rho c_p u^j \partial_j T + k J^{-1} \partial_j \left(J g^{ij} \partial_i T \right). \tag{47}$$

We now turn to a discussion of the best approach for computing the terms omitted from the above equations.

10. The computation of strain

We have not found any method of manipulating terms involving the contravariant components of the strain rate s^{ij} (as opposed to its divergence) which allows their computation without the use of the connection coefficients. The equation defining the strain in general co-ordinates is (18c). However, it should be noted that the components of the connection occur once only in this equation owing to the symmetry of s^{ij}. The covariant components may also be computed, by

$$s_{ij} = \tfrac{1}{2} \left(\partial_i u_j + \partial_j u_i \right) - \Gamma_{ij}^k u_k. \tag{48}$$

The "viscosity gradient" force,

$$2 \rho^{-1} s^{ij} \partial_j \mu \tag{49}$$

we suggest, may best be found by recomputing Γ_{ij}^k at each iteration (or time step), since it is possible to calculate the term using each of the eighteen independent components of Γ once only.

It may be possible to compute the viscous dissipation, should this also be non-negligible, in the same loop, so that each connection coefficient need only be found once. In a finite difference computation, a single recomputation of the eighteen components of Γ may not be too burdensome; in a spectral calculation, the use of the connection coefficients leads to a large number of additional Fourier (or other) transforms, and is expensive to implement.

11. Turbulent eddy viscosity

One type of computation that always involves a spacially varying viscosity coefficient is the turbulence model. Most simple turbulence closure models, or the subgrid scale models of large-eddy simulations, use Boussinesq eddy-viscosity terms. The eddy viscosity may be derived in a very simple or in a sophisticated manner, but if it is to model the turbulence at all adequately it must fall to zero at solid boundaries, and be large where the (unresolved) turbulent transport is high. Only extremely simple types of turbulence can be modelled with a constant eddy viscosity [10]. Near walls, the gradients of eddy viscosity are frequently large.

The conclusion that the methods given in this paper cannot be applied to Boussinesq turbulence models in general co-ordinates may nevertheless be premature. The general covariant form of the modelling term is as follows:

$$2\rho^{-1}\left(\mu_e s^{ij}\right)_{,j} = 2\rho^{-1}\mu_e s^{ij}_{,j} + 2\rho^{-1}s^{ij}\mu_{e,j}. \tag{50}$$

We point out that each half of the RHS of (50) is separately covariant, and that either could be included alone. If the viscosity gradient term is omitted, the result is a "local Boussinesq model":

$$2\nu_e s^{ij}_{,j} = -2\nu_e J^{-1}\partial_j\left(J\omega^{ij}\right). \tag{51}$$

Unless the turbulence modelling is critical, we suggest this may be an adequate replacement for the more usual form (50). The omitted term is significant in boundary layers. In the case of large-eddy simulations [11,12,13], which are believed to be less model-dependent than time-average closures [14], a subgrid scale model of the form (51) may be as useful as (50). The form (51) implies that the acceleration due to the presence of (unresolved) turbulence is parallel to the molecular viscous force (assuming μ to be constant), while (50) does not. We expect the magnitude of the turbulent momentum transfer to depend predominantly on the size of μ_e, and not to be affected greatly by the presence of the viscosity gradient term, except near boundaries. Some additional tuning of the decrease of the subgrid-scale eddy viscosity in boundary layers may be needed to allow for the effect of the omitted term.

Appendix A

Notation

c_p	specific heat
c^i_j	transformation matrix elements
$\mathrm{D}/\mathrm{D}t$	material derivative
∂_i	partial derivative with respect to x^i
∂^2	partial Laplacian
∂_t	partial derivative with respect to time
F^i	driving force

g	gravitation
g_{ij}	covariant metric tensor
J	transformation Jacobian
k	thermal conductivity
p	pressure
P	total pressure
s^{ij}	strain tensor
t	time
T	temperature
u^i	generalised velocity, contravariant components
v_i	Cartesian velocity components
x^i	general co-ordinates
y^i	Cartesian co-ordinates
Γ^k_{ij}	affine connection coefficients (Christoffel symbols)
θ	dilatation
λ	second coefficient of viscosity
μ	dynamic viscosity
ν	kinematic viscosity (first coefficient)
ρ	density
σ^{ij}	residual stress tensor
τ^{ij}	stress tensor
ω^{ij}	vorticity tensor

Sub and Superscripts
i, j, k, l and m take values 1, 2, 3. Summation over these values is implied wherever the sub or superscript is repeated in any term.

Appendix B

Relationships from General Tensor Calculus

A co-ordinate transformation is assumed to be defined as a mapping of some convoluted region of physical three-dimensional space into some simpler space (for instance a cuboid). The only restrictions on the mapping are that it should be one-to-one and onto. The natural co-ordinates x of the simpler image space are then related to the Cartesian co-ordinates y of the original space by an invertible function:

$$y^i = y^i(x^1, x^2, x^3), \quad (i = 1 \text{ to } 3). \tag{B1}$$

The transformation matrix elements are given by

$$c^i_j = \partial y^i / \partial x^j \tag{B2}$$

and their inverses by

$$\bar{c}^j_i = \partial x^j / \partial y^i = \left(c^i_j\right)^{-1}. \tag{B3}$$

The Jacobian of the transformation is

$$J = |c_j^i| = |g_{ij}|^{1/2}. \tag{B4}$$

The relationship between a Cartesian vector v^i in the object space and its counterpart general vector in the image space is simply

$$v^j = c_i^j u^i. \tag{B5}$$

This is the transformation law for vectors that transform in the same way as the co-ordinate differentials $\mathrm{d}x^i$. The gradient of a scalar of necessity transforms in an inverse manner:

$$\partial p/\partial y^j = \bar{c}_j^i \partial p/\partial x^i = \bar{c}_j^i \partial_i p. \tag{B6}$$

Vectors such as velocities, accelerations and forces transform as (B5), and are known as contravariant; those transforming like gradients (B6) are called covariant. The two types are distinguished by the position of the index, as indicated in (B5) and (B6). Higher-rank tensors may have all indices contravariant (up), all covariant (down), or be of mixed type.

The metric of a transformation is defined by

$$g_{ij} = \sum_{k=1}^{3} c_i^k c_j^k. \tag{B7}$$

It may be shown that g_{ij} transforms as a covariant tensor. Its inverse

$$g^{ij} = (g_{ij})^{-1} = \sum_{k=1}^{3} \bar{c}_k^i \bar{c}_k^j \tag{B8}$$

transforms as a contravariant tensor. Since the contraction (dot product) of one tensor with another is also a valid tensor, g_{ij} is used to derive covariant components for contravariant vectors or tensors, while g^{ij} reverses the process, thus:

$$u_i = g_{ij} u^j, \tag{B9}$$

$$u^j = g^{ij} u_i, \tag{B10}$$

$$\omega_j^i = g^{ik} \omega_{kj}. \tag{B11}$$

Note, however, that only covariant and contravariant indices may be contracted together (summed over). The following quantity, for instance, is *not* a well-defined scalar:

$$u_i \partial_i p. \tag{B12}$$

As shown in (B6), the gradient of a scalar quantity is a covariant vector. However, partial derivatives of higher rank tensors are not properly defined tensors, and in order to be able to write down equations in a co-ordinate independent form, a more general

concept, the covariant derivative, must be introduced. The covariant derivatives of tensors up to the second rank are as follows:

$$u^i_{,j} = \partial_j u^i + \Gamma^i_{jk} u^k, \tag{B13a}$$

$$u_{i,j} = \partial_j u_i - \Gamma^k_{ij} u_k, \tag{B13b}$$

$$s^{ij}_{,k} = \partial_k s^{ij} + \Gamma^i_{kl} s^{lj} + \Gamma^j_{kl} s^{il}, \tag{B13c}$$

$$\omega^i_{j,k} = \partial_k \omega^i_j + \Gamma^i_{kl} \omega^l_j - \Gamma^l_{kj} \omega^i_l, \tag{B13d}$$

$$s_{ij,k} = \partial_k s_{ij} - \Gamma^l_{ki} s_{lj} - \Gamma^l_{kj} s_{il}. \tag{B13e}$$

The quantities Γ^k_{ij} are the components of the affine connection, or connection coefficients, and are derived from the metric tensors:

$$\Gamma^k_{ij} = \tfrac{1}{2} g^{kl} \left(\partial_i g_{jl} + \partial_j g_{il} - \partial_l g_{ij} \right). \tag{B14}$$

The connection coefficients are also known as Christoffel symbols of the second kind, written:

$$\left\{ \begin{matrix} k \\ i\ j \end{matrix} \right\} = \Gamma^k_{ij}. \tag{B15}$$

From the examples above, it should be evident that in finding the covariant derivative of a tensor field the Γ's enter once for each index, with a positive sign for each contravariant index and with a negative sign for each covariant index. In Cartesian co-ordinates the components of the metric are those of a unit matrix ρ_{ij} everywhere, the Γ^k_{ij} vanish, and the covariant derivative becomes identical to the partial derivative. The Γ^k_{ij}, like the c^i_j, are not tensor components, but geometric objects dependent on the co-ordinate system in use.

Several important relationships are needed to understand the main text of this paper. In particular, the fact that the metric tensors commute with the covariant derivative operator

$$\left(g^{ij} p \right)_{,j} = g^{ij} p_{,j} \tag{B16}$$

follows from Ricci's lemma:

$$g_{ij,k} = g^{ij}_{,k} = 0. \tag{B17}$$

This follows immediately from the general principle of tensor calculus, that properly formulated tensor equations are equally true (or equally false) in all co-ordinate systems. (B17) is clearly true in Cartesian co-ordinates.

A second relationship of which we make much use is the following:

$$\Gamma^j_{ji} = \Gamma^j_{ij} = J^{-1} \partial_i J. \tag{B18}$$

The proof will be found in any standard text, [5].

Acknowledgements

This work was performed under a co-operative Research Grant between the Science and Engineering Research Council (grant GR/B41452 SRC), the United Kingdom Atomic Energy Authority (contract no. M2C3493 EMR) and the Department of Mechanical Engineering, The City University, London.

References

[1] J.F. Thompson, F.C. Thames and C.W. Mastin, Boundary-fitted curvilinear co-ordinate systems for solution of partial differential equations on fields containing any number of arbitrary two-dimensional bodies, *NASA CR-2729*, NASA, Washington D.C. (1977).

[2] R. Aris, *Vectors, tensors, and the basic equations of fluid mechanics,* Prentice-Hall, London, (1962).

[3] R. Peyret and H. Viviand, Computations of viscous compressible flows based on the Navier-Stokes equations, *AGARD-AG-212 (AGARDograph no. 212),* ed. J.J. Smolderen, AGARD (1975).

[4] J.F. Thompson, F.C. Thames and C.W. Mastin, Automatic numerical generation of body-fitted curvilinear co-ordinate system for field containing any number of arbitrary two-dimensional bodies, *J. Computational Physics* 15 (1974) 299–319.

[5] J.L. Synge and A. Schild, *Tensor calculus*, Univ. of Toronto Press, Toronto (1949).

[6] P.R. Voke, Fluid dynamic simulations in curvilinear co-ordinate systems, *U.K.A.E.A. Harwell report AERE R-10548*, H.M.S.O. (1982).

[7] T. Gal-Chen and R.J.C. Somerville, On the use of co-ordinate transformation for the solution of the Navier-Stokes equations, *J. Computational Physics* 17 (1975) 209–225.

[8] T. Gal-Chen and R.C.J. Somerville, Numerical solution of the Navier-Stokes equations with topography, *J. Computational Physics* 17 (1975) 276–309.

[9] P.R. Voke and M.W. Collins, Fluid dynamic simulations in general co-ordinates, *Proceedings of the Third International Conference on Numerical Methods in Laminar and Turbulent Flow, (Seattle 1983),* Pineridge Press, Swansea, U.K. (1983) 1204–1212.

[10] R.A. Clark, J.H. Ferziger and W.C. Reynolds, Evaluation of subgrid-scale turbulence models using a fully simulated turbulent flow, *Report no. TF-9*, Thermosciences Division, Dept. of Mechanical Engineering, Stanford University (1977).

[11] W.C. Reynolds, Computation of turbulent flows, *Ann. Rev. Fluid Mech. 8,* ed. Van Dyke, Vincenti and Wehausen (1976) 183–200.

[12] J.H. Ferziger, Large eddy numerical simulations of turbulent flows, *AIAA Journal* 15 (1977) 1261–1267.

[13] W.C. Reynolds and T. Cebeci, Calculation of turbulent flows, *Turbulence,* ed. P. Bradshaw, Topics in Applied Physics, Vol. 12, Springer-Verlag, (1978) 193–229.

[14] P.R. Voke and M.W. Collins, Large-eddy simulation: retrospect and prospect, *U.K.A.E.A. Harwell report AERE R-10716,* H.M.S.O. (1982).

[15] D.F. Roscoe, The numerical solution of the Navier-Stokes equations for a three-dimensional laminar flow in curved pipes using finite-difference methods. *J. Eng. Math.* 12 (1978) 303–323.