A Boussinesq Algorithm for Buoyant Convection in Polygonal Domains

K. B. McGrattan*
R. G. Rehm*
H. C. Tang*
H. R. Baum**

U.S. DEPARTMENT OF COMMERCE
Technology Administration
National Institute of Standards
and Technology
Computing and Applied Mathematics Laboratory*
Building and Fire Research Laboratory**
Gaithersburg, MD 20899

NIST

.U5/

4831
1992

A Boussinesq Algorithm for Buoyant Convection in Polygonal Domains

K. B. McGrattan*
R. G. Rehm*
H. C. Tang*
H. R. Baum**

U.S. DEPARTMENT OF COMMERCE
Technology Administration
National Institute of Standards
and Technology
Computing and Applied Mathematics Laboratory*
Building and Fire Research Laboratory**
Gaithersburg, MD 20899

A Boussinesq Algorithm for Buoyant Convection in Polygonal Domains

K.B. McGrattan, R.G. Rehm, H.C. Tang and H.R. Baum
National Institute of Standards and Technology
Gaithersburg, MD 20899

A 2-D Boussinesq model describing heat-driven buoyant convection in a polygonal enclosure is presented. The hydrodynamics is based on the time-dependent Navier-Stokes equations with constant viscosity and thermal conductivity; no turbulence model or other empirical parameters are introduced. The polygonal domain is mapped via a Schwarz-Christoffel transformation onto a rectangle. A finite difference scheme second-order in space and first-order in time is used to integrate the evolution equations, and an elliptic solver is used to solve the pressure equation. Computational results for high Reynolds numbers are presented through the use of Lagrangian particles which allow one to visualize the flow patterns.

1   Introduction

In computational aerodynamics, calculations of the steady-state flow over aircraft have routinely required the use of sophisticated grid-generating techniques [1]. In other areas, such as ship hydrodynamics, flow in machinery, and shore effects in oceanography, non-rectangular grids are routinely used. In fire research, however, the study of fluid flow in more complicated domains is a more recent activity. Many of the techniques developed by the CFD community for solving the flow equations on non-rectangular grids have one or more of the following limitations: 1) They are designed to compute only the steady-state solution. 2) They rely on various types of turbulence models. 3) They are inefficient due to the use of unstructured grids.

We have previously developed an algorithm for the computation of the time-dependent buoyant convection induced by a room fire [2] – [7]. Although this model has proven to be a useful tool in fire research, it is limited to flows in rectangular enclosures. In this document we generalize our algorithm to allow for polygonally-shaped domains, making it possible to study the geometrical effects of building elements such as windows, soffits and stairways. A conformal map from the polygon to a rectangle is used to transform the spacial coordinates of the Boussinesq equations. The advantage of this approach is that we retain the efficiency and resolution of the original algorithm. (Since no turbulence model is included, we must sufficiently resolve the boundary layer for a flow whose Reynolds number approaches 10^5, a reasonable value for enclosure fires.) The disadvantage of the conformal transformation is that it can lead to severe variations in grid cell size, and thus significantly limit the time-step

1

of our explicit scheme. For this reason, we consider domains which do not yield too large a variation in the Jacobian. We have chosen for the moment to retain the machinery built up from our previous work and implement the conformal transformation without changing the basic algorithm. However, for domains which exhibit a larger variation in the Jacobian (more than 2 or 3 orders of magnitude), a better approach would be to use an implicit, unconditionally stable time-stepping scheme [8].

The model of buoyant convection is presented in Sections 2 and 3 and a brief description of the numerical method in Section 4. Section 5 contains a detailed description of the algorithm. Section 6 presents the results of a few computations, including a verification of the "trench effect" [2] in a more realistic stairway geometry. An Appendix is also included which contains the algorithm used to introduce and track the Lagrangian particles, and also a linear stability analysis of the lagged-dissipation scheme.

2 Hydrodynamic Model

We consider a Boussinesq fluid with constant viscosity and thermal conductivity in a polygonal enclosure driven by a prescribed heat source. We start with the equations of motion for a thermally expandable ideal gas [3] in which we include viscous dissipation and thermal conductivity.

$$
\begin{aligned}
\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_i}(\rho u_i) &= 0 \\
\rho\left(\frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j}\right) + \frac{\partial p}{\partial x_i} - \rho k_i g &= \rho \nu \frac{\partial^2 u_i}{\partial x_j^2} \\
\rho C_p \left(\frac{\partial T}{\partial t} + u_j \frac{\partial T}{\partial x_j}\right) &= Q + K \frac{\partial^2 T}{\partial x_j^2} \\
p_0 &= \rho R T
\end{aligned}
\tag{1}
$$

Here, all symbols have their usual fluid dynamical meaning: ρ is density, u_i are the velocity components, p is pressure, g is the acceleration of gravity, k_i are the components of the vector describing the direction of the gravity vector, ν is kinematic viscosity, C_p is the constant-pressure specific heat, T is temperature and K is the thermal conductivity, t is time and Q is the spatially and temporally prescribed heat source.

If we combine these equations as described in [4], nondimensionalize them as described in the appendix of this reference and make the Boussinesq approximation, we obtain the following set of equations,

$$
\begin{aligned}
\frac{\partial \rho}{\partial t} + u_i \frac{\partial \rho}{\partial x_i} &= -\frac{\gamma - 1}{\gamma}Q + \kappa \nabla^2 \rho \\
\frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j} + \frac{\partial p}{\partial x_i} - (\rho - \rho_0(z))k_i &= \nu \nabla^2 u_i \\
\frac{\partial u_i}{\partial x_i} &= 0
\end{aligned}
\tag{2}
$$

Here, all symbols have the meanings given above but in dimensionless form, $\rho_0(z)$ is the initially stratified density profile assumed to depend only upon the vertical coordinate z, and κ is the dimensionless thermal conductivity. The dimensionless quantities are defined as follows: lengths are relative to the height of the enclosure, pressure is relative to ambient pressure, time is relative to the height of the enclosure and the velocity scale, and velocity is relative to a scale U, and the density perturbation is relative to ambient density with a small parameter β. If we denote temporarily the dimensional quantities by an asterisk, then these relationships can be written as follows:

$$
\begin{aligned}
x_i^* &= Hx_i, & p^* &= \rho_0(0)U^2 p \\
t^* &= Ht/U, & \rho^* &= \rho_0(0)(1 + \beta\rho) \\
u_i^* &= Uu_i, & \rho_0^*(z^*) &= \rho_0(0)(1 + \beta\rho_0(z))
\end{aligned}
$$

U and β are defined in terms of the magnitude of the heat source as follows:

$$
\begin{aligned}
\beta &= U^2/(gH) \\
U &= (Q_0 g/(\rho_0 C_p T_0 H))^{1/3} = (q_0 g/(\rho_0 C_p T_0))^{1/3}
\end{aligned}
$$

Here Q_0 is the strength of the three-dimensional heat source in units of energy per unit time, q_0 is the strength of the two-dimensional heat source in energy per unit length per unit time, H is the height of the enclosure and all other quantities have their usual meanings. See [4] and [7] for more details of the scaling. Henceforth, all variables will be regarded as dimensionless.

In the two-dimensional case, these equations can be rewritten as follows:

$$
\begin{aligned}
\frac{\partial\tilde\rho}{\partial t} + u\frac{\partial\tilde\rho}{\partial x} + w\frac{\partial(\rho_0 + \tilde\rho)}{\partial z} &= -\frac{\gamma - 1}{\gamma}Q + \kappa\left(\frac{\partial^2\tilde\rho}{\partial x^2} + \frac{\partial^2(\rho_0 + \tilde\rho)}{\partial z^2}\right) \\
\frac{\partial u}{\partial t} - w\omega + \frac{\partial\tilde p}{\partial x} - \tilde\rho g_x &= \nu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial z^2}\right) \\
\frac{\partial w}{\partial t} + u\omega + \frac{\partial\tilde p}{\partial z} - \tilde\rho g_z &= \nu\left(\frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial z^2}\right) \\
\frac{\partial u}{\partial x} + \frac{\partial w}{\partial z} &= 0
\end{aligned}
\tag{3}
$$

where

$$
\begin{aligned}
\tilde\rho &= \rho - \rho_0(z) \\
\tilde p &= p + q^2/2 \\
\omega &= \frac{\partial w}{\partial x} - \frac{\partial u}{\partial z} \\
q^2 &= u^2 + w^2
\end{aligned}
$$

Note that gravity need not be restricted to the z-direction (downward), but may point in any direction according to its components g_x and g_z. The norm of the gravity vector is unity. A no-flux condition is imposed at the boundary. In addition, one may choose a no-slip or a free-slip condition, as well as an adiabatic or constant temperature condition at the wall. The fluid is initially quiescent.

3

3 The Transformation

The physical domain may be taken as any simply connected polygon whose vertices may or may not extend to infinity. This polygon is mapped conformally into a rectangle whose aspect ratio is dependent on the shape of the polygon. Details of the Schwarz-Christoffel transformation may be found in [9] and a sample mesh is presented in Fig. 1. The equations above are written in terms of the physical coordinates x and z. We shall take the coordinates of the computational rectangle to be ξ and η. Eqs. (3) may then be transformed as follows:

$$
\begin{aligned}
J\frac{\partial \tilde{\rho}}{\partial t} + \hat{u}\frac{\partial(\rho_0 + \tilde{\rho})}{\partial \xi} + \hat{w}\frac{\partial(\rho_0 + \tilde{\rho})}{\partial \eta} &= -\frac{\gamma - 1}{\gamma}JQ + \kappa\left(\frac{\partial^2(\rho_0 + \tilde{\rho})}{\partial \xi^2} + \frac{\partial^2(\rho_0 + \tilde{\rho})}{\partial \eta^2}\right) \\
\frac{\partial \hat{u}}{\partial t} - \hat{w}\,\omega + \frac{\partial \tilde{p}}{\partial \xi} - \tilde{\rho}g_\xi &= -\nu\frac{\partial \omega}{\partial \eta} \\
\frac{\partial \hat{w}}{\partial t} + \hat{u}\,\omega + \frac{\partial \tilde{p}}{\partial \eta} - \tilde{\rho}g_\eta &= \nu\frac{\partial \omega}{\partial \xi} \\
\frac{1}{J}\left(\frac{\partial \hat{u}}{\partial \xi} + \frac{\partial \hat{w}}{\partial \eta}\right) &= 0
\end{aligned}
\tag{4}
$$

where

$$
\begin{aligned}
\tilde{p} &= p + q^2/2 \\
\hat{u} &= \frac{\partial x}{\partial \xi}u + \frac{\partial z}{\partial \xi}w \\
\hat{w} &= \frac{\partial x}{\partial \eta}u + \frac{\partial z}{\partial \eta}w \\
\omega &= \frac{1}{J}\left(\frac{\partial \hat{w}}{\partial \xi} - \frac{\partial \hat{u}}{\partial \eta}\right) \\
g_\xi &= \frac{\partial x}{\partial \xi}g_x + \frac{\partial z}{\partial \xi}g_z \\
g_\eta &= \frac{\partial x}{\partial \eta}g_x + \frac{\partial z}{\partial \eta}g_z \\
J &= \frac{\partial x}{\partial \xi}\frac{\partial z}{\partial \eta} - \frac{\partial x}{\partial \eta}\frac{\partial z}{\partial \xi}
\end{aligned}
$$

In transforming the viscous terms of the momentum equations, we have made use of the vector identity

$$
\nabla^2 \mathbf{u} = \nabla(\nabla \cdot \mathbf{u}) - \nabla \times (\nabla \times \mathbf{u})
$$

and the fact that $\nabla \cdot \mathbf{u} = 0$. Because the coordinate transformation is conformal, the boundary conditions for Eqs. (3) can be implemented in the computational domain in exactly the same way in which they would be for a rectangular enclosure.

4 Numerical Methods

Eqs. (4) are a mixed parabolic/elliptic system of partial differential equations; *i.e.*, the equations for the density and for the velocity components are parabolic, whereas that for the pressure is elliptic. The incompressible equations of hydrodynamics are well known to have this mixed character. Analytical studies of the ability of several candidate finite difference schemes to calculate internal gravity waves without dissipation [5] led to the conclusion that methods of second order accuracy in space and time are highly desirable for systems of this type.

We have chosen a lagged-dissipation scheme — all time derivatives are replaced by central differences over twice the time-step size (a leap-frog scheme) while all viscous and conduction terms are differenced at the lagged time level. Other terms in the evolution equations (the first three of Eqs. (4)) are evaluated at the mid level of the three level scheme. This scheme is not formally second-order accurate in time. However, for the Reynolds and Prandtl numbers capable of capturing large scale features in typical enclosure fire simulations ($Re \geq 10^4$), the dissipation terms are very small, and the formal first-order accuracy due to dissipation does not pose a practical limitation. The grid is taken to be uniform in the ξ and η directions, although the spacings $\delta\xi$ and $\delta\eta$ may be different. Within each rectangular cell, vector components are evaluated at the sides, scalar quantities at the center.

The density evolution equation in continuous form is the mass conservation equation minus the expression for the velocity divergence. Each of these two equations is approximated by central differences and then subtracted. The density at all faces is approximated by the mean of the density at the centers of adjacent cells. This procedure ensures global mass conservation as well as second order accuracy.

The momentum equations are differenced in the vector invariant form shown in Eqs. (4). This ensures nonlinear stability and complete compatibility between the "primitive variable" formulation presented here and a vorticity, stream-function formulation (in the dissipation-free case when the Jacobian is unity), see [6] for details.

The pressure equation is derived by differencing in time the centrally-differenced incompressibility equation, and then making use of the discretized momentum equations to eliminate the time differences. The result is an elliptic partial differential equation for the pressure. For the Boussinesq model, the linear algebraic system arising from its discretization has constant coefficients and can be solved by a fast direct method, see [7] for details. The solution to the pressure equation constitutes the bulk of the numerical computation since the density and the velocity are updated explicitly once the pressure gradients are known.

Stability of the computational scheme imposes a limit on the time-step size relative to the spatial mesh size, see [4], [5] and Appendix B. This may be extremely limiting due to the grid distortion brought about by the conformal mapping, whose Jacobian may vary up to two orders of magnitude. For geometries which are even more distorting than those we present here, another method of grid generation would probably have to be implemented, which would unfortunately reduce the efficiency of the present algorithm.

5 The Algorithm

5.1 The Density Equation

We now write out the details of the algorithm. For the density equation, return to the full continuity equation (in dimensional form):

$$J\frac{\partial \rho}{\partial t} + \frac{\partial \rho \hat{u}_j}{\partial \xi_j} = 0$$

Use the nondimensionalization presented earlier

$$J\frac{\partial \tilde{\rho}}{\partial t} + \hat{u}_j \frac{\partial(\rho_0 + \tilde{\rho})}{\partial \xi_j} + [1 + \beta(\rho_0 + \tilde{\rho})]\frac{\partial \hat{u}_j}{\partial \xi_j} = 0$$

where $\rho = 1 + \beta(\rho_0 + \tilde{\rho})$, where all variables are dimensionless, and where we have divided through by $\beta U/H$. If we now formally allow $\beta \to 0$, then we recover the Boussinesq equation for the density:

$$J\frac{\partial \tilde{\rho}}{\partial t} + \hat{u}_j \frac{\partial(\rho_0 + \tilde{\rho})}{\partial \xi_j} + \frac{\partial \hat{u}_j}{\partial \xi_j} = 0$$

The same procedure used above for the partial differential equations (PDE) can now be applied to the finite difference equations (FDE). We desire to keep conservation form for the FDE as well as for the PDE. Hence,

$$J_{\rho,ik}\frac{\partial \rho_{ik}}{\partial t} \quad + \quad \left[\frac{\rho_{i+1,k} + \rho_{ik}}{2}\hat{u}_{ik} - \frac{\rho_{ik} + \rho_{i-1,k}}{2}\hat{u}_{i-1,k}\right]\frac{1}{\delta\xi}$$

$$+ \quad \left[\frac{\rho_{i,k+1} + \rho_{ik}}{2}\hat{w}_{ik} - \frac{\rho_{ik} + \rho_{i,k-1}}{2}\hat{w}_{i,k-1}\right]\frac{1}{\delta\eta} = 0$$

or, rewriting,

$$J_{\rho,ik}\frac{\partial \rho_{ik}}{\partial t} \quad + \quad \frac{\rho_{i+1,k}\hat{u}_{ik}}{2\delta\xi} - \frac{\rho_{i-1,k}\hat{u}_{i-1,k}}{2\delta\xi} + \frac{\rho_{ik}}{2\delta\xi}(\hat{u}_{ik} - \hat{u}_{i-1,k})$$

$$+ \quad \frac{\rho_{i,k+1}\hat{w}_{ik}}{2\delta\eta} - \frac{\rho_{i,k-1}\hat{w}_{i,k-1}}{2\delta\eta} + \frac{\rho_{ik}}{2\delta\eta}(\hat{w}_{ik} - \hat{w}_{i,k-1}) = 0$$

Note that the Jacobian subscript ρ indicates that J is to be evaluated where ρ is, namely at the center of a grid cell. This convention will be followed from here on.

Now we subtract from the density equation the expression for the velocity divergence and obtain

$$J_{\rho,ik}\frac{\partial \rho_{ik}}{\partial t} + F_{\rho\xi,ik} + F_{\rho\eta,ik} + (1/2)\rho_{ik}\,\Delta_{ik} = 0$$

where

$$\Delta_{ik} = \left(\frac{\partial \hat{u}_j}{\partial \xi_j}\right)_{ik} = \left[\frac{\gamma - 1}{\gamma p_0}\left(JQ - \frac{p_0 K}{R\rho^2}\nabla^2\rho\right)\right]_{ik}$$

$$\nabla^2\rho = \frac{\rho_{i+1,k} - 2\rho_{ik} + \rho_{i-1,k}}{\delta\xi^2} + \frac{\rho_{i,k+1} - 2\rho_{ik} + \rho_{i,k-1}}{\delta\eta^2}$$

6

$$F_{\rho\xi,ik} = \frac{1}{2\delta\xi}\left(\rho_{i+1,k}\hat{u}_{ik} - \rho_{i-1,k}\hat{u}_{i-1,k}\right)$$

$$F_{\rho\eta,ik} = \frac{1}{2\delta\eta}\left(\rho_{i,k+1}\hat{w}_{ik} - \rho_{i,k-1}\hat{w}_{i,k-1}\right)$$

Finally, use the same nondimensionalization as cited above, using $\rho_{ik} = 1 + \beta(\rho_{0,ik} + \tilde{\rho}_{ik})$. Note that only $\tilde{\rho}_{ik}$ depends upon time. Also, note that we can divide by β and then allow it formally to vanish as done above for the PDE to obtain the finite difference equation for the Boussinesq model.

$$J_{\rho,ik}\frac{\partial\tilde{\rho}_{ik}}{\partial t} + \tilde{F}_{\rho\xi,ik} + \tilde{F}_{\rho\eta,ik} + D_{ik} = 0$$

where

$$\tilde{F}_{\rho\xi,ik} = \frac{1}{2\delta\xi}\left[(\rho_{0,i+1,k} + \tilde{\rho}_{i+1,k})\hat{u}_{ik} - (\rho_{0,i-1,k} + \tilde{\rho}_{i-1,k})\hat{u}_{i-1,k}\right]$$

$$\tilde{F}_{\rho\eta,ik} = \frac{1}{2\delta\eta}\left[(\rho_{0,i,k+1} + \tilde{\rho}_{i,k+1})\hat{w}_{ik} - (\rho_{0,i,k-1} + \tilde{\rho}_{i,k-1})\hat{w}_{i,k-1}\right]$$

and where D_{ik} is the dimensionless form of Δ_{ik} divided by β,

$$D_{ik} = \left[\frac{\gamma-1}{\gamma}\tilde{Q} - \kappa\nabla^2(\rho_0 + \tilde{\rho})\right]_{ik}$$

Here $\kappa = K/(\rho C_p)$. $\tilde{Q}$ will be described below.

At boundaries, the density fluxes are determined by the no-inflow, no-outflow conditions. Also, we must specify adiabatic or cold-wall boundary conditions, which determine the temperature and hence the density (since the perturbation density is the negative of the perturbation temperature) in ficticious cells adjacent to the boundaries.
At $i = 1$:

$$\tilde{F}_{\rho\xi,1k} = \frac{1}{2\delta\xi}\left(\rho_{0,2,k} + \tilde{\rho}_{2,k}\right)\hat{u}_{1,k}$$

$$\tilde{\rho}_{0,k} = \pm\tilde{\rho}_{1,k}$$

At $i = I$:

$$\tilde{F}_{\rho\xi,Ik} = -\frac{1}{2\delta\xi}\left(\rho_{0,I-1,k} + \tilde{\rho}_{I-1,k}\right)\hat{u}_{I-1,k}$$

$$\tilde{\rho}_{I+1,k} = \pm\tilde{\rho}_{I,k}$$

At $k = 1$:

$$\tilde{F}_{\rho\eta,i1} = \frac{1}{2\delta\eta}\left(\rho_{0,i,2} + \tilde{\rho}_{i,2}\right)\hat{w}_{i,1}$$

$$\tilde{\rho}_{i,0} = \pm\tilde{\rho}_{i,1}$$

At $k = K$:

$$\tilde{F}_{\rho\eta,iK} = -\frac{1}{2\delta\eta}\left(\rho_{0,i,K-1} + \tilde{\rho}_{i,K-1}\right)\hat{w}_{i,K-1}$$

$$\tilde{\rho}_{i,K+1} = \pm\tilde{\rho}_{i,K}$$

7

Here, the plus sign corresponds to adiabatic boundary conditions (BC) and the minus sign to cold wall BC. Adiabatic BC imply zero derivative in the temperature (density) perturbation across the boundary whereas cold wall BC imply the temperature (density) perturbation is zero at the boundary.

The discretization of the time derivative has been chosen as a leap frog. For the second-order time steps

$$\tilde{\rho}_{ik}^{n+1} = \tilde{\rho}_{ik}^{n-1} + 2\delta t R_{\rho,ik}^n / J_{\rho,ik}$$

and for the first-order time steps

$$\tilde{\rho}_{ik}^{n+1} = \tilde{\rho}_{ik}^n + \delta t \tilde{R}_{\rho,ik}^n / J_{\rho,ik}$$

where

$$R_{\rho,ik}^n = -\tilde{F}_{\rho\xi,ik}^n - \tilde{F}_{\rho\eta,ik}^n - \frac{\gamma - 1}{\gamma} \tilde{Q}_{ik}^n + \kappa \left((\nabla^2 \rho_0)_{ik} + (\nabla^2 \tilde{\rho})_{ik}^{n-1} \right)$$

$$\tilde{R}_{\rho,ik}^n = -\tilde{F}_{\rho\xi,ik}^n - \tilde{F}_{\rho\eta,ik}^n - \frac{\gamma - 1}{\gamma} \tilde{Q}_{ik}^n + \kappa \left((\nabla^2 \rho_0)_{ik} + (\nabla^2 \tilde{\rho})_{ik}^n \right)$$

The Laplacian operator ∇^2 has been defined above. Notice that we use a lagged-diffusion temporal discretization for the dissipation on the second-order time steps only.

It is more convenient to define the heat source in terms of the computational coordinates ξ and η, even though it would be preferable to use the actual physical coordinates x and z. The trouble with the latter approach is that it is difficult to interpolate a particle's position in the physical plane, because the mesh is obviously not uniform there. Therefore in these calculations, the following forms for the heat source have been used:

$$\tilde{Q}_{ik}^n = f^n \, J_{\rho,ik} \, Q_{\xi,i} Q_{\eta,k}$$

where f^n is either

$$f^n = \tanh(\alpha t_n)$$

or

$$f^n = \begin{cases} \sin(\alpha t_n) & 0 \le t_n \le \pi/\alpha \\ 0 & \text{otherwise} \end{cases}$$

The latter function represents a buoyant thermal. The spatial factors are

$$Q_{\xi,i} = \sqrt{\beta_\xi/\pi} \exp[-\beta_\xi(\xi_i - \xi_c)^2]$$
$$Q_{\eta,k} = \sqrt{\beta_\eta/\pi} \exp[-\beta_\eta(\eta_k - \eta_c)^2]$$

or

$$Q_{\eta,k} = \beta_\eta \exp[-\beta_\eta \eta_k]$$

8

5.2 The Momentum Equations

For the velocity equations, we use the following difference scheme:

$$\frac{\partial \hat{u}_{ik}}{\partial t} - \frac{\hat{w}_{\omega,ik}\,\omega_{ik} + \hat{w}_{\omega,i,k-1}\,\omega_{i,k-1}}{2} + \frac{\tilde{p}_{i+1,k} - \tilde{p}_{ik}}{\delta\xi} - g_{\xi,ik}\frac{\tilde{\rho}_{i+1,k} + \tilde{\rho}_{ik}}{2} \;=\; -\nu\frac{\omega_{ik} - \omega_{i,k-1}}{\delta\eta}$$

$$\frac{\partial \hat{w}_{ik}}{\partial t} + \frac{\hat{u}_{\omega,ik}\,\omega_{ik} + \hat{u}_{\omega,i-1,k}\,\omega_{i-1,k}}{2} + \frac{\tilde{p}_{i,k+1} - \tilde{p}_{ik}}{\delta\eta} - g_{\eta,ik}\frac{\tilde{\rho}_{i,k+1} + \tilde{\rho}_{ik}}{2} \;=\; \nu\frac{\omega_{ik} - \omega_{i-1,k}}{\delta\xi}$$

where

$$\omega_{ik} \;=\; \frac{1}{J_{\omega,ik}}\left[\frac{\hat{w}_{i+1,k} - \hat{w}_{ik}}{\delta\xi} - \frac{\hat{u}_{i,k+1} - \hat{u}_{ik}}{\delta\eta}\right]$$

$$\hat{u}_{\omega,ik} \;=\; \frac{\hat{u}_{ik} + \hat{u}_{i,k+1}}{2}$$

$$\hat{w}_{\omega,ik} \;=\; \frac{\hat{w}_{ik} + \hat{w}_{i+1,k}}{2}$$

$$g_{\xi,ik} \;=\; \frac{g_x}{2}\left(\frac{\partial x}{\partial \xi}_{ik} + \frac{\partial x}{\partial \xi}_{i,k-1}\right) + \frac{g_z}{2}\left(\frac{\partial z}{\partial \xi}_{ik} + \frac{\partial z}{\partial \xi}_{i,k-1}\right)$$

$$g_{\eta,ik} \;=\; \frac{g_x}{2}\left(\frac{\partial x}{\partial \eta}_{ik} + \frac{\partial x}{\partial \eta}_{i-1,k}\right) + \frac{g_z}{2}\left(\frac{\partial z}{\partial \eta}_{ik} + \frac{\partial z}{\partial \eta}_{i-1,k}\right)$$

The derivatives $\partial x_l / \partial \xi_m$ are evaluated at the corners, thus $g_{\xi,ik}$ is evaluated wherever $\hat{u}_{ik}$ is, and $g_{\eta,ik}$ is evaluated wherever $\hat{w}_{ik}$ is. To improve the computational efficiency, we set

$$G_{\xi,ik} \;\equiv\; -\frac{\hat{w}_{\omega,ik}\,\omega_{ik} + \hat{w}_{\omega,i,k-1}\,\omega_{i,k-1}}{2} - g_{\xi,ik}\frac{\tilde{\rho}_{i+1,k} + \tilde{\rho}_{ik}}{2} + \nu\frac{\omega_{ik} - \omega_{i,k-1}}{\delta\eta}$$

$$G_{\eta,ik} \;\equiv\; \frac{\hat{u}_{\omega,ik}\,\omega_{ik} + \hat{u}_{\omega,i-1,k}\,\omega_{i-1,k}}{2} - g_{\eta,ik}\frac{\tilde{\rho}_{i,k+1} + \tilde{\rho}_{ik}}{2} - \nu\frac{\omega_{ik} - \omega_{i-1,k}}{\delta\xi}$$

This notation will be of use in the pressure equation.

In order to compute these terms at the boundary we must either apply free-slip or no-slip boundary conditions. For $0 \le k \le K$,

$$\hat{w}_{0,k}^n \;=\; \pm\hat{w}_{1,k}^n$$

$$\hat{w}_{I,k}^n \;=\; \pm\hat{w}_{I-1,k}^n$$

and for $0 \le i \le I$

$$\hat{u}_{i,0}^n \;=\; \pm\hat{u}_{i,1}^n$$

$$\hat{u}_{i,K}^n \;=\; \pm\hat{u}_{i,K-1}^n$$

where the plus sign corresponds to free-slip and the minus sign to no-slip BC.

The temporal derivatives are handled by a leap-frog discretization. For the second-order time steps using the lagged-dissipation scheme,

$$\hat{u}_{ik}^{n+1} \;=\; \hat{u}_{ik}^{n-1} - 2\delta t\left(G_{\xi,ik}^n + \frac{p_{i+1,k}^n - p_{ik}^n}{\delta\xi}\right)$$

$$\hat{w}_{ik}^{n+1} \;=\; \hat{w}_{ik}^{n-1} - 2\delta t\left(G_{\eta,ik}^n + \frac{p_{i,k+1}^n - p_{ik}^n}{\delta\eta}\right)$$

and for the first-order time steps,

$$\hat{u}_{ik}^{n+1} = \hat{u}_{ik}^n - \delta t \left(\tilde{G}_{\xi,ik}^n + \frac{p_{i+1,k}^n - p_{ik}^n}{\delta \xi} \right)$$

$$\hat{w}_{ik}^{n+1} = \hat{w}_{ik}^n - \delta t \left(\tilde{G}_{\eta,ik}^n + \frac{p_{i,k+1}^n - p_{ik}^n}{\delta \eta} \right)$$

Here again, G and $\tilde{G}$ are the same except that the former has a lagged dissipation term, while the latter does not.

5.3 The Pressure Equation

The incompressibility condition is

$$\frac{1}{J_{\rho,ik}} \left[\frac{\hat{u}_{ik} - \hat{u}_{i-1,k}}{\delta \xi} + \frac{\hat{w}_{ik} - \hat{w}_{i,k-1}}{\delta \eta} \right] = 0$$

and this condition can be used to derive the linear algebraic equation system for the pressure. Away from boundaries, that is for $2 \le i \le I - 1, 2 \le k \le K - 1$, we have

$$\frac{\tilde{p}_{i+1,k} - 2\tilde{p}_{ik} + \tilde{p}_{i-1,k}}{\delta \xi^2} + \frac{\tilde{p}_{i,k+1} - 2\tilde{p}_{ik} + \tilde{p}_{i,k-1}}{\delta \eta^2} = -\frac{G_{\xi,ik} - G_{\xi,i-1,k}}{\delta \xi} - \frac{G_{\eta,ik} - G_{\eta,i,k-1}}{\delta \eta} \quad (5)$$

The lack of a superscript implies that all quantities are to be evaluated at time level n.

For cells adjacent to the left boundary, we have $\hat{u}_{0,k}^n = 0$ for all n, thus

$$\frac{\tilde{p}_{1k} - \tilde{p}_{0k}}{\delta \xi} = -G_{\xi,0k}$$

Subtracting this equation from the general equation for pressure with $i = 1$ yields the equation

$$\frac{\tilde{p}_{2,k} - \tilde{p}_{1,k}}{\delta \xi^2} + \frac{\tilde{p}_{1,k+1} - 2\tilde{p}_{1,k} + \tilde{p}_{1,k-1}}{\delta \eta^2} = -\frac{G_{\xi,1k}}{\delta \xi} - \frac{G_{\eta,1k} - G_{\eta,1,k-1}}{\delta \eta}$$

In a similar fashion, the above equation can be derived for the three other edges and the four corners. To improve the computational efficiency of the code, we set $G_{\xi,0k} = G_{\xi,Ik} = 0$ for $1 \le k \le K$, and we set $G_{\eta,i0} = G_{\eta,iK} = 0$ for $1 \le i \le I$. Then the index range of Eq. 5 may be extended to the boundary cells, and we solve the equivalent Poisson equation in which the normal derivative of the pressure is specified to be zero at the boundary.

6 Results

In this section samples of results generated using a code implementing the algorithm described above are presented and discussed. The results represent some problems of interest to scientists concerned with enclosure fires. The computations were performed on the IBM RISC System/6000 Model 550 in the Computational Combustion Facility, which is a joint facility of the Computing and Applied Mathematics Laboratory and Building and Fire Research Laboratory at NIST. The graphics were generated on a Silicon Graphics Personal Iris 4D20, which is also part of this facility.

The scale of these computations is substantial. Runs have now been performed using more than a quarter of a million grid points; they require at least 50 megabytes of memory and take up to 24 hours of CPU time. They are performed in 64 bit arithmetic. Typically, the CPU time per node per time-step is on the order of 2.0×10^{-5} seconds. If we were to save any major fraction of the data generated, we would be overwhelmed. Therefore, a significant effort has been expended in trying to select only those data required to understand the phenomena being studied. We have concluded that Lagrangian particle tracking for transient phenomena is the most convenient method for saving and examining data from our computations, and this procedure is described in [2].

The resolution of the computation determines the maximum Reynolds number of the flow. Since the size of the Reynolds number for the 2-D flow scales as K^2, flows with Reynolds numbers of nearly one million can in principle be computed. We have not yet computed flows with this large a Reynolds number, primarily because we have been examining non-rectangular, polygonal geometries. For transformations such as the stairway, we have computed results for flows with Reynolds numbers of up to 10^5, which is reasonably close to full scale for many fire scenarios. A big constraint in these computations is the time-step size, which is constrained by the CFL condition of the cells whose areas are extremely small. Fig. 1 displays a crude version of the mesh used for a simulation of smoke through a window. The Jacobian, which is a measure of the area of an individual cell, varies from about 10^{-2} to 10^1 for this geometry. Because the time-step size is proportional to the square root of the Jacobian (see Appendix B), the calculation is slowed considerably by the small cells. For this reason, the present methodology is limited to enclosures whose cell areas vary no more than those shown in Fig. 1. An alternative approach for difficult geometries would be to use an implicit time-stepping scheme.

6.1 The "Trench Effect"

Fires in buildings involve the transport of heat and mass by gravity-induced or buoyant convection. Generally, this convection occurs in rectangular enclosures where the direction of gravity is parallel to the surfaces of the enclosure, the walls. However, under certain circumstances, such as a fire in a stairwell or an escalator, the enclosure may be sloped relative to gravity. A very important example of a fire in a sloped enclosure was the devastating fire in the King's Cross underground station in England in 1987, where there was significant loss of life as well as property damage. Numerical simulation of this fire uncovered an unexpected phenomenon which caused a very rapid spread of the fire and led to much of the devastation [10], [11]. This phenomenon was termed the "trench effect", and caused some

controversy during investigations of the King's Cross fire in England. The phenomenon was ultimately confirmed by experiments and additional simulation, but transient aspects of the fire simulation are still of interest.

We repeat here an experiment performed with a rectangular domain tilted 35 degrees from the horizontal which was intended to model a stairwell [12]. The major difference in the present calculation is the addition of more realistic "landings" to the geometry. Fig. 2 presents a time sequence for the flow generated in an enclosure by a heat source (fire) located near the base of the stairwell. The mesh size is 1024×256 (262,144 grid cells), the Reynolds number based on the height of the landings is 4.0×10^4, and adiabatic, free-slip boundary conditions are imposed. The average time-step size is 8.6×10^{-4} dimensionless time units. The CPU time per node per time-step is 2.2×10^{-5} seconds. In this figure, the plume rises, but is bent back toward the lower landing. After the hot gases hit the ceiling, they progress both toward the back wall and up the ceiling toward the high end. However, the hot gases leaving the heat source are pinned along the floor and form a hot gas jet which progresses up along the floor, shedding hot gases near its front. This phenomenon we interpret as the "trench effect".

6.2   Smoke through a window

In Fig. 3, we present the flow induced by a fire in a small enclosure with a window. The mesh size is 256×128 (32,768 grid cells), the Reynolds number based on the height of the small room is 10^4, and adiabatic, no-slip boundary conditions are imposed. This geometry pushes the present capabilities to the limit. The average time-step size for this run is 7.7×10^{-4}. This is due to the extremely small cells around the window soffits. The CPU time per node per time-step is 1.6×10^{-5} seconds.

As before we notice that the vorticity of the escaping plume tends to pin the hot gas to the wall of the enclosure, a phenomenon which has been seen in actual fires where the flames and hot gases pour out of an enclosure opening and then spread rapidly up the side of the enclosure. In Fig. 4, we freeze in time four calculations with different boundary conditions. There is a dramatic difference between the free-slip and the no-slip cases. Ultimately, all demonstrate the above described phenomena, however, the free-slip cases allow for a faster escape of gas from the small room. This type of behavior has been seen in the case of a stairwell [12], where the "trench effect" of Fig. 2 is not as dramatic when no-slip boundary conditions are employed.

6.3   Corridor with soffits

Finally, in Fig. 5 we present the movement of smoke down a corridor with uniformly spaced soffits. The mesh size is 512×128, the Reynolds number is 1.0×10^4, and adiabatic, no-slip boundary conditions are imposed. This example points out two difficulties we have experienced with the method described above. First, the computation was slowed considerably by the small grid cells clustered around the soffits. This particular run required about 18 hours of CPU time; a similar run without the soffits required about a tenth of the time. Second, we observed that the calculation became unstable near the end of the run. Fig. 6 displays the perturbation density of the boundary cells, with the enclosure boundary serving

as the spacial axis. Where the curve falls *inside* the enclosure, the perturbation density is negative and thus the temperature is greater than its original value. Where the curve falls *outside* the enclosure, the temperature is less than its original value. Obviously, the density profile at time $t = 18$ reflects the instability. We suspect that the time-stepping scheme is responsible, although the linear stability analysis fails to identify the problem. Initially, it appeared as if the boundary layer was not being fully resolved, but we have noticed the instability for Reynolds numbers well below the theoretical limits. Also, the instability did not develop until tens of thousands of time steps had been completed, leading us to suspect some non-linear effect of the time-stepping scheme. We applied a second order Runge-Kutta scheme to replace the lagged-dissipation scheme, and the instability did not reoccur. The Runge-Kutta scheme, however, requires two calls to the pressure solver per time step instead of one, which nearly doubles the computing time. This is the price required to remove the long-time instability.

A   Appendix: Particle Tracking Algorithm

A.1   Particle Injection

The initial location of each particle is selected at random from a distribution function which is normal in the ξ-direction with mean value equal to the location of the center of the heat source and variance equal to the mean width of the heat source. In the η-direction the distribution function is either normal with characteristics similar to that in the ξ-direction or exponential. In all cases, the distribution functions mimic the spatial distribution of the heat source. Also, particles are injected at a rate that follows the rate of heat addition.

A.2   Interpolation of the Fields at Particles Locations

Take the 2-D case and let ξ_j^n and η_j^n be the coodinates of particle $j, 1 \leq j \leq N_p$ at time t^n. Let

$$\begin{aligned}
\mathcal{I}_j &\equiv [\xi_j^n/\delta\xi + 1/2] \\
\mathcal{K}_j &\equiv [\eta_j^n/\delta\eta + 1/2] \\
r_j &\equiv \xi_j^n/\delta\xi - (\mathcal{I}_j - 1/2) \\
s_j &\equiv \eta_j^n/\delta\eta - (\mathcal{K}_j - 1/2)
\end{aligned}$$

and let

$$\begin{aligned}
\mathcal{I}_j' &\equiv [\xi_j^n/\delta\xi] \\
\mathcal{K}_j' &\equiv [\eta_j^n/\delta\eta] \\
r_j' &\equiv \xi_j^n/\delta\xi - \mathcal{I}_j \\
s_j' &\equiv \eta_j^n/\delta\eta - \mathcal{K}_j
\end{aligned}$$

where [...] is the integer part of the quantity in square brackets.

Define the function F as follows:

$$F(A_{1,1}, A_{1,2}, A_{2,1}, A_{2,2}, r, s) \equiv A_{1,1}(1-r)(1-s) + A_{2,1}r(1-s) + A_{1,2}(1-r)s + A_{2,2}rs$$

and let $\bar{u}_j, \bar{w}_j, \bar{\rho}_j$ be the velocity components and the density respectively at the location of the jth particle. Then by bilinear interpolation:

$$\frac{\bar{u}_j^n}{J_u} = F\left(\frac{\hat{u}_{\mathcal{I}_j'+1,\mathcal{K}_j+1}^n}{J_u}, \frac{\hat{u}_{\mathcal{I}_j'+2,\mathcal{K}_j+1}^n}{J_u}, \frac{\hat{u}_{\mathcal{I}_j'+1,\mathcal{K}_j+2}^n}{J_u}, \frac{\hat{u}_{\mathcal{I}_j'+2,\mathcal{K}_j+2}^n}{J_u}, r', s\right)$$

$$\frac{\bar{w}_j^n}{J_w} = F\left(\frac{\hat{w}_{\mathcal{I}_j+1,\mathcal{K}_j'+1}^n}{J_w}, \frac{\hat{w}_{\mathcal{I}_j+2,\mathcal{K}_j'+1}^n}{J_w}, \frac{\hat{w}_{\mathcal{I}_j+1,\mathcal{K}_j'+2}^n}{J_w}, \frac{\hat{w}_{\mathcal{I}_j+2,\mathcal{K}_j'+2}^n}{J_w}, r, s'\right)$$

$$\bar{\rho}_j^n = F(\tilde{\rho}_{\mathcal{I}_j'+1,\mathcal{K}_j'+1}^n, \tilde{\rho}_{\mathcal{I}_j'+2,\mathcal{K}_j'+1}^n, \tilde{\rho}_{\mathcal{I}_j'+1,\mathcal{K}_j'+2}^n, \tilde{\rho}_{\mathcal{I}_j'+2,\mathcal{K}_j'+2}^n, r', s')$$

A.3   Time Advancement Scheme for Particles

Now, each particle is advanced through the differential equation:

$$\frac{d\xi_j}{dt} = \frac{\hat{u}_j(\xi_j, \eta_j)}{J}$$

$$\frac{d\eta_j}{dt} = \frac{\hat{w}_j(\xi_j, \eta_j)}{J}$$

This corresponds to the differential equation describing the movement of the particles in the physical domain:

$$\frac{dx_j}{dt} = u_j(x_j, z_j)$$

$$\frac{dz_j}{dt} = w_j(x_j, z_j)$$

The integration is carried out in the computational domain and the position of the particle is then mapped into the physical domain for graphical purposes.

The particles are advanced in time according to a second order Runge-Kutta time integration scheme as follows.

$$h_{\xi,j}^n \equiv \frac{\bar{u}_j^n(\xi_j^n, \eta_j^n)}{J_u}\delta t$$

$$h_{\eta,j}^n \equiv \frac{\bar{w}_j^n(\xi_j^n, \eta_j^n)}{J_w}\delta t$$

Then, with

$$\mathcal{I}_j \equiv [(\xi_j^n + h_{\xi,j}^n)/\delta\xi + 1/2]$$
$$\mathcal{K}_j \equiv [(\eta_j^n + h_{\eta,j}^n)/\delta\eta + 1/2]$$
$$r_j \equiv (\xi_j^n + h_{\xi,j}^n)/\delta\xi - (\mathcal{I}_j - 1/2)$$
$$s_j \equiv (\eta_j^n + h_{\eta,j}^n)/\delta\eta - (\mathcal{K}_j - 1/2)$$

and let

$$\mathcal{I}_j' \equiv [(\xi_j^n + h_{\xi,j}^n)/\delta\xi]$$
$$\mathcal{K}_j' \equiv [(\eta_j^n + h_{\eta,j}^n)/\delta\eta]$$
$$r_j' \equiv (\xi_j^n + h_{\xi,j}^n)/\delta\xi - \mathcal{I}_j$$
$$s_j' \equiv (\eta_j^n + h_{\eta,j}^n)/\delta\eta - \mathcal{K}_j$$

14

The new extrapolated velocity components at location $\xi_j^n + h_{\xi,j}^n, \eta_j^n + h_{\eta,j}^n$ are

$$\frac{\bar{u}_j^n}{J_u} = F\left(\frac{\hat{u}_{\mathcal{I}_j'+1,\mathcal{K}_j+1}^n}{J_u}, \frac{\hat{u}_{\mathcal{I}_j'+2,\mathcal{K}_j+1}^n}{J_u}, \frac{\hat{u}_{\mathcal{I}_j'+1,\mathcal{K}_j+2}^n}{J_u}, \frac{\hat{u}_{\mathcal{I}_j'+2,\mathcal{K}_j+2}^n}{J_u}, r', s\right)$$

$$\frac{\bar{w}_j^n}{J_w} = F\left(\frac{\hat{w}_{\mathcal{I}_j+1,\mathcal{K}_j'+1}^n}{J_w}, \frac{\hat{w}_{\mathcal{I}_j+2,\mathcal{K}_j'+1}^n}{J_w}, \frac{\hat{w}_{\mathcal{I}_j+1,\mathcal{K}_j'+2}^n}{J_w}, \frac{\hat{w}_{\mathcal{I}_j+2,\mathcal{K}_j'+2}^n}{J_w}, r, s'\right)$$

Finally, the location of the particle at time level $n + 1$ is

$$\xi_j^{n+1} = \xi_j^n + .5\left(\frac{\bar{u}_j^n}{J_u}\delta t + h_{\xi,j}^n\right)$$

$$\eta_j^{n+1} = \eta_j^n + .5\left(\frac{\bar{w}_j^n}{J_w}\delta t + h_{\eta,j}^n\right)$$

The accuracy of this particle-tracking routine was tested using a steady-state flow with vorticity in a rectangular enclosure. This flow field was originally derived to test the algorithms that integrate the fluid equations and is described in [13]. In the flow, particles follow streamlines which close upon themselves, a sensitive test of the quality of the particle-tracking integration scheme.

B   Appendix: Stability

The linear stability of the lagged-dissipation method has been checked for a single convection-diffusion equation. Derivation of the dispersion relation is straightforward. We start with a single convection-diffusion equation

$$\frac{\partial \theta}{\partial t} + \frac{U}{\sqrt{J}}\frac{\partial \theta}{\partial \xi} + \frac{W}{\sqrt{J}}\frac{\partial \theta}{\partial \eta} = \frac{\epsilon}{J}\nabla^2\theta$$

where θ is the single dependent variable, U and V are the local linearized vector velocity components, ϵ is the dissipation coefficient, ∇^2 is the Laplacian and J is the Jacobian of the spatial transformation. We define

$$\theta(t^n, \xi_j, \eta_k) \equiv \theta_{jk}^n$$

where $\xi_j = j\delta\xi, \eta_k = k\delta\eta, t^n = n\delta t$.

Consider the difference form of the convection-diffusion equation:

$$\frac{\theta_{j,k}^{n+1} - \theta_{j,k}^{n-1}}{2\delta t} + \frac{U}{\sqrt{J}}\frac{\theta_{j+1,k}^n - \theta_{j-1,k}^n}{2\delta\xi} + \frac{W}{\sqrt{J}}\frac{\theta_{j,k+1}^n - \theta_{j,k-1}^n}{2\delta\eta} -$$

$$\frac{\epsilon}{J\delta\eta^2}\left[r^2(\theta_{j+1,k}^{n-1} + \theta_{j-1,k}^{n-1}) + (\theta_{j,k+1}^{n-1} + \theta_{j,k-1}^{n-1}) - 2(1+r^2)\theta_{j,k}^{n-1}\right] = 0$$

where $r \equiv \delta\eta/\delta\xi$. Now let

$$\theta_{j,k}^n = A_n \exp\left[i\delta\eta\left(\frac{j}{r\lambda_\xi} + \frac{k}{\lambda_\eta}\right)\right]$$

15

The equation for the amplitude A_n becomes

$$A_{n+1} - A_{n-1} + iA_n \frac{2Q\delta t}{\sqrt{J}\delta \eta} \left[\frac{Ur}{Q} \sin(\frac{\delta \eta}{r\lambda_\xi}) + \frac{W}{Q} \sin(\frac{\delta \eta}{\lambda_\eta}) \right]$$

$$- A_{n-1} \frac{2\epsilon \delta t}{J\delta \eta^2} \left[2r^2 \cos(\frac{\delta \eta}{r\lambda_\xi}) + 2\cos(\frac{\delta \eta}{\lambda_\eta}) - 2(1+r^2) \right] = 0$$

or,

$$A_{n+1} + A_n iT - A_{n-1}[1 - 2d(1-P)] = 0$$

where

$$
\begin{aligned}
Q &= \sqrt{U^2 + W^2} \\
d &= \frac{2\epsilon \delta t(1+r^2)}{J\delta \eta^2} \\
T &= \frac{2Q\delta t}{\sqrt{J}\delta \eta} \left[\frac{Ur}{Q} \sin\left(\frac{\delta \eta}{r\lambda_\xi}\right) + \frac{W}{Q} \sin\left(\frac{\delta \eta}{\lambda_\eta}\right) \right] \\
P &= \frac{1}{1+r^2} \left[r^2 \cos\left(\frac{\delta \eta}{r\lambda_\xi}\right) + \cos\left(\frac{\delta \eta}{\lambda_\eta}\right) \right]
\end{aligned}
$$

Substitution of $A_n = \theta^n$ and division by θ^{n-1} yields the dispersion relation for the lagged-diffusion scheme

$$\theta^2 + \theta iT - [1 - 2d(1-P)] = 0$$

The roots of this quadratic equation, the amplification factors for the scheme, are

$$\theta = \frac{-iT \pm \sqrt{-T^2 + 4[1 - 2d(1-P)]}}{2}$$

If the Courant condition, $\frac{Q\delta t}{\sqrt{J}\delta \eta} \leq 1$, is satisfied and $d = \frac{2\epsilon \delta t(1+r^2)}{J\delta \eta^2} \leq 1$, then, for both roots,

$$|\theta| = \sqrt{1 - 2d(1-P)} \leq 1$$

since $|P| \leq 1$, and the scheme is stable.

References

[1] Jameson, A., "Computational Aerodynamics for Aircraft Design," *Science, 245,* pp. 361-371, July 1989.

[2] Rehm, R.G., H.C. Tang, H.R. Baum, J.S. Sims and D.M. Corley, "A Boussinesq Algorithm for Enclosed Buoyant Convection in Two Dimensions", *NIST Internal Report, NISTIR 4540*, March, 1991.

[3] Rehm, R.G. and H.R. Baum, "The Equations of Motion for Thermally Driven, Buoyant Flows", *Journal of Research of the NBS, 83,* pp. 297-308, May-June 1978.

[4] Baum, H.R., R.G. Rehm, P.D. Barnett and D.M. Corley, "Finite Difference Calculations of Buoyant Convection in an Enclosure", *SIAM J. Sci. Stat. Comput., 4*, pp. 117-135, March 1983.

[5] Baum, H.R. and Rehm, R.G., " Finite Difference Solutions for Internal Waves in Enclosures", *SIAM J. Sci. Stat. Comput., 5*, No. 4, pp. 958-977, 1984.

[6] Rehm, R.G., P.D. Barnett, H.R. Baum and D.M. Corley, "Finite Difference Calculations Of Buoyant Convection in an Enclosure: Verification of the Nonlinear Algorithm", *Applied Numerical Mathematics, 1*, pp. 515-529, North-Holland, 1985.

[7] Baum, H.R. and R.G. Rehm, "Calculations of Three-Dimensional Buoyant Plumes in Enclosures", *Combustion Science and Technology, 40*, pp. 55-77, Gordon and Breach Science Publishers, 1984.

[8] Gresho, P.M., "A Modified Finite-Element Method for Solving the Incompressible Navier-Stokes Equations," *Large-Scale Computations in Fluid Mechanics,* AMS Lectures in Applied Mathematics, Vol. 22, Part 1, pp. 193-240, 1985.

[9] Trefethen, L.N., "Numerical Computation of the Schwarz-Christoffel Transformation", *SIAM J. Sci. Stat. Comput., 1*, pp. 82-102, 1980.

[10] Simcox, S., Wilkes, N.S. and Jones, I.P., "Fire at King's Cross Underground Station, 18th November 1987: Numerical Simulation of the Buoyant Flow and Heat Transfer", Harwell Report AERE-G 4677, May 1988.

[11] Cox, G., Chitty, R. and Kumar, S., "Fire Modeling and the King's Cross Fire Investigation", Letter to the Editor, *Fire Safety Journal, 15* pp. 103-106, 1989.

[12] Rehm, R.G., Baum, H.R., Lozier, D.W., Tang, H.C. and Sims, J., "Buoyant Convection in an Inclined Enclosure," *Third International Symposium on Fire Safety Science,* The University of Edinburgh, Scotland, U.K., July, 1991.

[13] Rehm, R.G., Baum, H.R. and Barnett, P.D., "Buoyant Convection Computed in a Vorticity, Stream-Function Formulation," *Journal of Research of the NBS, 87*, No. 2, March-April 1982, pp. 165-185.
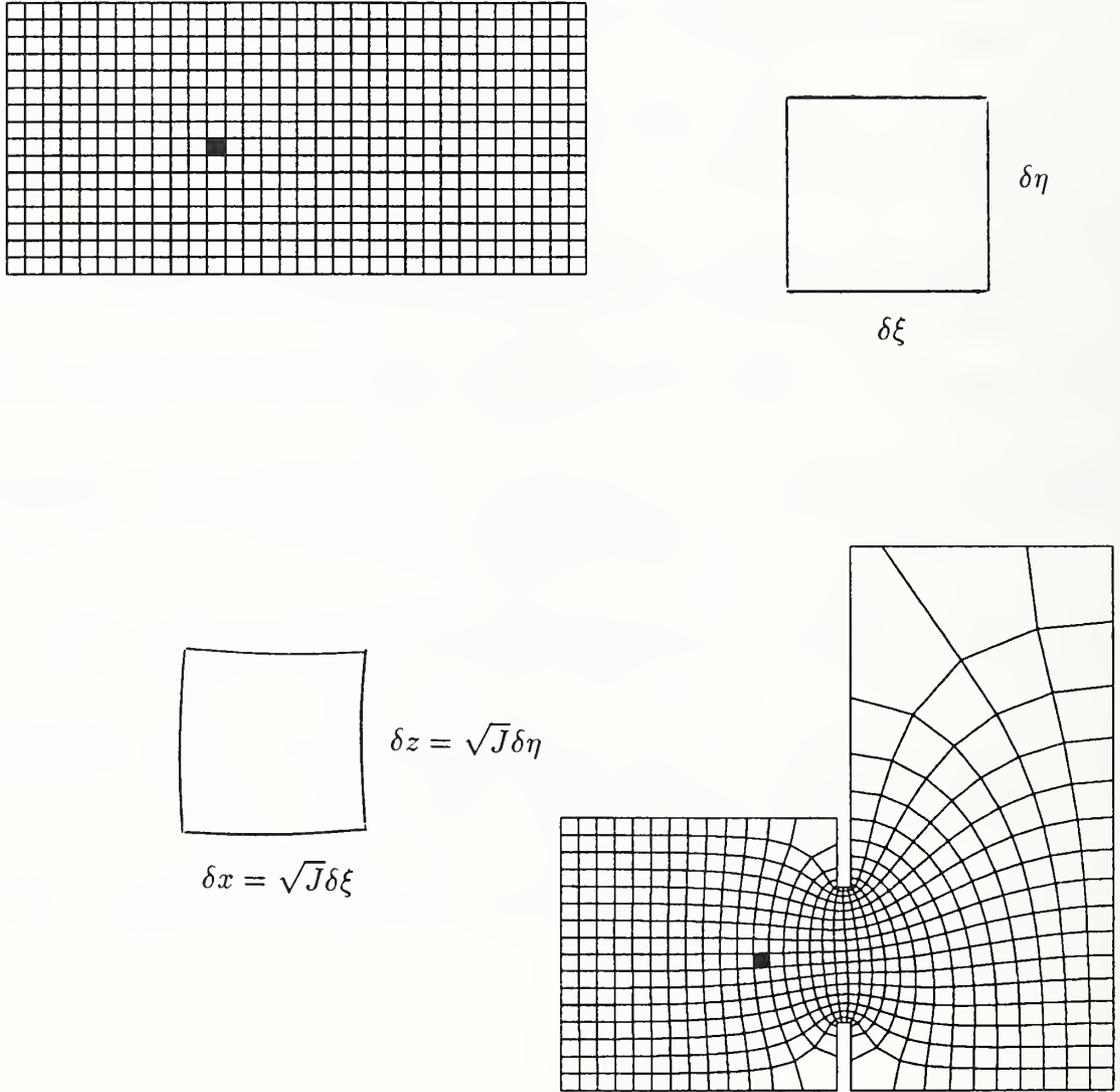
Figure 1. Window grid geometry demonstrating the conformal transformation of the computational into the physical domain.
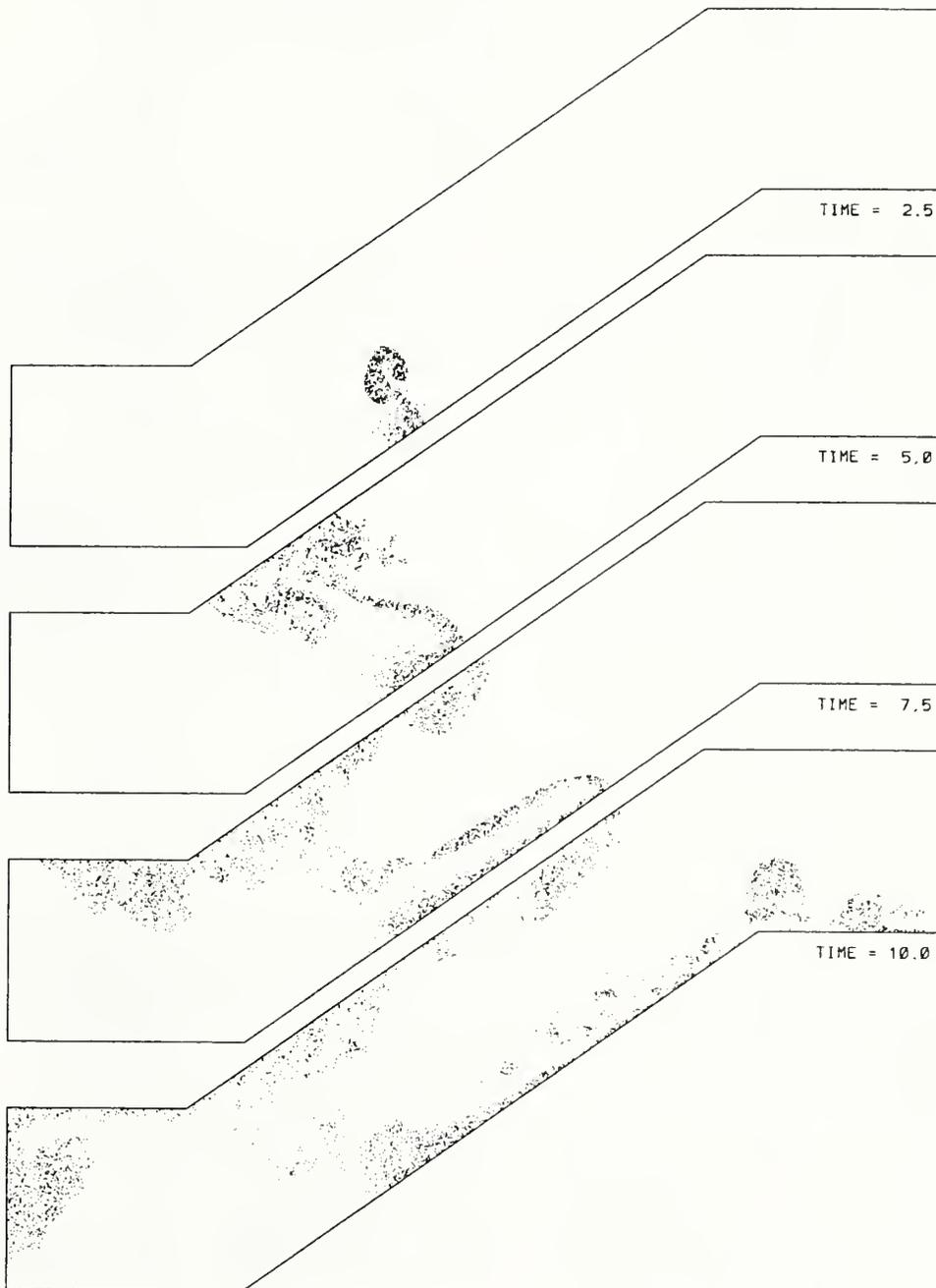
Figure 2. Simulation of fire in a stair well, demonstrating the "trench effect". Adiabatic, free-slip boundary conditions have been imposed. The Reynolds number is 4×10^4; the grid size is 1024×256.
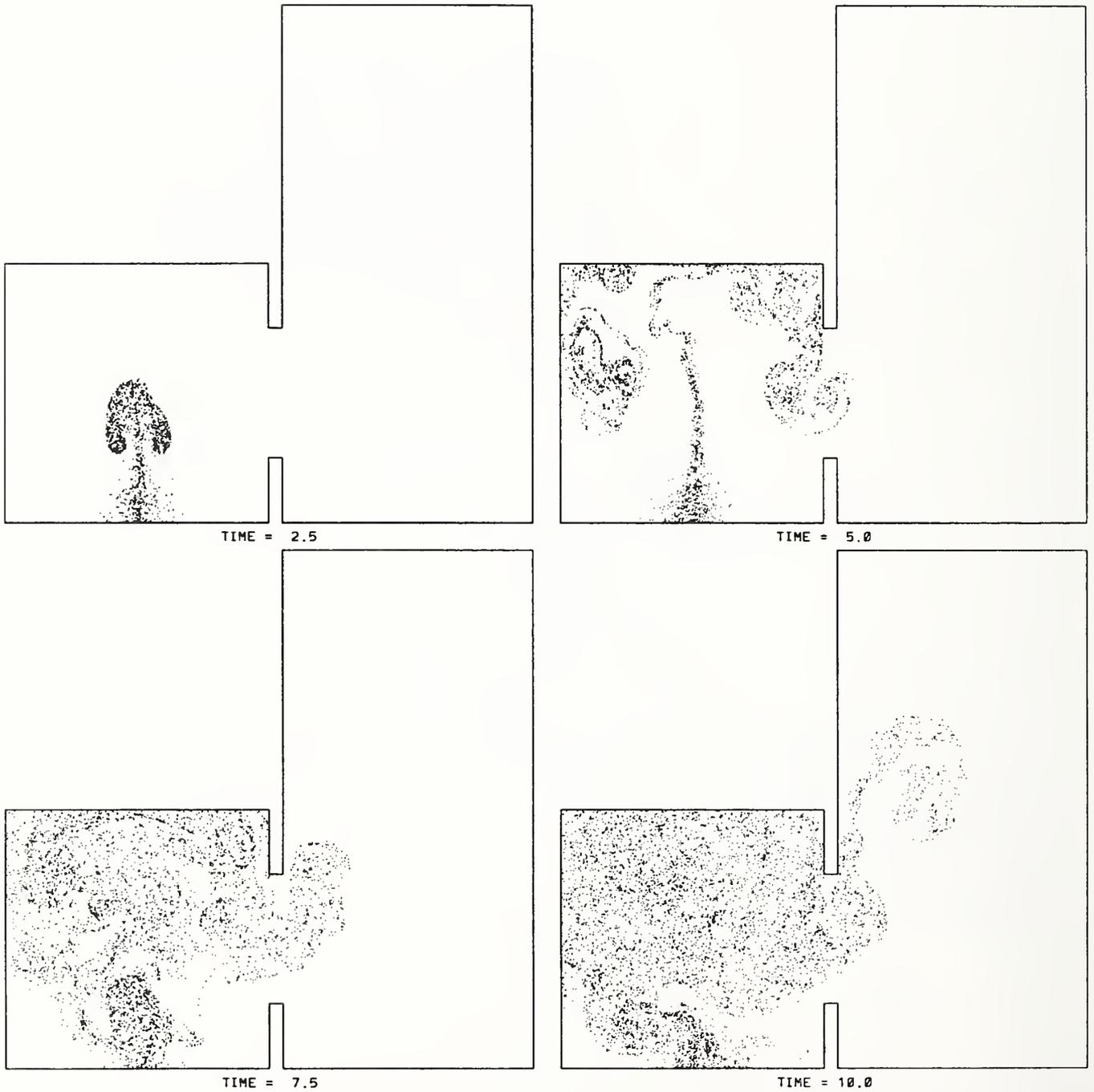
**Figure 3.** Simulation of smoke pouring from a window. Adiabatic, no-slip boundary conditions have been imposed. The Reynolds number is 1×10^4; the grid size is 256×128.

Figure 4. Comparison of four different boundary conditions for the room-window fire scenario. The no-slip condition represents the more realistic model, but is limited by low Reynolds number.

TIME = 5.0

TIME = 10.0

TIME = 15.0

TIME = 20.0

Figure 5. Flow of smoke down a corridor with soffits. Adiabatic, no-slip boundary conditions have been imposed. The Reynolds number is 1×10^4; the grid size is 512×128. This geometry points out the difficulties faced by the current methodology in handling the flow around sharp corners.
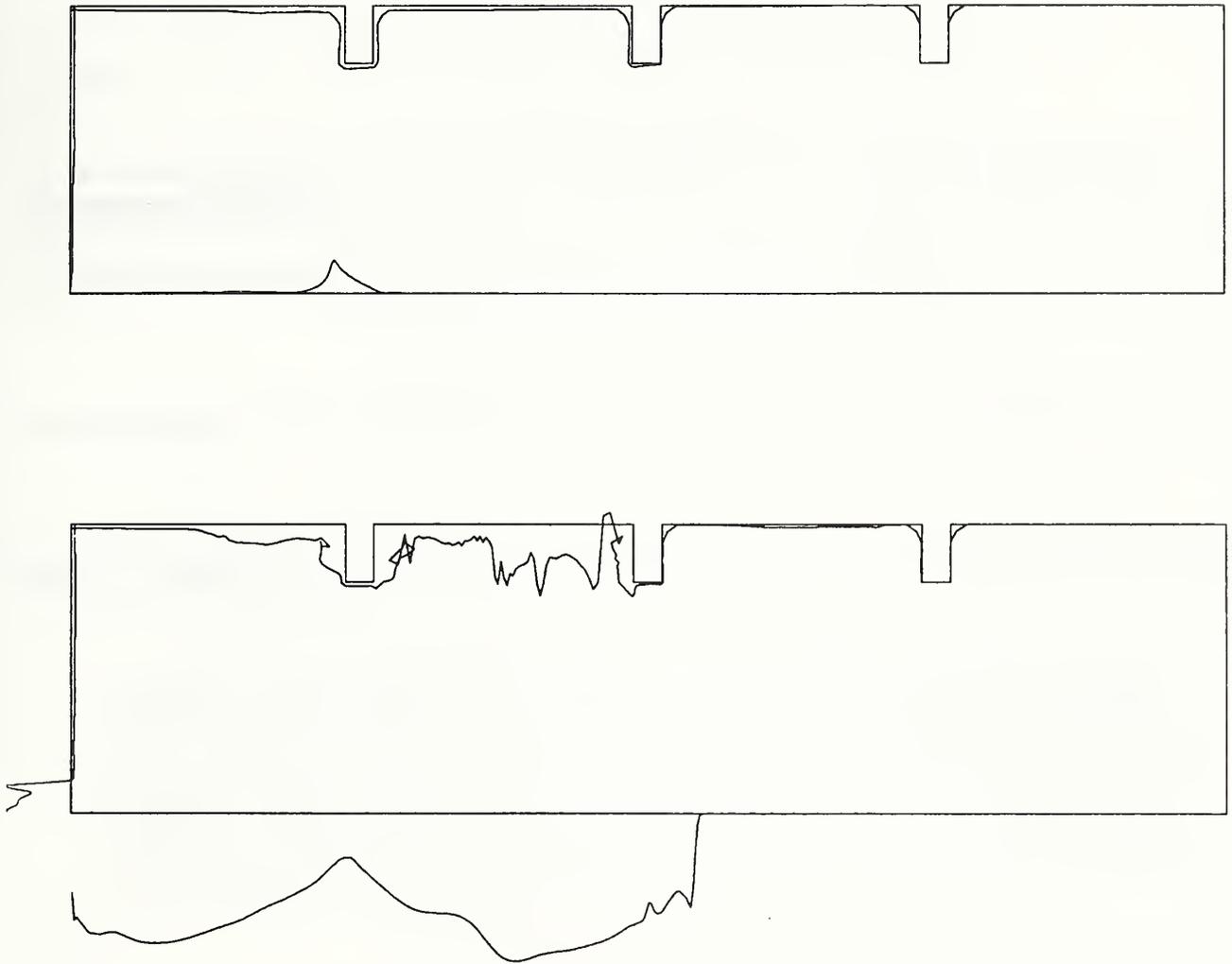
Figure 6. Density perturbation in the boundary cells at times $t = 16$ and $t = 18$. This instability of the method is not predicted by the linear stability analysis of Appendix B.

23

4. TITLE AND SUBTITLE

A Boussinesq Algorithm for Buoyant Convection in Polygonal Domains

5. AUTHOR(S)

11. ABSTRACT (A 200-WORD OR LESS FACTUAL SUMMARY OF MOST SIGNIFICANT INFORMATION. IF DOCUMENT INCLUDES A SIGNIFICANT BIBLIOGRAPHY OR LITERATURE SURVEY, MENTION IT HERE.)

A 2-D Boussinesq model describing heat-driven buoyant convection in a polygonal enclosure is presented. The hydrodynamics is based on the time-dependent Navier-Stokes equations with constant viscosity and thermal conductivity; no turbulence model or other empirical parameters are introduced. The polygonal domain is mapped via a Schwarz-Christoffel transformation onto a rectangle. A finite difference scheme second-order in space and first-order in time is used to integrate the evolution equations, and an elliptic solver is used to solve the pressure equation. Computational results for high Reynolds numbers are presented through the use of Lagrangian particles which allow one to visualize the flow patterns.

12. KEY WORDS (6 TO 12 ENTRIES; ALPHABETICAL ORDER; CAPITALIZE ONLY PROPER NAMES; AND SEPARATE KEY WORDS BY SEMICOLONS)

algorithm; Boussinesq flow; buoyant convection; computational fluid dynamics; conformal mapping; polygonal domains; Schwarz-Christoffel transformation