

GUIDANCE ON THE USE OF COMPUTATIONAL FLUID DYNAMICS FOR MODELLING BUOYANCY DRIVEN FLOWS

Malcolm J. Cook BSc MSc GIMA & **Kevin J. Lomas** PhD CEng MCIBSE MInstE
Institute of Energy & Sustainable Development, De Montfort University,
The Gateway, Leicester, LE1 9BH, UK
<http://www.iesd.dmu.ac.uk>

ABSTRACT

Buoyancy driven flows have always been, and still are, difficult to model using CFD programs. Much validation work is required along with guidelines for the CFD practitioner about how to model such flows. This paper makes a contribution to these two areas by considering buoyancy driven displacement ventilation in a very simple geometry.

The nature of buoyancy driven displacement ventilation is briefly described and guidance offered on how to represent the problem in a CFD program. Simulations were carried out using two eddy viscosity turbulence models. The results show good qualitative agreement with experimental data and analytical results but small discrepancies in some of the quantitative results.

INTRODUCTION

Buildings must be provided with adequate ventilation. Environmental concerns together with demands for energy efficiency are prompting designers to investigate natural ventilation as an alternative to mechanical ventilation or full air conditioning. Buoyancy driven displacement ventilation is proving to be a popular approach, and it is the modelling of this phenomenon, using Computational Fluid Dynamics (CFD), that is addressed in this paper.

Buoyancy driven displacement ventilation occurs when thermal plumes produced above occupants and equipment cause a layer of warm, buoyant air to form close to the ceiling of the space. This layer drives a flow through high-level openings and causes fresh, ambient air to be drawn in at low level. A prominent feature of this flow is the formation of a horizontal interface separating the warm, buoyant air above, from the ambient, cooler air below. A theory for this type of flow was described by Linden et al. (1990) who considered displacement ventilation of a simple rectangular box. To visualise these flows, the salt bath modelling technique was

employed. The results of these experiments compared favourably with the theory. The CFD approach is being used to explore the same idealistic conditions.

Buoyancy driven flows are notoriously difficult to model using CFD techniques for the following reasons: small driving forces lead to numerical instabilities; there is uncertainty regarding how to accurately model turbulence; and the flow is implicitly specified (i.e. sources of buoyancy are specified from which the program must calculate a velocity field). Consequently, much validation work is needed to evaluate the ability of these codes to model such flows.

It is the intention of this paper to offer guidance on how to model buoyancy driven flows and the accuracy which can be expected from a commercial CFD package. The simulations reported here comprise simple geometries with a single source of buoyancy, but it is hoped that the conclusions drawn and the experience gathered can be used in modelling more complex, real buildings.

THEORY OF BUOYANCY DRIVEN DISPLACEMENT VENTILATION

A good example of a buoyancy driven flow is displacement ventilation in which use is made of the buoyancy forces produced by naturally occurring heat sources such as occupants and equipment (e.g. computer terminals, projector units, etc).

The theory for this type of flow was elucidated by Linden et al. (1990) who considered displacement ventilation of a simple rectangular box (fig. 1).

The heat source produces a rising plume which entrains fresh (ambient) air taking it upwards into the upper region of the space where it is recirculated. As more and more light air accumulates in the upper region of the space, the rising plume begins to entrain this lighter air until a steady-state is achieved (fig. 1), in which a constant interface forms (at $y = h$) separating the dense, cool

air below the interface from the warmer, buoyant air above. In the steady-state, the layer of warm, buoyant air drives a flow through the upper openings since the hydrostatic pressure difference between the top and bottom of the layer is smaller inside the space than between the same heights outside the space. This causes fresh air to be drawn in through the lower openings.

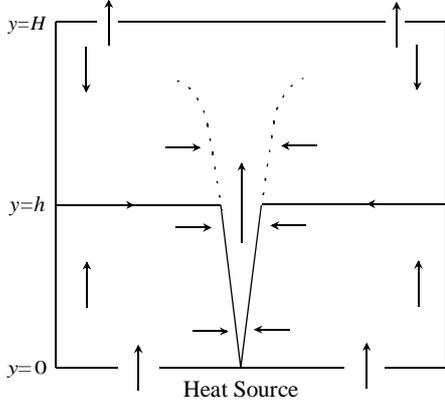


Figure 1. Steady displacement ventilation flow in a box containing a constant heat source (arrows show direction of flow), after Linden et al. (1990)

Linden et al. (1990) used the plume theory of Morton et al. (1956) to derive an expression for the interface height (eq. 1) by equating volume and buoyancy fluxes in the plume with those through the inlets.

$$\frac{A^*}{H^2} = C^{3/2} \left(\frac{(\frac{h}{H})^5}{1 - \frac{h}{H}} \right)^{1/2} \quad (1)$$

where A^* is an *effective opening area* defined as

$$A^* = \frac{a_u a_l}{\left[1/2(a_u^2/c + a_l^2) \right]^{1/2}} \quad (2)$$

where: a_u = total area of upper openings;
 a_l = total area of lower openings; and
 c = momentum theorem constant (= 1 throughout this work),

and $C = \frac{6}{5} \alpha (\frac{9}{10} \alpha)^{1/3} \pi^{2/3}$ is a constant dependent upon the entrainment constant α . The entrainment constant is an empirical value and can be thought of as a proportionality constant relating inflow velocity at the edge of the plume with the upward velocity on the axis of the plume. Measuring α is very difficult and several authors have suggested values (e.g. Morton et al. (1956), Baines et al. (1968)). The value used throughout this work is that used by Linden et al. (1990), ($\alpha = 0.1$).

Linden et al. (1990) verified their theoretical predictions using the salt bath modelling technique in which perspex models are inverted in fresh water

and brine injected to represent the effects of a heat source.

DETAILS OF THE CFD PACKAGE

The CFD package used for this work was CFX-F3D (1996). The code solves the conservation equations for mass, momentum and energy (enthalpy):

$$\frac{\partial}{\partial x_j} (\rho u_j \phi) - \frac{\partial}{\partial x_j} \left(\Gamma_\phi \frac{\partial \phi}{\partial x_j} \right) = S_\phi \quad (3)$$

The source terms and diffusion coefficients are given in table 1 for the variable ϕ . All transient terms have been omitted since the work sought steady-state solutions.

Table 1. Terms in the governing equations when using an eddy viscosity turbulence model.

Conservation equation	ϕ	Γ_ϕ	S_ϕ
Mass	1	0	0
Momentum	u_i	$\mu + \mu_\tau$	$\frac{\partial}{\partial x_j} \left(-p_0 \delta_{ij} + \mu_{eff} \frac{\partial u_i}{\partial x_j} \right) + \rho g_i$
Enthalpy	He	$\frac{\lambda}{C_p} + \frac{\mu_\tau}{\sigma_{te}}$	0

p_0 is a ‘modified’ pressure given by

$$p_0 = p + \frac{2}{3} \rho k - \rho_{Bref} g_i x_i \quad (4)$$

where ρ_{Bref} is a buoyancy reference density.

Turbulence modelling is a crucial aspect of all CFD modelling, but is particularly important in buoyancy driven flows in which the buoyancy forces increase turbulent activity. This paper considers two models. The first is the industry standard $k-\epsilon$ model of Launder and Spalding (1974) in which transport equations for k and ϵ are solved as follows:

$$\frac{\partial}{\partial x_j} (\rho u_j k) - \frac{\partial}{\partial x_j} \left(\left(\mu + \frac{\mu_\tau}{\sigma_k} \right) \frac{\partial k}{\partial x_j} \right) = P + G - \rho \epsilon \quad (5)$$

$$\frac{\partial}{\partial x_j} (\rho u_j \epsilon) - \frac{\partial}{\partial x_j} \left(\left(\mu + \frac{\mu_\tau}{\sigma_\epsilon} \right) \frac{\partial \epsilon}{\partial x_j} \right) = C_1 \frac{\epsilon}{k} P - C_2 \rho \frac{\epsilon^2}{k} \quad (6)$$

where $C_1 = 1.44$ and $C_2 = 1.92$ are empirical constants. P and G represent the production of turbulent kinetic energy due to shear stresses and buoyancy respectively:

$$P = \mu_{eff} \frac{\partial u_i}{\partial x_j} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right); \text{ and}$$

$$G = \frac{\mu_{eff}}{\sigma_\tau} \beta g_j \frac{\partial T}{\partial x_j}.$$

where σ_T is the turbulent Prandtl number for k or ε depending upon which equation is being considered.

The second is the more recent RNG $k-\varepsilon$ model based on Renormalised Group (RNG) theory established by Yakhot and Orszag (1992). This model assumes the same form of the k and ε equations as the model of Launder and Spalding (1974) but uses RNG theory to calculate the constants. The constants are represented in CFX-F3D as follows.

C_1 is replaced with $C_1 - C_{1RNG}$ where

$$C_{1RNG} = \frac{\eta \left(1 - \frac{\eta}{\eta_0}\right)}{(1 + \beta_1 \eta^3)},$$

$$\eta = \left(\frac{P}{\mu_T}\right)^{1/2} \frac{k}{\varepsilon}, \beta_1 = 0.015 \text{ and } \eta_0 = 4.38.$$

$C_1 = 1.42$, and $C_2 = 1.68$.

The models differ in the way the constants are derived. In the standard $k-\varepsilon$ model, all constants are found empirically. This has the disadvantage of the constants only being applicable to the types of flow from which they were derived. However, good results have been obtained for many air flow applications using this model. In the RNG model, the constants are derived mathematically such that the constants are valid for a very wide range of flow types including both high *and* low Reynolds number flows. It is interesting to note that the RNG theory gives rise to a new rate-of-strain term in the ε equation. This equation has long thought to have been responsible for inaccuracies in the standard $k-\varepsilon$ model when large deformation forces are present (Versteeg et al. (1995)).

In both models, the eddy viscosity, μ_T is calculated using

$$\mu_T = \rho C_\mu \frac{k^2}{\varepsilon}$$

$$\text{where: } C_\mu = \begin{cases} 0.09 & (\text{standard } k-\varepsilon) \\ 0.085 & (\text{RNG } k-\varepsilon) \end{cases}$$

CFX is a *multiblock* code in which one or more topologically rectangular blocks are ‘glued’ together to form the geometry. Variables are solved for at *cell centres* (a co-located grid) rather than the more traditional approach which uses a staggered grid. The co-located grid has the simple advantage of having both velocities and scalar quantities located

at the same position, thus producing a more readily understood solution field.

All the governing equations are discretised using hybrid differencing (except the mass conservation equation where central differencing is always used). The finite volume solution method is used (see Versteeg et al. (1995)), and pressure and velocity are coupled using the SIMPLEC algorithm with Rhie-Chow interpolation (Rhie and Chow (1983)) to prevent decoupling of pressure and velocity due to the co-located grid. The following (default) under-relaxation factors were used: momentum, 0.65; mass, 1.0; enthalpy, 1.0; k , 0.7; and ε , 0.7. These are similar to those used in most CFD codes.

Buoyancy is modelled using the Boussinesq approximation (see Turner (1973)) in which density is assumed to be constant except in the momentum equation where it is written

$$\rho = \rho_{Bref} (1 - \beta(T - T_{Bref})) \quad (7)$$

This gives rise to the buoyancy term in the modified pressure, p_0 in table 1. The Boussinesq approximation is only applicable in flows where the density differences are small. Those that occur in buildings due to temperature variations are sufficiently small.

Three types of boundary condition are used in this investigation: WALL boundaries; PRESSURE boundaries; and SYMMETRY PLANE boundaries.

WALL boundary conditions are placed at fluid-solid interfaces and enable the specification of velocities (normally zero), heat fluxes, and temperatures. Conventional wall functions are imposed at WALL boundaries (Versteeg et al. (1995)).

Fluid may flow into or out of the domain across a PRESSURE boundary. If fluid flows *into* the domain, Neumann conditions (i.e. zero normal gradient) are imposed on velocity and turbulence quantities, and values assigned explicitly to pressure and temperature (Dirichlet conditions). When fluid flows *out* of the domain across a PRESSURE boundary, Dirichlet conditions are imposed on pressure, and Neumann conditions on all other variables.

At SYMMETRY PLANE boundaries, all variables are set to be mathematically symmetric, except the component of velocity normal to the boundary which is *anti*-symmetric.

REPRESENTATION IN THE CFD MODEL

In order to evaluate the ability of the CFD code to model displacement ventilation, a simple case was

defined similar to that in figure 1. The space was 5.1m wide, 2.55m high and 1m deep with upper and lower openings to the ambient air at 18°C. The flow was driven by a point source of buoyancy placed in the centre of the floor. Both the source strength and the opening areas were varied.

Accurate modelling of the flow as it passed through the openings was considered important due to the loss in pressure head, or discharge, suffered by the flow as it expands and contracts. Therefore, instead of specifying boundary conditions at the openings, the flow through the openings was modelled explicitly by extending the computational domain beyond the room and specifying PRESSURE boundaries at some distance away. How much of the exterior to model was investigated in detail in Cook (1997), where a sensitivity analysis showed that extra space above and below the room (fig. 2) was sufficient.

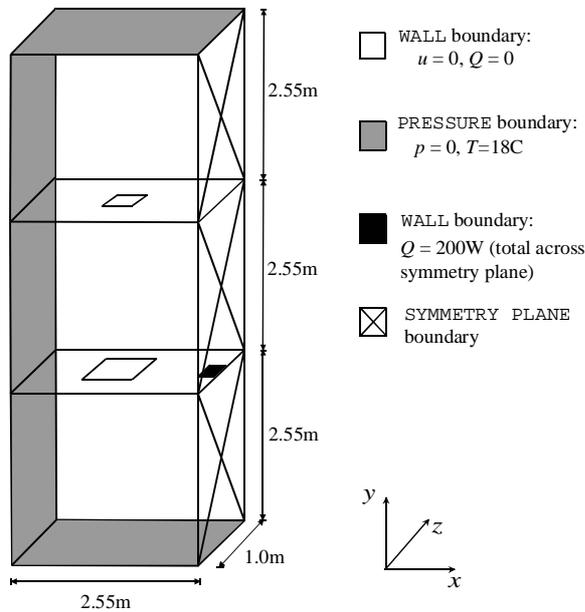


Figure 2. Geometry and boundary conditions used in the simulations

The distance of the PRESSURE boundaries from the room were such that any greater distance produced no further change in the flow *inside* the box, i.e. h/H . The value assigned to pressure at all PRESSURE boundaries was $p=0$. This was possible since in incompressible flows, pressure is arbitrary to within an additive constant, and the hydrostatic pressure difference is automatically taken care of in the momentum equation (see modified pressure (eq. (4)).

When prescribing a value for p at a PRESSURE boundary in a buoyancy driven flow, consideration must be given to the buoyancy reference density (eq. (4)), or the buoyancy reference temperature (eq.

(7)). By considering the behaviour of equation (4) at the pressure boundary, the following is obtained (by substituting eq. (7) into eq. (4)):

$$p_0 = p + \frac{2}{3}\rho k - \frac{\rho g_i x_i}{1 - \beta(T - T_{\text{ref}})} \quad (8)$$

At the PRESSURE boundary, in the absence of any local temperature gradient, p_0 should equal $\frac{2}{3}\rho k - \rho g_i x_i$, due to an extended Boussinesq relationship (see Versteeg et al. (1995)). In order for this to be so when $p=0$, the buoyancy reference temperature should be set equal to the temperature on the pressure boundary, i.e. the ambient temperature (see eq. (8)). Specifying any other temperature would generate an additional, unrealistic pressure force.

There exists a disadvantage of using *external pressure boundaries* such as those used here (fig. 2). There is a tendency, in all CFD codes, to over-predict entrainment across these boundaries (early simulations showed unrealistically high velocities predicted in the z -direction in the exterior). This occurs because there is nothing *in theory* to stop the flow from doing this (Newton's laws of motion). Thus a high velocity flow across these boundaries does indeed satisfy the governing equations. The problem is overcome simply by specifying WALL boundary conditions where this is a problem, i.e. at the high and low z faces in this case.

It is common practice to force numerical symmetry in problems that are geometrically symmetric by means of a symmetry plane (fig. 2). This not only reduces computational time, but in this case aids convergence by ensuring that the development of the plume is uninhibited by any unbalanced lateral forces that may arise during the solution procedure.

The mesh used in this investigation is shown in figure 3. It was found (Cook 1997) that further mesh refinement did not change the governing parameters of the flow, i.e. interface height and change in temperature across the interface. Accurate values for ΔT across the interface could be obtained using a coarser mesh. However, this produced a more diffuse stratification making it difficult to measure the interface height accurately.

In order to monitor the progress of the solution procedure it is necessary to define a monitoring point. The location of this point should be selected such that it provides the user with as much (useful) information as possible. For example, there would be little to be gained in placing it close to the heat source since this area is likely to converge with little difficulty as it is close to the driving force. Similarly it would be inadvisable to place it in the

core of a recirculation zone since this may take a very long time to reach convergence. In the simulations reported here, the monitoring point was located in the centre of the upper opening.

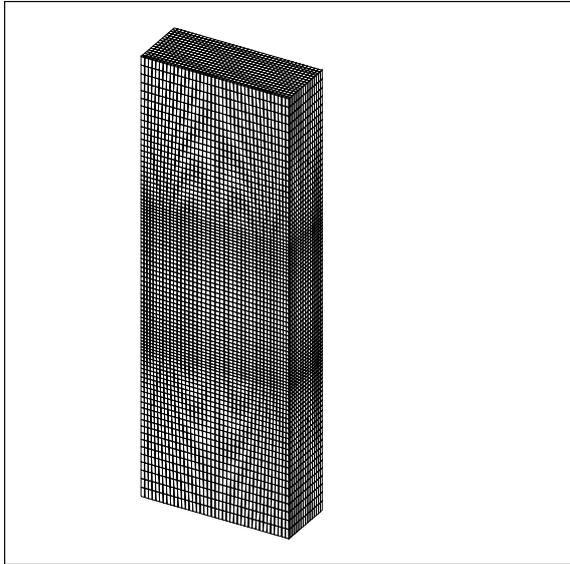


Figure 3. Mesh density used in all simulations
($40 \times 90 \times 16$)

RESULTS

In order to ensure that the mathematical model defined in the CFD code has been satisfactorily solved, one must identify some convergence criteria. An obvious criterion is that the values of the variables (u_i , p , k , ϵ , and He) are constant to within a very small tolerance, say 0.01%. This is a necessary condition but is not sufficient. Although the variables may be changing only very slightly, they could in fact be a long way from satisfying the governing equations. Consequently a convergence criterion is required on the residuals (the errors between the left and right hand sides of the discretised governing equations) to ensure they are sufficiently small. A suitable condition in this work is that the enthalpy residual (in watts) is small when compared with the total heat entering the domain (of order 1%, say).

In the simulations reported here, constant variables at the monitoring point could not be obtained using the default solution technique and under-relaxation factors. The enthalpy residual was too high and there were oscillations on all the variables. In an attempt to eliminate the oscillations, the amount by which a variable changed between subsequent iterations was reduced. This was done by imposing tighter under-relaxation, i.e. reducing the under-relaxation factors. However, convergence could still not be achieved. Another type of under-relaxation is false time-stepping. This is a particular

type of under-relaxation which reflects the time scale over which the flow evolves. Instead of specifying just one under-relaxation factor for each equation, false time-stepping enables under-relaxation factors to be specified in *each cell* in terms of a pseudo time-step, Δt , and the volume of that particular cell. This enables much more specific under-relaxation factors to be specified. The only difficulty lies in deciding what values to set for Δt . It was found by Cook (1997) that false time-steps of 0.1s on the two momentum equations eliminated the oscillations on all the variables. It is useful to note that this is approximately equal to the smallest cell size in the domain ([smallest cell area]^{1/2}) divided by the maximum velocity in the domain. Therefore, using the non-converged solution with default under-relaxation as an approximation to the converged result, an estimate could be found for Δt . It should be noted that Δt is only used to *control* the iteration procedure, it is not unique in determining the final result and only needs to be accurate to within the same order of magnitude.

Under-relaxation and false time-stepping have the effect of reducing the amount by which a variable changes between iterations. This slows down the rate of convergence as more iterations are needed. Consequently, in order to minimise run times, it was decided to begin the simulation using default under-relaxation and to 'switch on' false time-stepping when the residuals had stopped falling, i.e. when the iteration procedure had produced the best solution possible using the default under-relaxation factors. Convergence was achieved for both turbulence models by running for about 1000 iterations followed by about 2000 iterations, although the residuals were slightly higher for the RNG $k-\epsilon$ model compared with those for the standard $k-\epsilon$ model.

The resulting flow pattern (figs. 4, 5, 6 and 7) agreed favourably with the theoretical and experimental predictions of Linden et al. (1990). A rising plume forms above the heat source and produces stratification in the box. An interface forms across which there is a sharp change in temperature, and a flow is driven through the openings. When comparing the results showing the effects of various opening sizes, it can be seen that the RNG $k-\epsilon$ model gives a better prediction of the interface height than the standard $k-\epsilon$ model (fig. 8). This is thought to be due to the RNG model predicting a narrower plume than the $k-\epsilon$ model (c.f. figs. 4 and 5). The plume in figure 5 has to rise to a greater height before its volume flow rate matches that through the openings upon which it collapses and forms the interface. The narrower

In all flow field figures, a vector of length 1cm corresponds to a flow speed of 0.3m/s.

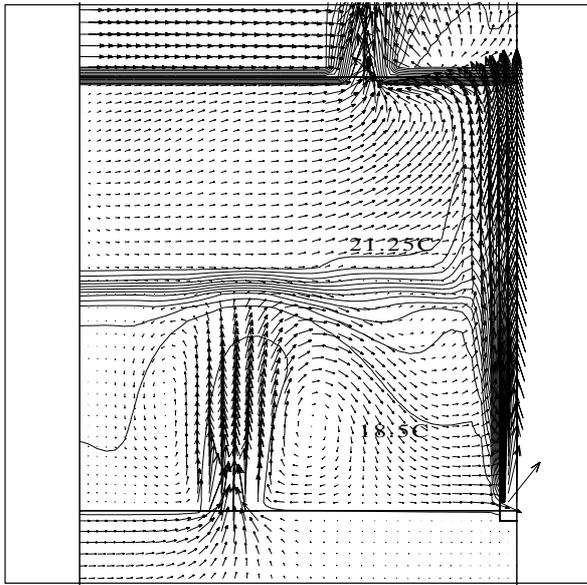


Figure 4. Flow pattern predicted using standard $k - \epsilon$ turbulence model (total heat input = 200W, $A^* / H^2 = 0.0304$)

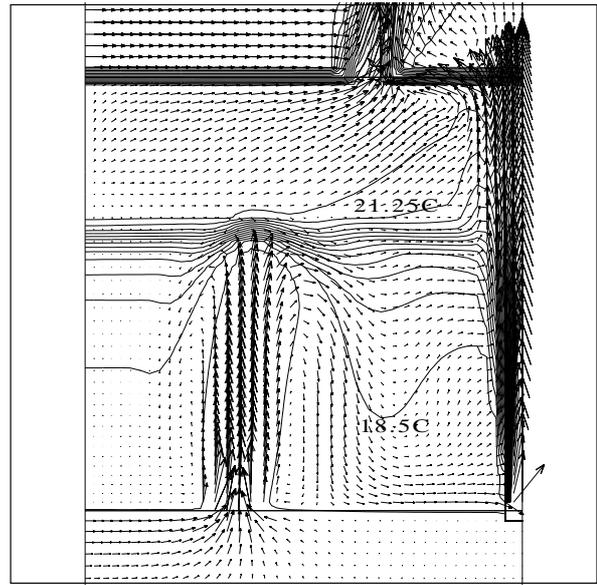


Figure 5. Flow pattern predicted using RNG $k - \epsilon$ turbulence model (total heat input = 200W, $A^* / H^2 = 0.0304$)

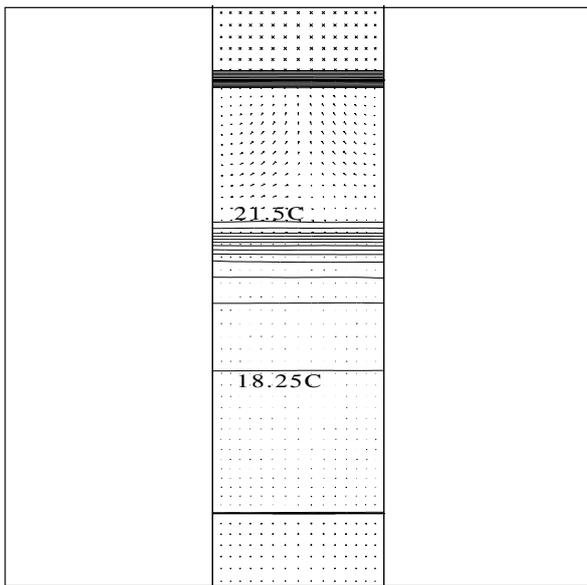


Figure 6. Flow pattern predicted using RNG $k - \epsilon$ turbulence model (total heat input = 200W, $A^* / H^2 = 0.0304$) showing plane $x = 0.2m$

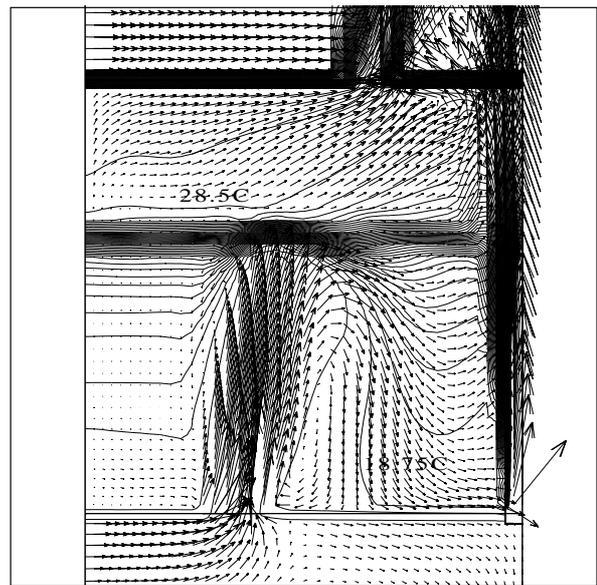


Figure 7. Flow pattern predicted using RNG $k - \epsilon$ turbulence model (total heat input = 1000W, $A^* / H^2 = 0.0304$)

plume suggests that the RNG $k-\epsilon$ model predicted a smaller rate of entrainment into the plume.

The work of Linden et al. (1990) concluded, amongst other things, that the height of the interface is *independent* of the source strength. Increasing the source strength only has the effect of inducing higher velocities and producing a larger temperature difference across the interface. CFD results for source strengths of 200W and 1000W verified these points (c.f. figs. 5 and 7).

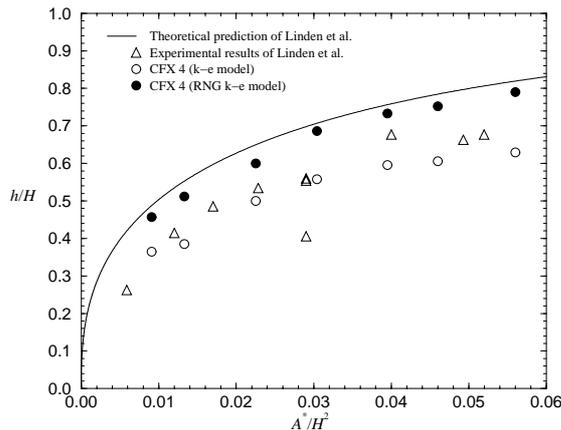


Figure 8. Variation of interface height with effective opening area for both turbulence models, using a heat source of 200W

CONCLUSIONS

This work has verified that buoyancy driven flows are difficult to model and tight control is required in the solution process of the equations. This is best done by use of under-relaxation factors which are both cell dependent and pseudo time dependent. Despite this, favourable qualitative agreement was obtained between the CFD results and the theoretical and experimental work of Linden et al. (1990) for a simple space undergoing buoyancy driven displacement ventilation.

Two turbulence models were investigated: the standard $k-\epsilon$ model; and the RNG $k-\epsilon$ model. Both gave qualitatively similar results, but the RNG $k-\epsilon$ model predicted quantitative results which were closer to the theoretical values for the height of the interface. This discrepancy is thought to be due to the way in which the plume is modelled, and in particular, the rate at which air is entrained into the plume. It highlights the importance of accurate modelling of the plume in such flows.

When using CFD programs, one must always bear in mind the accuracy with which the code was able to

solve the governing equations. In these simulations, it was noted that the RNG $k-\epsilon$ model produced higher numerical errors (residuals) in the enthalpy equation than the standard $k-\epsilon$ model. The RNG $k-\epsilon$ model is still a new technique and very little validation has been carried out. Small changes in how the RNG $k-\epsilon$ technique is implemented may still be required, and this will be the subject of further investigation.

ACKNOWLEDGEMENTS

This work formed part of the first named author's PhD work undertaken at De Montfort University. The authors wish to acknowledge the advice received from Dr. G. Whittle (Simulation Technology Ltd.) and Dr. P. Linden (University of Cambridge).

REFERENCES

- Baines, W. D. and Turner, J. S. "Turbulent Buoyant Convection from a Source in a Confined Region", *J. Fluid Mechanics* 37, pp.51-82, 1968
- Computational Fluid Dynamics Services, (AEA Harwell), "CFX 4.1 Flow Solver User Guide", pp.263-350, 1995
- Cook, M. J., "PhD Thesis", in press, 1997
- Linden, P. F., Lane-Serff, G. F., and Smeed, D. A. "Emptying Filling Boxes: The Fluid Mechanics of Natural Ventilation", *J. Fluid Mechanics* 212, pp.309-335, 1990
- Morton, B. R., Taylor, G. I., and Turner, J. S. "Turbulent Gravitational Convection from Maintained and Instantaneous Sources", *Proc. R. Soc. London A234*, pp.1-23, 1956
- Launder, B. E. and Spalding, D. B. "The Numerical Computation of Turbulent Flows", *Computer Methods in Applied Mechanics and Engineering* 3, pp.269-289, 1974
- Rhie, C. M. and Chow, W. L., "Numerical Study of the Turbulent Flow Past an Airfoil with Trailing Edge Separation", *AIAA J1* 21, pp.1527-1532, 1983
- Turner, J. S., "Buoyancy Effects in Fluids", Cambridge University Press, pp.165-206, 1973
- Versteeg, H. K. and Malalasekera, W., "An Introduction to Computational Fluid Dynamics - The Finite Volume Method", Longman Group Ltd. 1995
- Yakhot, V., Orszag, S. A., Thangam, S., Gatski, T. B. and Speziale, C. G., "Development of Turbulence Models for Shear Flows by a Double

NOMENCLATURE

C_p	specific heat
g_i, g	gravity force
He	enthalpy
k	turbulent kinetic energy
p	pressure
Q	heat input
S_ϕ	source term
T	temperature
T_{Bref}	buoyancy reference temperature
u_i	u, v, w (velocity components in x, y and z directions)
x_i	x, y, z (cartesian coordinate system where z is perpendicular to the page)
β	thermal expansion coefficient
δ_{ij}	$\begin{cases} 1 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$
ε	dissipation of turbulent kinetic energy
ϕ	arbitrary variable
Γ_ϕ	diffusion coefficient
λ	thermal conductivity
μ	dynamic (laminar) viscosity
μ_T	turbulent viscosity
μ_{eff}	$\mu + \mu_T$
ρ	density
ρ_{Bref}	buoyancy reference density
$\sigma_{He} = 0.9, \sigma_k = 1.0, \sigma_\varepsilon = 1.217$ (standard $k - \varepsilon$) or 0.7179 (RNG $k - \varepsilon$)	are turbulent Prandtl numbers for $He, k,$ and ε respectively