Improving the regulatory acceptance and numerical performance of CFD based fire-modelling software.

Angus Joseph Grandison

A thesis submitted in partial fulfilment of the requirements of the University of Greenwich for the degree of Doctor of Philosophy

December 2003

The University of Greenwich,
School of Computing and Mathematical Science,
Park Row,
Greenwich SE10 9LS

Acknowledgements

I wish to thank my supervisors, Prof. Ed Galea and Dr Mayur Patel, for all the help and advice they have given me throughout my time at the University of Greenwich. Dr Patel's impromptu creative brain storming sessions were both useful and interesting. Prof. Galea's constructive criticism has contributed to improving this thesis and my work in general.

I would like to thank the members of the FIREDASS consortium for their interest and advice. Thanks are due to Nigel Mawhinney, my friend and colleague at the University of Greenwich, for his endless streams of thought and discussions throughout FIREDASS and beyond.

Dr John Ewer and Dr Fuchen Jia need to be thanked for their help and advice concerning the SMARTFIRE CFD based Fire model. Dr Ewer's instruction in using the badger teaching tool was also extremely useful.

I am indebted to the European Union for the funding provided by the BRITE EURAM programme for the FIREDASS work. I am also indebted to the UK Home Office Fire Research Department for the funding provided for the software benchmarking. Particular thanks are due to Dr Brian Hume of the UK Home office for his enthusiasm and interest in this work.

Finally I would like to thank my friends and family that have supported me throughout my time at the University of Greenwich.

Abstract

The research of this thesis was concerned with practical aspects of Computational Fluid Dynamics (CFD) based fire modelling software, specifically its application and performance. Initially a novel CFD based fire suppression model was developed (FIREDASS). The FIREDASS (FIRE Detection And Suppression Simulation) programme was concerned with the development of water misting systems as a possible replacement for halon based fire suppression systems currently used in aircraft cargo holds and ship engine rooms. As part of this programme of work, a computational model was developed to assist engineers in optimising the design of water mist suppression systems. The model comprised of the following components: fire model; mist model; two-phase radiation model; suppression model; detector/activation model. The fire model uses prescribed release rates for heat and gaseous combustion products to represent the fire load. Typical release rates for heat and combustion products have been determined through experimentation. The radiation model is a six-flux model coupled to the gas (and mist) phase. The mist model is based on Lagrangian particle tracking. Only the fire and suppression model will be described in detail in this thesis as this constituted the author's contribution to FIREDASS. This work highlighted a number of issues associated with the application of CFD fire modelling software used in design of fire safety systems. The first issue was the reliability of CFD based fire predictions while the second was a practical issue associated with the amount of time required to run CFD fire models in a practical design environment. The remainder of the thesis is concerned with addressing these issues.

To address the first issue a set of procedures was developed to test the applicability of CFD fire modelling software. This methodology was demonstrated on three CFD products that can be used for fire modelling purposes. The proposed procedure involved two phases. Phase 1 allowed comparison between different computer codes without the bias of the user or specialist features that may exist in one code and not another by rigidly defining the case set-up. Phase 2 allowed the software developer to perform the test using the best modelling features available in the code to best represent the scenario being modelled. In this way it was hoped to demonstrate that in addition to achieving a common minimum standard of performance, the software products were also capable of achieving improved agreement with the experimental or theoretical results. A significant conclusion drawn from this work suggests that an engineer using the basic capabilities of any of the products tested would be likely to draw the same conclusions from the results irrespective of which product was used. From a regulators view, this is an important result as it suggests that the quality of the predictions produced are likely to be independent of the tool used – at least in situations where the basic capabilities of the software were used.

The second issue raised from FIREDASS was addressed by utilising Parallel Processing techniques on office based computer equipment. Parallel Processing has been used for many years in the field of computational modelling including fire modelling. Parallel processing distributes the computational task over a number of processors and therefore allows computational problems to be solved in a shorter timeframe essentially by utilising more computational power. The majority of this work has focussed on the use of specialised proprietary hardware generally based around the UNIX operating system. The majority of engineering firms that would

benefit from the reduced timeframes offered by parallel processing rarely have access to such specialised systems. However, in recent years with the increasing power of individual office PCs and the improved performance of Local Area Networks (LAN) it has now come to the point where parallel processing can be usefully utilised in a typical office environment where many such PCs maybe connected to a LAN. Harnessing this power for fire modelling has great promise. Modern low cost supercomputers are now typically constructed from commodity PC motherboards connected via a dedicated high-speed network. However, virtually no work has been published on using office based PCs connected via a LAN in a parallel manner on real applications. The SMARTFIRE fire field model was modified to utilise multiple PCs on a typical office based LAN. It was found that good speedup could be achieved on homogeneous PCs, for example for a problem composed of ~100,000 cells would run on a network of 12 PCs with a speedup of 9.3 over a single PC. A dynamic load balancing scheme was devised to allow the effective use of the software on heterogeneous PC networks. This scheme also ensured that the impact of the parallel processing on other computer users was minimised. This scheme also minimised the impact of other computer users on the parallel processing performed by the FSE.

Contents

1 Introduction

Traditionally Fire Safety Engineers (FSEs) tend to work with prescriptive regulations. These consist of a set of rules that if followed will ensure that the design is compliant and should achieve acceptable safety limits by specifying exits, number and location of sprinklers, fire doors, etc. Approval authorities follow prescribed codes when inspecting systems with a checklist (or tick box) type approach. Prescriptive codes are built up over time and represent our experience and knowledge of past fires under a set of "known" conditions. The problem with prescriptive codes is that they are not based on fundamental science. This leads to difficulties in applying codes to novel building designs outside the regime of experience. As the prescriptive codes are a simple set of applied rules they are unable to quantify a level of safety.

The current trend in fire safety is towards performance based codes. For instance, performance based codes have been adopted in the UK since 1993. In performance based codes the safety requirements are stated as goals rather than as absolute requirements. Performance based codes are scientifically based but the safety analysis conducted by the FSE is also more complex than would be necessary if a prescriptive based code was applied. This allows engineers to be more flexible in their designs but these designs need to be at least as safe as would have been obtained using a prescriptive code.

A typical example of a prescriptive code might state that "*the maximum distance to a building exit shall be less than 100 feet*". In a performance based code the equivalent requirement would read as "*all occupants must have time to reach a safe place*". The performance code requirement has the advantage of applying whether the occupants are high school students or nursing home residents. In a performance based environment the prescriptive code's checklist approach is not applicable and a more sophisticated risk analysis is necessary. Computer models are a tool for assessing the performance of a safety system. With the advent of high performance Personal Computers (PCs), computer based fire modelling techniques have become increasingly accessible to FSEs and approval authorities. In fire modelling there are

two basic computer models available, these are zone modelling and **C**omputational **F**luid **D**ynamics (CFD).

Zone models are based on empirical data and can produce useful results very quickly. This approach divides the compartment into a number of distinct zones in which the composition and temperature is considered to be relatively uniform within each zone. In the simplest model two zones are used. These are the "hot layer" where the hot gases would reside near the ceiling and the "lower layer" which represents the rest of the compartment. In each zone variables are assumed to be constant but can vary with time. Conservation equations for mass and energy are derived in the form of ordinary differential equations that can then be solved numerically. Zone models can also include more advanced effects of the plume, radiation, convective energy losses through walls and the fire spread over solid fuel surfaces [Kar1992, QRJ1995]. CFAST is an advanced zone model developed by National Institute of Standards and Technology (NIST) capable of predicting the fire environment in a multi-compartment structure [JFPR2000]. CFAST also has the capability to be extended by the user to include extra physical phenomena.

However, as more complex problems are addressed the weakness of the zone model approach becomes apparent. For a number of fire situations determination of a suitable zoning system may not be apparent. The model also has to include a number of empirical relationships and constants that may not be valid in certain situations. Furthermore the underlying assumption that the uniform fire characteristics such as air temperature, density and smoke concentration are held in each zone may break down in complex fire scenarios [Gal1989].

The second type of fire model available to the FSE is the CFD based fire field model, which allows far more complicated modelling but requires far greater computing resources than zone modelling. This has become increasingly more attractive as computing power becomes more affordable. Fire field modelling is more versatile and requires less empirical relations due to the model being based on more fundamental physics. The fluid flow (air) is described by a set of three-dimensional, partial differential equations. These equations include the continuity equations, the three momentum equations, the enthalpy equation, equations for turbulence modelling and

any other conserved quantity that the user may be interested in. A CFD model can solve these equations numerically giving a set of values at discrete points in time and space. CFD based fire field models have been used to model fires in a number of different environments.

Fire field modelling software has been used to model actual fire accidents for forensic examination. The use of such software has enabled insights to be made into how and why the fire spread and also allows investigation of how safety measures could be implemented to reduce fatalities or structural damage. Examples include the King's Cross Underground Station fire [SWJ1992], Gothenburg dance hall fire [YH2001], fire in a school [CF2001], a fire in a Cash and Carry warehouse [Cam2001] and a fire in a residential enclosure [MV2000].

Fire modelling software is more frequently required for the design process to test the fire safety of a particular design. This includes high rise buildings [GBH1996], aircraft [GGPM1998a], atrium [Cox1990] and power stations [Huh1989].

FIREDASS [Fir, Goo2000, MGG+2000, GGPM1998a, GGPM1998b, Odi1999, OM1998, OM1998, KB1997, Ker1997b, KSBM2000, MGPG1998, GGPM1999] was motivated by the need to find a suitable fire suppression agent to replace halon in aircraft cargo bays [Mau1990]; water mist was investigated as a possible replacement. This ambitious project resulted in the creation of a complex, state of the art, CFD based fire computer model [MGG+2000, GGPM1998a, GGPM1998b, Fir] that was capable of fully modelling the interactions between a fine water mist, the radiation field, oxygen concentration, and the fire load within an enclosure. The interaction between the fire and the mist is modelled using a suppression criterion developed by SINTEF and implemented by the author. In addition the model allowed the use of virtual sensors to monitor physical properties which could then be used to control the water mist suppression using a range of possible control algorithms.

An outcome from this work was that despite a suitable model being developed, the FSEs were unlikely to use the FIREDASS fire field model due to: -

- Lack of confidence in the fidelity of the results.

- The cost, in terms of both time and money, of running CFD based fire models.

The lack of trust in CFD methods is not limited to FSEs. The American Institute of Aeronautics and Astronautics reported "*One of the primary factors in the rate of growth of CFD as a research and engineering tool in the future will be the level of credibility that can be developed in the simulations produced*" [AIAA1998].

This outcome has motivated the work of this thesis. This work addresses: -

- The lack of confidence FSEs may have in a particular CFD model and the CFD fire modelling methodology in general.
- The time taken to run a CFD fire simulation by decreasing the wall clock time required ***using inexpensive and readily available computer equipment.***

1.1 Increasing trust and acceptance of CFD for Fire Modelling

Although CFD can be an extremely useful tool for a FSE, the problem that now faces the FSE is whether they can trust the results produced by CFD software. In order to lend confidence to CFD software there are two basic checks available, verification and validation. Verification is "*The process of determining that a model implementation accurately represents the developer's conceptual description of the model and the solution of the model*" [AIAA1998]. This is normally done by comparing the model to some known analytical solution. Validation is "*The process of determining the degree to which a model is an accurate representation of the real world from the perspective of the intended uses of the model*" [AIAA1998]. This is normally performed by comparison with benchmarking data derived from experiment. Similar definitions have been put forward by the International Standards Committee (ISO) for assessment of fire modelling software in the ISO/TR 13387-3 document [ISO1999].

Within the fire modelling community, testing of fire field models has usually completely ignored the underlying CFD engine and focussed on the fire model. Thus, when numerical fire predictions fail to provide good agreement with the benchmark standard, e.g. the experimental results, it is not certain if this is due to some underlying weakness in the basic CFD engine, the fire model or the manner in which

the problem was set-up (i.e. questions of user expertise). Furthermore, the case that was being used as the benchmark/standard was usually overly complex or cannot be specified to the precise requirements of the modellers. All of this is often to the benefit of the code developer/user as it allows for a multitude of reasons (some may say excuses) to explain questionable agreement.

Furthermore, what fire modelling testing has been undertaken was usually done in a non-systematic manner, performed by a single individual or group and is generally based around a single model. Thus it is not generally possible for other interested parties to exactly reproduce the presented results (i.e. verify the results) or to apply the same protocol to other models. This makes verification of the results very difficult if not impossible and the comparison of one model with another virtually impossible.

A set of benchmarks/standards is required which allow a judgement to be made on whether or not a particular piece of software is appropriate for a particular scenario. Currently there is no objective procedure that assists an approval authority in making such a judgement. The approval authority must simply rely on the reputation of the organisation seeking approval and the reputation of the software being used. In discussing this issue it must be clear that while these efforts are aimed at assisting the approval authorities, there are in fact three groups that are involved, the approvals authority, the general user population and the model developers. Ideally, the proposed standards/benchmark should be of benefit to all three groups. In proposing the standards/benchmark, it is not intended that meeting these requirements should be considered a SUFFICIENT condition in the acceptance process, but rather a NECESSARY condition. Finally, the benchmarks are aimed at questions associated with the software, not the user of the software.

1.2 CFD as a practical and economic tool for a Fire Safety Engineer

From a practical engineering viewpoint, producing reliable results from a fire field code is only useful if these results can be produced within in a reasonable and economic timeframe. During the FIREDASS project, circa 1998, a typical runtime for useful engineering application (e.g. the cargo bay model) was ~140hrs running on a 450Mhz DEC Alpha workstation. Despite the relatively high performance of modern

day PCs (e.g. Pentium III/IV or AMD Athlon), possibly three times faster and six times cheaper than the DEC Alpha of 1998, there will always be a demand for more computational power to run more complex and computationally intensive cases in the shortest possible time. CFD based fire field codes are exceptionally computationally expensive compared to most pieces of software that will be run by a FSE within the design engineer's environment. There is always a demand for faster turnaround of fire modelling CFD simulations. This issue is partly being addressed by the improvements in PC technology with faster computers being produced. However, this trend can not continue indefinitely and even if this were possible there will always be demand for even more processing power.

An obvious strategy available to reduce runtime is to improve the speed of the software. This can be achieved by using improved algorithms for the overall solving strategy or for the individual solvers for each dependent variable. Many numerical solvers exist to solve the matrix equations formed by CFD formulations. These include the CGM (Conjugate Gradient Method), BICG (BIConjugate Gradient), SOR (Successive Over Relaxation), JOR (Jacobi Over Relaxation) and TDMA (Tri Diagonal Matrix Algorithm) algorithms. These methods all have advantages and disadvantages when compared to one another and the most appropriate algorithm is largely problem dependent. In the SMARTFIRE [EGP+1999] software the use of the JOR and SOR solvers is generally recommended due to their greater numerical stability compared to the other methods.

Some speed up efficiencies can be obtained through optimised program coding strategies. A simple example demonstrates two methods to evaluate the same expression ($A = \cos^2\theta + 2\cos\theta + 5$). The following line of computer source code:-

```
A = cos(theta) * cos(theta) + 2 * cos(theta) + 5
```

Could be replaced with the following code fragment:-

```
C = cos(theta)
A = (C + 2)*C + 5
```

The first obvious improvement is `cos(theta)` is only evaluated once. The second improvement is that the first example consists of 2 multiplication and 2 addition operations. The second example requires 1 multiplication and 2 additions saving on 1 multiplication operation.

Group solving [Ewe2000] is another means of targeting the computational effort where it is required in the domain and reducing this effort in more stagnant regions. This method has lead to a 37.3% speedup [Ewe2000] for certain cases compared to the time normally taken by the CFD code with no detriment in the predicted values. Dynamic control of the solution procedure can also reduce the run time of a CFD code by optimising the relaxation factors of the solvers to reduce the number of iterations that are required to generate a converged result [JEG+2001].

The demand for increased computational power can also be met using parallel processing. Parallel processing techniques have been around for many years but have normally been the preserve of large-scale organisations and academic institutions due to the costs associated with purchasing the specialist hardware. Fire modelling has been performed using parallel processing techniques on specialised hardware in the past [IG1992, GI1992, GI1993, Stu1997]. These costs can be prohibitive for many FSEs, however with the current state of the personal computer market it may be possible to usefully utilise these techniques with equipment they already possess. The use of parallel processing methods is not an alternative to the above or other software improvements but offers additional speed improvement.

1.3 Research Questions

The work of this thesis begins with the tackling of a practical engineering problem, the interaction of water mist with fire and the eventual suppression of the fire. As a result of this project, the direction of the research broadens to address two main issues associated with the practical application of CFD fire modelling, that of the reliability of CFD based fire simulation results and the expense, both in terms of time and money, to perform CFD based fire simulation.

Can a suitable benchmark/standard for CFD based fire-modelling codes, which is free from manufacturer and user bias, be developed?

- Which types of problem should be selected for the benchmark process?
- What methods will reduce / eliminate user bias? Typically bias play a part in most validation processes as developers are allowed to use their own judgement in simulating a fire modelling scenario.
- How can software developers demonstrate features that differentiate their field model from other field models that exist?
- Can CFD methods in general be demonstrated to be useful for fire modelling?

Can parallel processing techniques be usefully applied to standard office based PCs to increase the computational power available to a FSE for the purposes of CFD based fire modelling?

- How much faster will the code run on a parallel processing system? Ideally this would be to the sum of the processing power of the computers involved in the parallel computation. In reality this is not even achieved on dedicated parallel processing hardware but can enough power be extracted from a network of PCs connected via a conventional LAN to make a useful resource for a FSE.
- What problem sizes can be tackled on a parallel processing system? Ideally this would be linearly related to the sum of the memory available on all the PCs, i.e. double the memory allows the problem size to be doubled.
- What are the limitations of PC parallel processing? At what point would the use of parallel processing become unprofitable. What is the potential future of using parallel processing on PCs with future developments of PC based technologies.
- Can methods be devised to efficiently take advantage of a network of heterogeneous (non-identical) PCs? A FSE's engineering environment may consist of non-identical computers that have no known performance benchmark. Ideally the parallel processing software itself should determine the performance of each of the computers used for parallel processing.
- Can performance improvements be made to the serial code without affecting the object-orientated structure of the code? There is little point in parallelising software if the performance can be easily speeded through modification to the serial code; this could include algorithm and implementation details.

- Can parallel processing be implemented to maximise the potential processing power without adversely affecting other computer users? Within a FSE's office environment other users will be using computers that could potentially be used as part of a parallel processing job. Methods need to be devised that ensure that neither the FSE using parallel processing of other computer users are adversely affected by each other.

1.4 Objectives

The objectives of this research were to answer the above research questions about the performance of CFD based fire modelling codes.

1.4.1 Can a suitable benchmark/standard for CFD based fire-modelling codes, which is free from manufacturer and user bias, be produced?

The objective here is to create a standard/benchmark procedure that would be useful to FSEs, approval authorities, and CFD code producers to assess the suitability of a particular code for fire modelling. A number of suitable test cases need to be selected to appropriately test the basic features of the CFD software and the application of that software to fire modelling. A methodology needs to be devised to eliminate, or at least reduce, the influence of the user in setting up and running the test cases. An appropriate scheme is also required to allow users / developers to demonstrate that their software has qualities that differentiate it from other software available for fire modelling. This work should also help demonstrate that not just particular software products but the underlying CFD methodology can be trusted.

1.4.2 Can parallel processing techniques be usefully applied to standard office based PCs to increase the computational power available to a FSE for the purposes of CFD based fire modelling?

The objective is to create a parallel CFD based fire field code that would run on standard office based PC equipment and be of benefit to a FSE. This parallel code should be a faithful reproduction of the original serial code. The code will be tested on a number of different possible PC network configurations, including homogeneous and heterogeneous networks, where a homogeneous network is composed of

'identical' machines and a heterogeneous network is composed of different machines. The performance of the code on a number of different test cases will be analysed to show the potential benefits and shortcomings of using parallel processing techniques on office based machines. As the code is designed to work on heterogeneous networks, a dynamic load-balancing scheme will also be devised to intelligently share the computational load amongst the computers. This load balancing scheme will also work interactively ensuring that computer resources are used fairly between the FSE and other possible computer users. An investigation of the source code will be undertaken to try to improve the performance of the serial code that would also improve the performance of the parallel code. The code should also allow maximum utilisation of the available memory of the machines in the parallel network.

1.5 Structure of Thesis

Chapter 1. This chapter has enumerated the research questions posed about the performance of Fire modelling CFD codes and the objectives of the work during this investigation.

Chapter 2. This chapter gives a more in depth background to the performance issues raised in chapter 1. An overview of fire modelling is given with reference to work related to the FIREDASS project including water mist modelling. Then a review of material relevant to the validation, verification and benchmarking of CFD fire field models is given. The last part of this chapter will detail the work relevant to parallel processing on office based PCs.

Chapter 3. This chapter details the equations and numerical modelling used by CFD techniques to solve fire modelling problems. The general conservation equation is detailed and methods of solving this equation by discretisation will be described. The SIMPLE solution method for the coupled equations of heat, mass, momentum and pressure will be detailed. Turbulence modelling will be described both in terms of the Reynolds average approach and for Large Eddy Simulation.

Chapter 4. This chapter will feature a practical example of fire modelling. The author performed this work as part of the FIREDASS project. The author was involved in the

generation of a suitable fire and suppression model. The model was intended to help FSEs in the design of a water mist suppression system. The model will be compared against experimental data as part if the verification process. The full model will be demonstrated with a suppression control algorithm devised by Cerberus Guinard. Two important issues to arise from FIREDASS were the reliability of CFD methods and the time taken to run a CFD simulation model. These issues provided the direction for the rest of the work contained in this thesis.

Chapter 5. In this chapter the procedures and test cases developed for establishing fire modelling standards, to raise confidence in CFD based fire models, will be detailed. Five basic CFD problems and five fire scenarios have been selected and will be described. The two phase nature of the standard will be detailed. Phase 1 of the standard requires all the code to reach a common standard by utilising a common specification. Phase 2 is free format allowing the modellers to model the problems using the more advanced features of their models if they wish.

Chapter 6. The results of applying the phase 1 standards, established in Chapter 5, on three commercial CFD codes will be given. These three codes were PHOENICS, CFX and SMARTFIRE. The results of applying the phase 2 standards, established in Chapter 5, using the SMARTFIRE fire modelling code are described.

Chapter 7. Details of the parallel CFD code developed to be used on conventional office hardware will be given. Both the shared memory and distributed memory methods of parallelism will be investigated. The domain decomposition strategy was employed to distribute the fire modelling problem between the computers. This introduces the concepts of load balancing and re-numbering which are explained in detail. The effects of parallel processing on the implementation of the SMARTFIRE CFD code are described.

Chapter 8. Results produced by the parallel CFD codes developed in the previous chapter will be given. Initially homogeneous networks are considered and illustrate the possible time savings that can be achieved using parallel processing techniques on conventional PCs. A dynamic load balancing mechanism is developed that allows a FSE to maximise the potential processing power available from a heterogeneous

network of computers. The dynamic load balancing mechanism also accounts for other computer users exploiting the same resources as the FSE and results from testing this will be given. Memory usage and the future of parallel processing will be explored. A case study on the possible benefits of parallel processing to a FSE is performed.

Chapter 9. Conclusions from the work done will be offered. Have the questions raised in chapter 1 been answered?

Chapter 10. Suggestions for areas of further work will be suggested.

2 Background and Literature Review

In this chapter the background to the performance issues raised by FSEs in chapter 1 will be detailed. Initially a brief review of CFD fire modelling is given. The FIREDASS model is described as a practical example of fire modelling in a later chapter (chapter 4) and therefore material relevant to water mist modelling is also briefly described. Literature directly relevant to the main issues of this thesis is then described. The main performance issues are the benchmarking of CFD based fire modelling codes and the use of parallel processing on office based equipment to reduce wall clock times of CFD based fire modelling codes.

2.1 Fire Field modelling

Simulation of fire growth and spread within enclosures is a difficult task. The computer model must contend with the interaction between turbulence, gas-phase combustion, solid-phase combustion and radiation. Over the past 20 years considerable effort has been expended in developing fire field models capable of predicting the development of hazardous conditions within fire enclosures [MC1984, Gal1989, SWJ1992, Cox1995, JGP1997, YH2001, MBR+2000, LSH+1999].

Many of the commercially available CFD based fire simulation models available utilise similar technologies, as described in chapter 3. These include PHOENICS [RST1983], JASMINE [CK1986], SOFIE [Rub1997, LMR1997, SRM1999], SOLVENT [LSH+1999] and SMARTFIRE [GKP+1999]. All these codes numerically solve the Reynolds Averaged Navier Stokes equations of transient fluid flow using a variant of the SIMPLE algorithm and utilise a buoyancy modified k-ε turbulence model. They all include some form of radiation modelling.

Two CFD codes used for fire modelling that are not based on the SIMPLE algorithm are NIST's FDS code [MBR+2001] and ANSYS's CFX [CFX2003]. CFX does not use the SIMPLE algorithm, but rather a coupled algorithm [Raw1994, Raw1996, CFX2003].

PHOENICS is a general purpose CFD code, developed by CHAM, which has been applied to fire modelling [Wat1986, KMG+1994]. PHOENICS possesses a variety of turbulence and gaseous combustion models. It is able to solve problems using Body Fitted Co-ordinates (BFC) grids. Radiation is modelled using a six-flux radiation model. PHOENICS was also used as a basis of development for the JASMINE code.

JASMINE is a long established fire field model, developed by the UK Fire Research Station (FRS), with an extensive validation history in fire modelling [KHC1985, CKM1986, PWC1989, FKE+1993, MC1996, MKC1999]. It features a number of sub-models for combustion and solid body heat transfer. It has been applied to a number of scenarios including warehouses [MC1996], hospital wards [KHC1985] and an air-supported structure [PWC1989]. The model is limited to orthogonal co-ordinate meshes. Radiation is modelled using a six-flux radiation model.

SOFIE is a model developed by a European consortium including the UK FRS, Technical Research Centre of Finland, Swedish National Testing and Research Institute, CSTB, Lund University, Health & Safety Laboratory and the Home Office Fire Safety Engineering Group and Cranfield University. This model incorporates the use of non-orthogonal curvilinear coordinates and also includes a laminar flamelet model and fire spread models. Radiation is modelled using the Discrete Transfer method.

SOLVENT is a model developed specifically for modelling fires in tunnels and includes special features such as the modelling of jet fans and other ventilation features of a tunnel. The model was developed from Phase IV of the Memorial Tunnel Fire Ventilation Test Program and has been validated against data from that program. Although the software is based on a staggered mesh discretisation it was unclear whether or not it could use BFC co-ordinate systems. Radiation is modelled using the six-flux model

CFX, developed by ANSYS Inc., is a general purpose package [RAW1994, RAW1996, CFX2003] that has been used for fire modelling. The CFX model has a number of features including unstructured or structured meshing, a coupled multigrid

solver, parallel processing, a number of gaseous combustion models (including EDM and flamelets), turbulence models (including LES and DES), radiation models (including Monte Carlo and Discrete Transfer), CHT (Conjugate Heat Transfer), and user Fortran utilities. CFX has been applied to fire problems, e.g. [SWJ1992, JEI+1997, Sin2000, SSEW2001, Sin2003], including the Millennium Dome and the King's Cross fire accident [SWJ1992]. Early versions of the software were named FLOW3D, CFDS-FLOW3D or HARWELL-FLOW3D, and were developed when the team was part of AEA Technology.

The Fire Dynamics Simulator (FDS) model [MBR+2001] produced by the National Institute of Standards and Technology (NIST) and is freely distributed. The FDS software uses an explicit predictor-corrector method and uses Large Eddy Simulation (LES) [Sma1963] to model the turbulence. It features a novel thermal element method for spreading heat from a fire source. It also contains combustion and radiation models.

The majority of practical fire modelling applications has been concerned with the spread of heat and smoke in complex structures and so combustion has either been ignored or greatly simplified. In cases where combustion is ignored the fire is treated as a simple prescribed volumetric source of heat and smoke. While this approximation may appear crude it can produce good agreement with experimentally derived temperature measurements [KMG+1994, KGHP1994, WJG+2000] for room fire scenarios.

Although the volumetric approach can be useful sometimes it is necessary to more accurately model the combustion process. This is particularly important when trying to model near field fire plumes [Kum2001, Jia2001]. The combustion model used within the SMARTFIRE field model will be described in section 3.8. Generally, if combustion is included, it is approximated using relatively simple one-step reaction mechanisms [YCL1995, KGC1991] for liquid or gaseous fuels such as methane.

While solid fuel pyrolysis models [YH1996; JGP1997; JGP1999] have been developed and incorporated into fire field models to simulate fire spread over flat solid fuel surfaces within compartments, it is difficult to use these pyrolysis models for complex mixtures of fuels such as may be found in aircraft cargo holds full of

luggage or in warehouse structures. In most practical engineering calculations, the conventional handling of this difficulty is to use the volumetric heat source model. In this it is assumed that the total heat released from the fuel source is released within a prescribed volume and the heat release rate is provided from experimental data or empirical relations. In other words, the model assumes that the mass loss due to the fuel burning, and the ensuing chemical reaction between the combustible pyrolysis products of the mass loss processes and oxygen, only take place in the prescribed region and outside this region there is no chemical reaction. In practice it is difficult to estimate the size and location of the prescribed heat release region even though the size and location of the fuel source is generally known. Furthermore, the combustible pyrolysis products released from the fire source may be transported outside the original region of the solid fuel and combustion may also occur outside of this region.

In these types of models, in addition to representing the heat release rate as a source term, it is also possible to treat the other fire products such as smoke, CO, etc as an imposed time dependent volumetric source term.

2.2 Water mist modelling

The modelling of water mist is a multiphase problem and could be modelled using either an Eulerian-Lagrangian or an Eulerian-Eulerian methodology within a CFD framework. In the Eulerian-Eulerian method, the mist is represented as an interpenetrating continuum in the air phase. A number of phases would be required to represent a range of different droplet sizes involved in the mist, each of these phases requiring at least a volume fraction, three momentum and enthalpy equations to represent it. These equations could then be solved and coupled using the IPSA method [Spa1983]. This method has been used in the past to model sprinkler systems [HGM1989, Has1996] and fluidised beds [PC1989, GDJ1990] for example.

Hoffman et al [HGM1989] utilised the Eulerian-Eulerian method to model sprinklers in a hospital ward. Only one phase was used to model the water droplets, due to the limitations of the computing resources at that time. This assumption is reasonable for large droplets with small changes in diameter. This would not be the case with a fine water mist where evaporation is a major component in cooling the fire environment.

Therefore this method has the disadvantage of requiring a lot of computation due to the number of phases that would be required to reasonably represent the range of droplet sizes within a mist.

Prasad et al [PLK1999] utilised the Eulerian-Eulerian methodology to model the suppression effects of a fine water mist on a theoretical small-scale ethanol pool fire. The domain measured 9.5cm in width and 64.0cm in height; the flame heights were approximately 2-3cm high. The pool fire was modelled using a sophisticated combustion model that allowed the suppression to be fully modelled. They found that the finest water mist gave the best suppression of the pool fire. The model was two-dimensional and they used five droplet size ranges to represent the mist. Their model typically took 10 hours to run on a single processor of a Cray C90.

In Eulerian-Lagrangian modelling the air phase is treated as a continuum but the mist is treated as a number of discrete particles that are tracked through the air phase. The main advantage of this approach is the ease of representing droplet sizes, velocity and temperature distributions compared to the Eulerian-Eulerian approach. The droplet phase is coupled to the air phase using the PSIC method of Crowe et al [CSS1977]. This method has been used to model water sprays from sprinkler systems [Alp1984, Alp1985, CF1991, Bil1993, Nam1996, KHL1997, Nam1999, HKKX2002]. A mist would consist of the order of 10^7 to 10^8 individual particles and would make it impossible to model such a system. To make the problem tractable a single particle actually represents an ensemble of thousands of similar particles. Other workers have also used the Eulerian-Lagrangian modelling approach in water mist suppression work [SSEW2001].

Sinai et al [SSEW2001] have developed a generalised fine water mist model which has been applied to halon replacement on warships in collaboration with the Warship Support Agency. Their model differs in a number of ways to the FIREDASS model although both are based on CFX and both utilise a PSIC [CSS1977] method to model the mist. They model radiation using a discrete transfer method; FIREDASS uses a six-flux radiation model. From the published material it is unclear how their misting nozzles work. In the FIREDASS work the nozzles are two phase with the water being forced through the nozzle with a gas propellant, this has important implications on

how to model the nozzle. Sinai et al model combustion and pyrolysis of a heptane pool fire allowing the suppression to be modelled by the reduction in pyrolysis rate. FIREDASS uses a more simple heat release model and empirical criterion to determine extinguishment based on experimental data for propane, kerosene and cardboard box fires [Wig1998]. Sinai performed a mesh sensitivity analysis using a very coarse mesh (9212 hexahedral cells) and a fairly fine mesh (101304 hexahedral cells). With a coarse mesh the fire was extinguished by the model, with a fine mesh the fire was not extinguished but controlled by the model. It was not surprising to find mesh sensitivity, a common feature of CFD codes, with the finer mesh being closer to reality as experimentally it was found that the fire was not extinguished but controlled by the water mist.

2.3 Assessment of Fire models

A number of publications concerning the assessment of fire models have been previously written and will be described in the following section. The purpose of the work in this thesis is not to try to address the whole assessment issue but to concentrate on a procedure that will reduce, and hopefully eliminate, user bias from the verification and validation of a CFD based fire model.

According to Kumar and Cox [KC2001] an evaluation of a fire model should address the following three aspects:

1) Scientific content, e.g. the representation of the important physical and chemical processes, the formulation of a mathematical problem and its solution in the model
2) Validation and Verification performed on the model using experimental and analytical data.
3) Practical realisation of the model, this includes Human Computer Interaction (HCI) issues including the user interface. This could also include software engineering issues

This is essentially the approach followed by a number of workers [ISO1999, Bea1997]. In this thesis the primary interest is in the verification and validation of CFD based fire models in an objective manner from the point of view of an approval

authority. Points 1 and 3 are beyond the scope of this thesis; these issues are described by other workers [ISO1999, Bea1997].

2.3.1 Verification and Validation of fire models

In the literature there are three broad categories of work related to the verification and validation of fire models.

(1) Comparison of a fire model with experimental / analytical data.

(2) Methodologies for validating and verifying fire model performance.

(3) Uncertainty.

These categories are not rigidly defined and many publications may refer to more than one category.

2.3.1.1 Comparison of fire models with experimental data

Beard [Bea2000] has described three forms of fire model comparison with experimental data. These are *A Priori*, *Blind* and *Open* comparison with experiment.

In *A Priori* comparison the modeller has no access to the experimental test results and no access to experimentally derived quantities such as mass loss rate or heat release rate. Furthermore there can be no *a posteri* adjustment to input parameters. The modeller must choose an appropriate set of parameters before details of the experimental results are known.

In *Blind* comparison the modeller has no access to the experimental test results but may have access to data such as mass loss rates or heat release rates. If this type of data is used it should be explicitly stated. As with *A Priori* testing the input parameters must be set before the comparison results are known. In some circumstances it may not be possible to test in an A Priori manner and therefore blind testing has to be adopted. For example it would be impossible to model fire induced flows from a gas burner without knowing the fuel supply rate.

In *Open* comparison the modeller is assumed to have full knowledge of the experimental test results. This may influence the modeller's choice of input parameters.

Generally speaking most of the comparison performed is *Open* with very little performed in the *Blind* and *A Priori* categories. This is hardly surprising as most of the comparison / validation work of a fire model is conducted by the author of the fire model. Beard describes the need for more A Priori and Blind comparison as these are more representative of a fire model's use by a FSE.

2.3.1.1.1 Examples of open testing

Validation and verification of individual CFD based fire models occurs in some form or other. However a lot of this work is carried out by the code vendors themselves and must therefore be judged with caution, as a vendor is hardly likely to publish any adverse results. Much published work in validation is concerned with comparing a single CFD software product with experimental data. These have included tunnel fires [FV1994, KC1985, Rho1996, YCL+2001, LSH+1999], forced ventilation enclosures [CK1986], naturally ventilated enclosures with fire induced flows [KGHP1994, BK1996, Sat1985, YCL+2001], aircraft cargo bays [GGPM1998a] and large scale enclosures [PWC1989, Tub1994, MC1996]. Generally this work lacks some of the details necessary to verify their findings.

2.3.1.1.1.1 Independent comparison of more than one CFD based fire model with experimental data

Some independent testing has been performed in the past comparing the results of more than one model on the same problem [KMG+1994, Cum1991, PB1996]. Kerrison et al [KMG+1994] compared PHOENICS and Flow3D (now known as CFX) against the experimental data of the Steckler Room experiments [SQR1982]. Both of the models were set up using the same computational mesh to try and ensure consistency in the comparison. However they did not solve the problem in the same way. The Flow3D (CFX) model was solved using 200 time steps of 1s. After 200 time-steps it was determined that the simulation had reached a steady state. The PHOENICS model was simulated as a steady state problem from the beginning so no time stepping was used. Both approaches were valid but demonstrate that expert users may adopt different approaches. They concluded that both models gave reasonable agreement with the experimental data. Pehrson [PB1996] compared five CFD codes;

Flow3D (now known as CFX), TASCFlow, PHOENICS, Jasmine and SOFIE against the Steckler room fire experiments. He found good agreement with the central door line velocity profile, however there were differences of about 30% for the temperature profiles.

2.3.1.1.2 Examples of A Priori and Blind Testing

There has been very little in the way of a priori and blind testing carried out on CFD based fire model. Some examples of a priori and blind testing are given below.

2.3.1.1.2.1 Review of Beard – "Evaluation of Deterministic Fire Models"

During 1989 and 1990, Beard performed an evaluation of deterministic fire models [Bea1992] sponsored by the UK home office. Beard tested four fire models. Three of these were zone models, ASET, HAZARD and FIRST. The remaining model was a CFD based fire model, JASMINE. The evaluation was conducted in two parts, a qualitative and a quantitative study. The qualitative study examined the assumptions of the model, the limitations of the model, the input required, the output that would be produced and a review of any existing literature of model comparison with experiment. The quantitative phase applied the models to three different fire scenarios. These fire scenarios were:

- A single room fire – polyurethane foam slab fire in domestic sized room
- House fire – armchair fire in lounge / dining room of 2-storey house
- Department Store fire – displayed furniture fire in centre of large sales area.

The models were run to try to predict the air temperature, smoke obscuration and carbon monoxide levels as the fire progressed. These models were run in an a priori manner. However Beard found it difficult to draw any firm conclusions about the models from his work due to difficulties associated with the experiments (section 2.3.1.3) and difficulty in determining the parameters that needed to be entered into the models in particular the zone models. He concluded that a Model Evaluation Group (MEG) should be established to approve (or not) models for particular applications. He further concluded that a methodology needed to be devised that could be applied to the models for evaluation purposes, see section 2.3.1.2.2.

2.3.1.1.2.2 Review of Freitas – "CFD Triathlon"

Although this work is not directed at fire modelling it does raise issues applicable to fire modelling. The CFD triathlon [Fre1995] was designed to appraise users and potential customers of the capabilities of commercial CFD codes. The problems simulated were designed to be fairly simple, unambiguous and were rather academic in nature. Despite the simple nature of the problems involved Roache [Roa1998] noted that *some of the discrepancy [between model prediction and experiment] can be attributed to the skill of the modellers as well as codes {CFD programs}.*

2.3.1.1.2.3 Canadian CFD society test case

Again this work is not directly relevant to fire models but is applicable to CFD models in general. This was intended to be an 'a priori' comparison of commercial and research CFD codes. There were a number of problems with this comparison. First of all the problem was underspecified and required experience and intuition of the modellers to fully specify the problem. Secondly, some of the modellers had identified that the results of the test case had actually been previously published. Rather unsurprisingly their results proved to be the best. Generally the model agreement with experiment was poor and agreement between different modellers using the same turbulence model was also poor.

2.3.1.1.2.4 Review of CIB W14 Scenario B

The CIB W14 is an informal workgroup consisting of informal group of approximately 75 voluntary fire expert members from 30 different countries working towards the use of performance based codes in building regulations. CIB was an abbreviation for "*Conseil International du Bâtiment*" which is now known as "*International Council for Research and Innovation in Building and Construction*". The Round Robin for Code Assessment Scenario B conducted by CIB W14 [HK1998] was an attempt to compare various computer models by blind and open testing against 3 experiments. This activity involved 9 zone models and 2 CFD models. The modellers were provided with all the details of an experiment including the geometry of the test rig, measurement locations, heat release rate, material properties, and ambient conditions. From this information, the modellers performed a simulation with no prior knowledge of the results. Scenario B consisted of 3 sub-

scenarios. Sub-scenario B1 represented an industrial building with burning material concentrated in one location. B2 represented a shop or office with a thin layer of burning material uniformly distributed inside the compartment. B3 was an example of a fire that spread from one source to another in a low space, which could represent fire spread from one car to another in a tunnel.

No final report from this activity has been generated to date but the draft [HK1998] reported the following four conclusions:-

1) *'All of the computer codes could reproduce, even blindly, the main qualitative features of the experiments'*
2) *'Quantitatively, there were deviations ranging from typically 20% up to a factor of 2.'*
3) *'All of the codes had features that indicated a discrepancy with the experimental data in the blind simulations, but which could be improved during the open round by choosing alternate submodels and/or changing optional parameters.'*
4) *'Where several persons used the same code, the dependence of the results on the user was demonstrated. It was indicated very clearly, that the user is the most critical link in the chain of using computer fire simulation models for fire safety engineering. This was true even though this group represented code developers, and other well educated fire science/engineering practitioners. The effect is expected to be much more pronounced when the whole group of computer code users is considered. Therefore, actions should be started to develop user interfaces to optimize the use of computer fire codes. This might include data banks and other auxiliary tools to help select appropriate models, input data and prevent the use of unsuitable models or data and prevent the use of unsuitable models or data. The results of these comparisons are not sufficient to provide much guidance on determining these limits. But they should start to motivate the development of simple user friendly, mainly graphical, tools for user interfaces.'*

Points 3 and 4 clearly raise concerns over the validation method because of problems of the user. Why were the initial submodels and optional parameters not chosen correctly for the blind testing? Surely 'tuning' results *a posteriori* should be regarded with a degree of suspicion? The bias caused by the user is problematic making the

cause of a computer model's failure difficult to determine, i.e. was it the user or the software at fault. These issues need to be addressed in any future validation exercises.

This activity probably also suffered from a competition orientated approach. Each group was probably more interested in proving that their software was best, than in reaching any agreed standard.

Miles et al [MKC1999] describes the comparison obtained between JASMINE and the CIB W14 tests. They report that their work is probably the first truly blind test of a CFD fire model. They concluded that the model was fit for the purpose of predicting gas phase conditions to better than 15% in a flashed over enclosure of the dimensions used within the study. They further described that the one dimensional solid heat conduction approximation used in the model was inadequate for the scenario modelled.

2.3.1.1.2.5 Review of Dey - 'Evaluation of Fire Models for Nuclear Power Plant Applications: Cable Tray Fires'

This document [Dey2002] detailed the results of the first task of an international collaborative project concerned with the use of fire modelling software within the nuclear industry. There were twenty two organisations from six countries participating in the collaborative project.

The first task was to evaluate the capability of fire models to analyze cable tray fires of redundant safety systems in nuclear power plants.

The following procedure was adopted for the benchmark exercise:-

- *Analysts should discuss and agree on the input data for the various fire codes that will be used in the benchmark exercise. The goal is to analyze the same problem and minimize the variation of results due to the different input parameters*
- *The form of the results to be compared should be agreed upon by the participants prior to the commencement of the exercise.*

- *Developers of the fire codes, and those not involved in the development of the codes, can conduct the code analyses for the benchmark exercise*
- *Blind simulation will be conducted, i.e., each analyst will independently conduct his or her analyses. The results will be shared between the participants when all the analyses by participants have been completed and the results are available. The results will be shared between participants when all the analyses by participants have been completed and the results are available. The results will be simultaneously posted on the collaborative project web portal prior to a meeting of the participants.*
- *If desired, the same code (e.g. CFAST) can be used by different organisations since this will provide useful information on whether the results vary with different users. However, the same version of the code should be used.*
- *A series of benchmark exercises will be defined and conducted in this project. This will allow the evaluation of the full spectrum of fire model features and applications, and facilitate the formation of a comprehensive technical reference for users on the capabilities and limitations of current fire models.*

A variety of fire models - including zone, field and lumped parameter - were used to examine the cases. The following organisations participated in the benchmark exercise:-

Organisation	Code
Institute for Protection and Nuclear Safety, France (IPSN)	FLAMME-S (zone)
Nuclear Regulatory Commission / National Institute of Standards and Technology, USA (NRC/NIST)	CFAST (zone), FDS (CFD)
Gessellscaft fuer Anlagen-und Reaktorsicherheit, Germany (GRS)	COCOSYS (lumped param), CFX (CFD)
Electricite de France (EdF)	MAGIC (zone)
Building Research Institute / H. M. Nuclear Installations Inspectorate, UK (BRE/NII)	CFAST (zone), JASMINE (CFD)
Institut fuer Baustoffe, Massivbau und Brandsschultz (iBMB), Germany and GRS	CFAST (zone)
CTICM, France	MAGIC (zone)

To perform this task a series of benchmark cases derived from a single room, a representative emergency switchgear room for a Pressurised Water Reactor, were constructed. The benchmark was conducted in two parts. The objective of Part I was to determine the maximum horizontal distance between a specified trash bag fire and

a cable tray that results in the ignition of the cable tray. Part II examined whether a target cable tray will be damaged by a fire of a cable tray stack that is separated by a horizontal distance, d. Within both of these parts the effect of an open or closed fire door and a mechanical ventilation system were further examined.

According to the interpretation of the results by the international panel it was concluded that: - '*The international panel determined that the analyses of the results of the benchmark exercise demonstrated that current zone, CFD, and lumped parameter fire models provide a comprehensive treatment of most physical phenomena of interest in the scenarios analyzed. The results indicate that the trends predicted by the sub-models are reasonable for the intended use of the models for analyzing the specified scenarios'.* This statement seems hard to justify from the work performed in the benchmark exercise. Firstly, there was no real data supplied to validate any of the model predictions against. Secondly, the model predictions reported varied significantly between modellers.

The results from the cases from the various organisations displayed a large amount of variation. Furthermore there were significant variations within the types (e.g. CFD based models) of fire models tested. The results from the Part I base case are reproduced below in Table 2-1.

Table 2-1 - Results from part I base case of Cable Tray Fires

	A	B	C	D	E	F	G	H	I
1	19.0	3.6	1770	0.54	1.37	359	1330	317	
2	19.3	4.7	2057	0.59	0.82	357	1257	322	
3	19.9	2.2	1444	0.41	1.83	347			303
4	22		961	0.39	1.37	336	1839		301
5						336		319	300
6	19.3	0.3	975	0.4		449	472	312	301
7			210	0.35	0.3	349			310
8	19.9		46	0.08		400	4287		
9			600			400	1197	333	
10						360	210	360	300

Key:

Program	Predicted variable
1 – CFAST (BRE)	A – O_2 Concentration in Hot Gas Layer at 600s (%vol)
2 – CFAST (NRC/NIST)	B – Peak Plume Flow (kg/s)
3 – FLAMME_S	C – Peak Pressure (Pa)
4 – MAGIC (EdF)	D – Peak Lower Layer outflow (kg/s)
5 – MAGIC (CTICM)	E – Layer Height at 240s (m)
6 – COCOSYS	F – Peak Hot Gas Layer Temperature (K)
7 – HADCRT	G – Peak Flux on Target (W/m^2)
8 – JASMINE	H – Peak Target Temperature (Surface) (K)
9 - FDS	I – Peak Target Temperature (CL) (K)
10 – CFX	

Although the results to be reported were apparently agreed prior to the analysis, many of the results, as can be seen from the above table, were not reported for the various codes. One example was the hot layer depth; this was not reported for any of the CFD codes. Apparently this was because CFD codes do not directly compute hot layer depth, why this value could not estimated from the results of a CFD code was unclear.

The results that were reported could also be inconsistent with one another, this presumable stemmed from a lack of rigid definition of what the required result actually was. For variable A the oxygen concentration was given in %volume by most of the codes but one code reported their result as %mass. There were large variations in variable G the peak flux on target; these were apparently due to the many different methods used to calculate the heat flux. This rendered any comparison between some of the results virtually meaningless.

A conclusion that can be drawn from the work of Dey et al is that despite the agreement on the input parameters and required results for the cases, the simulations were still subject to extensive 'user' bias.

It should be noted that the work of Dey et al is an ongoing project and it is possible that the procedures and cases used for evaluation may evolve and improve over time.

2.3.1.2 Methodologies for verification and validation

The need for improved verification and validation of CFD codes has been recognised and has led to a number of published guidelines [AIAA1998, Roa1998, ISO1999,

Bea1997]. These guidelines do not actually define a set of possible test cases but a methodology for how the codes should be tested and how the results should be analysed and reported.

These guidelines can have an extensive scope and may also include details concerning software quality assurance with code testing from a software engineering point of view and the need for audit trails and levels of documentation required for a software product. They may also discuss the need for improved experimental data.

2.3.1.2.1 Phased Validation Assessment

Both the AIAA [AIAA1998] and Roache [Roa1998] suggest a phased approach to CFD validation, which would be applicable to fire modelling, as shown in Figure 2-1.

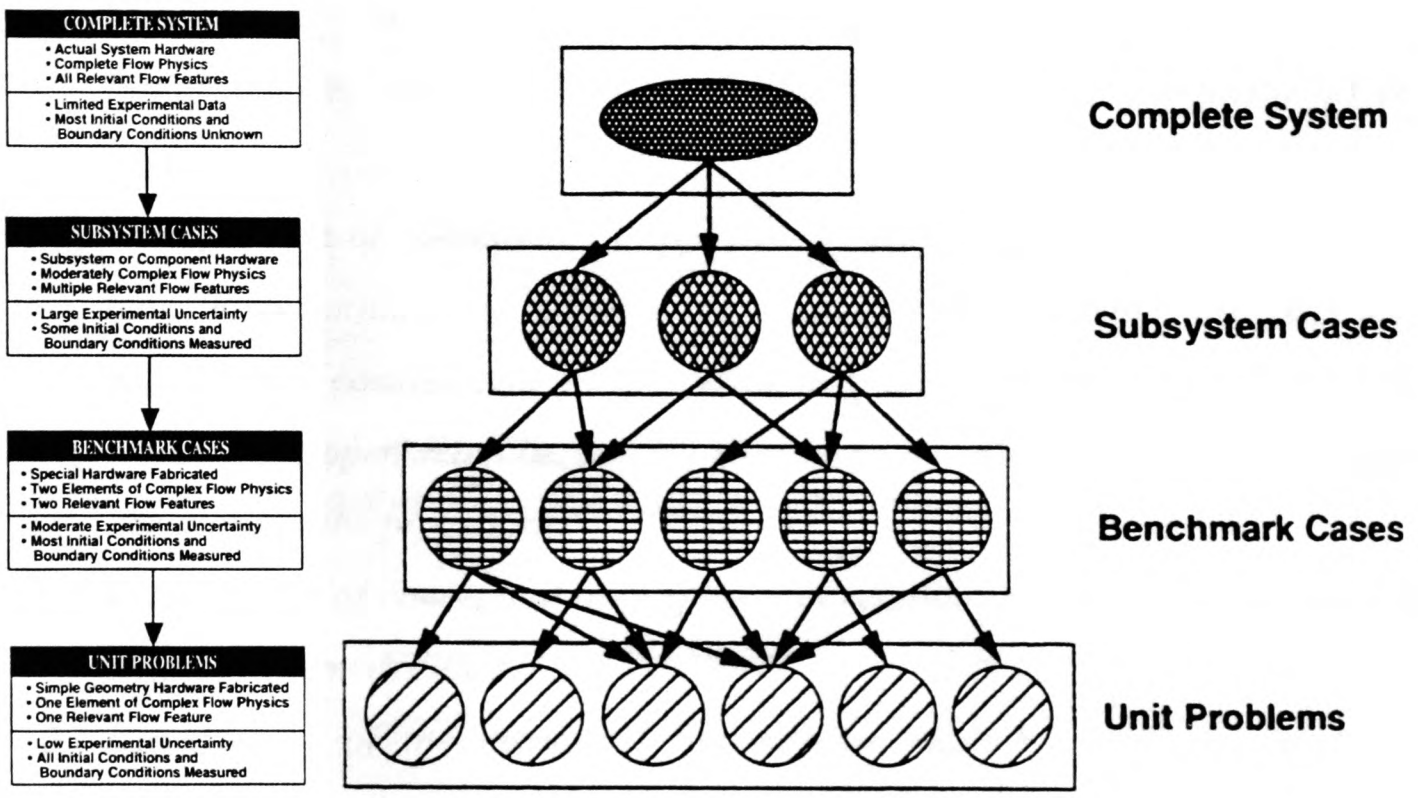


Figure 2-1 – Phased validation process

The basic philosophy of validation is by using a building block approach. Each phase of the validation represents varying complexity and varying accuracy. The unit problems are the simplest and are most accurately measured and defined. The complete system would represent the most complex problem that would also be the least accurately defined. The validation is performed from the bottom up with individual physical properties checked at the unit testing level and the coupling between the physical properties progressively increased as the cases become more complex as we ascend the validation structure. In addition this model is flexible to the

addition of new physical models or applications. New unit, benchmark, sub-system and complete system cases may be required but the existing validation is still applicable so that this work should be straightforward with minimal extra work required.

2.3.1.2.2 Beard's Assessment Procedure

Beard [Bea1997] has produced a tentative outline of a procedure for the complete assessment of a deterministic fire model. Beard advises that the assessment should be carried out by people who are independent as possible of the fire model. Beard's procedure is reproduced in full below and illustrates how large a scope for the proper assessment of a fire model is required. Point 10 deals with verification and validation.

1. *Description of the model and the state-variables it is intended to predict.*
2. *Initial examination of the documentation provided by the producer of the model.*
3. *Identification of conditions of applicability for which the model is likely to have the potential to be valuable (i.e. building and occupant characteristics). As an initial consideration, the conditions of applicability as specified by the model developer may be taken. This may be altered during subsequent iterations of the procedure.*
4. *Examination of conceptual assumptions in relation to 1 and 2 above (e.g. the physics / chemistry etc.).*
5. *Examination of the numerical assumptions implicit within the model in relation to 1 and 2. (i.e. without those which a user may insert or alter).*
6. *Examination of the numerical solution techniques employed: conceptual and numerical aspects.*
7. *Examination of the source code of the program and the software as a whole. Assessment of the likelihood of errors.*
8. *Assessment of the likelihood of hardware faults for the types of computers on which the program might be used.*
9. *Sensitivity study of the model.*
10. *Comparison between theoretical predictions and empirical data*

a. *Assessment of uncertainty / flexibility within the available appropriate experimental data.*
b. *Comparison with the results of replicate sets of experimental tests. Both a priori and a posteriori comparisons should be carried out.*
c. *Comparisons between theoretical predictions and other sources of empirical data; e.g. from a real fire.*
d. *Assessment of the ability of the model to predict quantitative results*
e. *Assessment of the ability of the model to predict qualitative results (e.g. trends).*

11. *Assessment of the limitations of the model in the light of the foregoing considerations.*
12. *Identification of the conditions under which the model may have the potential to be valuable.*
13. *Assessment of the documentation in the light of the foregoing considerations (Return to Step 1.)*

In terms of verification and validation Beards procedure is reasonable although there are some unresolved issues. If a software product fails in some a priori testing how is this reported? Can the software be retested using the same a priori test? Given the level of expertise required to assess the code it is unlikely that any assessor will be truly independent and is likely to be associated to some software product. Many software developers, particularly commercial developers, would be unwilling to reveal their source code to an assessor who could be a potential competitor.

2.3.1.2.3 ISO/TR 13387-3 document

The ISO/TR 13387 document is concerned with the provision of fire safety in buildings. Part 3 of this document is concerned with the Assessment and verification of mathematical fire models. This ISO document discusses the full assessment of fire models and includes:

a) *guidance on the documentation necessary to assess the adequacy of the scientific and technical basis of the model;*
b) *a general methodology to check a model for errors and test it against experimental data;*

c) *guidance on assessing the numerical accuracy and stability of the numerical algorithms of a model;*

d) *guidance on assessing the uncertainty of experimental measurements against which a model's predicted results may be checked;*

e) *guidance on the use of sensitivity analysis to ensure the most appropriate use of a model.*

Part (b) addresses techniques in detecting errors of a fire model can be classified as:

1) *review of the theoretical basis of the model;*
2) *code checking;*
3) *analytical tests;*
4) *inter-model comparison;*
5) *empirical validation.*

In the section on empirical validation the ISO document states that computer predictions should be performed without reference to the results of the experiment that is being simulated (a priori or blind testing).

2.3.1.3 Uncertainty

Uncertainty is an issue for both model predictions and experimental results and makes comparison between the two problematic [Bea1992].

2.3.1.3.1 Experimental data

With experiment data there are a number of uncertainties and flexibilities:

- Uncertainty due to a lack of knowledge of controlled conditions. For example a set of 'identical' experiments may produce substantially different results due to differences in ambient conditions. These ambient conditions may be measured but in many cases they may not be. This has been illustrated in the past by work at the Factory Mutual Research Corporation on full scale bedroom tests [Cro1975] where the time for "Full Room Involvement" in the fire experiment was 17.5 minutes after ignition in test 1 and 7 minutes after ignition in test 2. It has been speculated that the only substantial difference between the experiments was the humidity. Clearly there is a need for experimentalists to perform repeat tests and preferably as many as possible to lend confidence to their experimental results.

- Uncertainty associated with the design of the experiment. In a single experiment it is quite possible for a single parameter to have different results dependent on the measurement method used. For example a bare thermocouple may record a different temperature to a radiation shielded thermocouple. Furthermore there is an error associated with any single measurement device.

- Data processing can also have an effect on the reported values. For example time series data could be averaged over small or large time intervals and an oscillation apparent on the small scale time interval maybe smoothed out when a larger time interval is used.

2.3.1.3.2 Model predictions

- Uncertainty with the mathematical model can lead to incorrect theoretical prediction. For example there are uncertainties in the 'constants' for turbulence models including both the k-ε model and the LES SGS model.

- Uncertainty with the numerical method used in solving the problem. It is quite likely that different mesh resolutions will produce different predictions although a more resolved mesh is likely to produce more accurate model predictions.

The uncertainty in mesh resolution can be tackled using grid convergence studies. The method described by Roache [Roa1998] is detailed below.

For reporting and analysing CFD model predictions Roache describes the use of Systematic Grid Convergence studies for estimating grid convergence error on simulated problems and presents some case studies. The problem should be simulated on at least two different grids although a minimum of three grids is preferred. Three grids are typically produced by halving the initial grid and then halving the grid again, and using Richardson extrapolation on these three values to get the 'exact' value, i.e. the solution that should be obtained on a very fine mesh provided the results lie in the

asymptotic limit. This method also allows an error estimation to be associated with a computed value. For three dimensional problems this would lead to a mesh ratio of 1:8 for the first level of refinement and 1:64 for the second level of refinement which could lead to impractical simulation times for some, if not all, complex scenarios. Roache further describes that the mesh refinement need not be achieved by mesh doubling but using a smaller factor r (refinement) as long as it is greater than 1.1 and this would help alleviate the problem of excessive runtimes. The exact solution can be estimated from the fine mesh solution using the following formula:-

$$f_{exact} \cong f_1 + \frac{f_1 - f_2}{r^p - 1} \qquad (2.3.1.3.2.1)$$

$$E = \frac{\varepsilon}{r^p - 1} \qquad (2.3.1.3.2.2)$$

$$\varepsilon = \frac{f_2 - f_1}{f_1} \qquad (2.3.1.3.2.3)$$

where E is the estimated fractional error, ε is the standard error estimator, r is the grid refinement ratio and p is the order of the computational method. The Grid Convergence Index is defined as

$$GCI = F_s \frac{|\varepsilon|}{r^p - 1} \qquad (2.3.1.3.2.4)$$

where F_s is the "factor of safety" over the estimated fractional error. Roache uses a value of 3 so that the GCI is directly related to the error estimator on a grid doubling scheme when a second order method is used (r = 2, p = 2). The AIAA [AIAA1998] similarly describes the use of grid convergence studies and is essentially a summary of the work of Roache. Kumar and Cox [KC2001] also advocate the use of three grids to perform a mesh sensitivity analysis although no guidance on how to perform this analysis is given.

However, there is presently no way, that the author is aware of, to apply these formulas / methods generally on grids that are generated independently of one another [Roa1998, Roa2001] which would be extremely useful if this method was to be

applied to fire modelling applications. This is also a problem for unstructured meshes and non-uniformly refined meshes.

Due to the large computational cost involved in using Richardson's extrapolation it is normally only used on simplified or model problems. The more usual, though not rigorous, definition of convergence is that there is little change in important dependent variables observed through the refinement process. Quantifying uncertainty is still an active research topic in CFD and as yet there is no general consensus [Kar2002] on the subject.

2.3.1.3.3 Review of Peacock et al – Quantifying fire model evaluation using functional analysis.

Peacock et al [PRDW1999] noted that many published comparisons of fire models, zone and field models, with experiment tend to be qualitative in nature. Typically phrases such as "satisfactory", "favourable", "reasonable" or "well predicted" are used to describe the comparisons. Peacock et al describes the use of functional analysis to try and quantify the level of agreement particularly with respect to time series data. This work is still a topic of research and more work needs to be done before this method could be widely adopted and therefore qualitative comparisons will still be used in the meantime.

2.3.2 Case study of user bias - Nielson SMARTFIRE model comparison with experimental data

Another example of user bias is the work performed by Nielson [Nie2000]. Nielson used the SMARTFIRE fire field model to simulate a fire experiment and compared these predictions with experimental values. Unfortunately due to some inadequacies in the user's modelling, the results were not as good as would have been anticipated. When a more experienced user (Jia) attempted to reproduce the experiment, the code gave a much closer comparison with the experimental results [Jia2001]. Jia concluded the following:-

- Nielson's mesh specification was not refined enough for the purposes of experimental comparison.

- Neilson did not take into account the obstruction nature of the gas burner that was elevated above the floor. This lack of obstruction to the air flow has a significant effect on the plume behaviour.
- The use of steady state modelling can lead to difficulty in obtaining converged results for some fire simulations. Convergence can be more easily achieved by time stepping towards a steady-state solution.
- In order to get good predictions in the fire near field a combustion model should be used instead of a volumetric heat source.

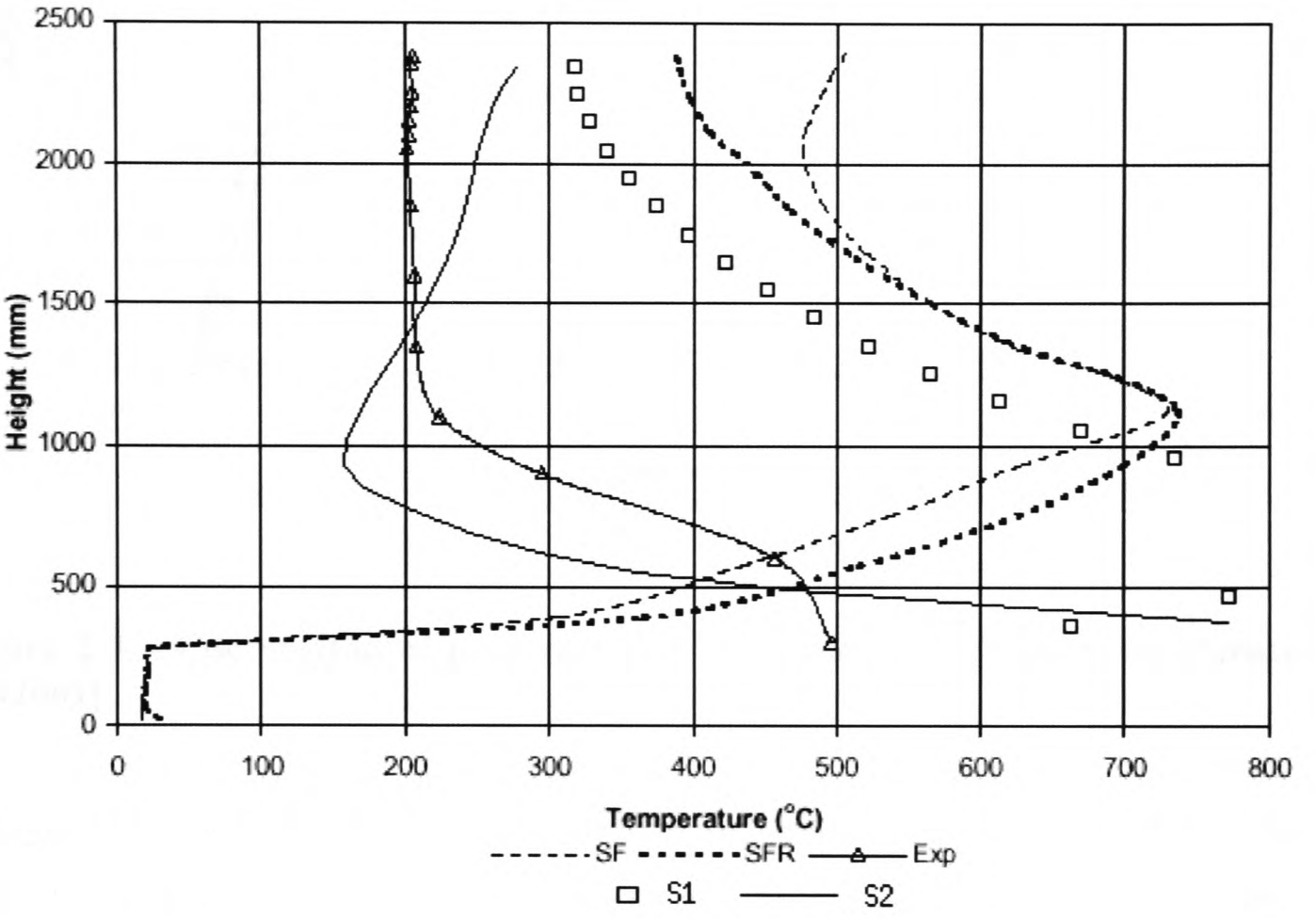


Figure 2-2 – Comparison of predicted and experimental temperatures above gas burner [Jia2001]

In Figure 2-2 above the predictions of Nielson and Jia are compared against the Canterbury experiment [Nie2000]. The figure illustrates the comparison of predicted and experimental temperatures of a thermocouple tree located above a 110 kW gas burner. Nielson's predictions are denoted by SF and SFR where SFR is the same as SF but with the inclusion of the six-flux radiation model. S1 and S2 denote Jia's predictions, where S1 is a volumetric heat source, as used by Nielson, and S2 uses a combustion model to represent the heat release from the gas burner, Jia used a six-flux radiation model throughout. Exp represents the experimental values. It can be seen

that Jia's S1 prediction is much lower than either of Nielson's predictions towards the ceiling although Jia's S1 prediction is arguable worse nearer the floor level. The major improvement is due to the introduction of the combustion model where Jia's S2 predictions are far closer to the experimental results.

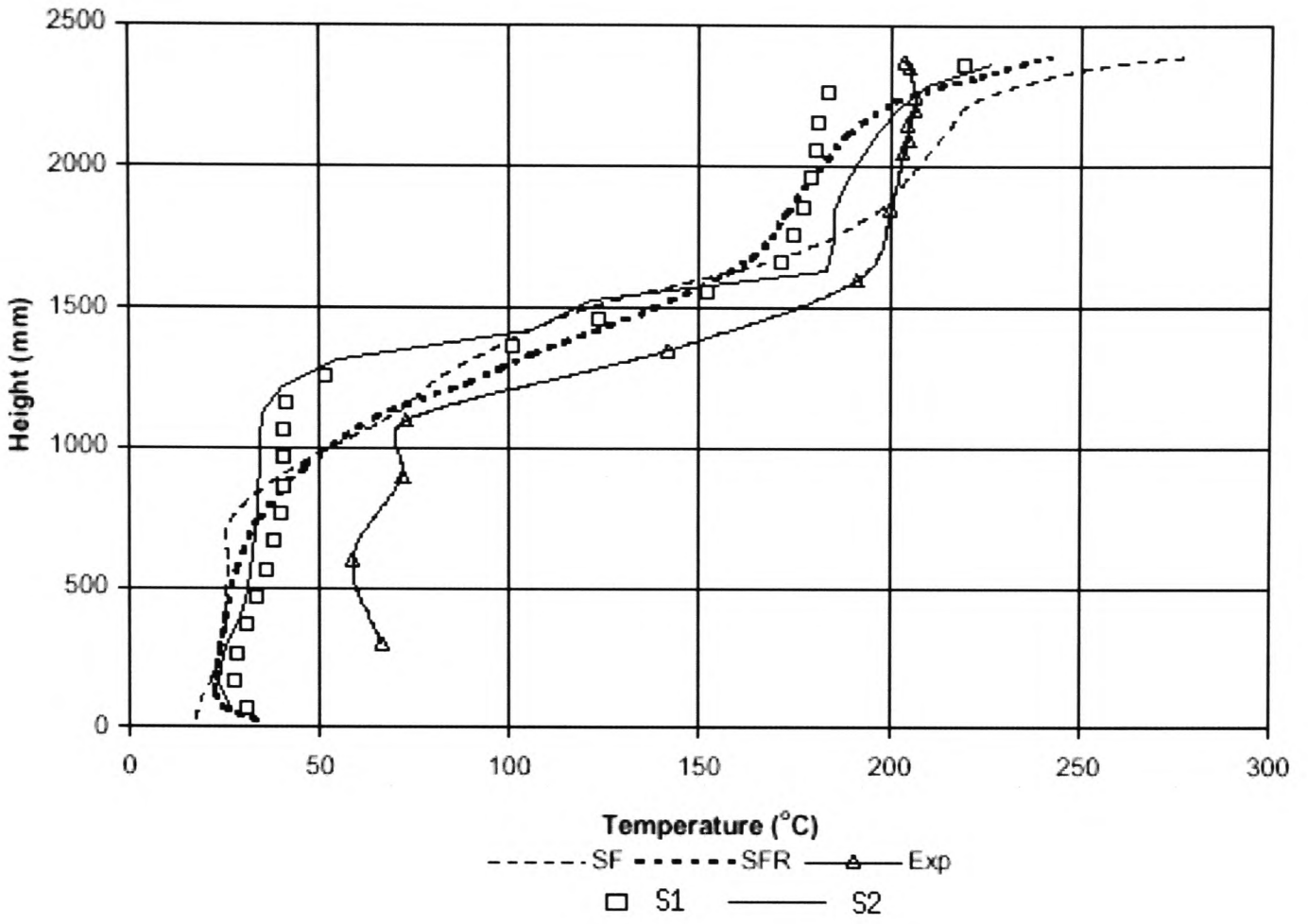


Figure 2-3 – Comparison of predicted and experimental far field temperatures [Jia2001]

In Figure 2-3 above the far field temperature comparison is made. It can be seen that both of Jia's predictions (S1 + S2) showed a hot layer stratification at approximately 1600mm similar to that seen in the experiment. This stratification was not observed in the predictions of Nielson (SF + SFR).

2.3.3 Sensitivity analysis

Many workers advocate the use of sensitivity analysis on both fire models and fire experiments [Bea1997, ISO1999, PRFK1998]. Some of the purposes of a sensitivity analysis are to determine:

- the important input variables in the model;
- the required accuracy of the input variables;
- the sensitivity of the output variables to the input data.

A sensitivity analysis can have an almost limitless scope for a CFD based fire model due to the number of parameters involved so the studies conducted must be selectively chosen with care. The grid convergence study is a form of sensitivity analysis with respect to the variation of predicted values caused by differing computational meshes. Possible sensitivity analysis could include varying the ambient conditions, varying the position of the fire or varying strength of the fire source.

2.3.4 Discussion of "Fire modelling assessment" review

Previous work on guidelines for accessing fire models [Bea1997, ISO1999], presented above, has much to commend it and indeed much of their work should probably be adopted for assessing fire models. It is not the intention of this thesis to address the whole fire model assessment issue. However the problem of user bias within the verification and validation phase of any assessment process has been reported [Bea2000, HK1998] and described in section 2.3.1.1.2 and section 2.3.2 but there appears to be little work on accounting for and eliminating user bias from the evaluation of a fire model. User bias can come in many forms that include tuning results, modelling expertise and application expertise. Using blind and a priori testing should remove result tuning [Bea2000] although it does leave open issues of the user in terms of modelling and application expertise. Allowing the software product vendors to evaluate their own software could result in bias due to the possible usage of undocumented features or by addition of extra coding to improve the results. Independent evaluation could lead to bias due to lack of expertise with that software product. Bias can also be introduced when the assessment procedure is conducted by different people whose judgements may differ from one another. Different users may use different sub-models, grid refinement levels, levels of convergence which can lead to differing results without any certainty as to whether their model is right or wrong due to errors caused by user bias. A procedure to reduce, and hopefully eliminate, user bias from the validation and verification of a CFD based fire model is presented in chapter 5. Testing of fire models in the past little has been done in checking the basic CFD capabilities of the fire model. It must be noted that in the past zone models have been included in such tests so these tests are not applicable in these cases. However the CFD capabilities of a CFD based fire model should be checked to ensure that the fundamental mathematics is functioning correctly.

2.4 Parallel Processing using Standard PCs

2.4.1 Parallel Architecture

Parallel processing has been used in many computationally intensive tasks such as weather prediction, genome sequencing, computational chemistry, ray tracing and even playing a very good game of chess. In the past, parallel processing techniques have been successfully applied to CFD fire simulations on specialised parallel processing computer equipment [IG1992, GI1992, GI1993]. Although the equipment used in this work has long since become redundant the argument is still valid for the profitable use of parallel processing in the realm of fire modelling applications.

CFX has also been used to model fire in parallel on a 4-processor SGI PowerChallenge [Eur, Stu1997]. They modelled a set of fire experiments performed on a third scale shuttle train carry a HGV. They obtained a speedup of approximately 3.5.

The history of parallel computers can arguably be traced back to the 1960s when vector processors had been used in the supercomputing technology of the time. These machines could act on a vector of data as opposed to a single data item of a standard scalar processor. This concept was extrapolated into the array structured SIMD (Single Instruction Multiple Data) in which a whole array of data could be subjected to the same instruction at the same time. These machines have proved difficult to program when mapping irregular problems and has lead to decline of this platform.

Currently the most popular parallel platform is MIMD (Multi Instruction, Multi Data); here the parallel machine is built by connecting a number of processors together. There are two main variants, Distributed Memory (DM) and Shared Memory (SM). In SM the processors share the same memory (see Figure 2-4 below).

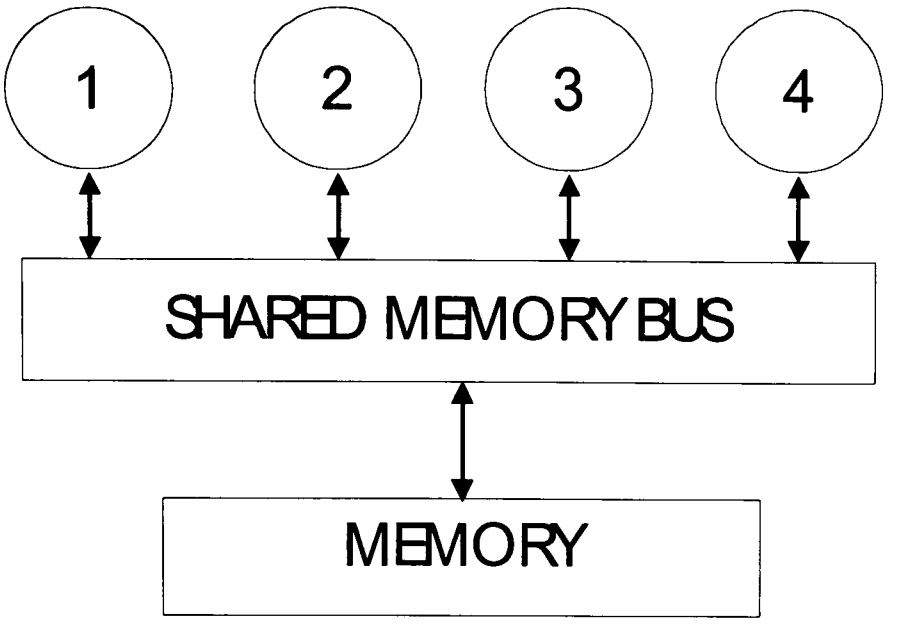


Figure 2-4 – Shared memory Parallel Processing architecture

The shared memory bus can become congested when processors attempt to access the shared memory at the same time. This problem restricts the maximum useful number of processors on a SM machine. There are also problems with data contention when processors may be writing to memory whilst another process reads or writes the same bit of memory. In PC terms a SM machine could be a PC equipped with a dual or quad processor motherboard. In the future, motherboards with larger numbers of processors may become available with methods devised to overcome problems associated with a congested memory bus.

In DM computers, each processor has its own memory and communicates with the other processors via some form of network (see Figure 2-5 below). This system has the advantage of each processor having its own memory which removes the data congestion possible on a SM machine. This however leads to other disadvantages such as more complex programming requirements and the need for data communication between the processors. The communication link is now a possible source of bottleneck. Generally, the DM approach is far more scalable than the SM approach; DM machines consisting of 100s of processing nodes have been created, such as Deep Blue [IBM1997]. DM in PC parlance can be most simply thought of as a series of networked PCs working on the same problem that has been distributed between them.

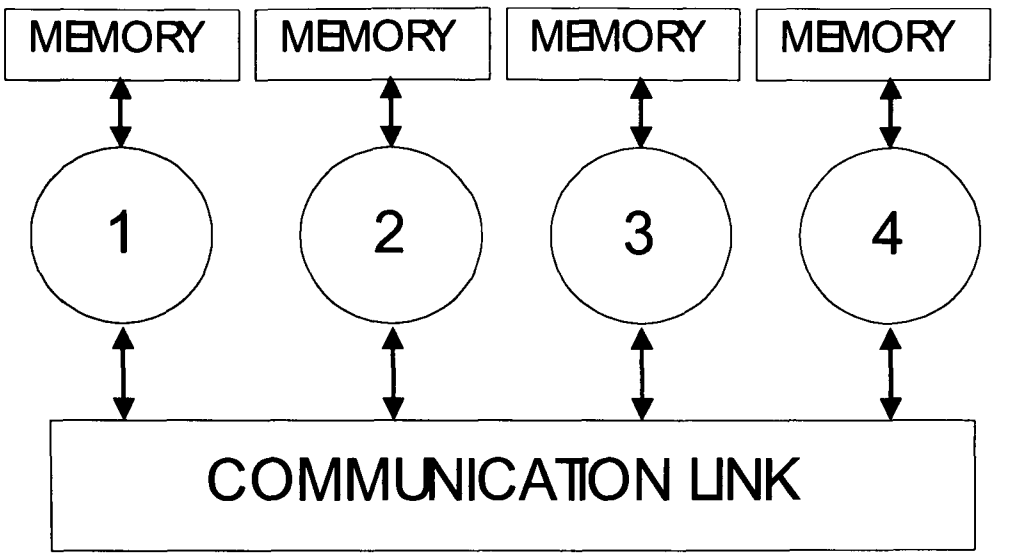


Figure 2-5 – Distributed Memory parallel processing architecture

The communication link can be implemented in a number of ways and processor topology mapping can be an issue in some circumstances. These topologies include 1D-pipeline, 2D-pipeline, ring, hypercube, and star. With these methodologies it can be seen that if information may need to be communicated via several processors to get to its ultimate destination.

However when a general Local Area Network (LAN) is used as the communication link all the machines are effectively connected to every other machine by a router (c.f. star). The disadvantage of this communication link is that communication is not as fast as a more optimised topology such as a Beowulf parallel cluster [SBS+1995] that utilises a dedicated fast network connecting commodity PC parts. There seems to be some work, mostly web based, about specialised number crunching PC clusters with their authors eulogising about their high performance to cost ratio while little attention has been given to standard networked PCs. This is hardly surprising because of the relatively inferior performance of networked PCs, compared to these clusters, but this potential computing resource may be freely available within a FSE's office environment with little to no additional investment in hardware and therefore deserves further investigation. No work appears to have been published concerning the direct use of typical non-specialised office / laboratory based PCs for the purpose of CFD based fire modelling and has therefore motivated the research described within this thesis. Some investigations into parallel processing on office based PCs have been conducted and they are reviewed below.

2.4.2 Review of CFD parallel processing on conventional office / laboratory based PCs

The European Commission's HPCN-TTN (High Performance Computing and Networking – Technology Transfer Node) has initiated a number of projects utilising low cost parallel processing for SMEs (Small and Medium sized Enterprises). A number of these projects utilised clusters of workstations. Most of this work appears to have been performed on UNIX based machines with some work performed on specialised PC clusters mostly operating Linux although some did work with Windows NT. There only appears to be one documented piece of work using a non-dedicated network. This was parallel PHOENICS developed as part of the DOWN-PORT project

2.4.2.1 Parallel PHOENICS – DOWN-PORT project

A number of CFD codes will work in parallel on Windows NT; these include PHOENICS, STAR-CD and CFX. However little data appears to be available concerning their actual performance on a normal Windows NT based network.

For Parallel PHOENICS some data exists from the DOWN-PORT project [Tho1999, Tho2000]. That project was concerned with utilising clusters of workstations for parallel processing. Most of the work of this project was performed on UNIX platforms with some on Linux platforms and a small amount of work reported on Windows NT based machines. Parallel PHOENICS was evaluated by three companies; Roger Preston and Partners, Atos, and RWTUEV Anlagentechnik. Roger Preston conducted simulations of a HVAC problem on a special cluster comprised of two 333MHz machines and three 300MHz machines attached via an isolated 100Mbps hub. They reported that the use of three processors, which were not specified, reduced the problem runtime of 39 hours to 19 hours. The problem was also not specified beyond the fact that it was a HVAC type problem. No results were reported for the Atos case apart from '*The speed up experienced by Roger Preston and Atos was as expected*'. RWTUEV Anglentechnik utilised a range of PCs attached to a 10Mbps LAN, this was the only test which did not isolate the machines used from the general LAN. They found that the software performed unsatisfactorily with runtimes taking longer than would be expected for a single processor.

On a specialised cluster of four Pentium 200MHz computers a speedup of approximately 3.2 for parallel PHOENICS was reported, on the Cham web site [Cha] although details of the test case were not apparent. Parallel PHOENICS was also reported to have heterogeneous load balancing although whether this can cope with sharing computer resources with other users was unclear. CFX also possesses static heterogeneous load balancing but there appears to be no documentation mentioning a dynamic load balancing feature.

2.4.2.2 Review of Law and Turnock – "Utilising Existing Computational Resources to Create a Commodity PC Network suitable for fast CFD Computation"

Law and Turnock [LT2001] investigated and utilised a set of student laboratory PCs, ≥ 350Mhz Pentium II/III, as a parallel computing resource for solving an Euler based problem, which utilises similar CFD modelling techniques. One inconvenience of their approach was that the parallel machine used Linux as its operating system. The general lab machines used Windows NT and therefore required a reboot to the Linux operating system when parallel computing was required. They examined the performance of an unstructured Euler solver on a problem consisting of 48672 cells. They found that a network of 16 processors would run the parallel code up to 10 times faster than the serial code on a single processor (see Figure 2-6 below).

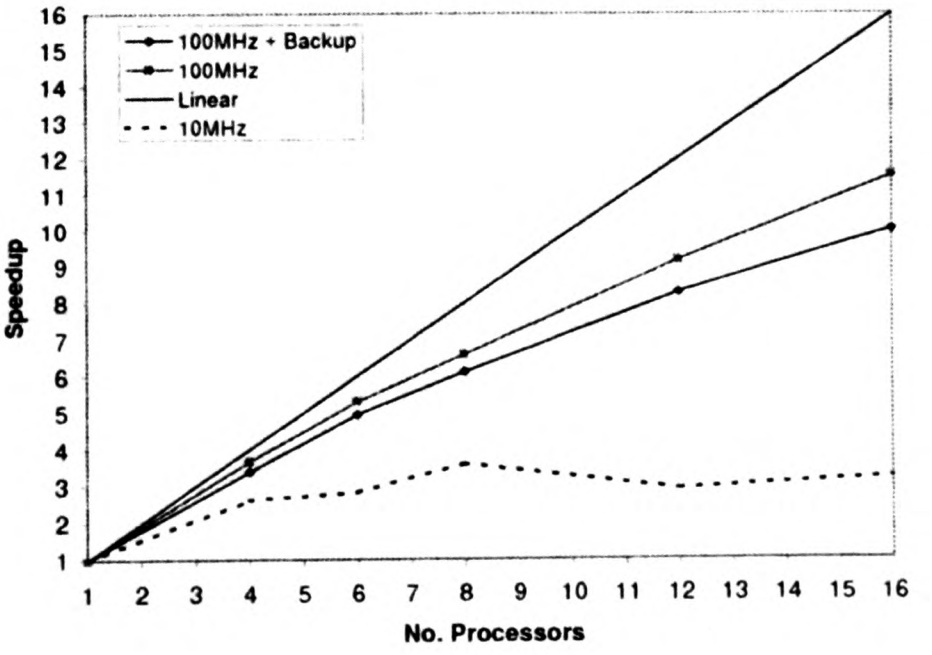


Figure 2-6 - Parallel Speedup performance for PentiumII/III network [LT2001]

Another important observation they made was the difference in performance when a 10MHz network was used compared to a 100MHz network as the communication link between the PCs. It can be seen from Figure 2-6 that the 10MHz network delivers very poor speedup and is impractical for parallel computation.

2.4.3 Concluding remarks on review of parallel processing on conventional office based PCs

Virtually all of the work done on PC networks/clusters [LT2001, SBS+1995, KRSK2001, Tho1999, Tho200] has tended to be performed using Linux [Lin], possibly the preferred operating system of the parallel processing community. Linux has much to commend it and may be superior to Windows NT in a number of areas. However, Windows NT is also quite suitable for parallel processing and the majority of 'ordinary' computer users, including FSEs, are much more likely to have access to Windows NT workstations as opposed to Linux workstations, although it is virtually impossible to get any agreed figures on comparative usage. The published work with Windows NT type machines has only used a small number of computers, typically less than six computers. Presently (circa 2003) the DOS based windows products Windows95, Windows98 and WindowsMe are the most popular operating systems in use but they generally can not support the present parallel processing technologies. This is a moot point as these operating systems have now been phased out by Microsoft in favour of NT based technologies, i.e. Windows2000 and WindowsXP, which can support the parallel technology. Furthermore if parallel processing is required the operating system can always be upgraded at minimal cost compared to the potential benefits gained.

From the DOWN-PORT project it was apparent that there would be scope for parallel processing on a Windows NT PC network. Unfortunately the only evaluation done on conventional equipment, all the other evaluations were effectively performed on specialised clusters, did not demonstrate that parallel processing was practical on conventional office based PCs.

2.4.4 Parallel Processing Strategies

There are several possible strategies for parallelising a CFD based fire modelling software, which include Task Farming, Algorithmic parallelisation and geometric decomposition.

Task farming has potentially a very high parallel efficiency, as all the processors are constantly being kept busy. In this case when a processor finishes a particular task the master process initiates a new task on that processor. This method is only suitable for tasks that would be unrelated to one another. This is the strategy used for seti@home [SWB+1997] and has created the largest and most powerful parallel virtual computer in existence with the co-operation of millions of computer users around the world. Unfortunately, this method does not lend itself to parallelisation of CFD based fire simulation software. This is due to the high degree of coupling that exists within the fire field model and the tasks would therefore be highly related to each other.

Algorithmic parallelisation involves each processor working on different parts of the algorithm. In the fire field modelling case this could be one processor solves the u-momentum equation while another solves the pressure correction equation and so on. This scheme has little to commend it, as it requires a high level of inter-processor communication and it would not be possible to scale such a scheme to an arbitrary number of processors efficiently.

In geometric decomposition the problem is divided up between the processors. Each processor then solves its part of the problem using the same algorithm. In CFD based fire modelling the variables are dependent on the value of that variable at neighbouring locations due to the discretisation process. This requires communication between the processors but this is relatively small as only the variables at the boundary of the each processor's sub-domain need to be communicated. This is a form of Domain Decomposition (DD), where DD is the division of a problem space into two or more parts that can be usefully operated on separately. DD is a major area of research in its own right, with much interest in ways of optimising the shape of the sub-domains to minimise inter-processor communications [Fox1988, RVD1993, Far1988, Wal1995, KK1999]. DD was the strategy selected for this research and has been successfully used for many CFD codes in the past such as PUIFS [McM1996],

CFX [CFX2003] and PHOENICS [BS1994]. This approach has also been successfully utilised for fire modelling using a CFD code [IG1992, GI1992, GI1993].

SMARTFIRE was selected as the starting point for the creation of a parallel CFD based fire code. SMARTFIRE was selected because:-

- SMARTFIRE has some validation history in Fire Modelling [WJG+2000, EGP+1999] and is designed specifically for fire modelling.
- The entire source code was freely available to the author.
- Utilises modern Object Orientated program design methodologies and implemented in C++.

2.4.5 Implementation of Parallel Processing

Parallel processing can be implemented using either specialised parallel languages or by using parallel add-ons to general serial languages. Ada, Occam, HPF and possibly Java are amongst the specialised languages used for parallel processing. Given that a legacy serial code, SMARTFIRE, was the starting point there was little point in totally re-implementing the software in a specialised parallel language. This would require a substantial amount of work in the initial conversion and would also require additional maintenance with additional features being added both to a serial and parallel code. The most economical approach was to choose a parallel add-on that has bindings for C++, the implementation language of SMARTFIRE. This reduces the amount of work required to obtain a parallel version and simplifies code maintenance by utilising a single source code that can be compiled as a serial or parallel version.

2.4.6 Parallel Communication Package

A parallel communications package was needed to implement the communications on the Windows NT PCs. There are a number of such packages available for Windows NT, the most notable examples being PVM (Parallel Virtual Machine) [GBD+1994] and MPI (Message Passing Interface) [For94]. Both systems have their advocates and have been compared against each other [GL1997, GKP1996]. It is generally considered that PVM is best for networked machines and MPI is best for MPP (Massively Parallel Processing) due to the particular features each possesses. MPI is intended primarily for data-parallel problems (which are closely related to the

geometric decomposition method). Therefore, it does not have the flexibility of PVM's dynamic process spawning, but its collective operations (like gather-scatter operations) and asynchronous message passing capabilities (asynchronous sends and receives) are much more sophisticated and configurable than those in PVM. As found by McManus [McM1996] the use of asynchronous communications is necessary to deliver good speedup for a parallel CFD code and thus makes MPI the most appropriate choice for a network of PCs in this instance. Furthermore, MPI has now established itself as the de facto standard message passing interface amongst the Parallel Computing community, some of the reasons for this success are discussed by Hempel and Walker [HW1999]. Essentially it is the forum nature of the MPI specification with the involvement of many highly respected members of the Parallel Processing Community (PPC) that has led to the standard; being well tailored to the needs of the PPC, and also being widely adopted by the PPC in general.

2.4.6.1 Overview of MPI

MPI is a non-vendor Application Programming Interface (API) standard that facilitates high level parallel programming. The MPI forum (MPIF) was born out of the need for standardisation of message passing interfaces in the 1990s to allow easy software porting between parallel platforms.

The MPI API provides:-

- Blocking communication modes, both global and point to point communication.
- Asynchronous communication modes allowing the overlap of computation and communication.
- Language bindings for C and FORTRAN.

Additional features for MPI have been recommended by the MPI-2 forum, this has added language bindings for C++, dynamic processes, one-sided communication and parallel I/O to the MPI-2 standard. At the time of writing only MPI-1.2 was readily available and this was the version used to implement parallel SMARTFIRE. Details of the MPI commands used in the parallelisation of SMARTFIRE are described in Appendix A.

3 Mathematical Modelling

In this chapter, the mathematical methodologies adopted by most CFD codes and in particular the SMARTFIRE field model are described. The general conservation equation is described and is integrated to illustrate the general method of discretisation. The discretisation of the physical equations that are solved along with details of the handling of the various boundary conditions and special treatment of the source terms will be elaborated. The equations solved are the momentum, enthalpy and the mass continuity differential equations. In fire modelling, turbulence, combustion and radiation need to be effectively modelled and relevant methods are described. The SIMPLE method will be detailed which is the basis of the solution algorithm used in SMARTFIRE computational model as well as many other CFD codes. Two notable exceptions are the Fire Dynamics Simulator (FDS) which utilises a second order predictor-corrector method [MBR+2001] and CFX which utilises a coupled solver [Raw1994, Raw1996, CFX2003].

The Rhie and Chow (or Pressure Weighted) interpolation method will be described including a discussion on the relative merits and demerits compared to an alternative approach that utilises a staggered formulation. Finally, the numerical solvers used to solve the equations will be described.

3.1 The General Conservation Equation

The general conservation equation takes the following form:

$$\frac{\partial(\rho\phi)}{\partial t} + \nabla.(\rho \underline{u}\phi) = \nabla.(\Gamma_\phi \nabla\phi) + S_\phi \qquad (3.1.1)$$

Transient + Convection = Diffusion + Source

where ϕ represents the dependent variable to be solved, ρ is the density of the fluid and $\underline{u}$ is the velocity of the fluid. Γ_ϕ is the diffusion coefficient for the variable ϕ, which may represent such quantities as viscosity and conductivity. S_ϕ is the additional source terms for the variable ϕ.

3.1.1 The Transient term

The first term in equation (3.1.1) is the transient term which represents the rate at which ϕ accumulates per unit volume.

3.1.2 The convection term

The second term in equation (3.1.1) is the convection term, which is the accumulation of ϕ per unit volume due to the divergence in its convective flux field.

3.1.3 The diffusion term

The third term in equation (3.1.1) is the diffusion term, which is the accumulation of ϕ per unit volume due to the divergence in its diffusive flux field.

3.1.4 The source terms

The last term in equation (3.1.1) is the source term, this term includes all the additional sources of ϕ per unit volume which is not covered by the previous terms. It can then include such external sources of ϕ such as gravity is an external source to the momentum equation.

3.2 Numerical procedure

The process of discretisation is to transform a set of simultaneous continuous non-linear partial differential equations of ϕ into a set of simultaneous linear algebraic equations which describe the value of the variable ϕ at a number of discrete points in time and space.

This process is performed as the original partial differential equation cannot normally be solved by analytical methods. This set of equations can be represented in matrix and vector form as $\underline{\underline{A}}\ \underline{\phi} = \underline{B}$. This matrix equation can now be solved using a number of different methods; it is sometimes possible to solve this set of equations directly e.g. by finding $\underline{\underline{A}}^{-1}$, but it is more generally solved by using an iterative method such as the Successive Over Relaxation (SOR) method which will be described, along with some other numerical schemes later in this chapter.

3.3 Discretisation scheme

There are a number of possible discretisation approaches available; these include the control-volume (CV) or finite-volume (FV) method, the finite element (FE) method, the finite difference (FD) method and the boundary element (BE) method. The finite-volume method will be described in the following sections.

The FV approach is based on Gauss' divergence theorem:

$$\iiint_V \nabla . \underline{F}\, dV = \oiint_S \underline{F} . \underline{\hat{n}}\, dS \tag{3.3.1}$$

where V is some arbitrary volume in space, $\underline{F}$ is a vector field extending throughout the volume, S is the surface that encases the volume V and $\underline{\hat{n}}$ is the unit normal vector over the surface S. It is clear from (3.3.1) that there exists a relationship between a volume integral to a surface integral. This is the crucial part of the FV method. It allows the large computational domain to be divided up into a number of smaller sub-domains referred to as finite volumes (or control volumes) around which the surface integral can be used. The main advantage of this method is that for each control volume the variable of interest in conserved for that control volume. This means no matter how coarse or fine the mesh is the variable is conserved both locally and globally.

3.3.1 The Computational Grid

The computational domain is divided into a number of smaller sub-control volumes known as cells. These cells are chosen so that they completely fill the domain and that the control volumes do not overlap one another. These control volumes are considered to surround a point in the domain and it is the value at this point that the solution is obtained. The boundaries of the cells are said to form a grid (also known as a mesh).

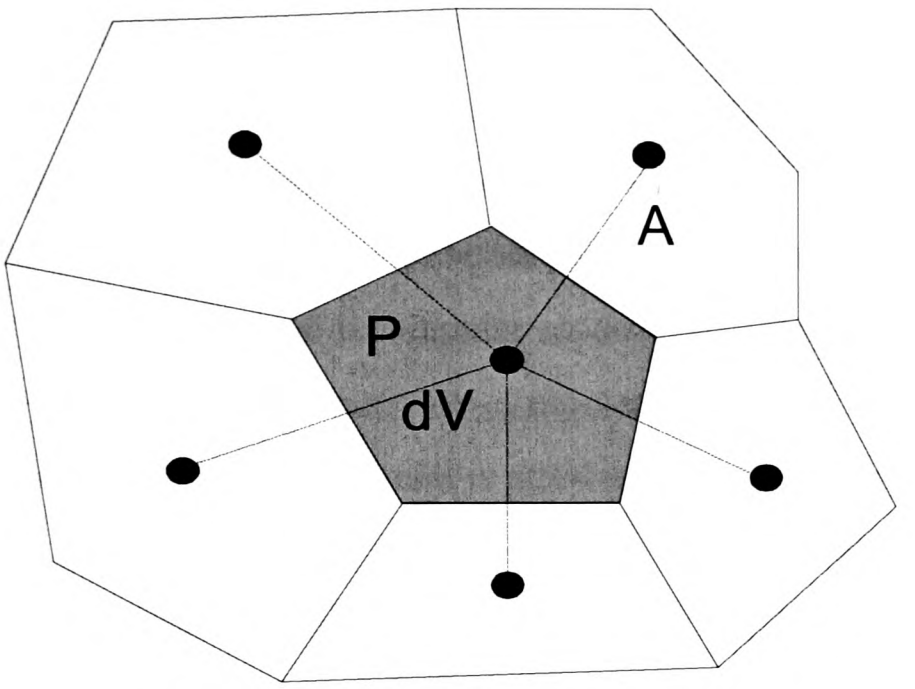


Figure 3-1 – A finite difference grid

The cells may be any arbitrary shape in principle, but more complex shapes do introduce more complexities to the calculations involved. An example of a finite difference grid is illustrated in Figure 3-1 above. This is a graphical representation of the computational molecule for the point P. It is assumed that the value of ϕ at P is related to the values of ϕ at the neighbouring points.

3.3.1.1 Structured and Unstructured grids

There are typically two types of grid used in computational fluid dynamics these are structured and unstructured grids.

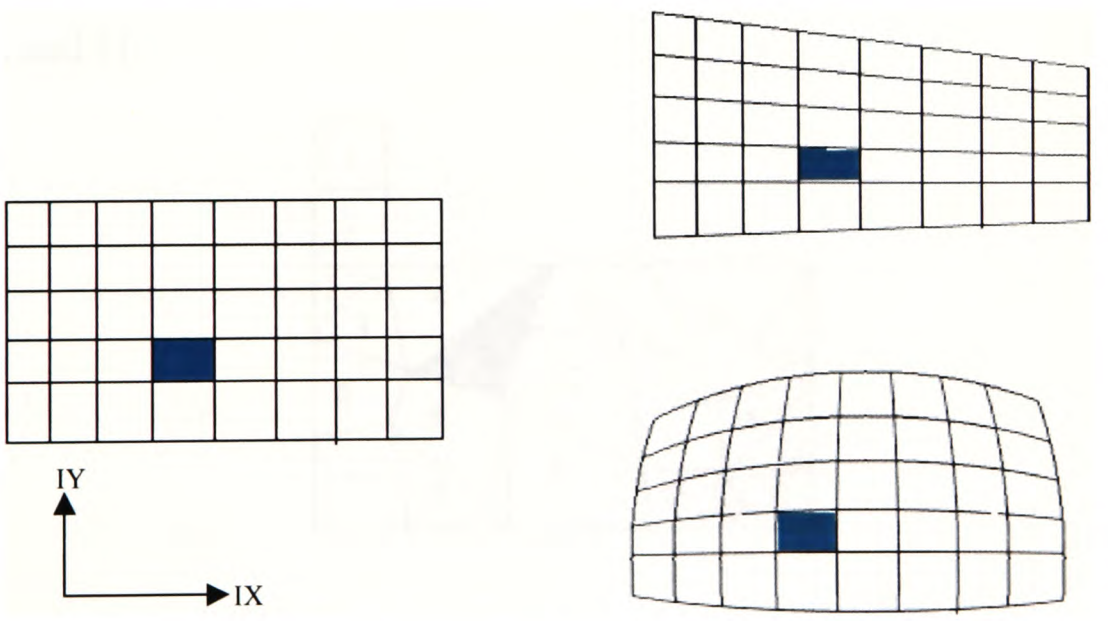


Figure 3-2 - Structured Grids

A structured grid (see Figure 3-2 above) is formed by the intersection of families of curvilinear lines that coincide with the shape of the physical domain at its boundaries. It can be viewed as a regular, rectangular grid that has been pulled and stretched until

it fits the dimensions of the physical domain. The simplest case is when the physical domain is a regular rectangular grid. The cells are referenced with respect to their computational rectangular co-ordinate system. From Figure 3-2 it can be seen that the structure of the mesh means that the neighbours of a computational cell can be found with reference to the computational rectangular co-ordinate scheme. All three meshes in Figure 3-2 are effectively the same in computational space although they are different in physical space. The shaded cell is at IX = 4 and IY = 2 in all cases. The 4 neighbouring cells for a cell located at (IX, IY) are identified using the following simple relations (3.3.1.1.1). Codes such as PHOENICS and CFX4 utilise structured grids.

$$\left.\begin{aligned} WEST\_NEIGHBOUR &= (IX-1, IY) \\ EAST\_NEIGHBOUR &= (IX+1, IY) \\ NORTH\_NEIGHBOUR &= (IX, IY+1) \\ SOUTH\_NEIGHBOUR &= (IX, IY-1) \end{aligned}\right\} \qquad (3.3.1.1.1)$$

An unstructured grid (see Figure 3-3) can be formed from an arbitrary set of cells provided they are non-overlapping. There is no structured co-ordinate system to refer to so each cell must therefore have a unique identifier, typically a number ranging from 1 to n where n is the number of cells in the domain. Each cell must have an adjacency table of its neighbours. For cell 10, in Figure 3-3, the adjacency table would be 5, 8, 12, and 11.

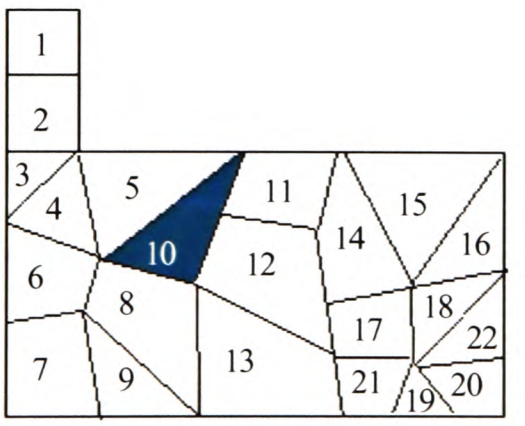


Figure 3-3 – An unstructured grid

The unstructured grid allows for a greater flexibility in the formulation of the grid but requires an extra overhead of keeping an adjacency table for each cell. The structured mesh can take advantage of its inherent structure to utilise faster solution techniques. Unstructured grids are used by CFX5, Physica and SMARTFIRE.

3.3.2 The Discretised general conservation equation

In this section, the FV formulation will be applied to the general conservation equation. To obtain the finite difference equations the general conservation equation is integrated, for a general variable ϕ located at a point P, (see Figure 3-1) over a typical cell volume δV located around that point P, and over a small period of time δt.

By rearranging the general conservation equation (3.1.1) and integrating with respect to volume the following is obtained:

$$\iiint_V \frac{\partial(\rho\phi)}{\partial t} dV + \iiint_V \nabla.\left(\rho \underline{u}\phi - \Gamma_\phi \nabla\phi\right) dV = \iiint_V S_\phi \, dV \tag{3.3.2.1}$$

In equation (3.3.1) replace $\underline{F}$ with the quantity ($\rho\underline{u}\phi$ - $\Gamma_\phi \nabla\phi$) to obtain:

$$\iiint_V \nabla.\left(\rho \underline{u}\phi - \Gamma_\phi \nabla\phi\right) dV = \oiint_S \left(\rho \underline{u}\phi - \Gamma_\phi \nabla\phi\right).\underline{\hat{n}} \, dS \tag{3.3.2.2}$$

and,

$$\oiint_S \left(\rho\underline{u}\phi - \Gamma_\phi \nabla\phi\right).\hat{\underline{n}} \, dS = \sum_{\text{all faces}} \iint_f \rho\phi\underline{u}.\underline{\hat{n}} df - \iint_f \Gamma_\phi \nabla\phi.\underline{\hat{n}} df \tag{3.3.2.3}$$

and,

$$\iint_f \rho\phi \underline{u}.\underline{\hat{n}} df \approx \rho_f \left(\underline{u}.\underline{\hat{n}}\right)_f A_f \phi_f \tag{3.3.2.4}$$

and,

$$\iint_f \Gamma_\phi \nabla\phi.\underline{\hat{n}} df = \Gamma_\phi A_f \left(\frac{\partial\phi}{\partial x} n_x + \frac{\partial\phi}{\partial y} n_y + \frac{\partial\phi}{\partial z} n_z \right) \tag{3.3.2.5}$$

Using (3.3.2.4) and (3.3.2.5) and substituting into (3.3.2.1) equation (3.3.2.6) is obtained:

$$\frac{\left[(\rho\phi) - (\rho\phi)^{old}\right]_P}{\delta t} V_P + \sum_{\substack{all\ \text{faces} \\ \text{of FV}}} \left(\rho_f \left(\underline{u}.\underline{\hat{n}}\right)_f \phi_f - \Gamma_\phi \left(\frac{\partial\phi}{\partial x} n_x + \frac{\partial\phi}{\partial y} n_y + \frac{\partial\phi}{\partial z} n_z \right) \right) A_f = S_{\phi,P} V_P \tag{3.3.2.6}$$

In the above formulation it is assumed that the transient term can be approximated by the average rate of change between the current and previous time steps. The above is an implicit formulation for the variable ϕ_P. In the implicit formulation the value of ϕ_P is dependent on the values of the neighbouring ϕ at the new time step. An alternative formulation, the explicit temporal discretisation, is described in section 3.3.3.

The next stage is to obtain a computational molecule by converting ϕ_f and $\left.\frac{\partial \phi}{\partial k}\right)_f$ into values approximated by ϕ_P and ϕ_A, where A is a neighbouring node.

By assuming orthogonality the partial derivatives of ϕ can be estimated by using the following relation.

$$\frac{\partial \phi}{\partial x} = \frac{\phi_A - \phi_P}{d} \hat{d}_x \qquad (3.3.2.7)$$

where $\underline{d}$ is the unit vector from node P to the adjacent node A and d is the distance between these two nodes. Similar relations exist for $\frac{\partial \phi}{\partial y}$ and $\frac{\partial \phi}{\partial z}$.

If the mass flux through a surface is represented by FA ($= \rho_f(\underline{u}.\underline{n})_f A_f$) where ρ_f is the upwinded density, $(\underline{u}.\underline{n})_f$ is calculated using Rhie and Chow interpolation [RC1983].

The diffusion coefficient can be represented by

$$\mathrm{DA}\left(= \Gamma_{\phi,f} A_f \frac{n_x d_x + n_y d_y + n_z d_z}{d}\right) \qquad (3.3.2.8)$$

where $\Gamma_{\phi,f}$ is an average of the cell centred values.

Equation (3.3.2.6) can now be written as:

$$\frac{\left[(\rho\phi) - (\rho\phi)^{old}\right]_P}{\delta t} V_P + \sum_{\substack{all \text{ faces} \\ \text{of FV}}} \left((FA)_f \phi_f - (DA)_f (\phi_A - \phi_P)\right) = S_{\phi,P} V_P \qquad (3.3.2.9)$$

In the above equation (3.3.2.9) there still exists a facial value for ϕ_f. This is most easily estimated using the upwind scheme [Pat1980]. If the convection flux is *leaving* the cell then $\phi_f = \phi_P$; if the convection flux is *entering* the cell then $\phi_f = \phi_A$. Therefore (3.3.2.9) becomes:

$$\frac{\left[(\rho\phi) - (\rho\phi)^{old}\right]_P}{\delta t} V_P + \sum_{\substack{all\text{faces} \\ \text{ofFV}}} \left[\max\left((FA)_f, 0\right)\phi_P - \max\left((-FA)_f, 0\right)\phi_A - (DA)_f (\phi_A - \phi_P)\right] = S_{\phi,P} V_P \qquad (3.3.2.10)$$

There are a number of possible estimates for the value of ϕ_f [Pat1980] in (3.3.2.9). These estimates are introduced by the addition of a function A(|P|) where P is the Peclet number ($= FA_f/DA_f$) and leads to:

$$\frac{\left[(\rho\phi)-(\rho\phi)^{old}\right]_P}{\delta t}V_P + \sum_{\substack{all\ faces\\ of\ FV}}\left[\begin{array}{l}\max\left((FA)_f,0\right)\phi_P - \max\left((-FA)_f,0\right)\phi_A \\ -(DA)_f A(|P|)(\phi_A-\phi_P)\end{array}\right] = S_{\phi,P}V_P \quad (3.3.2.11)$$

The various differencing schemes used for A(|P|) in (3.3.2.11) are given below in Table 3-1.

Table 3-1 - Table of Differencing Schemes

Differencing scheme	Formulae for A(|P|)
Upwind	1
Hybrid	Max(0, 1 – 0.5 |P|)
Central Difference	1 – 0.5 |P|
Power Law	Max(0, $(1 - 0.1\ \|P\|)^5$)
Exponential	|P| / $e^{|P|}$ - 1

Now by using the following notation $a_A = A(|P|).(DA)_f - \max((-FA)_f,0)$ and noting that $\max((FA)_f,0) = \max((-FA)_f,0) + FA_f$ equation (3.3.2.11) becomes:

$$\left\{\frac{(\rho V)_P}{\delta t} + \sum_{All\ faces} a_A + \sum_{All\ faces}(FA)_f\right\}\phi_P = \sum_{\substack{All\ adjacent\\ nodes}} a_A\phi_A + \frac{(\rho\phi)_P^{old}V_P}{\delta t} + S_{\phi,P}V_P \quad (3.3.2.12)$$

The source term S_ϕ can be a function of ϕ and should be written in the linearised form:

$$S_{\phi,P} = S_C + S_P\phi_P \quad (3.3.2.13)$$

The continuity equation has the following form:

$$\frac{\partial\rho}{\partial t} + \nabla(\rho\underline{u}) = 0 \quad (3.3.2.14)$$

The discretised form of the continuity equation is therefore:

$$\frac{\left[(\rho)-(\rho)^{old}\right]_P}{\delta t}V_P + \sum_{\substack{all\ faces\\ of\ FV}}\left(\rho_f(\underline{u}.\underline{\hat{n}})_f\right)A_f = 0 \quad (3.3.2.15)$$

By subtracting the discretised continuity equation (3.3.2.15) from the first coefficient of (3.3.2.12) the following is obtained

$$\left\{\frac{(\rho V)_P^{old}}{\delta t}+\sum_{All\ \text{faces}} a_A - S_P V_P\right\}\phi_P = \sum_{\substack{All\ \text{adjacent}\\ \text{nodes}}} a_A\phi_A + \frac{(\rho\phi)_P^{old} V_P}{\delta t} + S_C V_P \qquad (3.3.2.16)$$

The above equation can be written as:

$$a_P\phi_P = \sum_{\substack{All\ \text{adjacent}\\ \text{nodes}}} a_A\phi_A + b \qquad (3.3.2.17)$$

where

$$a_P = \left\{\frac{(\rho V)_P^{old}}{\delta t}+\sum_{All\ \text{faces}} a_A - S_P V_P\right\}$$
$$b = \frac{(\rho\phi)_P^{old} V_P}{\delta t} + S_C V_P \qquad (3.3.2.18)$$

This is the computational molecule for point p (c.f. Figure 3-1).

A set of equations like (3.3.2.17) exists at every discrete point in the computational domain and this can be written in matrix form as:

$$\underline{\underline{A}}\,\underline{\phi} = \underline{b} \qquad (3.3.2.19)$$

3.3.3 Explicit discretisation

In the preceding work an implicit temporal discretisation scheme has been used. An alternative, the explicit discretisation, is described here. The explicitly discretised general conservation equation (c.f. implicit equation 3.3.2.6) is given in (3.3.3.1).

$$\frac{[(\rho\phi)-(\rho\phi)^{old}]_P}{\delta t} V_P + \sum_{\substack{all\ \text{faces}\\ \text{of FV}}} \left(\rho_f (\underline{u}.\hat{\underline{n}})_f \phi_f^{ols} - \Gamma_\phi \left(\frac{\partial\phi}{\partial x} n_x + \frac{\partial\phi}{\partial y} n_y + \frac{\partial\phi}{\partial z} n_z\right)^{old}\right) A_f = S_{\phi,P} V_P \qquad (3.3.3.1)$$

which leads to (3.3.3.2) (c.f. implicit equation 3.3.2.17):-

$$a_P\phi_P = \sum_{\substack{All\ \text{adjacent}\\ \text{nodes}}} (a_A\phi_A)^{old} + b \qquad (3.3.3.2)$$

The difference between the explicit (see equations 3.3.3.1 and 3.3.3.2) and implicit (see equations 3.3.2.6 and 3.3.2.17) formulation is that the value of ϕ_p can now be calculated explicitly from the previously calculated old values of neighbouring ϕ and therefore there is no coupling of the neighbouring values of ϕ^{new} as found in the implicit formulation. This results in a set of independent ordinary differential

equations at each point that can be solved using methods such as a corrector-predictor or Runge-Kutta methods.

However this scheme can result in unphysical results if sufficiently small time steps are not utilised [Pat1980]. The size of these time step δt is limited by the Courant-Friedrichs-Levy (CFL) limit ($\delta t < \left.\frac{\delta x}{u}\right|_{\min}$) where δx is grid cell size and u is the velocity within that grid cell. The implicit formulation is not limited in this way so that longer time steps may be used to advance the temporal discretisation. Generally implicit methods are faster than explicit methods due to the longer time step length that can be utilised. However the use of explicit methods can be useful when small time steps are inherently required such as in the case of Large Eddy Simulation (LES).

3.4 The momentum equation

The conservation of momentum is shown below in (3.4.1).

$$\frac{\partial(\rho u_i)}{\partial t} + \nabla(\rho u_i \underline{u}) = \mu div(grad(u_i)) - \frac{\partial P}{\partial x_i} + S_i \qquad (3.4.1)$$

In terms of the general conservation equation (3.1.1) $\phi = u_i$. The source term absorbs all the other terms, although the pressure gradient term is treated separately and is described in section 3.4.4.

Each term of the momentum equation (3.4.1) will be described in the following sections.

3.4.1 Transient term

The transient term of the momentum equation can be discretised over the control volume V as follows

$$\begin{aligned} \iiint_V \frac{\partial(\rho u)}{\partial t} dV &\approx V\left(\frac{\partial(\rho u)}{\partial t}\right)_P \\ &\approx V\left(\frac{(\rho_P u_P)_P - (\rho_P u_P)_P^{old}}{\Delta t}\right) \end{aligned} \qquad (3.4.1.1)$$

In equation (3.4.1.1) V is the volume of the control volume about the node P. ρ_P is the density at the node P, Δt is the time step and the old superfix refers to the values at the previous time step.

3.4.2 Convection term

$$\begin{aligned}\iiint_V \nabla(\rho u\underline{u})dV &= \oiint_S \rho u\underline{u}.d\underline{S} \\ &= \sum_j \int_{F_j} \rho u\underline{u}.\underline{n}dS\end{aligned} \tag{3.4.2.1}$$

The volume integral has been changed to a surface integral using the divergence theorem. The summation is over all the faces that form the surface S of the control volume V. The integration over the face is

$$\int_{F_j} \rho u\underline{u}.\underline{n}\, dS = A_F \rho_F (\underline{u}.\underline{n})_F u_F \tag{3.4.2.2}$$

The convection flux at the face is defined as $FA = \rho_F(\underline{u}.\underline{n})_F A_F$. If the face is internal to the domain then Rhie and Chow interpolation (see section 3.10.1) is used to provide an estimate for the velocity at the face. If the face is a symmetry or wall boundary then there is no mass flux across the boundary so that FA = 0. If the boundary face is an inlet boundary condition then

$$FA = A_F \rho_F \sum_{k=x,y,z} v_k n_k \tag{3.4.2.3}$$

where $\underline{v}$ is the inlet velocity. There is no contribution due to an outlet fixed pressure boundary condition.

3.4.3 Diffusion term

$$\begin{aligned}\iiint_V \nabla(\mu\nabla u)dV &= \oint_S \mu\nabla u.\underline{n}dS \\ &= \sum_j \int_{F_j} \mu\nabla u.\underline{n}dS\end{aligned} \tag{3.4.3.1}$$

By substitution in equation (3.3.2.8) in section 3.3.2 the following is obtained

$$DA = (\mu A)_f \frac{n_x d_x + n_y d_y + n_z d_z}{d} \tag{3.4.3.2}$$

The following relation handles the boundary condition of a wall or an inlet.

$$DA = \frac{\mu_f A_f}{d_1} \tag{3.4.3.3}$$

3.4.4 Pressure Gradient

The pressure gradient is calculated using Gauss' divergence theorem:

$$\begin{aligned} \iiint_V \nabla_k P dV &= \oiint_S \underline{P}.\underline{n}_k dS \\ &= \sum_j \int_{F_j} \underline{P}.\underline{n}_k dS \end{aligned} \tag{3.4.4.1}$$

3.5 The mass continuity equation

The mass continuity equation takes the form:

$$\frac{\partial \rho}{\partial t} + \nabla(\rho \underline{u}) = \dot{m} \tag{3.5.1}$$

where $\dot{m}$ is a source of mass

In the terms of the general conservation equation $\phi = 1$ and $\Gamma = 0$.

3.6 Enthalpy

For a Fire model it is generally necessary to solve the enthalpy equation to calculate the spread of heat throughout the domain. The enthalpy equation has the same form as the general differential equation and is detailed below:-

$$\frac{\partial (\rho h)}{\partial t} + \nabla.(\rho \underline{u} h) = \nabla.\left(k \nabla \left(\frac{h}{c_p} \right) \right) + S_h \tag{3.6.1}$$

where k is the thermal conductivity, c_p the specific heat and S_h represents any source of enthalpy.

3.7 Turbulence model

3.7.1 Time Averaged Approach

Previously the conservation equations for continuity, momentum and the general conservation equations have been defined. These equations describe an instantaneous

value for the flow quantities but modellers are generally more interested in the time averaged quantity than the values due to random fluctuations. The instantaneous equations are therefore time averaged. The time dependent quantities can be considered to consist of a time averaged component $\tilde{\phi}$ and a random fluctuating component ϕ''.

The mean value $\bar{\phi}$ of a variable ϕ can be defined as

$$\bar{\phi} = 1/\Delta t \int_{t-1/2\Delta t}^{t-1/2\Delta t} \phi dt \tag{3.7.1}$$

The Favre (density weighted) average for compressible flows is defined as

$$\tilde{\phi} = \frac{\overline{\rho\phi}}{\bar{\rho}} \tag{3.7.2}$$

and the instantaneous value of ϕ can be described by

$$\phi = \tilde{\phi} + \phi'' \tag{3.7.3}$$

Applying Favre averaging to the conservation equations leads to the following, with the tilde dropped for clarity and superposed bars to indicate the Favre averaging.

$$\frac{\partial \rho}{\partial t} + \nabla.(\rho \underline{u}) = 0 \tag{3.7.4}$$

$$\frac{\partial (\rho u_i)}{\partial t} + \nabla(\rho u_i \underline{u}) = \mu div(grad(u_i)) - \frac{\partial P}{\partial x_i} + S_i - \frac{\partial}{\partial x_j}\left(\rho \overline{u_i'' u_j''}\right) \tag{3.7.5}$$

$$\frac{\partial (\rho h)}{\partial t} + \nabla.(\rho \underline{u} h) = \nabla.\left(k \nabla\left(\frac{h}{c_p} \right)\right) + S_h - \nabla.\left(\rho \overline{\underline{u}'' h''}\right) \tag{3.7.6}$$

The effect of Favre averaging on the general conservation equation (3.1.1) is similar to the enthalpy equation.

$$\frac{\partial (\rho\phi)}{\partial t} + \nabla.(\rho \underline{u}\phi) = \nabla.\left(\Gamma_\phi \nabla\phi\right) + S_h - \nabla.\left(\rho \overline{\underline{u}''\phi''}\right) \tag{3.7.7}$$

After the averaging process the continuity equation remains the same. The momentum (3.7.5) and general conservation equation (3.7.7) have both gained an additional term on their right hand sides. The $\rho\overline{(u_k'' u_l'')}$ term is known as the Reynolds (turbulent) stress. The $\rho\overline{\underline{u}''\phi''}$ term is known as the Reynolds (turbulent) flux. These terms need to be calculated via some form of turbulence model. The standard buoyancy modified k-ε turbulence model is used in SMARTFIRE and will be described below. Kumar [Kum1983] reviews the various methods of modelling turbulence from the point of view of a FSE.

The k-ε turbulence model is based on the eddy viscosity hypothesis [Bou1877], and two additional variables k (the turbulent kinetic energy) and ε (the dissipation rate) and leads to the following equations

$$\frac{\partial(\rho k)}{\partial t}+\nabla.(\rho\underline{u}k)=\nabla.\left(\left[\mu_{lam}+\frac{\mu_t}{\sigma_k}\right]\nabla k\right)+P+G-\rho\varepsilon \qquad (3.7.8)$$

$$\frac{\partial(\rho\varepsilon)}{\partial t}+\nabla.(\rho\underline{u}\varepsilon)=\nabla.\left(\left[\mu_{lam}+\frac{\mu_t}{\sigma_\varepsilon}\right]\nabla\varepsilon\right)+\frac{\varepsilon}{k}\left[C_{1\varepsilon}(P+G)(1+C_3)-C_{2\varepsilon}\rho\varepsilon\right] \qquad (3.7.9)$$

where (3.7.8) is the turbulent kinetic energy equation and (3.7.9) is the dissipation rate equation. In these equations, P represents the turbulent production rate:

$$P=2\mu_t\left\{\left(\left[\frac{\partial u}{\partial x}\right]^2+\left[\frac{\partial v}{\partial y}\right]^2+\left[\frac{\partial w}{\partial z}\right]^2\right)+\left(\frac{\partial u}{\partial y}+\frac{\partial v}{\partial x}\right)^2+\left(\frac{\partial u}{\partial z}+\frac{\partial w}{\partial x}\right)^2+\left(\frac{\partial w}{\partial y}+\frac{\partial v}{\partial z}\right)^2\right\} \qquad (3.7.10)$$

and G represents the buoyancy term:

$$G=-g\frac{\mu_t}{\rho}\frac{\partial\rho}{\partial y} \qquad (3.7.11)$$

The eddy viscosity hypothesis assumes that the turbulent stresses are proportional to the mean velocity gradients in a similar manner to viscous stresses in laminar flows. This is expressed as,

$$\rho\overline{(u_i''u_j'')} = \mu_t\left(\frac{\partial u_i}{\partial x_j}+\frac{\partial u_j}{\partial x_i}\right)-\frac{2}{3}\rho k\delta_{ij} \tag{3.7.12}$$

Similarly, in the general conservation equation the turbulent flux is assumed proportional to the gradient of the variable ϕ.

$$\rho\overline{u''\phi''} = \frac{\mu_T}{\sigma_T}\left(\frac{\partial \phi}{\partial x}\right) \tag{3.7.13}$$

The unknown fluctuating quantities have been replaced with known time averaged values. The last term in (3.7.12) is effectively a pressure term and becomes absorbed into the pressure gradient term of the momentum equation. The value of turbulent viscosity, μ_t, is calculated from (3.7.14).

$$\mu_t = \rho C_\mu \frac{k^2}{\varepsilon} \tag{3.7.14}$$

The constants used with the k-ε model are detailed in Table 3-2 below.

Table 3-2 - Constants used in k-ε model

C_μ	σ_k	σ_ε	$C_{1\varepsilon}$	$C_{2\varepsilon}$	C_3
0.09	1.0	1.3	1.44	1.92	1.0

3.7.2 Direct Calculation

Time averaging is not necessary if the modeller is prepared to resolve the simulation to the finest time and length scales to resolve all the turbulent eddies. This approach is known as Direct Numerical Simulation (DNS) and is the most realistic approach to modelling turbulence as a turbulence model is no longer needed. However with this approach very fine meshes and small time step sizes are required to resolve all the turbulent eddies. This means that even with modern computing power of a PC even a relatively simple fire enclosure would be prohibitively expensive to model. Large Eddy Simulation (LES) [Sma1963] is an attempt to maintain the accuracy of DNS but without the mesh sizes required to resolve the finest eddies.

3.7.3 Large Eddy Simulation model

In the LES model the instantaneous equations are not time averaged but filtered to remove turbulent eddies smaller than a certain spatial scale, normally the resolution of the computational cell. Mathematically this is represented by (3.7.3.1).

$$\bar{f}(x) = \int_D f(x')G(x,x';\overline{\Delta})dx' \qquad (3.7.3.1)$$

where the overbar represents the filtered function, D is the entire domain, G is the filter function and $\overline{\Delta}$ is the filter width. The filter function determines the size and structure of the small scales. Similarly to the Reynolds averaging process the variable can be decomposed into two parts.

$$\phi = \bar{\phi} + \phi' \qquad (3.7.3.2)$$

however, the definition is different. ϕ' represents the fluctuation of ϕ at length scales less than $\overline{\Delta}$. $\bar{\phi}$ represent the fluctuations of ϕ at length scales greater than $\overline{\Delta}$. Applying the filtering to the momentum equation and dropping the overbars leads to (3.7.3.3).

$$\frac{\partial(\rho u_i)}{\partial t} + \nabla(\rho u_i \underline{u}) = \mu div(grad(u_i)) - \frac{\partial P}{\partial x_i} + S_i - \frac{\partial}{\partial x_j}(\rho\tau_{ij}) \qquad (3.7.3.3)$$

The filtered fields do not need to resolved at scales less than $\overline{\Delta}$ and can therefore be calculated properly. The effect of the small scales appears through the last term, $\tau_{ij} = \left(\overline{u_i u_j} - \bar{u}_i \bar{u}_j\right)$ the sub-grid-scale (SGS) stress tensor. This term needs to be modelled.

Using an eddy-viscosity model the stress term can be modelled using (3.7.3.4).

$$\tau_{ij} - \frac{1}{3}\tau_{kk}\delta_{ij} = -\nu_t\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) = -2\nu_t \bar{S}_{ij} \qquad (3.7.3.4)$$

where S_{ij} is the deformation tensor of the filtered velocity (velocity gradients). Similar to the time-averaged approach the unknown variables have been replaced with known filtered values. The only term that is now required is the LES turbulent viscosity. The most widely used model was proposed by Smagorinsky [Sma1963]. A local mixing-length assumption is made, in which the eddy viscosity is assumed to be proportional to the sub-grid-scale length $\overline{\Delta}$, and the characteristic turbulent velocity determined by $\overline{\Delta}\left|\overline{S}_{ij}\right|$. The LES turbulent viscosity is given by (3.7.3.5).

$$\nu_t = \left(C_S \overline{\Delta}\right)^2 \left|\overline{S}_{ij}\right| \tag{3.7.3.5}$$

C_S is a 'constant' although this may be variable depending on the modelling chosen. Frequently this term is damped where the turbulence is not totally developed, during transition or near walls.

With LES modelling small time steps are required for time accuracy. This means that the advantages of an implicit scheme, which can use larger time steps are lost, so that the use of explicit methods becomes more attractive. By using an explicit method a set of ordinary differential equations are formed for the thermodynamic quantities. In the NIST FDS [MBR+2001] model a predictor-corrector method is employed for time advancement solution of these ordinary differential equations. The size of the time step δt is limited by the CFL limit that is defined by (3.7.3.6).

$$\delta t < \left.\frac{\partial x}{v}\right|_{\text{minimum}} \tag{3.7.3.6}$$

where δx is the grid spacing and v is the velocity in that grid cell. Due to this method being employed each time step is solved relatively quickly but each time step only represents a small advance in time by small fractions of a second. This leads to the method being potentially computationally expensive for modelling large time scales. The main attraction of LES is that the large eddies formed by fire plumes are lost, or averaged out, when a time averaged approach is used. When visually compared with time averaged simulations, LES simulations look more realistic with the turbulent nature of the plume apparent. However from an engineering viewpoint time averaged results are still extremely useful for accessing fire dynamics. The main advantage of

the time averaged approach is the faster computation time. LES may also be useful in researching the properties of time averaged turbulence models.

3.8 Combustion modelling

In most combustion processes the actual mechanisms of combustion are extremely complex involving many intermediate chemical species. In some cases, the exact nature of this process is unknown, and even if it is well characterised, the modelling requirements would be prohibitive. Generally, a one step chemical reaction is used to represent the combustion process, allowing the process to be more easily modelled. This can be represented by (3.8.1).

$$1 \text{ kg fuel} + s \text{ kg oxidant} \rightarrow (1+s) \text{ kg product} \ (+ \text{ heat}) \tag{3.8.1}$$

alternatively this can be described by

$$f_{st} \text{ kg fuel} + (1-f_{st}) \text{ kg oxidant} \rightarrow 1 \text{ kg product} \tag{3.8.2}$$

where $f_{st} = 1/(1 + s)$ and is known as the stoichiometric value.

The mass fractions of fuel, oxidant and product can be calculated over the entire domain using this one step global reaction scheme. These values may then be used to calculate the fluid properties depending on the assumptions that have been made. Typically, the ideal gas law (3.8.3) is used to calculate the density.

$$\rho = \frac{PW}{RT} \tag{3.8.3}$$

where p is the pressure, T the temperature, R the gas constant and W the mixture molecular weight calculated using the equation

$$\frac{1}{W} = \frac{m_{fu}}{W_{fu}} + \frac{m_{ox}}{W_{ox}} + \frac{m_{pr}}{W_{pr}} \tag{3.8.4}$$

where the subscripts stand for fuel (fu), oxidant (ox) and product (pr). m is the mass fraction for each of these components.

Similarly a specific heat capacity for the mixture can be calculated using

$$c_{p,mixture} = m_{fu}c_{p,fu} + m_{ox}c_{p,ox} + m_{pr}c_{p,pr} \tag{3.8.5}$$

The enthalpy of the mixture, h, is now modified so that it includes the contribution from the heat of reaction of the fuel.

$$h = c_p T + m_{fu}\Delta H_C \tag{3.8.6}$$

A model is now required to calculate the distribution of the mass fractions m_{fu}, m_{ox} and m_{pr}.

The Eddy break-up model is described by the following equations.

$$\frac{\partial(\rho f)}{\partial t} + \nabla.(\rho \underline{u} f) = \nabla.\left(\Gamma_\phi \nabla f\right) \tag{3.8.7}$$

where the conserved quantity f is the mixture fraction. Γ is the diffusion coefficient for the mixture fraction. Its value is conventionally taken to be the same as the value of the diffusion coefficient in the enthalpy equation, i.e. the thermal conductivity divided by the specific heat. The mixture fraction f is defined as:

$$f = \frac{\chi - \chi_{ox}}{\chi_{fu} - \chi_{ox}} \tag{3.8.8}$$

where

$$\left.\begin{aligned} \chi &= m_{fu} - \frac{m_{ox}}{s} \\ \chi_{ox} &= -\frac{1}{s};\ \chi_{fu} = 1 \end{aligned}\right\} \tag{3.8.9}$$

As well as solving the mixture fraction, the mass fraction of fuel is also solved.

$$\frac{\partial(\rho m_f)}{\partial t}+\nabla.(\rho \underline{u} m_f)=\nabla.\left(\Gamma_\phi \nabla m_f\right)+R_{m_f} \tag{3.8.10}$$

where the source term R_{mf} represents the reaction rate of fuel and the diffusion coefficient, Γ, takes the same value as the mixture fraction diffusion coefficient. The Eddy Break-up model assumes that the reaction rate is controlled by the turbulent eddies which cause the species to mix. This is modelled by making the reaction rate a function of the levels of the turbulent variables k and ε (3.8.11) [MH1977]. This expression takes into account both lean oxygen and lean fuel scenarios.

$$R_{m_f}=A\frac{\varepsilon}{k}\times\min\left(\overline{C}_f,\overline{C}_o/s\right) \tag{3.8.11}$$

where A is a model constant, C is the species concentration.

The mass fraction of the oxidant and product are calculated using the following formulae.

$$m_{ox}=s\left(m_{fu}-\frac{f-f_{st}}{1-f_{st}}\right) \tag{3.8.12a}$$

$$m_{pr}=1-m_{ox}-m_{fu} \tag{3.8.12b}$$

3.9 Radiation Modelling

Radiation is a significant form of heat transfer in many fire scenarios and therefore needs to be modelled. What follows is a brief description; a fuller description is given by Jia [Jia1999]. The radiative transfer is described by the following equation. In this equation, it is assumed that it is a grey system, i.e. no wavelength dependence.

$$\frac{\partial}{\partial l}I(\Omega,r)=-(\alpha+s)I(\Omega,r)+\alpha I_b(r)+\frac{s}{4\pi}\int_{\Omega'=4\pi}I(\Omega',r)\Phi(\Omega'\rightarrow\Omega)d\Omega' \tag{3.9.1}$$

where Ω is a specified direction; l represents the physical pathlength along Ω; r is the position of a point; $I(\Omega,r)$ is the radiation intensity along Ω at position r; α is the

absorption coefficient and s is the scattering coefficient of the medium; $I_b(r)$ is the blackbody radiation intensity; and $\Phi(\Omega' \rightarrow \Omega)$ is the scattering phase function.

The above equation is extremely difficult to solve and a number of simplifying assumptions has led to many different radiation models [How1988]. These include the zone method [HS1967], Monte Carlo methods [How1968], composite flux models (six-flux) [Ham1947a, b; Sch1905] and discrete transfer models [LS1981]. The zone and Monte Carlo methods are both computationally very expensive unless coarse meshes are used. Due to the fine meshes that are generally used in fire modelling neither of these techniques is suitable.

The six-flux model and the discrete transfer model are more suitable for fire modelling because of the reduced computational cost. These methods will be briefly described below.

3.9.1 Six flux model

The six-flux model has been successfully used in fire application in the past [HM1988; JGP1997, 1999]. In this model the discretisation around the solid angle 4π results in 6 fluxes along the positive and negative co-ordinate directions. The resulting equations are given below.

$$\frac{dI}{dx} = -(\alpha + s)I + \alpha E + \frac{s}{6}(I + J + K + L + M + N) \qquad (3.9.1.1)$$

$$\frac{dJ}{dx} = +(\alpha + s)J - \alpha E - \frac{s}{6}(I + J + K + L + M + N) \qquad (3.9.1.2)$$

$$\frac{dK}{dy} = -(\alpha + s)K + \alpha E + \frac{s}{6}(I + J + K + L + M + N) \qquad (3.9.1.3)$$

$$\frac{dL}{dy} = +(\alpha + s)L - \alpha E - \frac{s}{6}(I + J + K + L + M + N) \qquad (3.9.1.4)$$

$$\frac{dM}{dz} = -(\alpha + s)M + \alpha E + \frac{s}{6}(I + J + K + L + M + N) \qquad (3.9.1.5)$$

$$\frac{dN}{dz} = +(\alpha + s)N - \alpha E - \frac{s}{6}(I + J + K + L + M + N) \qquad (3.9.1.6)$$

$$E = \sigma T^4 \qquad (3.9.1.7)$$

where α is the absorption coefficient, s is the scattering coefficient, E is the black body emissive power of the fluid and I, J, K, L, M and N are the are radiative fluxes in the six co-ordinate directions. I, K and M are the positive fluxes in the x, y and z co-ordinate directions and J, L and N are the negative fluxes in the x, y and z co-ordinate directions. The six-flux model is traditionally implemented as described in section 3.9.1.1. and is the method utilised in the FIREDASS model. Jia's alternative implementation is also described, as this is the six-flux model used within the SMARTFIRE fire modelling code.

3.9.1.1 Traditional Six-Flux model

In the traditional discretisation, but not in the Jia modified model [Jia1999], of the six-flux model the fluxes in the positive and negative directions are combined using the following definition (3.9.1.1.1) .

$$RX = \frac{(I+J)}{2}; RY = \frac{(K+L)}{2}; RZ = \frac{(M+N)}{2} \qquad (3.9.1.1.1)$$

with this definition the six-flux equations (3.9.1.1)- (3.9.1.6) can be manipulated to produce three second-order ordinary differential equations (3.9.1.1.2)-(3.9.1.1.4) in terms of the composite fluxes RX, RY and RZ.

$$\frac{d}{dx}\left[\frac{1}{\alpha+s}\frac{d(RX)}{dx}\right] = (\alpha+s)RX - 2\alpha E - \frac{s}{3}(RX+RY+RZ) \qquad (3.9.1.1.2)$$

$$\frac{d}{dy}\left[\frac{1}{\alpha+s}\frac{d(RY)}{dy}\right] = (\alpha+s)RY - 2\alpha E - \frac{s}{3}(RX+RY+RZ) \qquad (3.9.1.1.3)$$

$$\frac{d}{dz}\left[\frac{1}{\alpha+s}\frac{d(RZ)}{dz}\right] = (\alpha+s)RZ - 2\alpha E - \frac{s}{3}(RX+RY+RZ) \qquad (3.9.1.1.4)$$

The net radiative heat fluxes are defined as follows

$$Q_{RX} = I - J = \frac{2}{(\alpha+s)}\frac{d(RX)}{dx} \qquad (3.9.1.1.5)$$

with similar expressions for Q_{RY} and Q_{RZ}.

The radiation source to the enthalpy equation is given by the following expression.

$$S_{h,rad} = -\nabla(Q_{RI}) = 2\alpha[(RX + RY + RZ) - 6E] \qquad (3.9.1.1.6)$$

The main attraction of this method is that it is easily incorporated into the finite volume scheme used by many CFD based fire models.

The boundary condition at a wall is given by the following equation

$$\left.\frac{d(RX)}{dx}\right|_{w_x} = \frac{(\alpha + s)[\varepsilon_w (RX_w - 2E_w)]}{2 - \varepsilon_w} \qquad (3.9.1.1.2)$$

with similar expressions for RY and RZ.

3.9.1.2 Jia's modified Six-Flux Radiation model

In the SMARTFIRE modified six-flux radiation model [Jia1999] scattering is neglected (i.e. s = 0) as the effect of scattering is unimportant for the applications SMARTFIRE is intended for [Sar1986] e.g. enclosure fires.

Instead of forming 3 second order ordinary differential equations from the original six-flux equations (3.9.1.1)-(3.9.1.6), Jia solves these original six-flux first order ordinary differential equations. Jia argues that the traditional formulation can lead to ill-posed problems and this is avoided by solving the original first order equations instead.

The transfer of heat through radiation gives the following source to the enthalpy equation.

$$S_{h,6-flux} = a((I - E) + (K - E) + (M - E) + (J + E) + (L + E) + (N + E)) \qquad (3.9.1.2.1)$$

In this model the boundary condition is given by the following expression

$$I = \varepsilon_w E_b + (1 - \varepsilon_w) J \qquad (3.9.1.2.1)$$

Due to the dependence of I and J, iteration is required to solve both of these equations unless ε_w is 1. Similar expressions exist for K and L, and M and N.

The main disadvantage of this method over the traditional method is that there are now six equations to solve instead of three.

3.9.2 Discrete Transfer Model

The six-flux model is not intended for applications where the accuracy of the heat flux at a solid surface is a crucial component of the calculations, such as situations involving flame spread over solid surfaces or when structural interaction with the fire is being predicted. It is intended for applications where the dominant factor is the radiative heat loss from the flame. This is commonly the situation when representing non-spreading fires. The discrete transfer model provides the increased angular resolution required for situations such as flame spread.

The discrete transfer model was proposed by Lockwood and Shah [LS1981]. The RTE is solved along a number of prescribed ray directions that discretise the 4π solid angle. By neglecting scattering, the radiation along a ray direction is given by

$$\frac{\partial}{\partial l} I(\Omega, r) = -\alpha I(\Omega, r) + \alpha I_b(r) \qquad (3.9.2.1)$$

If the domain is divided into a number of zones where the optical properties and temperature are constant the above can be integrated to give the following recurrence equation.

$$I_{n+1} = I_n e^{-\alpha \delta l} + I_b\left(1 - e^{-\alpha \delta l}\right) \qquad (3.9.2.2)$$

Where I_{n+1} is the radiation intensity leaving the zone and I_n is the radiation intensity entering the zone via the ray. δl is the length of the ray segment in the zone. In SMARTFIRE, each zone is simply mapped (1 to 1) to each of the computational cells used to calculate the other field properties. In some other codes the zone maybe more coarsely defined with its properties defined by an average of the computational cell properties encapsulated within the zone.

The solid angle is divided up into n parts and in each part a ray is projected to represent the radiation. Generally the more rays that are used the better the radiation is distributed but at the cost of higher computational effort. There are a number of rules associated with the ray distribution which are described by Lathrop and Carlson [LC1965].

The wall boundary condition is calculated using the equation below.

$$I = \frac{\varepsilon_w I_B}{\pi} + (1-\varepsilon_w) I^- \tag{3.9.2.3}$$

where I^- is the incident radiation on the wall.

The following relation represents the energy transfer into the enthalpy equation.

$$S_{h,DTM} = \sum (I_{n+1} - I_n)\vec{\Omega} \cdot \Delta\vec{A}\Delta\Omega \tag{3.9.2.4}$$

rays

3.10 Staggered and Co-located meshes

When the solution scheme for the pressure and velocity equations was originally devised it was discovered that there was an oscillation in the pressure and velocity fields. This was due to the strong coupling between the pressure and velocity and when the equations were discretised alternate nodes would be decoupled leading to the so-called checkerboard effect. To overcome these difficulties two methods have been devised.

The first method devised was the staggered grid approach. In this approach all the values are solved at the cell centre except the velocity components which would be solved at the cell faces and is described as the staggered grid approach. The main advantages of the staggered grid approach are:

- The velocity is situated at the cell faces so there is no need interpolate to the faces.

- The difference between adjacent velocity nodes is used in the discretisation instead of the difference between alternate nodes. This removes the problem associated checkerboard effect.
- The velocity is situated between adjacent pressure nodes which is the driving force of the velocity

The main disadvantage of this approach is the need to store a grid for each velocity component. This increases the amount of memory and book-keeping required. In the case of an unstructured mesh it may be difficult to define a suitable staggered mesh.

An alternative approach is the use of a co-located (unstaggered) grid with the use of a special velocity interpolation scheme devised by Rhie and Chow.

The main advantages of this method are:

- Less storage space is required.
- The geometric bookkeeping is much simpler making the use of unstructured meshes more viable
- Removes the decoupling of alternate nodes.

The main disadvantage is that the cell centred value calculated need not satisfy continuity. Only the facially interpolated values are constrained to this.

3.10.1 Rhie and Chow interpolation

The Rhie and Chow interpolation method [RC1983] is used to interpolate cell centred velocities to the cell faces but removing the checkerboard effect reported by Patankar [Pat1980] which was previously avoided by using a staggered grid approach. The explanation of the interpolation method has been deferred until now as the method makes use of the discretised momentum equation.

The discretised form of the u momentum equation for a control volume about a node P can be expressed in the following manner

$$a_P u_P + (\nabla P)_P = \left(\sum a_{nb} u_{nb}\right)_P + S_P \qquad (3.10.1.1)$$

Similarly for an adjacent node

$$a_A u_A + (\nabla P)_A = \left(\sum a_{nb} u_{nb}\right)_A + S_A \qquad (3.10.1.2)$$

From the conservation principle of the finite volume formulation the u velocity component at a point on the face between these two nodes must also have a corresponding discretised momentum equation of a similar form

$$a_f u_f + (\nabla P)_f = \left(\sum a_{nb} u_{nb}\right)_f + S_f \qquad (3.10.1.3)$$

The Rhie and Chow interpolation uses the equations (3.10.1.1) and (3.10.1.2) to approximate a solution to equation (3.10.1.3). It is assumed that the terms on the RHS of (3.10.1.3) may be approximated by a linear weighted average of (3.10.1.1) and (3.10.1.2).

$$a_f u_f + (\nabla P)_f = \overline{\left(\sum a_{nb} u_{nb}\right)_f} + \overline{S_f} = \overline{a_f u_f} + \overline{(\nabla P)_f} \qquad (3.10.1.4)$$

The overbar represents the weighted linear average for the variable. It is now further assumed that $a_f \approx \overline{a_f}$ so that

$$u_f = \overline{u_f} + \overline{d_f}\left(\overline{(\nabla P)_f} - (\nabla P)_f\right) \qquad (3.10.1.5)$$

where

$$\begin{aligned} \overline{u}_f &= \alpha u_P + (1-\alpha) u_A \\ \overline{\nabla P}_f &= \alpha (\nabla P)_P + (1-\alpha)(\nabla P)_A \\ \nabla P_f &= A_f n_x (P_P - P_A) \\ a_f &= \alpha a_P + (1-\alpha) a_A \\ \overline{d_f} &= 1/a_f \end{aligned} \qquad (3.10.1.6)$$

3.11 Solution method

The set of equations $\underline{\underline{A}}\underline{\phi} = \underline{b}$ now needs to be solved. All the values of ϕ are generally unknown except those specified at the boundary conditions. Because there is a large amount of coupling between the equations so they can not be solved sequentially but must be solved iteratively.

The main problem with the physical equations involved is due to the strong coupling between the pressure and velocity fields. The momentum equation is dominated by

the pressure term so an accurate knowledge of the pressure field is required. However, the pressure field is hard to determine as it is only indirectly specified by the continuity equation. An estimate of the pressure field requires an accurate measurement of the velocity field for the continuity equation. Hence the velocity and pressure fields are strongly coupled.

The problem was solved by Patankar and Spalding [PS1972] with the iterative solution scheme known as the Semi Implicit Method for Pressure Linked Equations (SIMPLE). The procedure has been widely adopted by the CFD community and has led to variations such as SIMPLER [Pat1980] and SIMPLEC [VR1984].

The SIMPLE method is the basic method used for solving the physical equations used.

1. Start with an initial guess for all the variables.
2. Solve the enthalpy equation
3. Solve radiation, combustion, soot, turbulence and any other extra equations, if required
4. Solve the momentum equations for each phase obtaining a new velocity field. This velocity field will generally not satisfy the joint continuity equation
5. Solve the continuity equation to get the pressure correction.
6. Calculate the pressure correction to the velocity components of each phase. These corrected velocity field will generally not satisfy the momentum equations.
7. Goto 2 until a converged solution is achieved
8. Increment to the next time interval and goto 2 until all the time steps have been completed.
9. End

3.11.1 The mass continuity equation

The mass continuity equation takes the following form:

$$\frac{\partial \rho}{\partial t} + \nabla(\rho \underline{u}) = 0 \qquad (3.11.1.1)$$

This discretises to:

$$\frac{\rho_p V_p - \rho_p^{old} V_p^{old}}{\Delta t} + \sum_{faces} (\rho \underline{u}.\underline{n})_f A_f = 0 \qquad (3.11.1.2)$$

During the iterative cycle (3.11.1.2) will not generally sum to zero, so a correction is applied to the pressure field so that mass continuity is satisfied. The pressure correction is brought in via the Rhie and Chow interpolation method. The Rhie and Chow interpolation states that

$$u_f = \bar{u}_f + \overline{d_f}\left(\overline{\nabla P} - \nabla P\right)_f \qquad (3.11.1.3)$$

this leads to:

$$u_f = \frac{n_x A_f}{\overline{a_{P,u}}}\left((P+P')_P - (P+P')_A\right) + \bar{u}_f + \overline{d\nabla P_f} \qquad (3.11.1.4)$$

where P' is the pressure correction at the node. Using the above relation (3.11.1.4) a computational molecule for P' can be formed, which can then be solved like any other matrix equation.

3.11.2 Pressure and Velocity correction

The pressure corrections obtained from the mass continuity equation need to be added to the previous iterations pressure field and the corrections to the velocity field must also be calculated. It is known that u = u* + u', where u is the exact velocity field, u* is the current velocity field and u' is the error. Referring to the discretised momentum equations (3.11.2.1) and (3.11.2.2):

$$a_P u_P = \sum a_A u_A - \nabla P + S_P \qquad (3.11.2.1)$$

$$a_P u_P^* = \sum a_A u_A^* - \nabla P^* + S_P \qquad (3.11.2.2)$$

subtracting (3.11.2.1) from (3.11.2.2) gives:

$$a_P u_P' = \sum a_A u_A' - \nabla P + \nabla P^* \qquad (3.11.2.3)$$

The $\sum a_A u_A'$ term is now dropped from the equation. This is a reasonable assumption as this term will tend to zero as convergence is approached and will not effect the final solution. This leads to the following equation

$$a_P u'_P = -\left(\nabla P - \nabla P^*\right) \qquad (3.11.2.4)$$

For an internal cell

$$\nabla P = \sum_{All\ \text{faces}} A_f \left(\alpha P_P + (1-\alpha) P_A\right) n_{x,f} \qquad (3.11.2.5)$$

and

$$\nabla P^* = \sum_{All\ \text{faces}} A_f \left(\alpha P_P^* + (1-\alpha) P_A^*\right) n_{x,f} \qquad (3.11.2.6)$$

then

$$\nabla P' = \nabla P - \nabla P^* = \sum_{All\ \text{faces}} A_f \left(\alpha P_P' + (1-\alpha) P_A'\right) n_{x,f} \qquad (3.11.2.7)$$

The pressure correction to the v and w velocity components are similarly derived.

3.11.3 FDS solution procedure

NIST's FDS CFD model uses explicit time discretisation and therefore requires a different solution strategy [MBR+2001]. Briefly summarised the equations are solved using a predictor-corrector scheme except for the pressure equation which is derived by applying divergence to the momentum equation which forms a Poisson equation which is solved using a fast Fourier transform.

3.12 Boundary conditions

The definition of the fire problem also requires a set of boundary conditions to be specified. The boundary conditions that are typically used by a FSE are:

- Inlet
- Wall
- Pressure Boundary

3.12.1 Inlet

At an inlet, the velocity into the domain is specified. At this inlet, it is also necessary to specify the value of the other fluid properties (ϕ_{inlet}). In a cell containing an inlet the FA_{inlet} term is specified and DA_{inlet} can be calculated and leads to the calculation of the a_{inlet} coefficient. This leads to the following source being added to the any of the discretised equations.

$$S_P = a_{inlet}$$
$$S_C = a_{inlet}\phi_{inlet} \qquad (3.12.1.1)$$

This is a fixed value boundary condition.

3.12.2 Wall Boundary condition

Walls are used to confine the flow within an enclosure. No mass flux passes through a wall, therefore FA_{wall} is zero. It is further assumed that at the wall, the velocity parallel to the wall is zero.

3.12.2.1 Momentum Equation

In the discretised momentum equation the shear force, F_s, in a cell containing a wall is described by

$$F_S = -\tau_{wall} A_{wall} \qquad (3.12.2.1.1)$$

where $\tau_{wall} = \mu \frac{u_P}{\Delta y_P}$ in the near wall laminar flow. Δy_p is the distance of the wall from the node P.

This leads to the following source term being added to the discretised momentum equation.

$$S_P = -\frac{\mu}{\Delta y_P} A_{wall} \qquad (3.12.2.1.2)$$

If the flow is turbulent, the node P may lie in the laminar sub-layer in which case the above relation for laminar flow is used. If node P lies in the turbulent layer then special log-law functions are used to model the shear force. The following relation is used to determine where node P lies.

$$y^+ = \frac{\sqrt{\rho \tau_w}}{\mu} \Delta y_P \qquad (3.12.2.1.3)$$

If y^+ (3.12.2.1.3) is less than 11.63 then the near wall flow is laminar. If y^+ is greater than 11.63 then the near wall flow is turbulent. The turbulent shear stress is given

below. The source term in the discretised momentum equation is derived in the same fashion as that for the laminar shear force.

$$\tau_{wall} = \frac{\rho C_{\mu}^{1/4} k_P^{1/2} u_P}{u+} \qquad (3.12.2.1.4)$$

where u^+ is defined as

$$u^+ = \frac{1}{\kappa} \ln(Ey^+) \qquad (3.12.2.1.1)$$

where κ and E are model constants.

3.12.2.2 Enthalpy Equation

For the enthalpy equation for a wall surface temperature, T_w, the convective heat flux into the wall is

$$q_s = h_c (T_P - T_w) \qquad (3.12.2.2.1)$$

where the laminar heat transfer coefficient is

$$h_c = \frac{k}{\Delta y_P} \qquad (3.12.2.2.2)$$

where k is conductivity coefficient.

The turbulent heat transfer coefficient is

$$h_c = \frac{\rho c_P C_{\mu}^{1/4} k_P^{1/2}}{T+} \qquad (3.12.2.2.3)$$

where k is turbulent kinetic energy and

$$T^+ = \frac{\sigma_T}{\kappa} \ln(E_T y^+) \qquad (3.12.2.2.4)$$

T_w can either be prescribed or it can be calculated based on the material and physical properties of the wall and the net heat flux on the wall. This leads to an iterative process as the net heat flux is dependent on T_w. Details of an efficient calculation of T_w has been described by Jia [Jia1999].

3.12.2.3 Turbulence Equations

3.12.2.3.1 Kinetic energy equation

No special modification is required for the k equation apart from the turbulent viscosity at the wall is dependent on the wall shear stress instead of the standard formula (3.7.14).

3.12.2.3.2 Dissipation Rate equation

For the dissipation rate the value of ε in the near wall cell is given by

$$\varepsilon = \frac{C_{\mu}^{3/4} k^{3/2}}{\kappa \Delta y_P} \qquad (3.12.2.3.2.1)$$

3.12.2.4 Pressure Boundary

A pressure boundary is used when the exact details of the flow are unknown but the boundary values of the pressure are known. This boundary condition is typically used in fire modelling for allowing mass to vent in and out of a specified fire domain. Care must be taken to ensure that the pressure boundary is far enough away from the area of interest so that the boundary does not unduly influence the flow in the area of interest. This generally leads to an extended region beyond the enclosure being modelled. At a pressure boundary, the value of P is prescribed. As P is prescribed the pressure correction, P', is zero at the pressure boundary.

3.13 Solvers

The above equations have now been discretised, using an implicit formulation, and now need to be solved via an appropriate numerical matrix solver. Direct methods, such as Gaussian elimination, are generally not usable due to the high memory requirements and the amount of time involved in obtaining a solution. Many CFD codes, including SMARTFIRE, utilise iterative solvers. These solvers calculate a value based on a set of previously calculated values. This process is repeated until the values achieve convergence. The two solver methods generally used by SMARTFIRE are the Jacobi Over Relaxation (JOR) method and the Successive Over Relaxation method (SOR). Other methods are available such as the conjugate gradient method [HS1952] but they will not be discussed in this thesis.

3.13.1 JOR method

The JOR (Jacobi Over Relaxation) method is based on the Jacobi iterative method. In this method the i+1 iteration value of ϕ in the p'th element (ϕ_p^{i+1}) is obtained using the formula:-

$$\phi_p^{i+1} = \frac{1}{A_{pp}}\left(b_p - \sum_{k \neq p} A_{pk}\phi_k^i\right) \qquad (3.13.1.1)$$

The JOR method adds an over relaxation term

$$\phi_p^{i+1} = (1-\alpha)\phi_p^i + \frac{\alpha}{A_{pp}}\left(b_p - \sum_{k \neq p} A_{pk}\phi_k^i\right) \qquad (3.13.1.1)$$

Although over relaxation term is used to accelerate convergence ($\alpha > 1$) it is more usual in CFD based fire codes to under relax the solver ($\alpha < 1$) to give greater solution stability. The JOR method is slow compared to most methods but is very stable.

The above iterative formula is used with the following algorithm.

1) loop over the cells p = 1 to N_{cell} applying equation (3.13.1.1) to calculate ϕ_p for each cell.
2) repeat 1) until the prescribed number of iterations have been performed or convergence has been achieved.

The new value of ϕ_p^{i+1} is dependent on the previous neighbouring values of ϕ_k^i and is independent of ϕ_k^{i+1}. This means that the cell ordering makes no difference to the obtained value of ϕ_p^{i+1}. There may be slight differences due to machine precision but this is generally a very minimal effect.

3.13.2 SOR method

The SOR (Successive Over Relaxation) method is based on Gauss-Seidel iterative scheme

$$\phi_p^{i+1} = \frac{1}{A_{pp}}\left(b_p - \sum_{k \neq p} A_{pk}\phi_k^m\right) \qquad (3.13.2.1)$$

where m = i + 1 if k < p; m = i if k > p

$$\phi_p^{i+1} = (1-\alpha)\phi_p^i + \frac{\alpha}{A_{pp}}\left(b_p - \sum_{k \neq p} A_{pk}\phi_k^m\right) \qquad (3.13.2.2)$$

The SOR method is generally faster than the JOR method. The same algorithm as used in the JOR case is utilised but equation (3.13.2.2) is used instead. The new value of ϕ_p^{i+1} is dependent on the neighbouring values of ϕ_k^{i+1}. This means that the cell ordering does makes difference to the obtained value of ϕ_p^{i+1}. However, whatever the cell ordering the value of ϕ_p will be the 'same' at convergence. There may be slight differences due to machine precision but this is generally a very minimal effect.

As with the JOR solver it is more usual to under relax the solver ($\alpha < 1$) to increase solution stability.

In SMARTFIRE the JOR solver is generally used for the momentum equations and the SOR solver is used to solve the other equations.

4 The development of a CFD based simulator for water mist fire suppression systems (FIREDASS)

In this chapter, the FIREDASS (FIRE Detection And Suppression Simulation) project [Fir, Goo2000, MGG+2000, GGPM1998a, GGPM1998b, Odi1999, OM1998, OM1998, KB1997, Ker1997b, KSBM2000, MGPG1998, GGPM1999] will be described. This ambitious research and development project resulted in the creation of a complex, state of the art, CFD based fire simulation model that was capable of fully modelling the interactions between a fine water mist, the radiation field, oxygen concentration, and the fire load. In addition the model allowed the use of virtual sensors to monitor physical properties which could then be used to control the water mist suppression using a range of possible control algorithms. Emphasis will be given to the sub-models developed by the author, namely the fire sub-model and the suppression sub-model and the verification and validation of these models. Finally, the full FIREDASS model will be demonstrated on a full A340 type C cargo bay mock up using all the sub-models developed within the project.

4.1 Introduction

The advent of the Montreal Protocols (1986) where a world-wide ban on the production and use of Chlorofluorocarbons (CFCs) [Mau1990] was agreed has generated an urgent requirement for alternative fire control and suppression systems. This requirement is particularly urgent for aircraft manufacturers and operators who use halon based suppression and extinguishment systems exclusively in aircraft cargo holds due to their light weight and proven effectiveness. The aviation industry has therefore been investigating alternative suppression systems, including powders, other gases and water mist.

An associated problem with aircraft fire safety concerns the fire detection systems installed in aircraft cargo holds. Currently approximately 95% of all reported smoke warnings are false alarms [SBS1991]. Thus, if a water mist system were selected as a replacement for halon, improved detection systems would be essential in order to improve reliability and reduce the likelihood of unnecessary activations. In addition, an improved detection system could be used by an intelligent activation system to

target the delivered water to the appropriate location, thereby assisting in the optimisation of the overall system.

The FIREDASS (FIRE Detection And Suppression Simulation) [MGG+2000; etc] research project was a European Union funded BRITE/EuRam research project set up to address this requirement for alternative suppression and improved detection systems. This project lasted for three years, 1996-1999, and consisted of the following organisations: The University of Greenwich (UK), The National Technical University of Athens (Greece), SINTEF (Norway), Cerberus Guinard (France), DLR (Germany), GEC Marconi (UK), and the Civil Aviation Authority (UK). The total manpower used on the project was 322 man-months. The author of this thesis contributed approximately 30 man-months to the project. This work involved implementing the fire and suppression model, and developing and integrating the other modules into the final software. The author was also responsible for running the model, liaising with other consortium members concerning model development, and interpretation of the results and validating the component modules. In addition to the development of a state of the art CFD fire-suppression computer simulation model an extensive experimental campaign was conducted by some of the consortium members to test the water mist nozzles and the activation system this included a full size mock-up of a A340 type C cargo bay. Some of this data was used to develop and validate the computer model. Its specific aim was to develop a combined system comprised of a fine water mist suppression system, an improved detection system and an intelligent control system for aircraft cargo holds.

However, the task of developing an optimised version of such a system would require extensive physical experimentation and hence significant costs. Thus, one of the objectives of the FIREDASS project was to develop sophisticated engineering computational modelling tools which could be used both in the project and later routinely by industry to optimise such combined systems. This would allow the necessary physical experimentation to be better targeted and thereby reduce development time and cost. Such a computational model has been developed. It was tested and validated by comparing its predictions with the results of a series of fire tests performed as part of the project. These included full-scale tests in an A-340 cargo test cell.

In order for such a model to be routinely used the model predictions must be as convincing, to both FSEs and regulators, as a large experimental campaign and also more economic both in terms of time taken and financial cost to the FSEs.

The first issue concerning the confidence in the fidelity of predictions from CFD based fire models was briefly addressed in the FIREDASS project. This issue is more extensively examined in chapter 5 where a procedure to assess CFD based fire modelling is presented and this procedure is then applied on three CFD codes used for fire modelling purposes.

The second issue is the runtime required for practical use of the model. It was found throughout the project that the model had very long runtimes. When a full scale simulation, utilising 2-fold symmetry, of an A340 cargo bay was conducted the runtime was >140hrs on a 433Mhz DEC Alpha. If symmetry was not utilised, which would typically be the case, the run time would be in excess of 500 hours. A FSE would need to run many different configurations and scenarios to assess the performance of the candidate fire safety system. From the runtimes mentioned this makes the routine use of such software unlikely. The potential benefits of utilising parallel processing techniques on typical office based equipment, which would be commonly available to a FSE, has been investigated in chapters 7 and 8.

4.2 Structure of the FIREDASS Computational Model

The primary aim of the computational model development component of the FIREDASS research project was to generate and validate a number of submodels which, when integrated with a CFD engine, would allow the simulation of the interaction of fire, thermal radiation and water mist with the detection and activation systems and with each other. The complete, integrated model would be able to address a number of different compartment configurations, primarily representing aircraft cargo holds but with possible extensions to other configurations such as ship engine rooms. The fire types that the model needed to address were gas burner fires, pool fires and fire loads generated by cardboard boxes stuffed with shredded paper.

The submodels developed during the project were developed as separate computational modules. They were: the fire module (developed by the author/UoG where UoG stands for The University of Greenwich); the two-phase thermal radiation module (developed by NTUA, the National Technical University of Athens); the water-mist module (developed by UoG); the fire suppression module (developed by SINTEF and author/UoG); and the detector/activation module (developed by CG, Cerberus Guinard). The framework for the computational model and the integration of the additional modules with the CFD engine was the responsibility of UoG. The modules were integrated through the CFD environment provided by the commercial software CFX 4.1 [CFX1997]. The interactions between the submodels are shown in Figure 4-1 and the submodels were as follows.

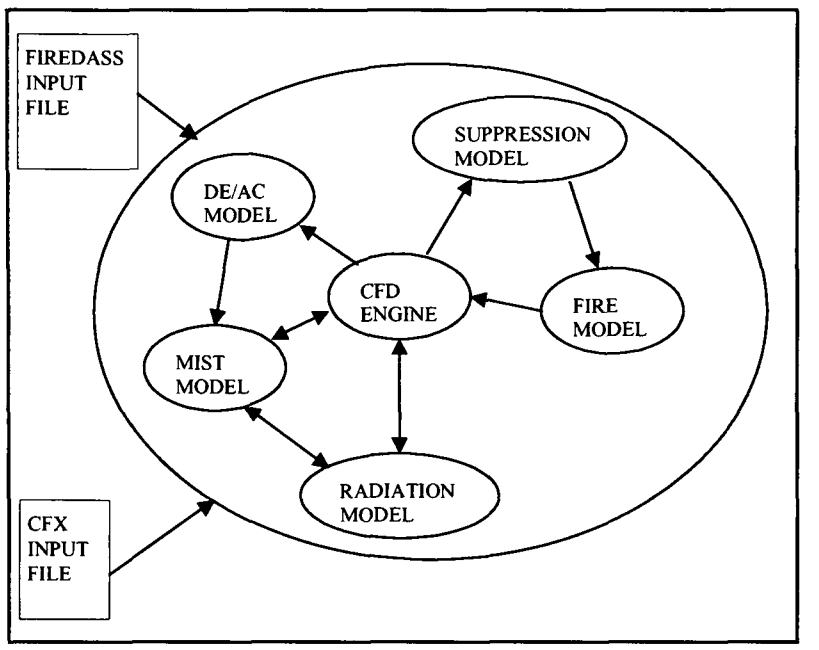


Figure 4-1 - Interactions between FIREDASS submodels.

The fire submodel simulates the fire [GGPM1998a]. It supplies heat, smoke and gaseous combustion product (CO_2, CO, H_2O and O_2) production-consumption rates to the CFD engine. These rates are not predicted by the fire submodel but are provided as inputs from rate tables held in external files. These files contain data generated from experimental test fires for various types of fuel under various test conditions [WAD1997]. The test fires were part of a test series undertaken by SINTEF as part of the project. It was necessary to prescribe experimental data as the fire submodel does not perform combustion calculations. The inclusion of complex combustion models, while possible, was not considered viable for use in this engineering model. The fire submodel is described further in section 4.3.1.

The fire suppression submodel was developed by UoG [GGPM1998b] using an empirical criterion derived by SINTEF [Wig1998]. By considering the average temperature and oxygen concentration of the air entrained into the fire it determines the point at which the fire is extinguished. This information is passed to the fire submodel which subsequently reduces the heat release rates. The fire suppression submodel is described further in section 4.6.

The detection/activation (DE/AC) submodel monitors the state of the environment (i.e. smoke concentrations and temperature) within the compartment and simulates the response of the sensors. With the predicted sensor responses, the activation routine determines which of the spray nozzles are to be activated and the nature of the pulsed activation sequence [OM1998]. This information is passed to the mist submodel which generates the mist.

The mist submodel simulates the behaviour of the water mist and its interaction with the fire atmosphere and the radiation field [MGPG1998]. It determines the effect of the mist upon the temperature distribution within the compartment during the course of the fire. The mist submodel can support any number of nozzles that are activated by the DE/AC submodel though it does not predict the formation of the mist by the nozzle. The initial state of the mist was determined experimentally and is provided to the mist submodel as input parameters [NACB1997].

The mist submodel uses an Euler-Lagrange methodology, is transient and three-dimensional and fully two-way couples momentum, heat and mass using the PSI-Cell method [CSS1977]. This is a particle tracking model which tracks discrete particles through the domain. Heat transfer includes exchange of heat with the radiation field. This is achieved by the mist submodel passing a summary of the mist to the radiation submodel which uses it to calculate how much heat the mist absorbs from the radiation field. These heat sources are then passed back to the mist submodel which applies them to the mist. Note that the mist also interacts with the fire through the suppression submodel, though only indirectly. The mist submodel is beyond the scope of this thesis but it has been described previously [MGPG1998, GGPM1998b].

The radiation submodel simulates the fire generated radiation field in the compartment and its interactions with the air, smoke, water mist and compartment surfaces [KSBM2000, KB1997]. It is a multiphase model based on the traditional six-flux radiation model [Ham1947a, b; Sch1905]. The heat sources representing the interaction of the radiation field with the air and smoke are passed directly to the CFD engine. The heat sources representing the interaction with the mist are passed to the mist submodel.

The CFD engine used was CFX 4.1 [CFX1997]. This provides the computational models that convect and diffuse the fire products through the compartment i.e. that calculate the fluid interactions. It also provides the framework for linking in the FIREDASS interface routines. These are routines that sit between the CFD engine and the FIREDASS modules and link them together. By accessing the CFX data structures and converting them to the data structures used in the FIREDASS modules and vice versa they also maintain the independence of the FIREDASS modules from the CFD engine and permit a highly modular structure to be adopted for the overall package. In principle, it should only be necessary to modify these interface routines to allow the FIREDASS modules to be ported to other CFD engines. CFX also has validation background in fire modelling [SWJ1992, SO1995, JGP1999, SSEW2001] and has some limited independent testing and comparison (e.g. [KMG+1994]).

4.3 The Fire and Radiation Submodels

4.3.1 The Fire Submodel

The governing equations for all fluid variables, as described in section 3.1, can be expressed in the general form:

$$\frac{\partial \rho \phi}{\partial t} + \mathrm{div}(\rho \underline{U} \phi) = \mathrm{div}(\Gamma_\phi \nabla \phi) + S_\phi \qquad (4.3.1.1)$$

where S_ϕ is the source term and ϕ stands for any one of the following variables: the velocities *u, v, w* in the three co-ordinate directions, the enthalpy *h*, the turbulent kinetic energy *k*, its dissipation rate ε and other scalar product concentrations (i.e. CO_2, H_2O, CO and O_2). For the continuity equation ϕ takes the value of one.

The FIREDASS fire submodel [GGPM1998a] generates the sources of heat, smoke and gaseous combustion products (CO_2, H_2O, CO and O_2) representing the fire and adds them into the appropriate gas phase equations. The heat source is added into the enthalpy equation; the smoke source is added as a concentration flux to a scalar equation; the gaseous combustion gases are modelled as simple mass fractions and an appropriate mass flux is added to these equations. In addition the combined mass flux from the combustion gases is added into the pressure correction (mass continuity) equation as a source of mass. It is assumed that the source terms all act in the same specified volume. When the CFD engine solves the gas phase equations these sources are convected and diffused throughout the compartment. In this way the effect of the fire on the flows, temperatures etc. in the compartment are accounted for. As the fire submodel does not directly simulate combustion, accurately prescribed release rates, derived from experimental data are required. It is also difficult to apply a combustion model to complex fuel types which would be typically found in a cargo bay.

The fire submodel allows all the parameters specifying the fire scenario to be entered via input files so that the user does not need to modify or recompile the software. Parameters specified via these input files include enclosure geometry, details of wall fluxes-temperatures, air leakage rates and the release rates of the various fire products.

A range of methods are provided for handling solid wall boundary conditions. These include:

- fixed flux (including adiabatic);
- fixed temperature;
- imposed variable convective heat flux and imposed variable radiative heat flux;
- imposed variable temperature;
- full boundary modelling using solid structures.

The fire source may be set at any arbitrary location and to posses any arbitrary volume within the geometry. The fire submodel fits this volume to the constraints of the grid. Alternatively the user may allow the fire submodel to vary the volume of the fire

according to the Heskestad relation [Hes1981]. Here the horizontal cross sectional area of the fire is kept constant but the height is varied according to:

$$L = 0.23\dot{Q}_C^{2/5} - 1.02D \qquad (4.3.1.2)$$

where L is the plume height (in m), $\dot{Q}_c$ is the convective heat flux generated by the fire (in kW) and D is the equivalent diameter of the surface area of the fire (in m). The area of the fire and its original height are specified in the input file. This height is taken as a minimum value. The height required according to the above formulation is calculated at each time step. If it exceeds the original height then the number of cells included in the volume of the fire vertically is increased until the calculated height is just exceeded. The new heat source is uniformly distributed over this increased volume.

4.3.2 The Thermal Radiation Submodel

Thermal radiation is the dominant mode of heat transfer in compartment fires. The radiation submodel [KB1997, KSBM2000, Ker1997a] simulates the generation of thermal radiation by the fire and its transport throughout the compartment. Absorption and scattering by the combustion gases, smoke, water-mist and compartment surfaces are accounted for. Heat sources representing absorption from the radiation field are passed to both the CFD engine for inclusion in the gas field calculations and the water-mist submodel for inclusion in the droplet temperature calculations. The radiation interaction with the water-mist phase is outside the scope of this thesis and is discussed elsewhere [Ker1997b; KB1997; KSBM2000].

As a first approximation, the traditional six-flux model (see section 3.9.1.1 for details) is used to describe thermal radiation. This model has already been successfully applied to a number of practical problems involving radiation [HM1988]. The six radiation fluxes modelled are those in the positive and negative x, y and z co-ordinate directions.

The primary attraction of this model is that it is easily incorporated into the finite volume scheme of the numerical solution procedure adopted by fire field models and is much less computationally intensive than other advanced radiation models such as

the discrete transfer method [LS1981]. This later point is of prime importance as the model, if it is to be used in an engineering context, must be as computationally efficient as possible.

4.3.3 Fire - Radiation Coupling

The fire submodel adds the heat output from the fire to the gas phase enthalpy equation in the computational cells in the volume of the domain specified as being occupied by the fire (see (4.3.1.2)). Consequently high temperatures are obtained in this volume. The radiation equations include an energy term representing the addition to the radiation field of the black body radiation emitted by the hot gas. This energy source is the driving force for the radiation field. Since this source depends upon the fourth power of the gas temperature it is dominated by the contribution from the very hot gas in the computational cells containing the fire. In this way the radiation field is produced by the coupling of the fire and radiation submodels. Note that, as the FIREDASS model does not account for soot, it is assumed that the soot temperature is identical to the gas temperature. This is not strictly correct as the soot will generally be hotter than the gas.

4.4 Validation of the Fire and Radiation Submodels

Due to the life safety nature of the model a thorough testing strategy was developed and implemented. The fire and radiation submodels were validated against available experimental data in two stages. Firstly, as each submodel was developed it was tested against experimental data. The submodels were then combined and the full model tested against more challenging experimental scenarios. For the tests of the component submodels a set of experimental data readily available in the fire literature, one commonly used for this purpose, that of Steckler et al. [SQR1982], was used. This data set consists of the results of a series of fire tests in a small well ventilated compartment. Predictions made by the separate fire and radiation submodels were compared with this experimental data set. The results generated were in good agreement with the experimental data and were consistent with predictions produced by other fire field models including PHOENICS and FLOW3D 2.3 (now known as CFX) [GGPM1998a, Ker1997a]. Results for CFX CFD code can also be found in section 6.1.2.1. For the tests of the combined fire submodel, radiation submodel and

CFD engine the experimental results obtained from fire tests conducted as part of the FIREDASS programme were used. These fire tests consisted of a systematic set conducted by SINTEF [WAD1997] and were designed and performed specifically for generating data for the validation of these models. These trials were jointly specified by the author/UoG and SINTEF.

4.4.1 The SINTEF Fire Trials

The SINTEF experiments [WAD1997] were conducted with and without the presence of water mist. While these experiments were not conducted in an aircraft cargo hold mock-up, the fire scenarios were intended to resemble as far as possible the target application environments. The SINTEF trials were performed in a metal container equipped with a chimney to allow the combustion products to escape and a forced air supply to represent leakage into an aircraft cargo hold, arranged as shown in Figure 4-2. Instrumentation included (see Figure 4-3):

- 72 thermocouples to measure gas temperatures;
- 6 velocity probes to measure gas velocities;
- 3 total heat flux meters to measure the total heat flux to a target wall;
- 3 radiative heat flux meters to measure the radiative heat flux to a target wall.

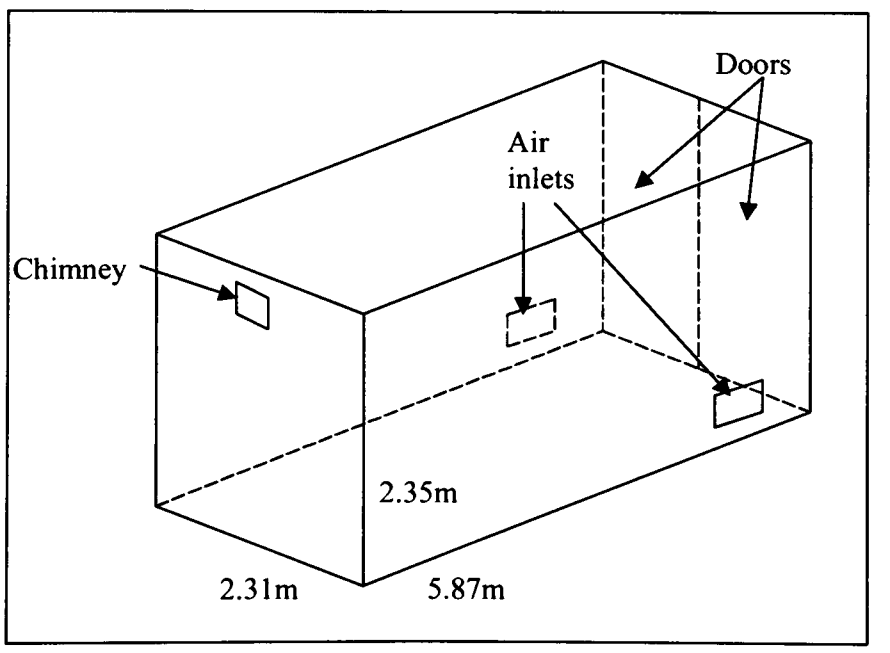


Figure 4-2- Schematic of SINTEF test compartment.

In addition to the above, the following measurements were also made:

- concentrations of O_2, CO_2, CO in the chimney and centrally in the compartment (G1);
- water vapour concentration in the chimney and centrally in the compartment (G1);
- particle (soot) concentration in the chimney;
- wall temperature at two locations;
- pressure in the compartment.

A total of 35 experiments were performed involving different fire types, fire locations, numbers of obstacles and numbers of nozzles. The fire types implemented consisted of gas burners, cardboard boxes and kerosene pool fires. The experiments were all simulated using the FIREDASS numerical model and the predicted and experimental results compared [GGPM1998a, b].

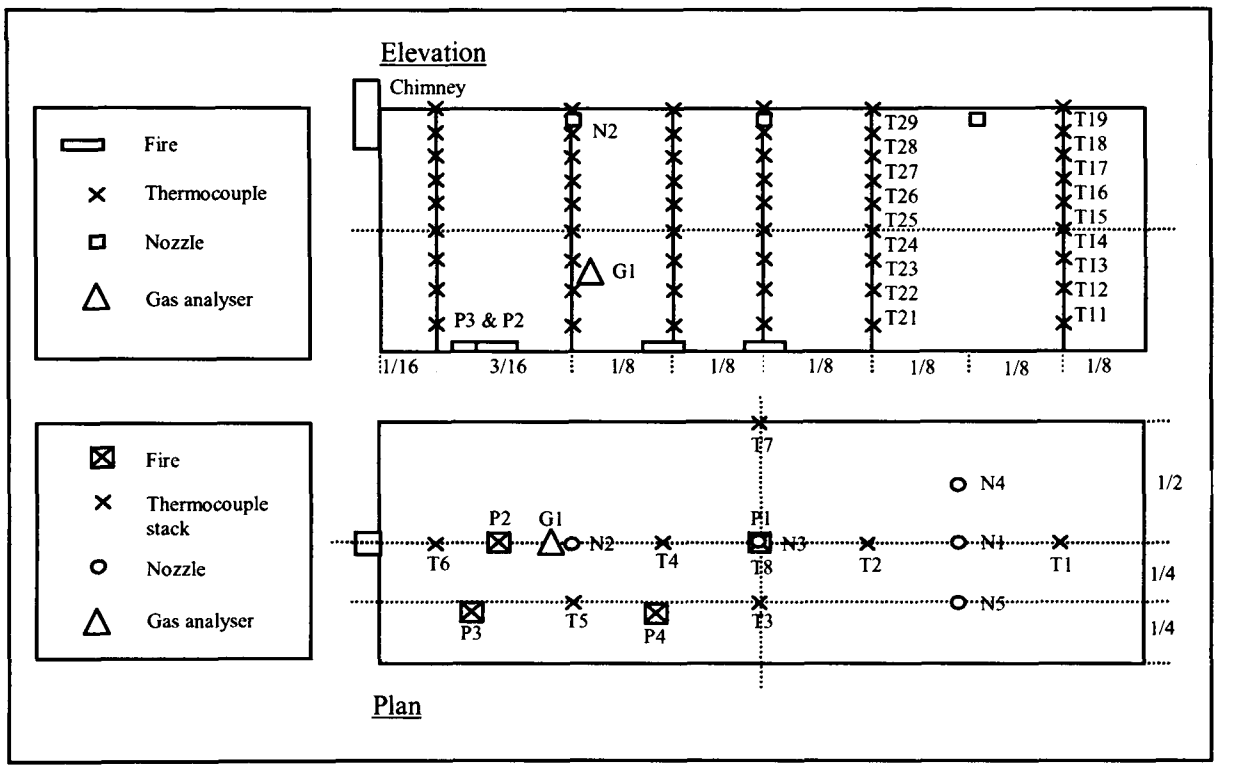


Figure 4-3 - Diagram showing instrumentation of SINTEF test compartment.

4.4.2 Comparison of FIREDASS Numerical Predications with SINTEF Experimental Results

A selection of the results will be presented here; interested readers can find the full results presented in the project report [GGPM1998a]. Comparisons will be presented for the gas and smoke concentrations in the chimney, for the gas and smoke concentrations in the centre of the room at location G1 (see Figure 4-3) and for the temperature at four representative thermocouple locations (see Figure 4-3), namely:

- T12 – near the floor in the vicinity of the compartment doors;
- T18 – in the same thermocouple stack as T12 but just below the ceiling;
- T58 – just below the ceiling between fire positions P3 and P4;
- T36 – located centrally and at approximately ¾ height.

These comparisons will be presented for three different fire types. These are as follows:

- experiment 02HS utilised a propane gas burner;
- experiment 05PM utilised a small kerosene pool;
- experiment 06BXB utilised a line of three cardboard boxes.

In all three experiments the fire was located at position P2 (see Figure 4-3) and there were no obstacles in the compartment. In the 02HS case, gas was supplied to the burner for a predetermined period of time, after which the supply ceased. For the 05PM and 06BXB cases, the fuel or boxes were lit and then allowed to burn until self-extinguishment occurred, either through consumption of all of the oxygen in the compartment or all of the fuel.

In all three simulations the following boundary conditions were imposed:

- all compartment surfaces (including the floor and ceiling) were modelled as isothermal surfaces at 288K;
- for the gas velocities the non-slip condition was applied on all surfaces;
- the k-ε turbulence model was used with the standard wall functions;
- with the exception of the chimney, all boundaries were sealed allowing zero mass transfer;
- the chimney was treated as a zero pressure boundary with ambient conditions outside, so in the event of reverse flow, i.e. air being drawn into the compartment, fresh air was entrained;
- in the radiation model all compartment surfaces (including the floor and ceiling) were assumed to have an emissivity of 0.9 and a temperature of 288K.

4.4.2.1 02HS – The Propane Burner Case

In this experiment the fire source consisted of a propane burner which had a fixed supply of gas. The burner was allowed to burn for a predetermined period of time before the gas supply was terminated. All measurements were taken for the duration of the burn and for 15 minutes after the gas supply was terminated.

As the experimental setup was symmetrical only half the compartment was simulated in order to save computational time. The full geometry was initially simulated using a cruder mesh and the numerical results were found to be symmetrical. The symmetrical mesh used had 39 cells along the long axis, 22 vertically and 13 in the half width of the compartment, a total of 11,154 cells. Cells were concentrated in the vicinity of the surfaces and of the fire. Figure 4-4 shows the mesh in the vertical plane along the long central axis of the compartment, the symmetry plane.

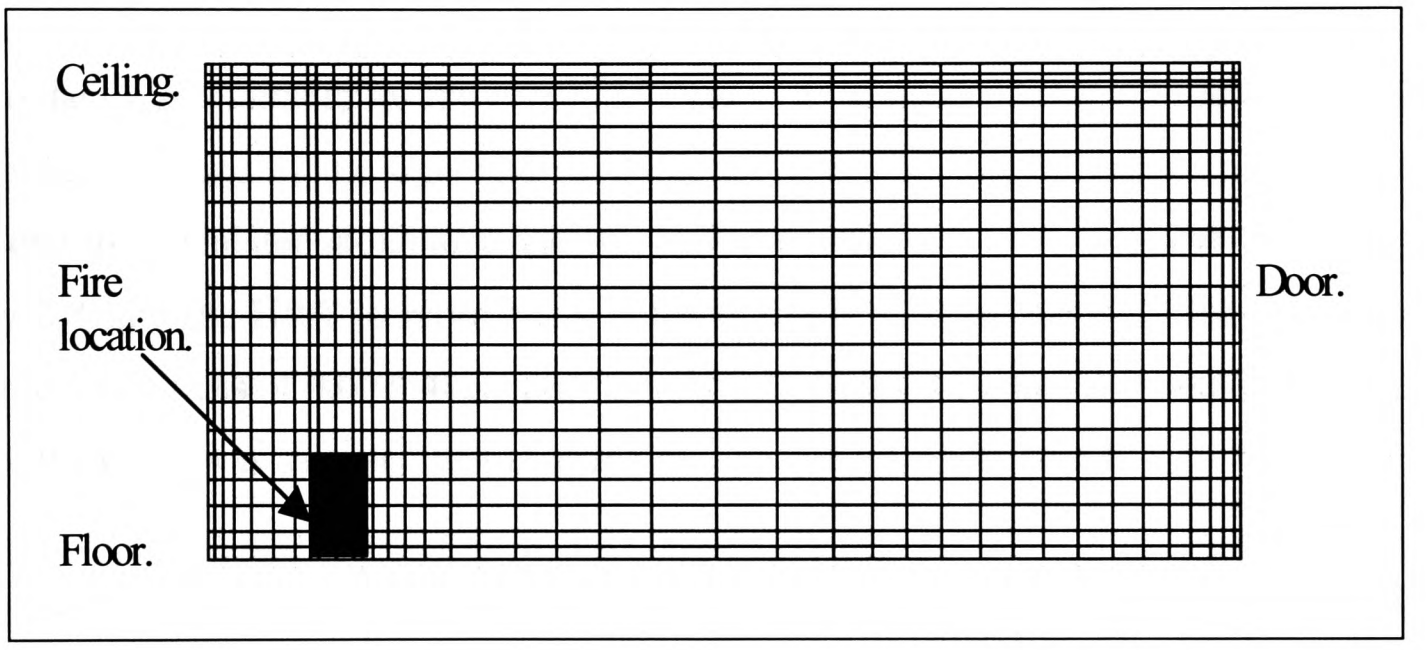


Figure 4-4 - Mesh on the symmetry plane for 02HS problem.

In the experiment the fire started at t=24s, so this is the point at which the simulation was begun. Simulation time therefore ran from t=24s to t=724s, i.e. a total of 700s of real time was simulated. This required 35 hours of CPU time on a DEC Alpha 466MHz processor. It should be noted that this is a relatively small case and this runtime indicates that larger problem sizes the model may be impractical for everyday usage by a FSE.

As described in section 4.3.1, the fire was represented as a volumetric source of heat, CO_2, H_2O and a sink for O_2. As the simulation uses a Cartesian grid the volume used to contain the release rate sources was cuboidal in shape with a square horizontal

cross section whose area was equal to the area of the gas burner (~0.09m^2). Its flame height was determined automatically by the fire submodel using the Heskestad formulation [Hes1981] (see equation (4.3.1.2)). Note that as the heat release rate of the fire was constant throughout the simulation, the height also remained constant (~1.5m). All sources were distributed uniformly throughout the volume.

The heat release rate (HRR) for the gas burner was determined from the propane supply rate and the heat of combustion of propane with an assumed average combustion efficiency of 80%. An oxygen depletion method was also utilised in the experimental setup and suggested a HRR ~80% less than that obtained from the fuel supply measurement. This produced a uniform HRR of 117kW for the duration of the burn i.e. from 48s to 421s. The HRR ramped to this value between 24 and 48 seconds and back to zero between 421 and 429 seconds.

No data was available for the production rates of the other combustion products. The production rates of the CO_2 and the H_2O and the consumption rate of the O_2 were therefore calculated stoichiometrically from the propane supply rate and then scaled by 0.8 like the HRR to represent a reduced combustion efficiency. This gave peak production rates of 0.00764kg/s for the CO_2, 0.00416kg/s for the H_2O and -0.0088kg/s for the O_2. It was assumed that neither CO nor smoke was produced.

In reality the combustion efficiency is not constant but varies throughout the experiment. The efficiency is likely to be large at the start of the fire and then tend to decrease as the oxygen within the compartment is consumed. Representing the combustion efficiency as an average value is therefore a crude approximation. This has implications for all the release rates used in the model. For the combustion products, CO_2 and water vapour, the production rates will be over-predicted as will be the consumption rate of oxygen. It was also assumed that generation rates for the products of incomplete combustion, i.e. CO and soot, are zero. It was noted from visual observation however that as the experiment progressed small quantities of smoke started to be produced.

The main results for this simulation are presented in Figure 4-5 to Figure 4-11. These show comparisons between the experimental results and the model predictions for various parameters at various locations within the compartment as a function of time.

Figure 4-5 to Figure 4-7 depicts the temperature distribution at three locations. Figure 4-5 is thermocouple T12 (located on the long symmetry axis, towards the front of the compartment and near the floor, see figure 3); Figure 4-6 is thermocouple T18 (same thermocouple tree as T12 but located just below the ceiling); and Figure 4-7 is thermocouple T36 (located on the short symmetry axis, half way between the compartment centre and the wall, approximately 1.75m above the floor, see Figure 4-3). It can be seen that, in the experiment, once the fire has started, temperatures in the upper part of the compartment rise rapidly before starting to level out as equilibrium is reached between the heat sourced by the fire and the heat lost through the walls and chimney. Equilibrium would be possible in this case as the fire has a steady heat release rate. When the fire is extinguished the temperatures fall rapidly again due to continued heat loss through walls and chimney and the flow of cold fresh air in at floor level through the inlets. Temperatures lower in the compartment rise more slowly but are still rising when the fire is extinguished. This is due to the hot layer continuing to deepen as the fire burns.

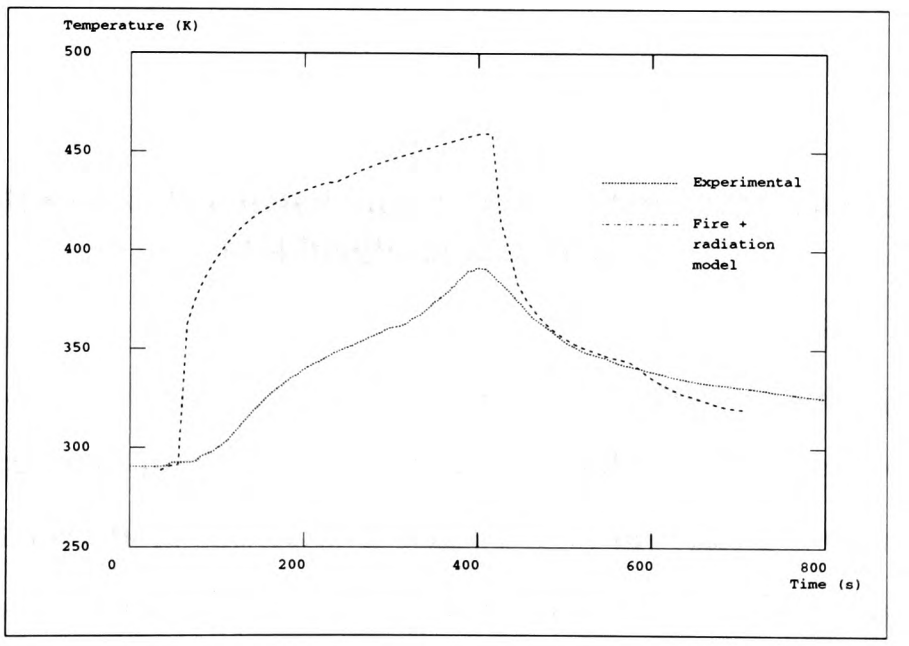


Figure 4-5- 02HS case. Predicted and measured temperature variation at T12 (near floor and doors).

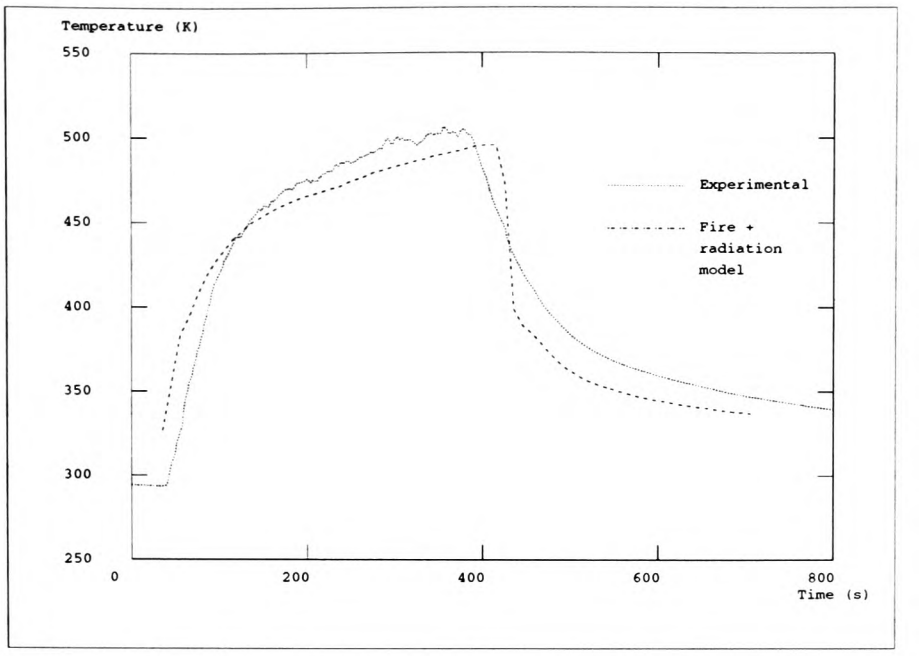


Figure 4-6 - 02HS case. Predicted and measured temperature variation at T18 (near ceiling and doors).

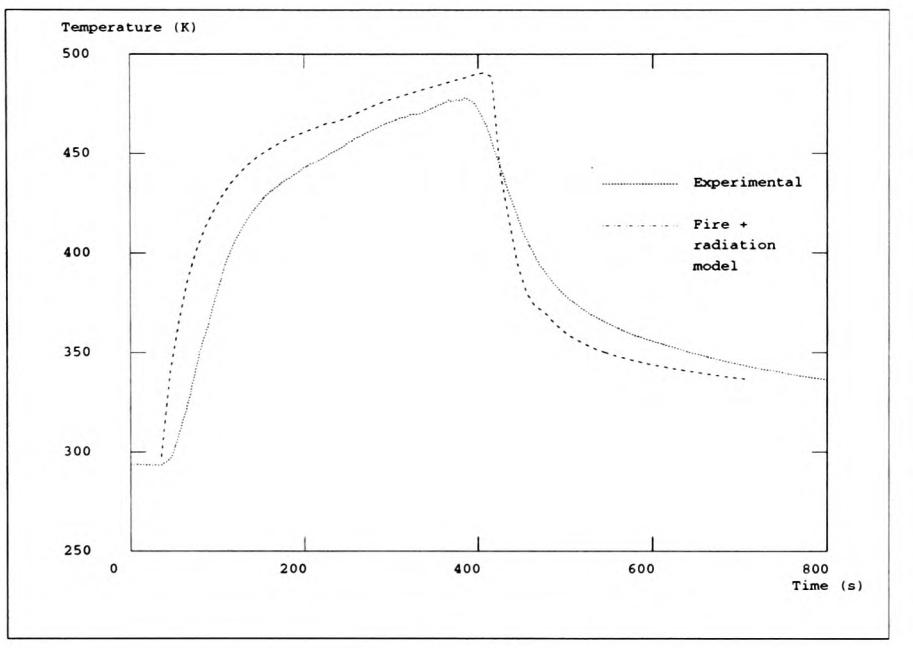


Figure 4-7 - 02HS case. Predicted and measured temperature variation at T36 (3/4 height and central).

As can be seen from Figure 4-6 and Figure 4-7, there is quite good agreement between measured and predicted temperatures high up in the compartment, the temperature prediction near the ceiling, Figure 4-6, providing the best match to the experimental results. From the point of view of creating a ceiling based, temperature operated, detection system this is an important result. The growth and decay portions of the temperature curves and the peak temperatures are well captured by the numerical predictions. However, temperatures near the floor are severely over-predicted. Here we find that the predicted temperature growth is much more rapid than that measured and that the peak temperature is over-predicted by 17%. The exact reason for this discrepancy has not yet been determined and is the subject of

continuing research. It is likely however that it is due to the use of approximate thermal wall boundary conditions, the assumption of zero smoke production and the interaction of these factors with the radiation field. Such a discrepancy leads to skepticism amongst FSEs about the overall reliability of CFD based methods and results. Such a discrepancy is not unexpected by modellers due to approximations and assumptions made in modelling but these arguments can hold little weight with experimentalists and FSEs.

In discussing the comparison between measured and predicted temperatures it should be noted that there is an uncertainty of "several degrees" [WAD1997] in the temperatures measured by the thermocouples. This is partly due to the fact that the thermometer is not only subjected to gas temperatures but also acts as a receiver and emitter of radiation.

Figure 4-8 to Figure 4-11 compare predicted and measured concentrations for the combustion gases. Figure 4-8 depicts the CO_2 concentration in the chimney, while Figure 4-9 to Figure 4-11 depict the CO_2, O_2 and H_2O concentrations at G1, a point located on the long symmetry axis half way between the compartment centre and the chimney wall, approximately 0.59m above the floor. In this case CO was not considered as the values measured were very small and hence involved large uncertainties.

From Figure 4-9 it can be seen that as the fire burns the CO_2 it produces builds up in the compartment, the concentration increasing from less than 1% to over 7%. When the fire is extinguished, at 420s, the concentration immediately starts to drop. This is as would be expected since the sampling point is low down in the compartment and the compartment is being supplied with cold fresh air at floor level. After the fire has been extinguished this flushes all the combustion products out of the compartment and returns gas concentrations to ambient levels. As can be seen from the graph this process occurs slowly due to the volume of the compartment and the turbulent mixing which takes place, combustion products from high in the compartment being mixed with the fresh air entering the compartment lower down.

From Figure 4-8 it can be seen that the increase in the CO_2 concentration in the compartment, Figure 4-9, is matched by a corresponding increase in the concentration of CO_2 in the flow through the chimney. Since this flow is drawn from high in the compartment and away from the fire the combustion products have had time to mix with the gas in the bulk of the compartment so the CO_2 concentration is slightly lower then at the sampling point in the compartment, which is near the fire. However, as the fire starts to die due to oxygen depletion and is then extinguished the temperature in the compartment starts to drop, see Figure 4-6. As the air in the compartment cools it contracts, resulting in the flow through the chimney reversing. Since the chimney is long, mixing between fresh air entering and hot buoyant gases trying to escape occurs. This results in the measured CO_2 concentration in the chimney dropping slowly towards ambient levels as the chimney is cleared of combustion products. Once the compartment stops cooling the flow returns to normal due to the inflow of fresh air through the vents creating a flow out through the chimney. The CO_2 concentration measured in the chimney therefore returns to the value in the compartment and then shows the same slow decline to ambient values as in the compartment.

Simulated results show excellent qualitative agreement with this behaviour. The main difference occurs due to the fact that in the simulation the chimney is modelled as an exit and not as a long pipe. Consequently when the flow reverses the predicted CO_2 concentration immediately drops to ambient values. The quantitative agreement is also quite good though there does seem to be a systematic over-prediction. This suggests that there may be a problem with the source terms, which it will be recalled were assumed and not measured.

As can be seen from Figure 4-11 the H_2O concentration in the compartment shows the same behaviour as the CO_2, there is a steady rise up to 420s when the fire is extinguished followed by a slow return to ambient values as the combustion products are flushed out of the compartment. As before the simulation results show excellent qualitative agreement, accurately capturing the behaviour of the system. The quantitative agreement is also quite good though there may be a tendency to under-predict the concentration. Again it should be recalled that these predictions are based upon assumed and not measured production rates.

Figure 4-10 depicts the O_2 concentration in the compartment. Here the concentration falls steadily as the fire progresses as the O_2 is being consumed. As with the other combustion products, when the fire is extinguished at 420s the concentration starts to return to ambient values as the compartment is flushed with clean air.

The concentrations of CO_2, O_2 and H_2O were obtained by drawing the air out of the compartment down suction lines to gas analysers. For the CO_2 and O_2 measurements the gas analyser has a nominal accuracy of ±1%. It is likely however that the experimental error is in fact substantially greater than this due to the length of the suction lines which had to be used and the time taken for the sample to get to the analyser (5m with a sample to result time of 38s in the case of the samples from the centre of the compartment). For the H_2O the measurements were based upon a technique involving heating the sample to a high temperature (around 150°C) and then measuring its relative humidity and temperature and using these to calculate the absolute humidity. The exact experimental error is hard to quantify but is likely to be substantial both as a consequence of the need to use long suction lines to obtain the sample and the difficulty of making the actual measurements. When comparing the experimental and theoretical results therefore, it should be remembered that the experimental results have considerable uncertainty associated with them.

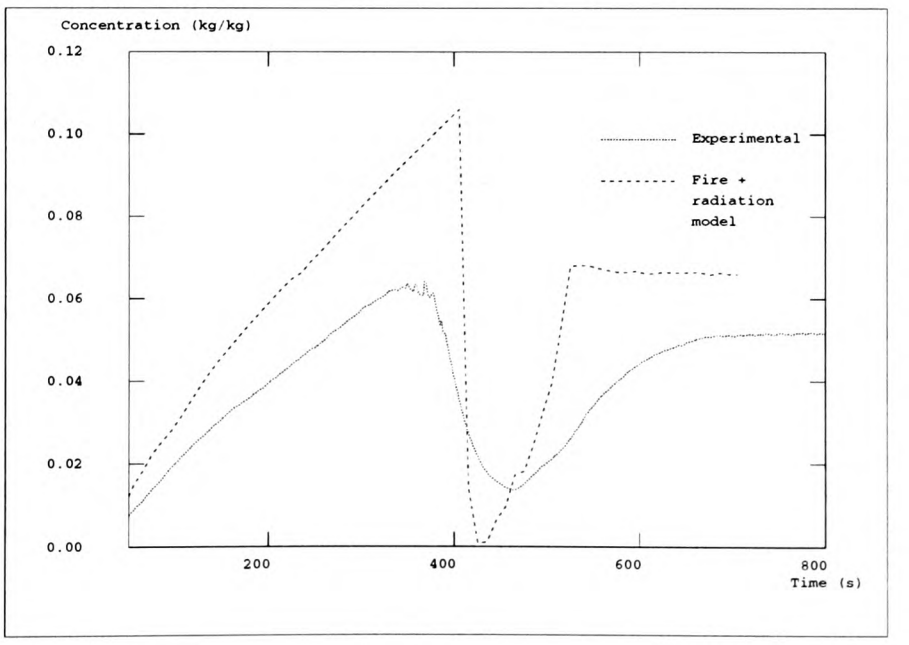


Figure 4-8 - 02HS case. Predicted and measured CO_2 variation at the chimney.

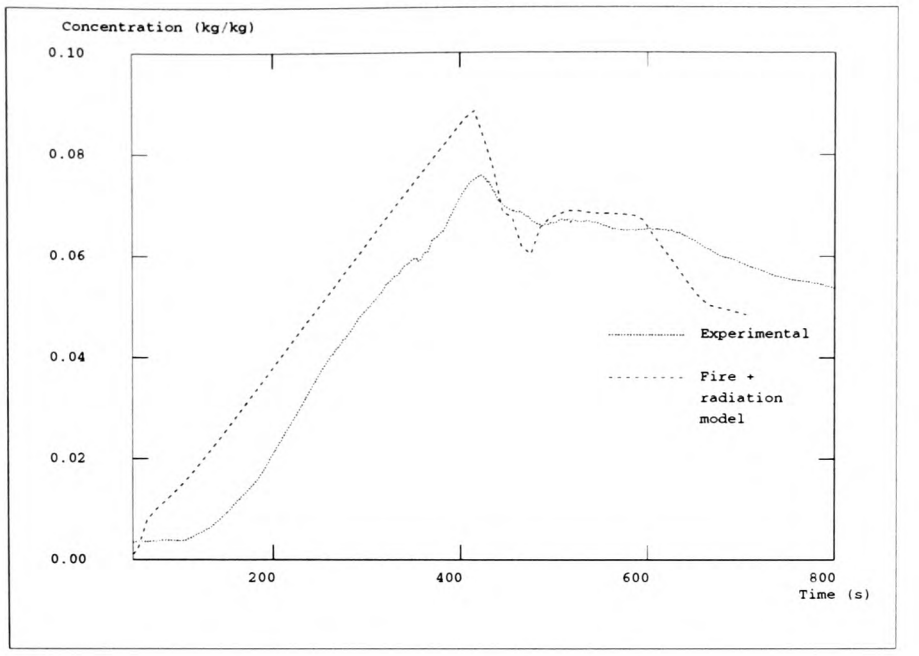


Figure 4-9 - 02HS case. Predicted and measured CO_2 variation in the room at location G1.

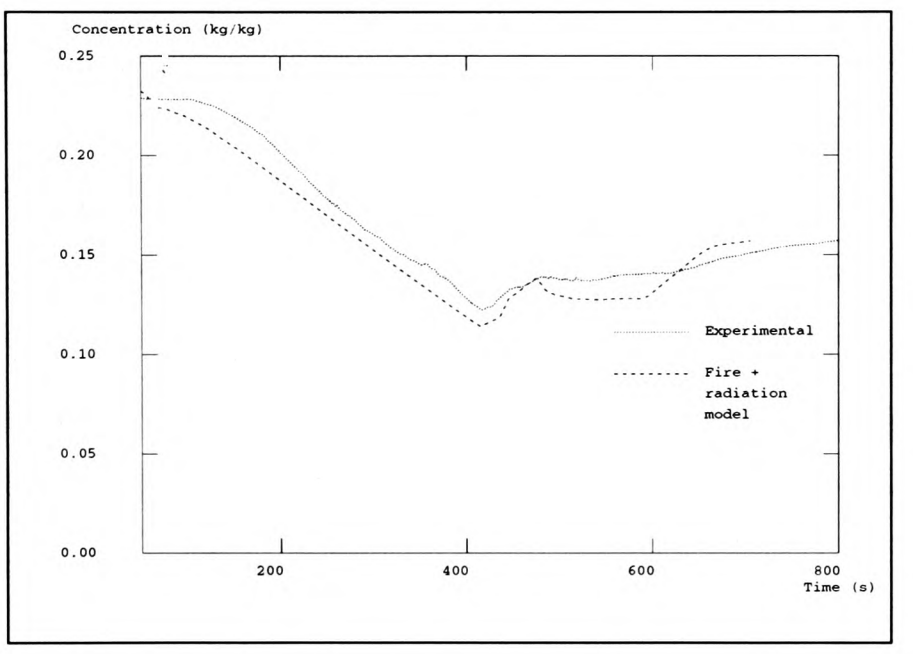


Figure 4-10 - 02HS case. Predicted and measured O_2 variation in the room at location G1.

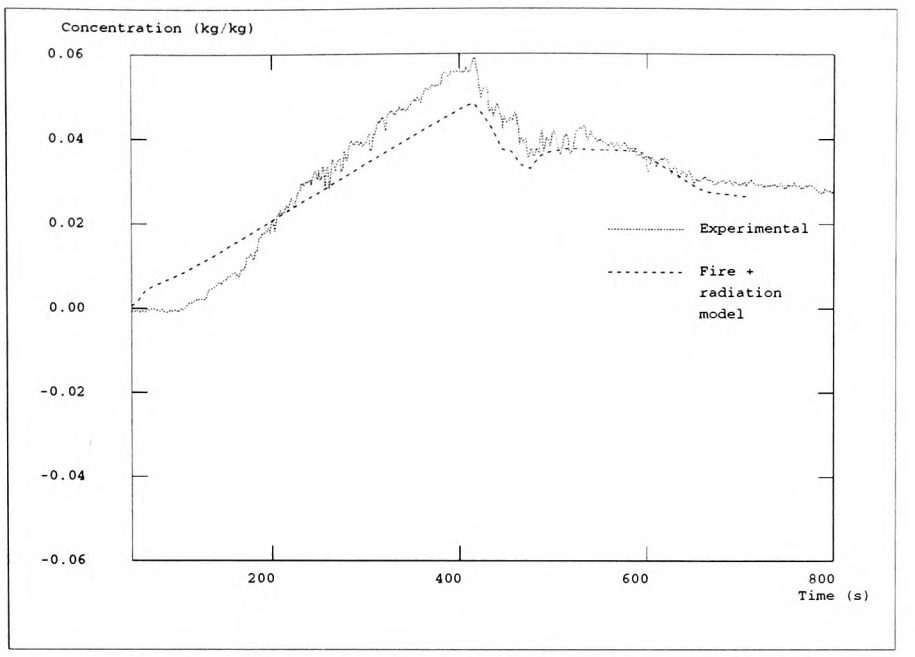


Figure 4-11 - 02HS case. Predicted and measured H_2O variation in the room at location G1.

4.4.2.2 05PM - The Kerosene Pool Fire Case

The fire source in this experiment consisted of a 0.5m square tray of kerosene. The kerosene was supplied from a reservoir at a rate adjusted to keep a constant liquid level in the tray with the overflow led to a second reservoir. The pool was ignited and allowed to burn until it self extinguished through oxygen starvation. This required approximately seven minutes. The data was recorded throughout the burn and for approximately 10 minutes afterwards.

For the simulation essentially the same mesh was used as for the 02HS case, only the number of cells along the long axis was increased to provide a better resolution of the physically larger fire. The mesh was 41 by 22 by 13, a total of 12,584 cells. In the experiment the fire started at t=50s, so this is the point at which the simulation was begun. Simulation time therefore ran from t=50s to t=895s, i.e. a total of 845s of real time was simulated. This required 34 hours of CPU time on a DEC Alpha 400MHz processor.

As described in section 4.3.1 and in the 02HS case, the fire was represented as a volumetric source of heat, CO_2, H_2O, and sink of O_2. For this case sources of CO and smoke were also included. The sources were imposed in a volume determined using the Heskestad formulation.

The nature of the prescribed heat source was determined in the following way. The reservoir supplying the kerosene to the fire and the reservoir collecting the over-spill were both mounted on scales and weighed continuously during the experiment. The difference between the quantity supplied and the quantity collected, after allowing for the thermal expansion in the tray and the time delay between leaving the supply reservoir and arriving in the collection reservoir, was taken to be the quantity of oil evaporated. To get the perfect combustion efficiency heat release rate it was assumed that 100% of this was burned and that the oil had a constant net heat of combustion. This allowed the heat release rate of the fire to be calculated as a function of time. As with the 02HS case no direct measurements of the sources of CO_2, H_2O and O_2 were available so their production-consumption rates were calculated stoichiometrically from the amount of kerosene consumed. The source terms therefore become multiples of the heat release rate.

Also as with the 02HS case the simulations were run with all sources multiplied by a scaling factor of 0.8 to represent an average real combustion efficiency of 80% over the course of the burn. As with the 02HS case it was found that the oxygen depletion method gave a heat release rate ~80% of that derived from the fuel supply rate. The resultant peak rates used were heat, 80.8kW, oxygen, -0.006kg/s, water vapour, 0.0025kg/s and CO_2, 0.0052kg/s and the resultant HRR curve is shown in Figure 4-12.

The CO and smoke sources were determined by using standard tables of the yields of combustion products for well ventilated fires [SFPE1995]. These give the source of CO and smoke in grams per gram of kerosene burned, which was known. When the usual 80% efficiency factor was included this resulted in peak production rates of 7.7×10^{-5}kg/s for the smoke and 2.2×10^{-5} kg/s for the CO.

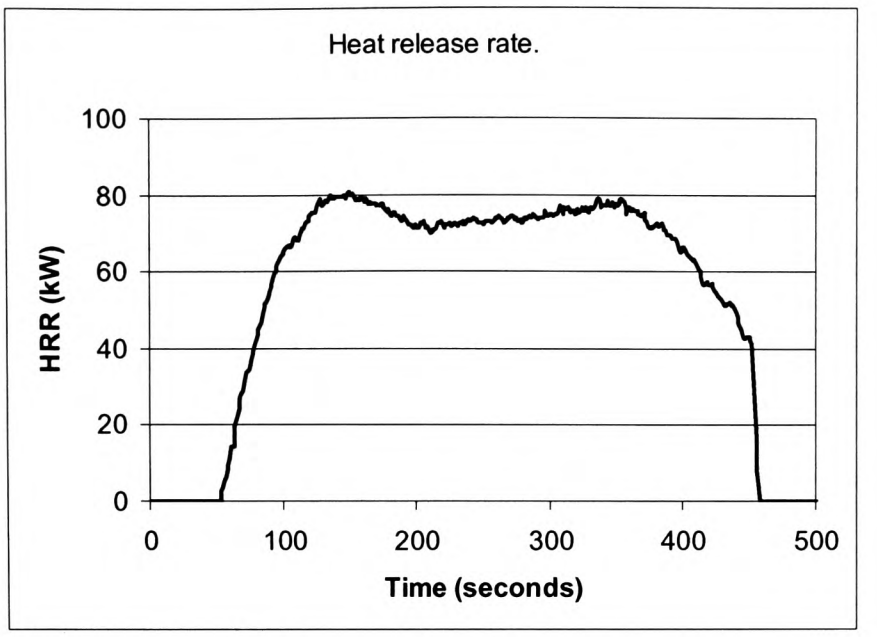


Figure 4-12 - Heat release rate for kerosene pool fire.

The main results for this simulation are presented in Figure 4-13 to Figure 4-21. These show comparisons between the experimental results and the model predictions for various parameters at various locations within the compartment as a function of time.

Figure 4-13 to Figure 4-15 depict the temperature distribution at the same three locations as used in the previous case i.e. Figure 4-13 is at thermocouple T12, Figure 4-14 is at thermocouple T18 and Figure 4-15 is thermocouple T36. It can be seen that the temperature histories are similar to those in the 02HS case, though here the curves are flatter once the initial rapid increase is over. This is because of the slight fall off in the HRR after its peak is reached. This will combine with the increased heat losses through the compartment boundaries to significantly slow any further increase in the temperature in the compartment.

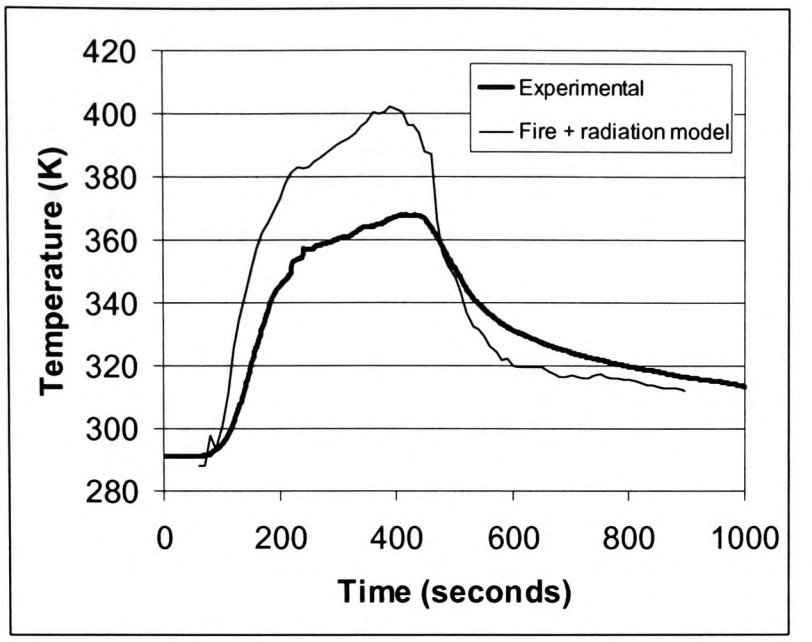


Figure 4-13 - 05PM case. Predicted and measured temperature variation at T12 (near floor and doors).

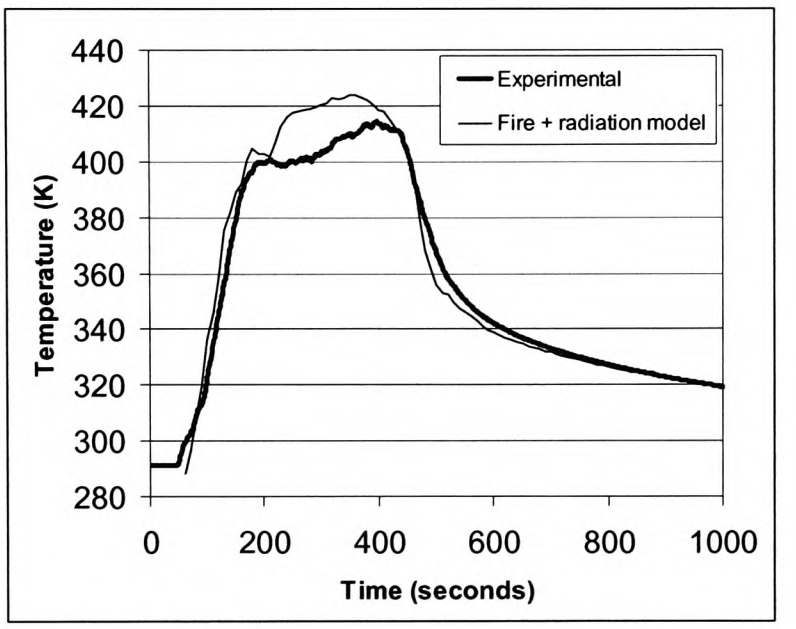


Figure 4-14 - 05PM case. Predicted and measured temperature variation at T18 (near ceiling and doors).

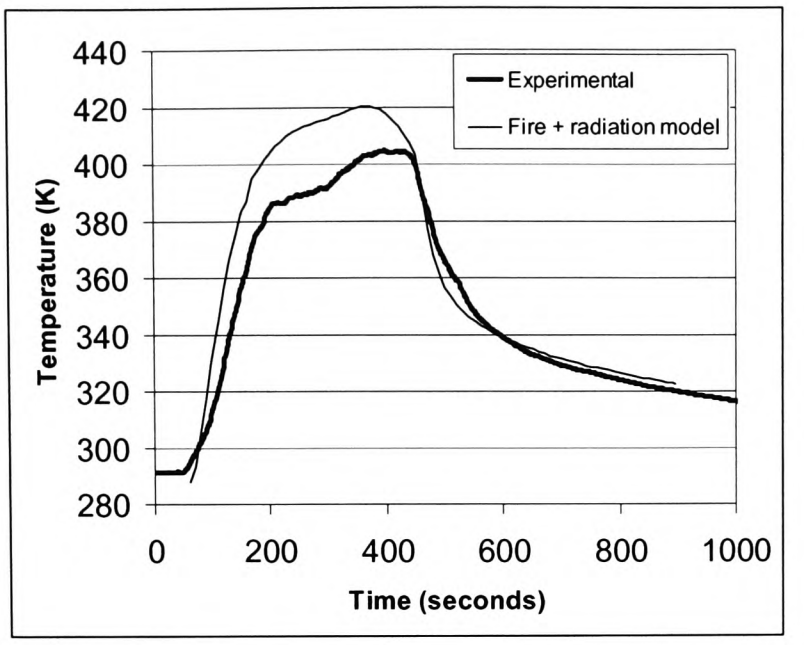


Figure 4-15 - 05PM case. Predicted and measured temperature variation at T36 (3/4 height and central)

As in the gas burner (02HS) case, there is quite good agreement between measured and predicted temperatures high up in the compartment. The shape of these temperature distributions corresponds to the heat release rate curve in Figure 4-12. The growth and decay portions of the temperature curves and the peak temperatures are well captured by the numerical predictions. However, once again, temperatures near the floor are over-predicted. Near the floor we find that the growth of the predicted temperature is more rapid than that measured and that the peak temperature is over- predicted by some 9%. Similarly to the 02HS case, the crude thermal boundary conditions and the approximate smoke release rates are thought to contribute to these discrepancies. In particular, while in this case smoke has been included, the source used assumed a well ventilated fire. In reality, as the fire proceeded and became under ventilated, the rate of smoke production per gram of fuel consumed will have risen.

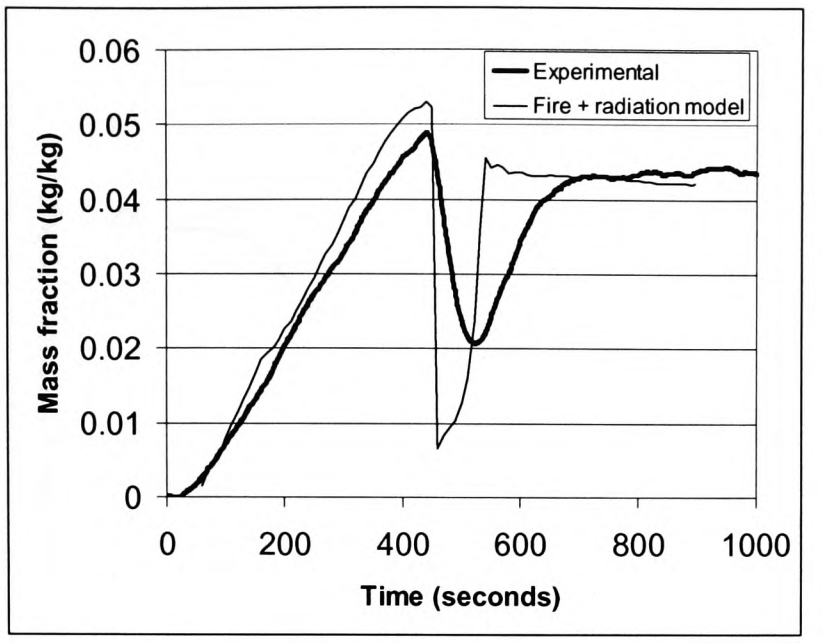


Figure 4-16 - 05PM case. Predicted and measured CO_2 variation at the chimney.

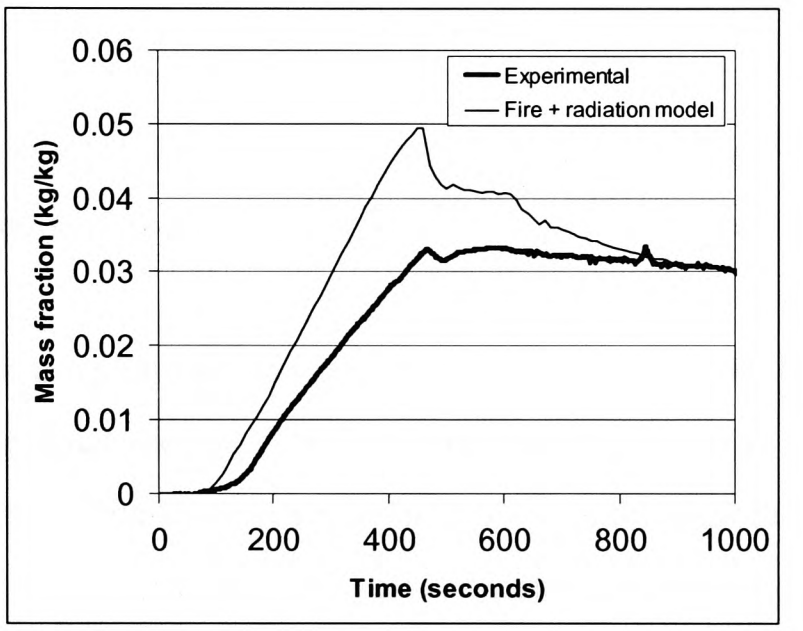


Figure 4-17 - 05PM case. Predicted and measured CO_2 variation at room location G1.

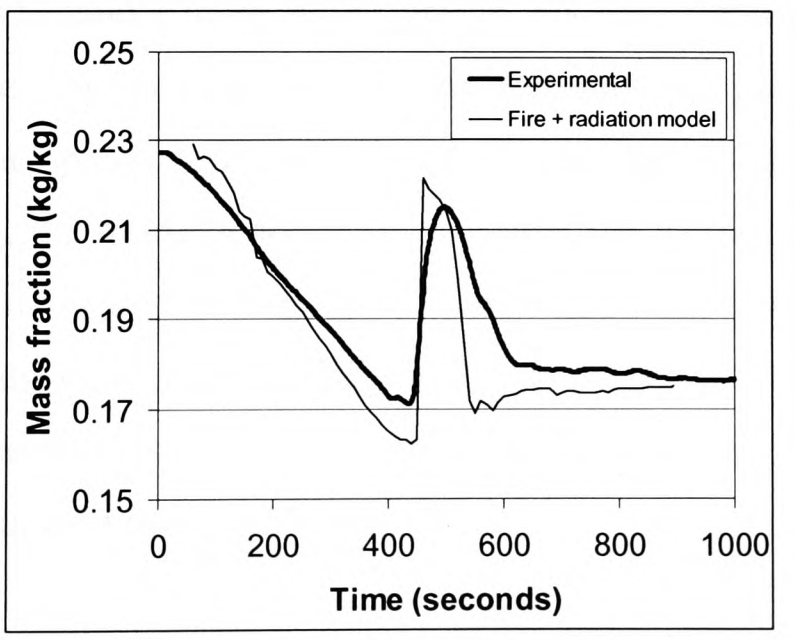


Figure 4-18 - 05PM case. Predicted and measured O_2 variation at the chimney.

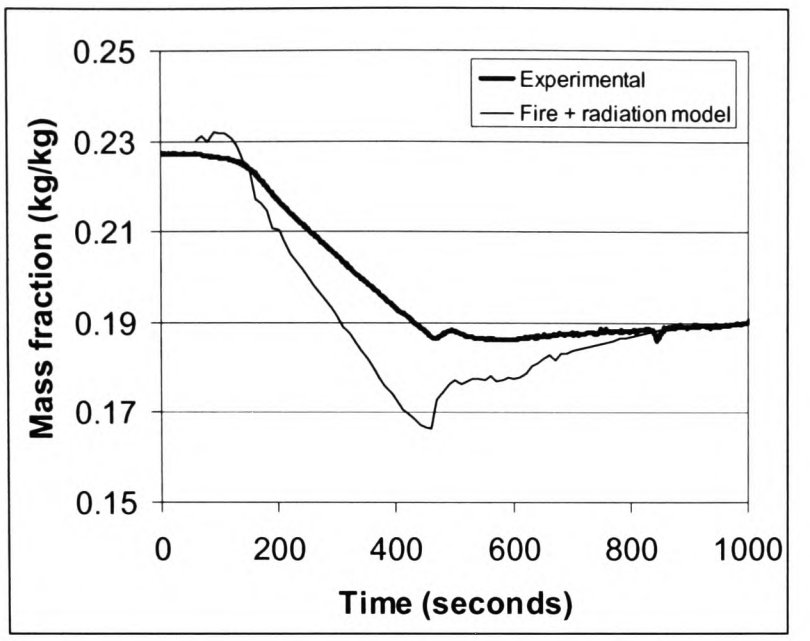


Figure 4-19 - 05PM case. Predicted and measured O_2 variation at room location G1.

Figure 4-16 to Figure 4-19 compare predicted and measured concentrations for CO_2 and O_2 in the chimney and at location G1.

The behaviour here is qualitatively the same as with the gas burner test, see the discussion in the previous section. There is the same build up in CO_2 and decline in O_2 as the fire progresses, the same reverse flow back in through the chimney when the compartment starts to cool after the fire self extinguishes and the same slow return to normal values as the compartment is flushed with clean air.

It can be seen that the numerical predictions for the CO_2 and O_2 concentrations at the chimney (Figure 4-16 and Figure 4-18) are in good qualitative and quantitative agreement with the experimental values. At the internal gas analyser location however, while the predictions are in good qualitative agreement with the experimental values, the model tends to over-predict the amount of CO_2 and under-predict the amount of O_2 present (Figure 4-17 and Figure 4-19). The predicted values do however tend to the experimental values with time. It is likely that this discrepancy is due to the assumption of constant combustion efficiency when in reality it will have varied during the course of the fire.

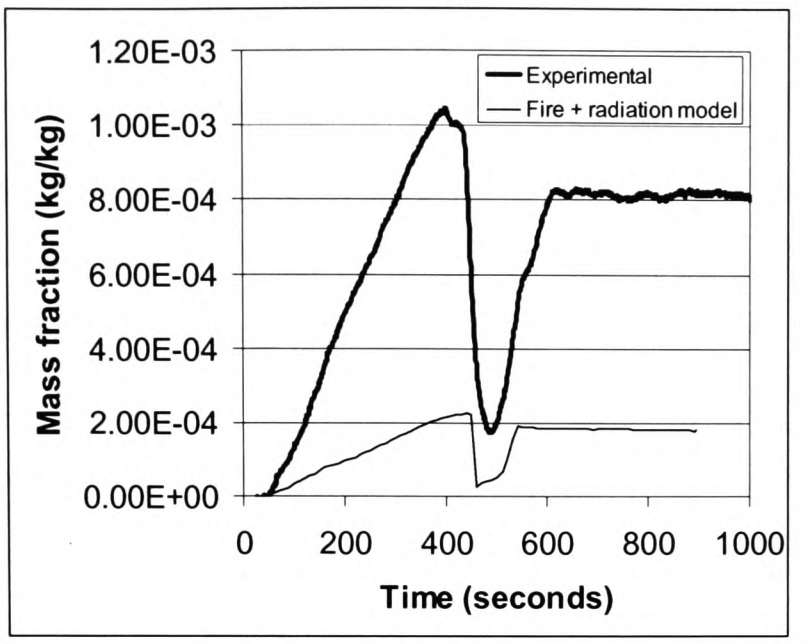


Figure 4-20 - 05PM case. Predicted and measured CO variation at the chimney.

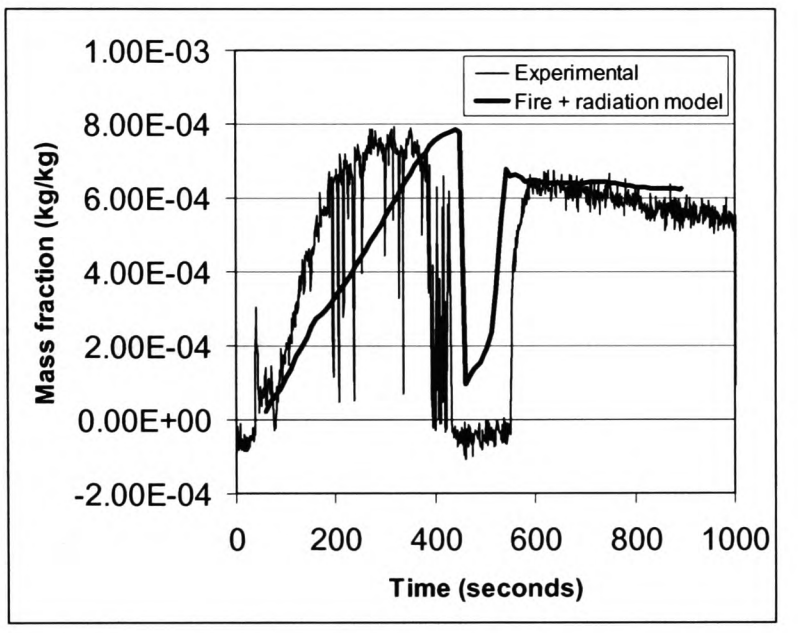


Figure 4-21 - 05PM case. Predicted and measured smoke variation at the chimney.

Figure 4-20 and Figure 4-21 compare predicted and measured concentrations of CO and smoke in the chimney. The same behaviour can be seen as for the other combustion products and in the 02HS case. That is, the concentration of combustion products increases as the fire burns, drops to ambient when the reverse flow through the chimney occurs and then increases again before slowly declining towards ambient values as the compartment is flushed.

The prediction of the smoke concentration is in good qualitative agreement with the experimental results. It is perhaps not surprising that there is a discrepancy in the level of quantitative agreement as the smoke concentration measurement is prone to error.

The level of CO in the chimney however is severely under-predicted, though the qualitative agreement is good. The explanation for this under- prediction of CO in the chimney (and over-prediction of temperature and CO_2 within the compartment) lies in the nature of the combustion processes occurring within the compartment. The fire within the compartment is ventilation controlled and as such does not burn as efficiently as a free burning pool fire. The release rates imposed on the calculations however are essentially derived from free burning pool fires scaled uniformly by a factor of 0.8 across time and also across all species. This is only an approximation however as the efficiency will have varied with time. It will also have varied across species. In particular, the rate of production of CO will increase and not decrease as the fire becomes oxygen depleted. Since no data was available to determine how much this increase was or how it varied with time the CO production rate had the same scaling factor applied as the other sources for consistency, i.e. 0.8. From Figure 4-20 it can be seen that an average scaling factor of around 5 may have been more appropriate for the CO.

4.4.2.3 06BXB - The 3 Box Fire Case

The fire source in this experiment consisted of three cardboard boxes measuring 0.5m by 0.5m by 0.5m arranged in a row. They were filled with shredded paper, ignited and allowed to burn out. This required approximately three minutes. The data was recorded throughout the burn and for approximately 10 minutes afterwards.

To simulate the problem the same mesh was used as for the 05PM case. The model was run for 885s, starting at 100s into the experiment. This required 48 hours of CPU time on a DEC Alpha 466MHz processor. As has been noted earlier this is a long time that may make the routine usage of such a model uneconomic for FSEs and regulators.

As with the pool fire case the fire was represented as a volumetric source of heat, CO_2, H_2O, CO and smoke and sink of O_2. The heat release rate was determined by SINTEF using the Oxygen Depletion Technique. The resultant curve is shown in Figure 4-22, the maximum heat release rate achieved was 405kW. Since this HRR was determined from the oxygen consumed rather than the fuel used it was not scaled as in the

previous simulations to allow for efficiency. The CO_2, H_2O and O_2 production/consumption rates were determined, as before, by assuming stoichiometric consumption rates, in this case that of cellulose. This results in maximum rates of 0.0181kg/s of CO_2, 0.0657kg/s of H_2O and -0.0372kg/s of O_2. The production rates of CO and smoke are discussed later.

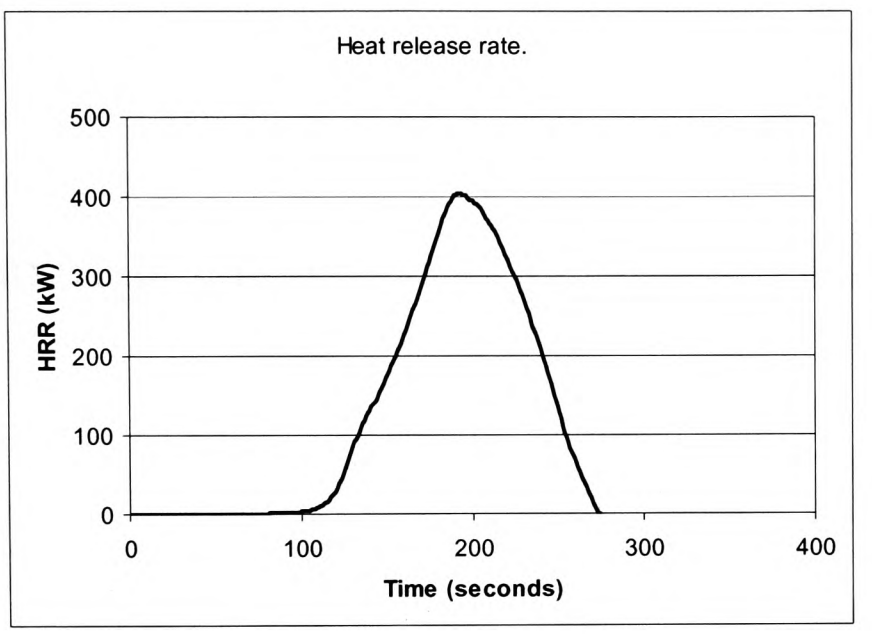


Figure 4-22 - Heat release rate for three box fire.

The main results for this simulation are presented in Figure 4-23 to Figure 4-31. These show comparisons between the experimental results and the model predictions for various parameters at various locations within the compartment as a function of time.

Figure 4-23 to Figure 4-25 depict the temperature distribution at the same three locations used in the previous cases i.e. Figure 4-23 is at thermocouple T12, Figure 4-24 is at thermocouple T18 and Figure 4-25 is thermocouple T36. As the HRR curve is strongly peaked the temperatures in the compartment follow a similar trend - a rapid rise to a high temperature followed by a slower decay as the fire self-extinguishes and the compartment starts to cool.

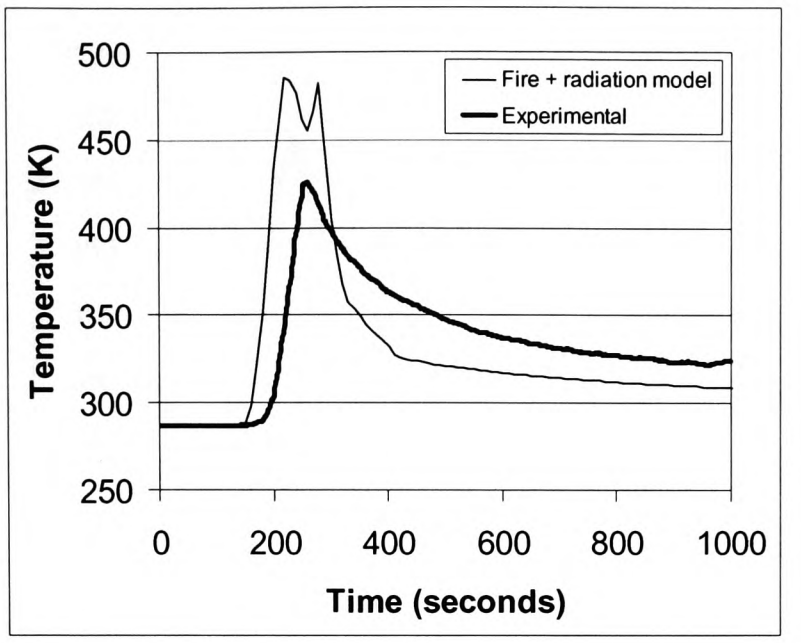


Figure 4-23 - 06BXB case. Predicted and measured temperature variation at T12 (near floor and doors).

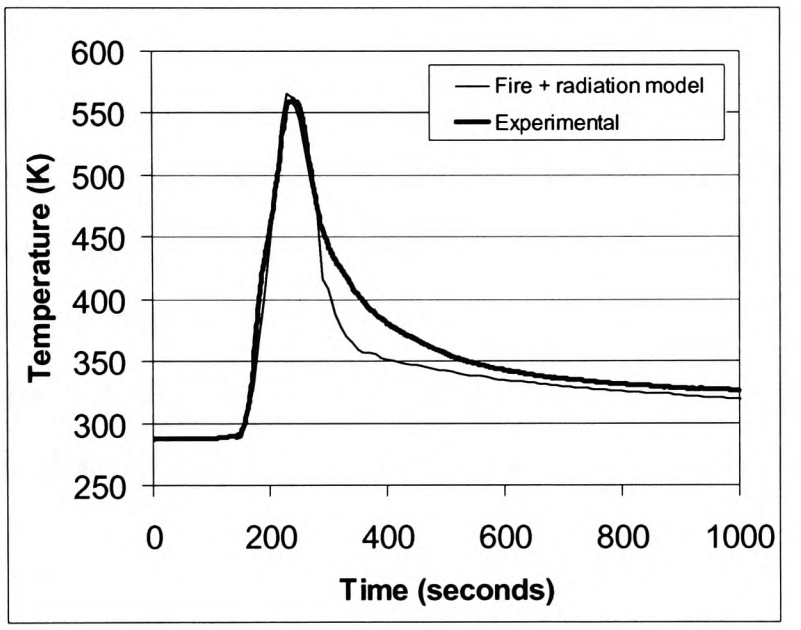


Figure 4-24 - 06BXB case. Predicted and measured temperature variation at T18 (near ceiling and doors).

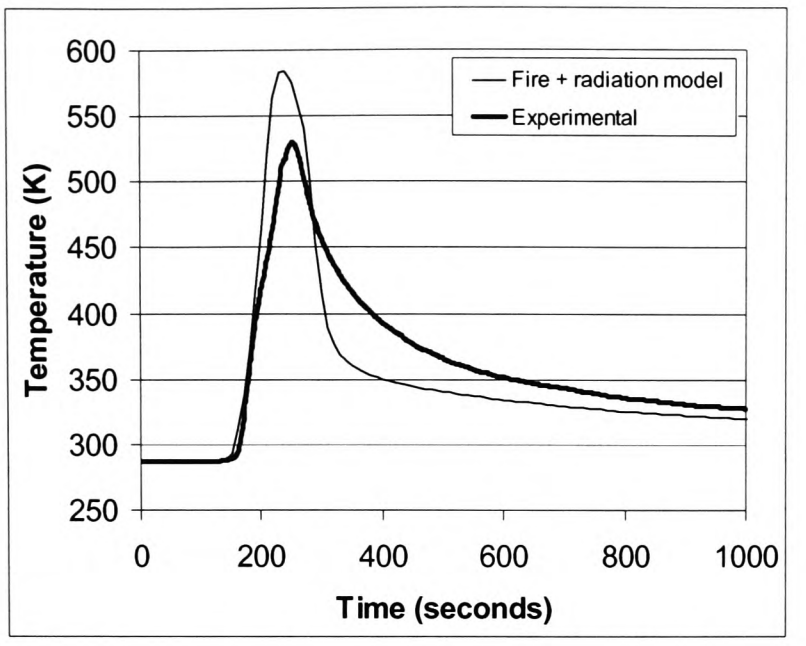


Figure 4-25 - 06BXB case. Predicted and measured temperature variation at T36 (3/4 height and central)

As in the previous two cases, there is quite good agreement between measured and predicted temperatures high up in the compartment. The growth and decay portions of the temperature curves and the peak temperatures are well captured by the numerical predictions. However once again, temperatures near the floor are over-predicted. Here we find that the growth rate of the predicted temperature is similar to that of the measured temperature but that the peak temperature is over- predicted by 14%. The main sources for this discrepancy are as explained previously for the 02HS and 05PM cases.

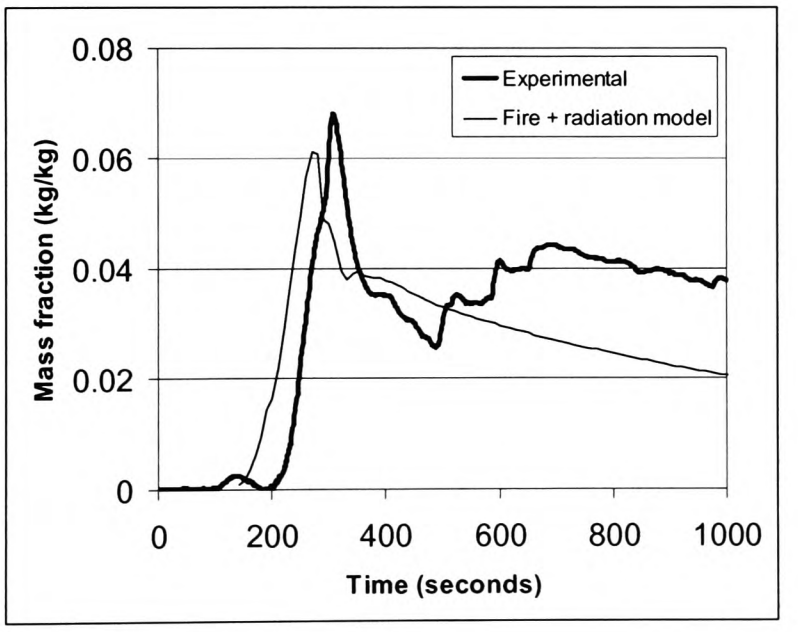


Figure 4-26 - 06BXB case. Predicted and measured CO_2 variation at room location G1.

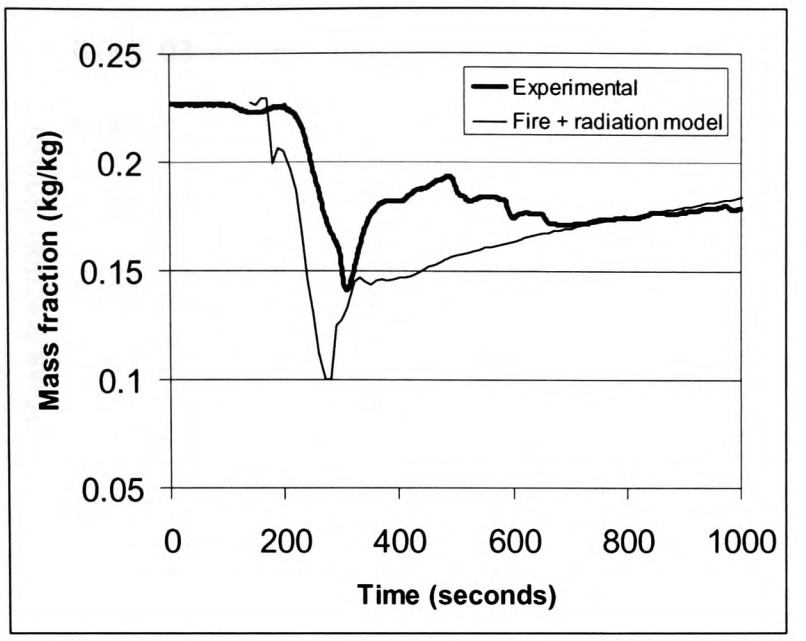


Figure 4-27 - 06BXB case. Predicted and measured O_2 variation at room location G1.

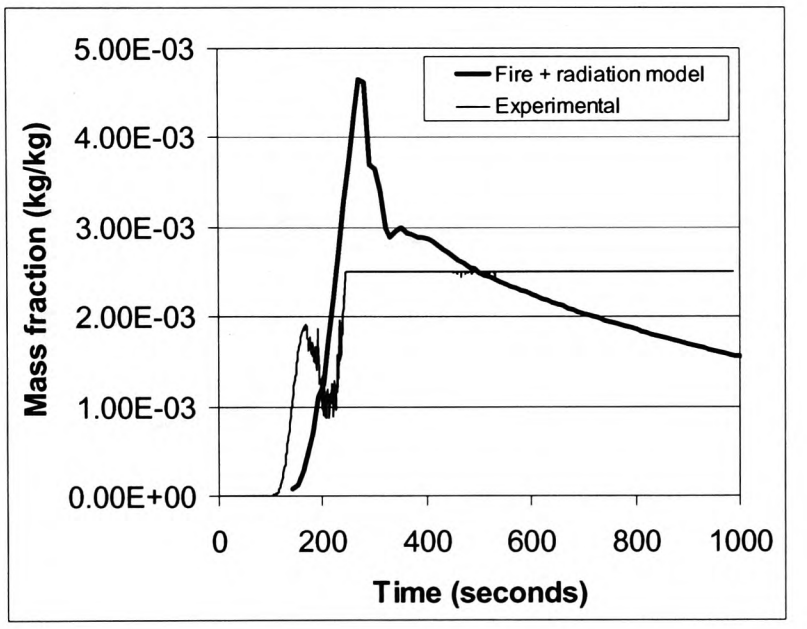


Figure 4-28 - 06BXB case. Predicted and measured CO variation at room location G1.

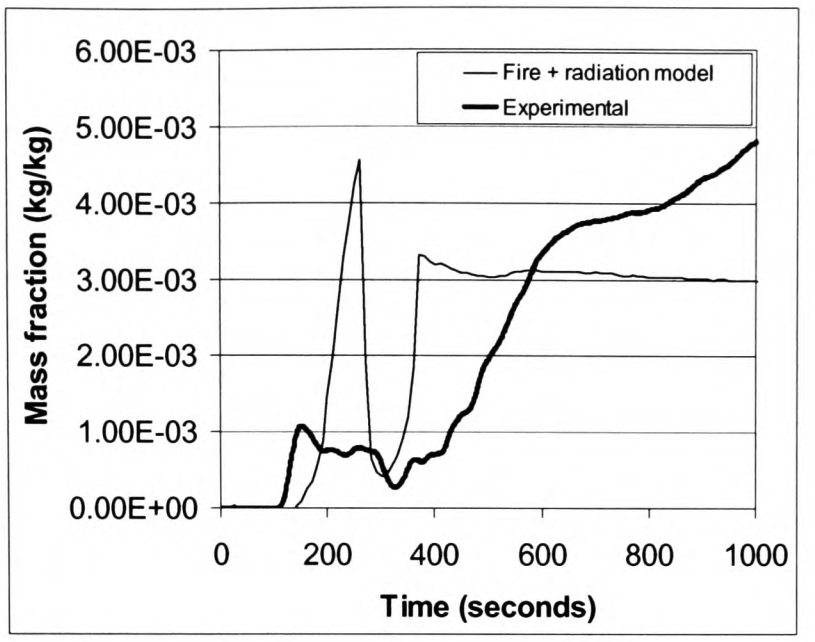


Figure 4-29 - 06BXB case. Predicted and measured CO variation at the chimney.

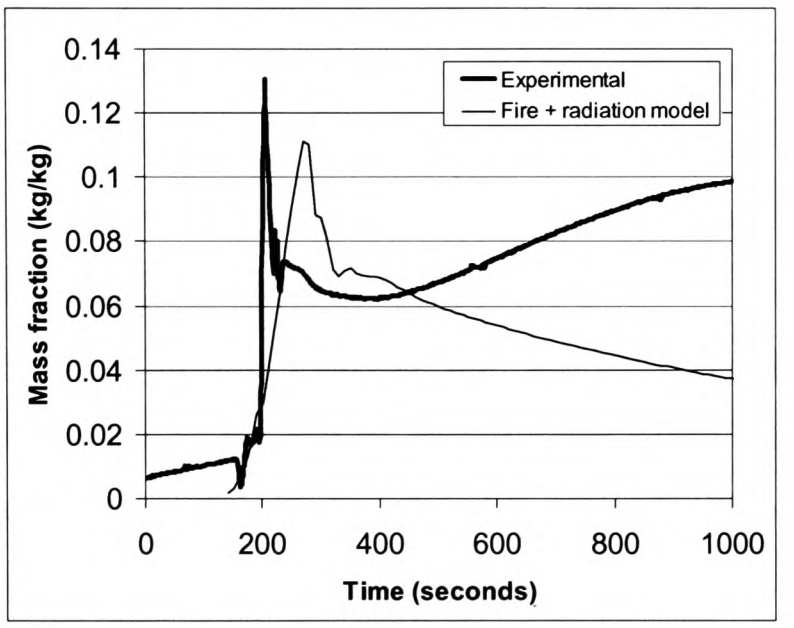


Figure 4-30 - 06BXB case. Predicted and measured water vapour variation at room location G1.

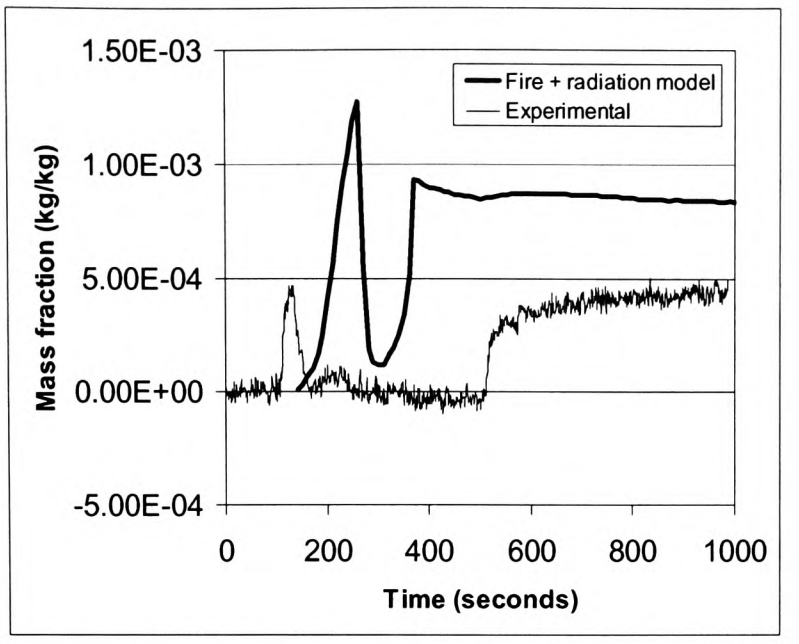


Figure 4-31 - 06BXB case. Predicted and measured smoke variation at the chimney.

Figure 4-26 to Figure 4-31 compare predicted and measured values for the combustion products. It can be seen from the experimental results that, as before, there is a short term build up in the concentration of CO_2 and decline in the concentration of O_2 in the room during the fire, Figure 4-26 and Figure 4-27. These changes are strongly peaked as the fire is strongly peaked. The return to ambient values however is much slower than in the previous cases, probably due to smouldering by the remains of the boxes. In the early stages of the fire reasonable agreement is obtained between the predicted and measured values for the CO_2 and O_2 concentrations. In the later stages however the agreement is poorer due to the sources going to zero while in reality the fire probably continued to smoulder. Hence the return of the predicted values towards ambient conditions while the measured values do not. This level of agreement in the first stages of the fire is expected as the sources used in the model for these species have been tuned using the experimental derived heat release rate. At the chimney, however, the agreement obtained is poorer for both species. This may be due to some experimental error or some phenomenon taking place in the compartment that was not modelled.

For the other combustion products the agreement is generally poorer. The experimental readings for the CO do not give a clear indication as to how the source of this species must be varying in order to produce the readings seen simultaneously at the chimney and in the centre of the room. As a compromise a constant yield was used to represent the source term. This was chosen to produce results that are of the correct order of magnitude at both the room centre and chimney measuring locations.

The generation of the soot was also not clearly understood so again a simple constant yield was used. The results obtained suggest that the yield used is of the right order of magnitude.

As in the other cases these discrepancies can be partially attributed to the modelling of the chimney. The model assumes that any air coming through the chimney from outside has mass concentrations of 23% oxygen and 77% nitrogen and a temperature of 288K (ambient). In reality the air that is sucked back through the chimney will be a mixture of ambient air and the exhaust air from the compartment. It will therefore be hotter and will contain some combustion products. This will mix with the air in the compartment and affect both the values measured in the compartment and the values measured in the chimney.

The successful outcome of the fire model development enabled the development of the other key components of the FIREDASS system, namely the mist, extinguishment, and detection/activation models.

4.5 Mist Model

The mist model is based on the PSI method developed by Crowe [Cro1977]. In this

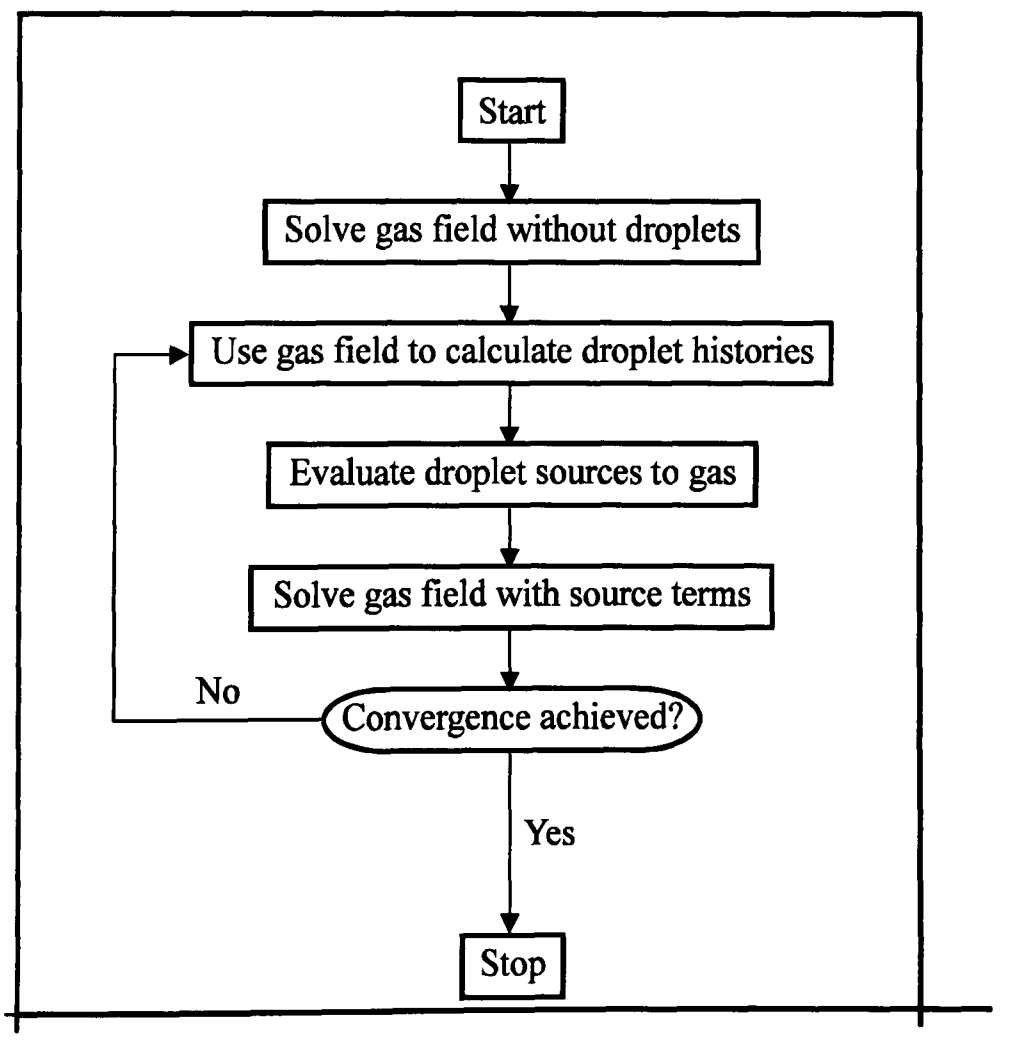


Figure 4-32 – PSI flow diagram

method a particle (water droplet) is tracked through the fluid domain. The physical properties of each of the particles are described by a set of Ordinary Differential Equations. These equations represent the droplet's velocity, diameter and temperature as it moves through the gas field. As the water droplet moves through the gas field it will exchange momentum, energy and mass with that gas field. This is represented as added source terms to the gas field equations. This changes the gas field that the particles originally traversed so the particles are now tracked through this new gas field which generates new sources to the gas field equations. This process is repeated until there is no change in the gas field (see Figure 4-32). The author did not write this particular section of the FIREDASS model so it will not be described in any further detail. Interested readers can find more information regarding the implementation and validation of the mist model in Mawhinney et al [MGPG1998].

4.6 FIREDASS Suppression model

The fire suppression submodel was developed by UoG [GGPM1998b] using an empirical criterion derived by SINTEF [Wig1998]. By considering the average temperature and oxygen concentration of the air entrained into the fire it determines the point at which the fire is extinguished. The original intention was to have detailed information relating combustion efficiency to temperature and oxygen concentration. However this data could not be assessed from the experiment and the less ambitious extinguishment criterion was developed instead.

Another possible method of suppression would be inclusion into a sophisticated combustion model. This possibility was not pursued due to the time constraints of the project and the possible impact on overall runtimes of the full model.

In this validation study the suppression model was not directly used, however the parameters of oxygen concentration and temperature have been accessed to give an indication of when the extinguishment criteria is met. This study is far more useful because it was possible to determine when the fire will be extinguished using a variety of parameters; this would not be possible if the suppression model was active during the simulations. It should be noted that even though in this study the extinguishment

criterion has not been used it was nonetheless correctly implemented within the FIREDASS computational model.

4.6.1 SINTEF Relation

The suppression model [GGPM1998b; Wig1998] supplied by SINTEF (hereafter referred to as the 'SINTEF relation') uses the following relation to determine when a fire is extinguished

$$O_{2,ext} = 20.9 - kT^n \tag{4.6.1}$$

where,

$O_{2,ext}$ is the critical volumetric oxygen concentration below which the fire can not be sustained if the entrained oxygen concentration is below this value.

T is the representative temperature of the air being entrained into the fire (K).
k and n are empirically determined quantities

The values for k and n suggested by SINTEF are given in Table 4-1 below.

Table 4-1 - SINTEF suggested values for extinguishment criterion

	K	N
Upper limit	0.000032	2.0
Lower limit	0.000067	2.0
Suggested Average	0.00004	2.0

The time of extinguishment was determined as the instant at which the predicted entrained oxygen concentration was less than the critical oxygen concentration as defined by the SINTEF relation. The 3 values of k will therefore give a range of extinguishment times. The upper limit defines the highest critical oxygen concentration and therefore the quickest extinguishment time.

4.6.2 Localised Suppression Zone

For the above relationship suitable representative values had to be determined for temperature and oxygen concentration of the air entrained into the fire. From the experimental values, from the SINTEF test campaign [WAD1997], the average

temperature was determined using all the thermocouples 600mm above floor level but excluding the thermocouples which were directly influenced by the flames. The gas analyser inside the compartment measured the average oxygen concentration (at 600mm above floor level). In the model a more accurate localised suppression zone can be defined (see Figure 4-33).

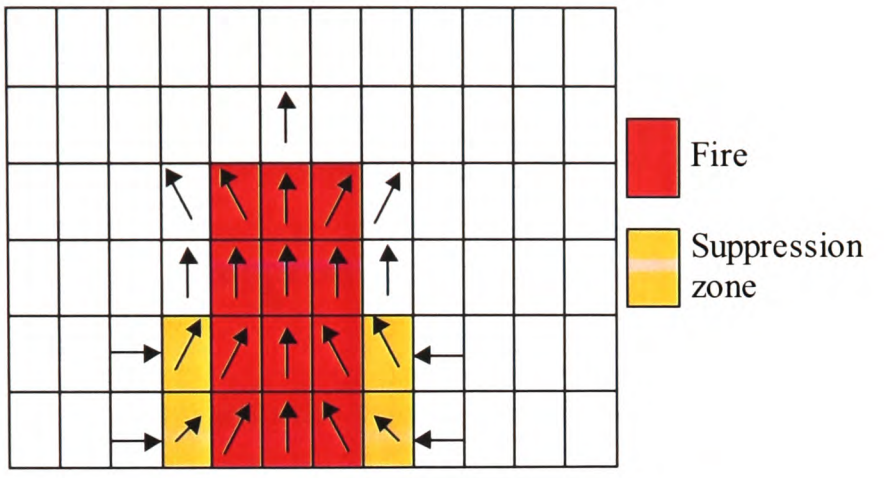


Figure 4-33 - 2D representation of suppression zone within a CFD fire scenario

The zone is defined to be the control volumes which faces contain the fire and further must have air flowing *into* the fire. This zone has the advantage of the fire only being influenced by local conditions. The localised zone therefore gives a much more accurate picture of the temperature and oxygen concentration of the air being entrained into the fire.

In a real fire the plume may lean over influencing the entrainment into the plume. This is not well represented by a volumetric heat source as the shape of the plume is prescribed by the volume and the entrainment may not be accurately represented within the suppression zone. The shape of the plume could be more accurately described by using a standard combustion model, such as the eddy break up model, or by using particle tracking to trace out the shape of the plume and adding a volumetric heat source to the plume envelope formed by massless tracer particles (c.f. FDS [MBR+2001] thermal elements).

For the combustion model the suppression zone could be defined to be all the cells containing no fuel that are adjacent to a cell containing some fuel with flow into the plume (see Figure 4-34).

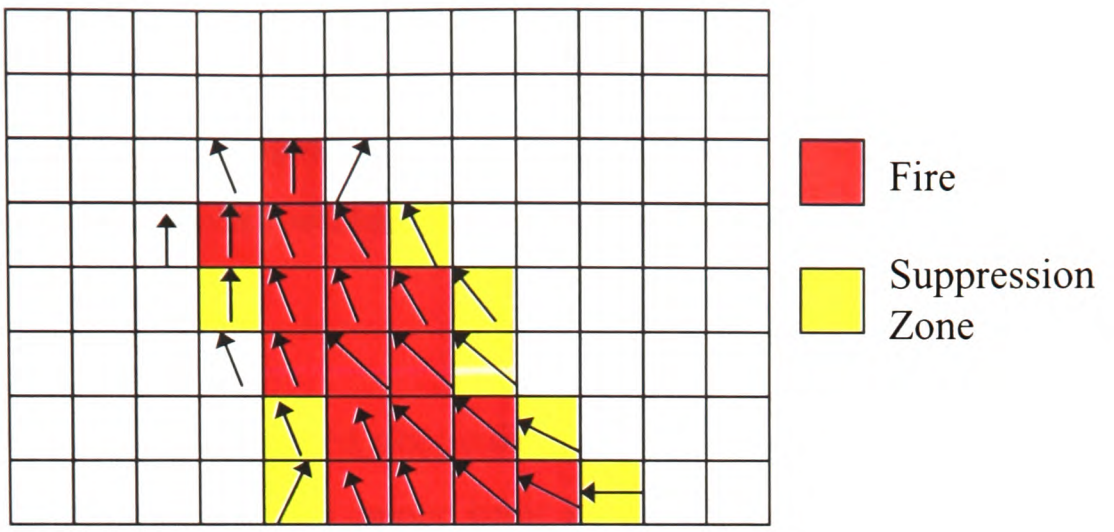


Figure 4-34 - Suppression zone with leaning plume

The average quantities flowing into the zone are calculated using a standard weighted average.

$$\bar{\phi} = \frac{\sum \dot{m}\phi}{\sum \dot{m}} \tag{4.6.1}$$

where,

$\dot{m}$ is the mass flux of air *flowing into* the fire through the control volume face

ϕ is the variable of interest (i.e. temperature or oxygen concentration)

4.6.3 Results

The following SINTEF test cases [WAD1997] were used for the analysis of the suppression model

1. 02HS – Self extinguishing gas burner case
2. 06BXB – Self extinguishing 3 boxes case
3. 11HS – Gas burner with single water mist nozzle active

The 02HS and 06BXB case were previously described in section 4.4.1, the 11HS case is the same as the 02HS apart from the mist nozzle at position N1 (see Figure 4-3) was activated 60s after ignition of the gas burner and then continuously operated until the end of the experiment.

Post processing of numerical simulations for 02HS, 06BXB and 11HS are presented below. A table is presented which contains the time of extinguishment, the percentage difference between experimental and numerically determined extinguishment time and the representative temperature and oxygen concentration at this time for the following:

1) The experimentally observed extinguishment.
2) Numerical simulation with upper limit of SINTEF relation (average values are computed in the same way as in SINTEF experiment).
3) Numerical simulation with middle value of SINTEF relation (average values are computed in the same way as in SINTEF experiment).
4) Numerical simulation with upper limit of SINTEF relation (average values are computed in the same way as in SINTEF experiment).
5) Numerical simulation with upper limit of SINTEF relation (average values are computed using localised suppression zone).
6) Numerical simulation with upper limit of SINTEF relation (average values are computed using localised suppression zone).
7) Numerical simulation with upper limit of SINTEF relation (average values are computed using localised suppression zone).

In addition to the table, two figures are presented for each test. In each figure the following quantities as they vary with time are plotted:

- The representative oxygen concentration
- The SINTEF relation (upper limit)
- The SINTEF relation (lower limit)
- The SINTEF relation (middle limit)

The intersection of the oxygen concentration line and the SINTEF relations will be the time of extinguishment according to the particular SINTEF relation intersected, thus a range of possible extinguishment time is created. The first figure represents the extinguishing limits and O_2 concentration when using spot values from the numerical tests with the same locations as the experimental measurements (the oxygen

measurement and the thermocouples at 600mm level). The SINTEF measurement points used in the numerical simulation allows the direct comparison with the experimental procedure with the numerical one. The second figure uses representative values for oxygen concentration and temperature taken from a localised zone around the fire (see Figure 4-33).

4.6.3.1 02HS

In the 02HS case the fire in located in the P2 position with a propane fire source.

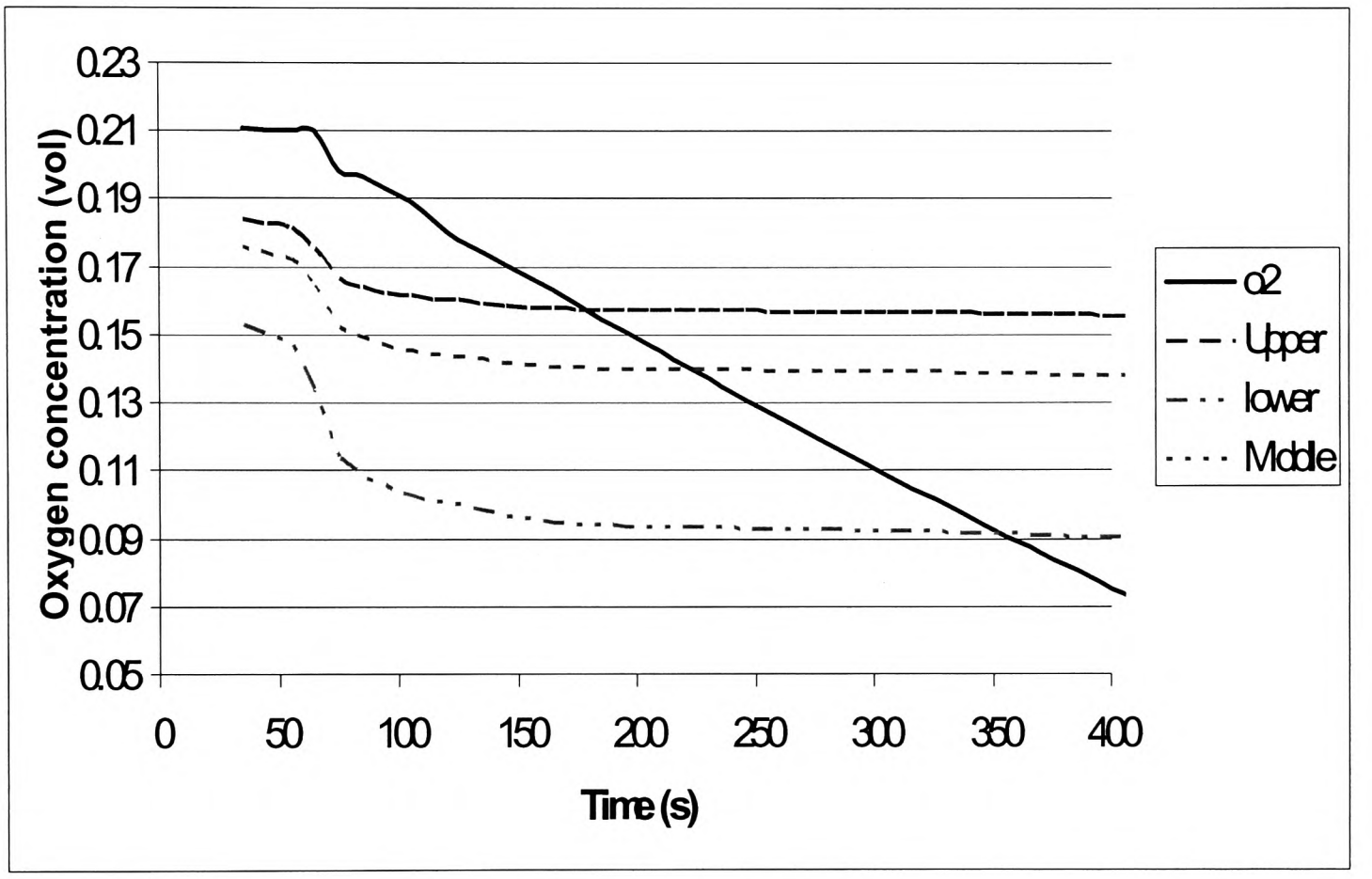


Figure 4-35 – Oxygen concentration and extinguishment criteria Vs time using SINTEF measurement points for 02HS

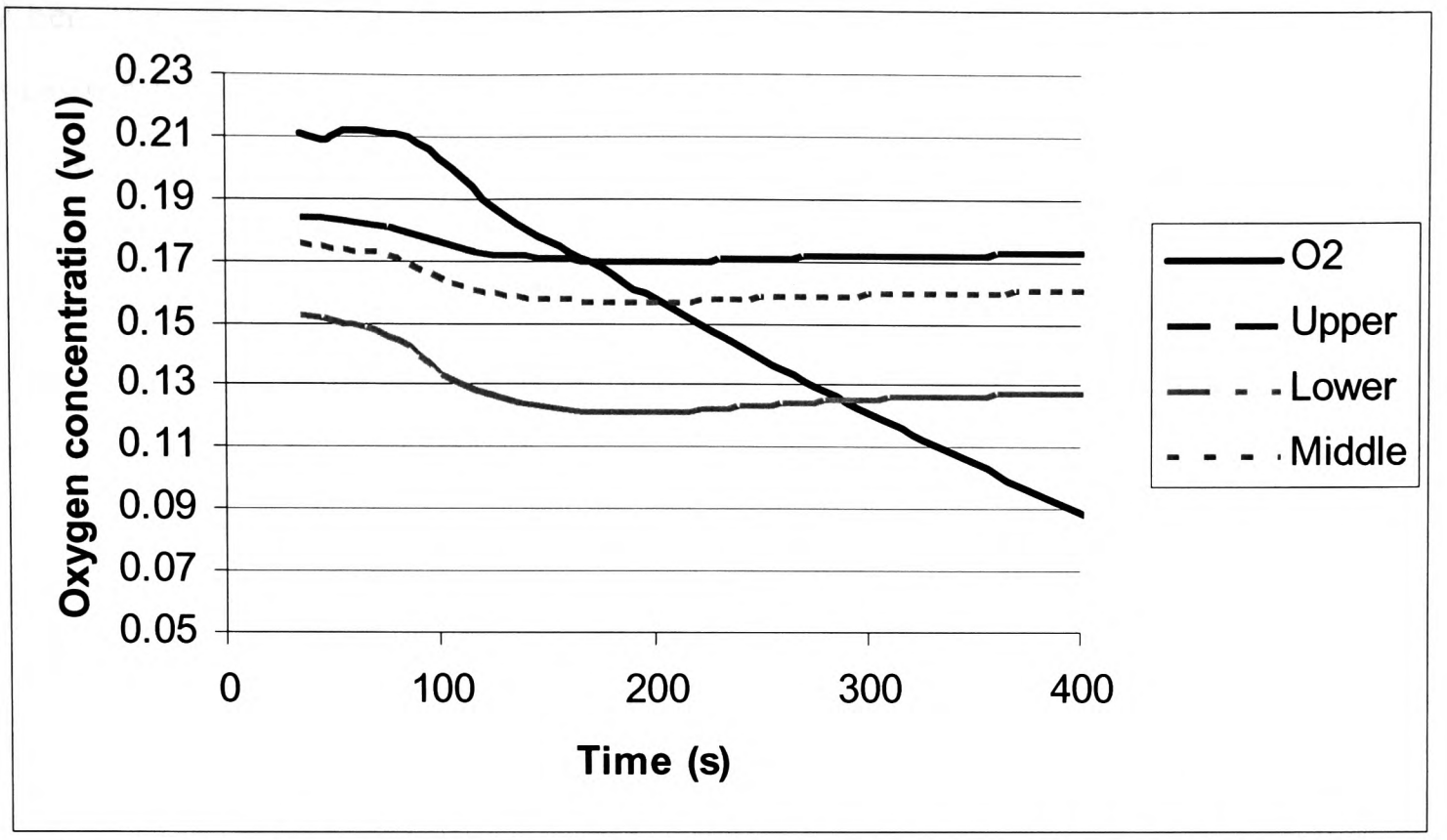


Figure 4-36 – Oxygen concentration and extinguishment criteria Vs time using localised zone for 02HS

Table 4-2 – Comparison of extinguishment for 02HS

		Time (s)	%diff in time	Temperature (K)	O_2 conc^{n} (%)
1	Experimental	382	0	392	10.8
2	Upper (SINTEF)	155	59	414	15.7
3	Middle (SINTEF)	205	54	415	14.0
4	Lower (SINTEF)	325	15	419	9.0
5	Upper (zone)	135	65	361	17.0
6	Middle (zone)	175	54	360	15.6
7	Lower (zone)	250	35	353	12.6

From Table 4-2 and Figure 4-35 it can be seen that using the SINTEF relation gives good agreement between the lower and middle values for oxygen concentration when the SINTEF measuring points are used. The time predicted by the lower limit was less than the experimentally measured result but numerically determined value was within 15% of the experimental value and this test experimentally defined the lower limit. The temperature was 23-28K higher than suggested by experiment. This was due to the overprediction of temperature near the floor level, which has been previously described (see section 4.4.2).

When the zone values were used much lower temperatures than the SINTEF measuring points were predicted; therefore higher critical oxygen concentrations were predicted and this leads to predicted quicker extinguishment times. The lower zone temperature can be partly attributed to entrained air at lower levels than 600mm being taken into account. Another reason for the difference is temperatures at the 600mm level are higher further away from the fire. This is due to the recirculation of air within the chamber (see Figure 4-37).

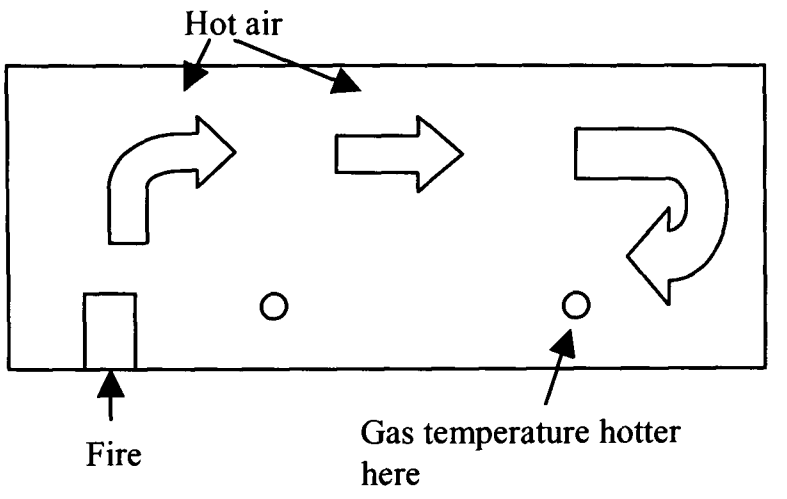


Figure 4-37 – Recirculation of hot air in SINTEF chamber

4.6.3.2 6BXB case

The 3 box case located in the P2 position with no mist. The 3 box fire has a peak output of ~300kW.

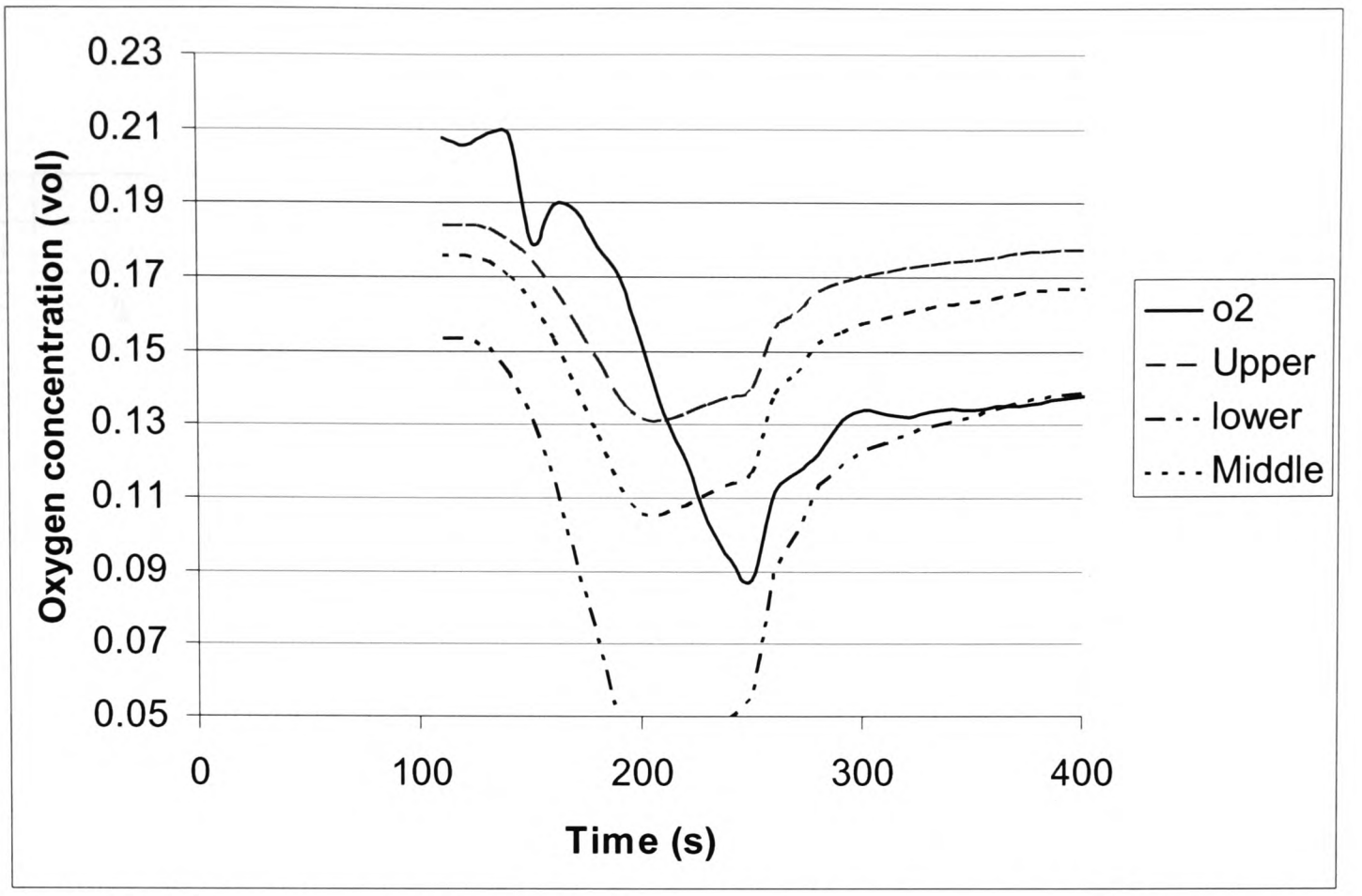


Figure 4-38 – Oxygen concentration and extinguishment criteria Vs time using SINTEF measurement points for 06BXB

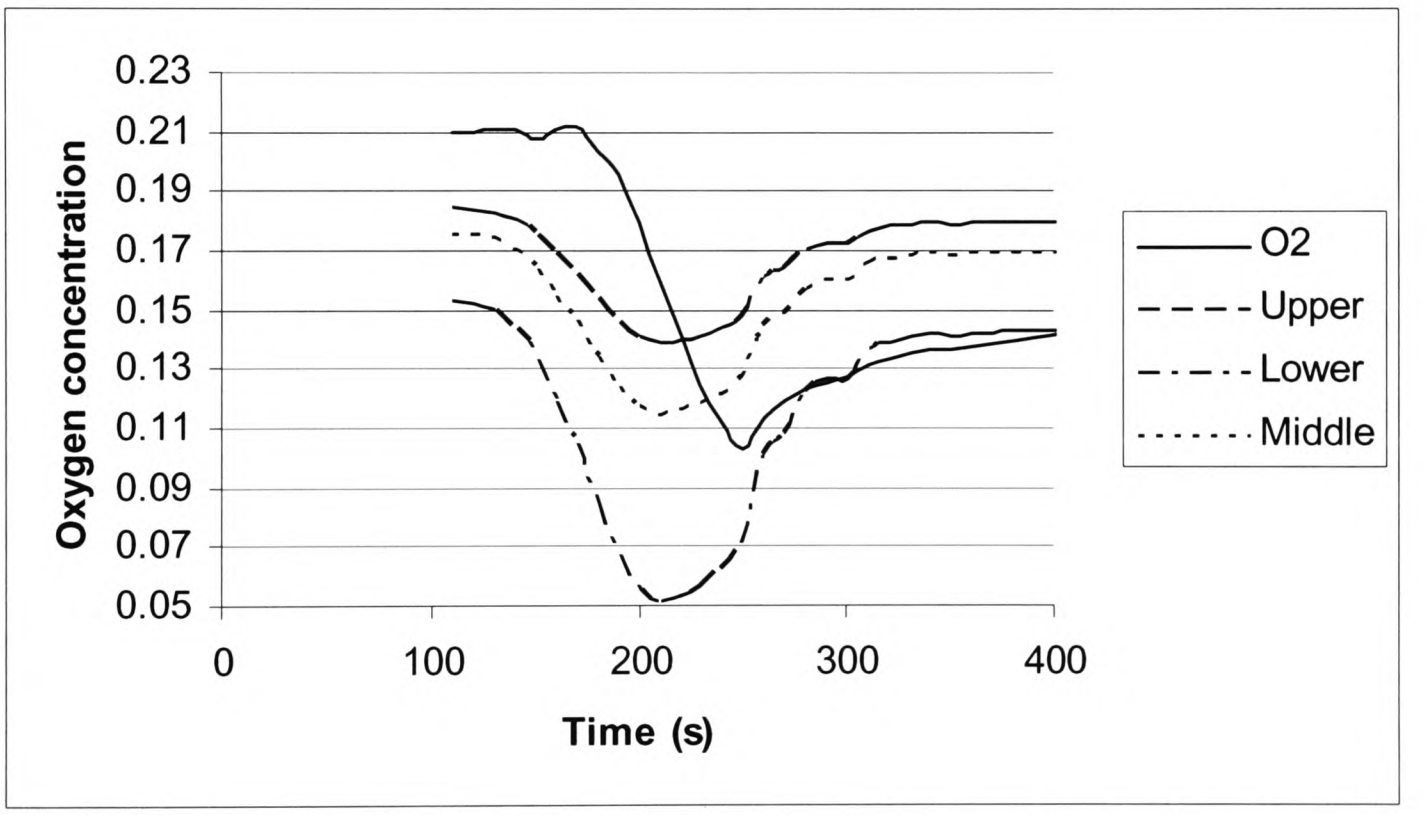


Figure 4-39 – Oxygen concentration and extinguishment criteria Vs time using localised zone for 06BXB

Table 4-3 – Comparison of extinguishment for 06BXB

		Time (s)	%time diff	Temperature (K)	O_2 conc^n (%)
1	Experimental	245	0	442	11.7
2	Upper (SINTEF)	132	46	502	11.9
3	Middle (SINTEF)	152	28	494	10.3
4	Lower (SINTEF)	278	-13	333	13.1
5	Upper (zone)	128	48	485	15.8
6	Middle (zone)	138	44	482	14.0
7	Lower (zone)	198	19	360	12.1

From Table 4-3 and Figure 4-38 it can be seen that using the SINTEF relation gave good agreement between the lower and middle values for time and oxygen concentration when the SINTEF measuring points were used. The lower and middle range bracketed the experimental temperature. This temperature drop between 230s and 360s was due to the decay of the heat release between these points. This case highlights the problem with the extinguishment criterion; a small variation of the k parameter at the lower level may lead to large variations in the extinguishment time.

In the case of the 3 boxes there is only a small variation between the localised zone (Figure 4-39) and the SINTEF measurement locations (Figure 4-3). The trends exhibited in the 02HS case still remain with lower temperatures predicted by the zone, which implies a higher critical oxygen concentration. Although the difference to the upper and middle values was small the lower value predicts extinguishment 80s earlier due to the extremely sensitive nature of the SINTEF relation.

4.6.3.3 11HS case

This case was setup identically to 02HS (see section) but now nozzle N1 was activated 60s after ignition of the burner to observe the influence of the watermist in the far field.

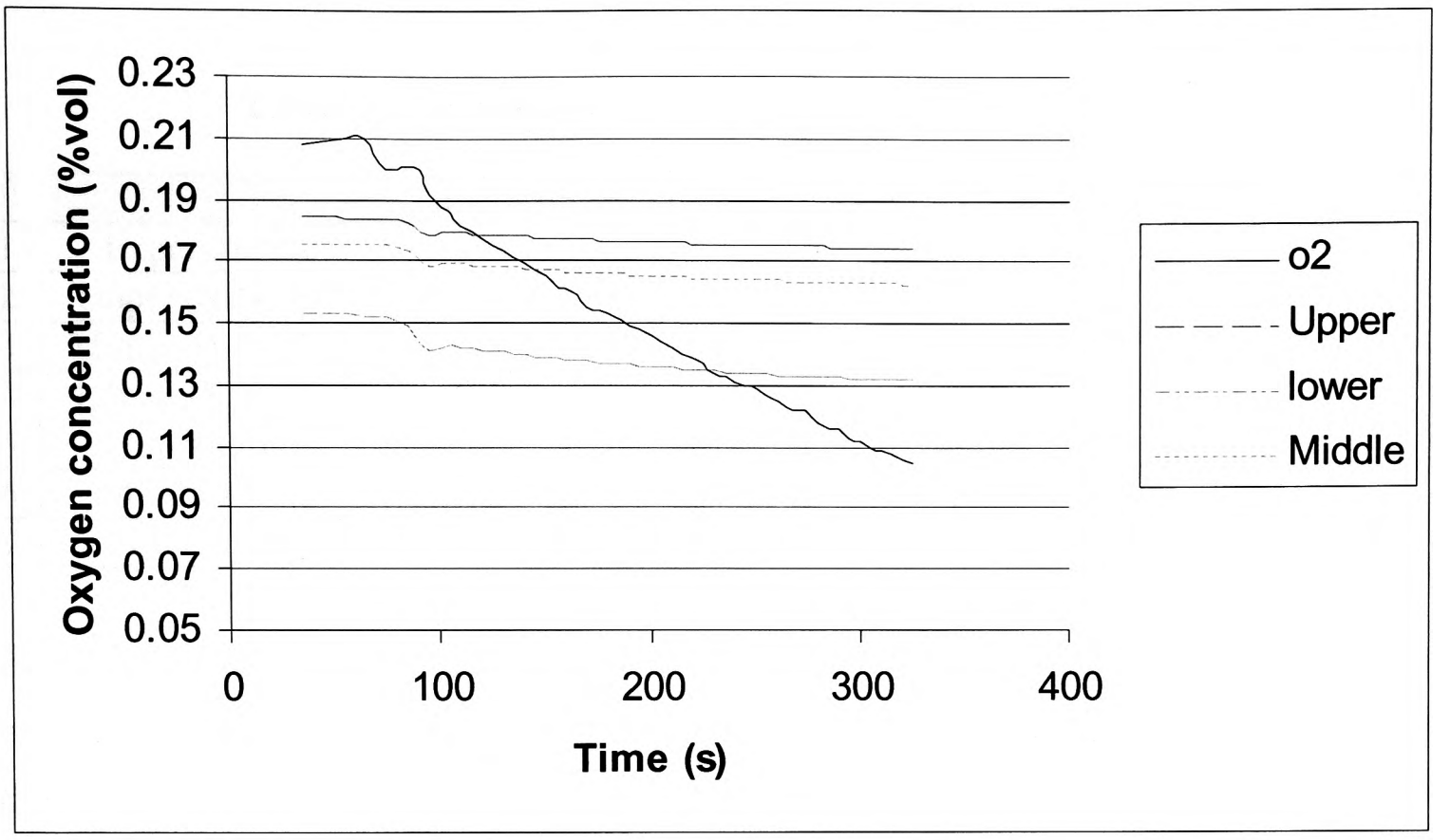


Figure 4-40 – Oxygen concentration and extinguishment criteria Vs time using SINTEF measurement points for 11HS

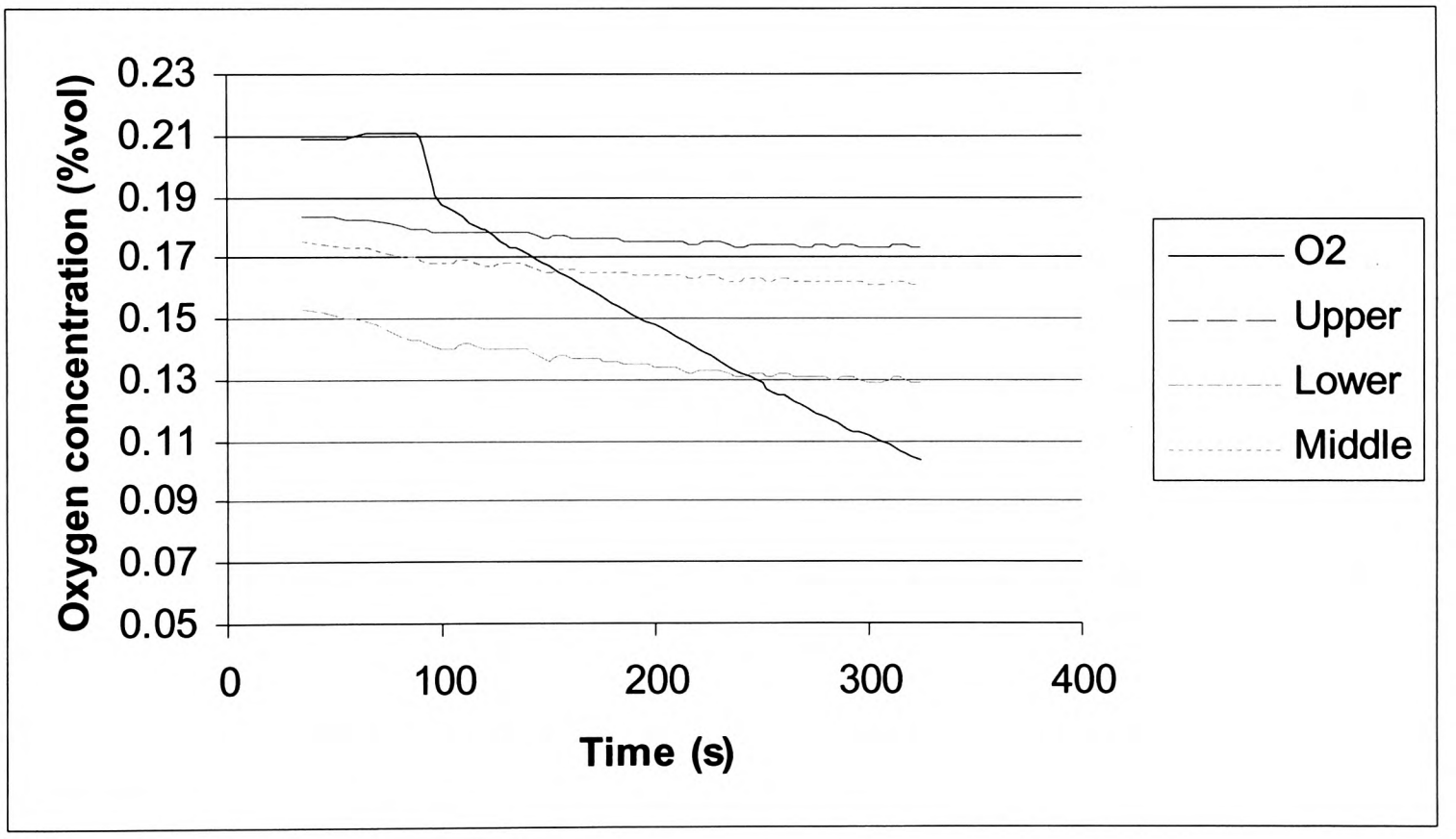


Figure 4-41 – Oxygen concentration and extinguishment criteria Vs time using localised zone for 11HS

Table 4-4 – Comparison of extinguishment for 11HS

		Time (s)	%time diff	Temperature (K)	O_2 conc^n (%)
1	Experimental	296	0	352	13.1
2	Upper (SINTEF)	85	71	317	17.8
3	Middle (SINTEF)	120	59	323	16.5
4	Lower (SINTEF)	205	31	333	13.3
5	Upper (zone)	95	68	320	17.8
6	Middle (zone)	125	58	325	16.7
7	Lower (zone)	215	27	339	13.2

From the above table and figures it can be seen that using the SINTEF relations do not bracket the experimental results. However, the results for the lower limit were close to those of the experimental result. The predicted oxygen concentration was very close to the experimental value although the temperature was lower than expected

The results using the suppression zone were very similar to the results obtained using the SINTEF measuring points.

4.6.4 Comparison of 02HS and 11HS cases

Comparing 11HS and the 02HS case which is the gas burner but no mist the following features are observed:

1) The temperature of the entrained gas is much lower when the mist is activated.
2) There is some oxygen displacement observed due to the presence of water vapour

These 2 factors combine to reduce the time to extinguishment. The most important factor is the cooling of the entrained gas.

4.6.5 Discussion

The SINTEF criterion when used in the FIREDASS model tends to underpredict the extinguishment time for propane burner cases, overpredicts the extinguishment time in the case of the large pool fire, while for the 3 box fire, the extinguishment time was bracketed by the lower and middle values. The difference observed between experimental and computed results were reasonable given all the approximations/assumptions which need to be entered into the numerical model. It

would be possible to adjust the parameters of the SINTEF relation to fit our results but there is still some degree of uncertainty about release rates and boundary conditions. The 02HS case also highlights the possible discrepancy between using SINTEF locations and the localised suppression zone for the model. In order to get better values of k and n for the SINTEF relation more detailed experiments would be required. As noted in section 2.3.1.3.1 it is important to conduct repeat experiments but this was not possible due to the cost of the additional experiments.

The SINTEF relation is only a criterion that states whether the fire is extinguished or not. Within the model it behaves as a switch, turning the fire model off. However, in reality as conditions near the extinguishment criteria, the fire begins to die down. This important behaviour is not included in the SINTEF formulation. A possible procedure to extend the method to produce a partial suppression near the point of extinguishment. This would dynamically ramp down the fire release rates according to the difference between the critical oxygen concentration and the measured oxygen concentration. The following relation (which incorporates the SINTEF criteria) is proposed:

$$P = \frac{O_2 - (20.9 - kT^n)}{O_2} * 100 \qquad (4.6.5.1)$$

Where P is the percentage to extinguishment

If $P > P_{crit}$ then Combustion efficiency = 1

If $0 < P < 10$ then Combustion efficiency is P/P_{crit}

If $P < 0$ then combustion efficiency = 0

Where P_{crit} is a definable critical percentage where the ramp off in combustion efficiency is started.

This method has been implemented and tested in the model but no work has been presented here due to the uncertainties of the k and n parameters and also the

uncertainty associated with arbitrary P_{crit} value. Considerable extra work is required for both the accurate use of the SINTEF style relation and the use of this ramping procedure.

4.7 DLR Model Prediction

The full FIREDASS model was used to simulate the fire and suppression of the fire within a mock up of a A340 cargo bay. This included the use of the Fire submodel, the mist submodel, the radiation submodel and the DE/AC submodel. The suppression model was used in a post-processing mode to predict the time of extinguishment.

4.7.1 Development

This Model prediction was based on the DLR test chamber [Oli1998]. The DLR test chamber was a full scale mock up of an A340 fuselage containing the front cargo compartment. The cargo compartment was based on the type C cargo compartment and measured 14.80m x 4.2m x 1.73m. It was covered using the same cargo liner material that is used in the A340.

Two computer models of the DLR chamber were created.

1) The Quarter chamber – 2 planes of symmetry were assumed so that only a quarter of the chamber is simulated.
2) The full chamber – No symmetry was assumed and the whole compartment is simulated. This requires far greater computational effort than the quarter chamber.

The DLR model was a specific extension of the generic interaction model [GGPM1998b]. The extension required the following parameters

- Geometry and CFD mesh of cargo compartment
- The nature of the walls (specified heat fluxes, temperature)
- The position and behaviour of the fire
- The position and behaviour of the misting nozzles
- Ventilation characteristics

The DLR test cell was meshed on a single simple block using 67 x 19 x 19 cells (24187 total cells) for the ¼ chamber arrangement and 134 x 19 x 38 cells (96748 total cells) for the full chamber. The cells were meshed in such a way that a higher density of cells existed around the fire and the walls of the test chamber. The ventilation was modelled using a novel method for ensuring mass balance for the domain (see section 4.7.1.1). This system is particularly useful when there are no obvious inlets or outlets into the domain and the leakage can be ascribed to the small cracks and openings that would exist within the chamber.

As the FIREDASS model assumes weakly compressible flow, mass balance into and out of the chamber must be achieved; this is not necessary the case in the more general fully compressible flow. The weakly compressible flow option is good for flows less than 0.3 Mach and was therefore suitable for the FIREDASS model.

4.7.1.1 Mass balance system

The total expansion and contraction of air in the compartment can be defined by the following equation.

$$\dot{m}_{total} = \left(\frac{\sum_{i=1}^{n} \left(\rho_i^{new} - \rho_i^{old} \right) V_i}{dt} \right) + \dot{m}_{ventilation} + \dot{m}_{fire} + \dot{m}_{vapour} + \dot{m}_{nozzle} \qquad (4.7.1.1.1)$$

where,

$\frac{\sum_{i=1}^{n} \left(\rho_i^{new} - \rho_i^{old} \right) V_i}{dt}$ is the mass rate due to thermal density changes and mass fraction changes.

ρ_i^{old} is the density of cell i at the beginning of the computational time step.

ρ_i^{new} is the density of cell i at the end of the computational time step.

V_i is the volume of the computational cell i.

dt is the length of the computational time step.

$\dot{m}_{fire}$ is the additional mass flux due to the combustion products of the fire

$\dot{m}_{ventilation}$ is the additional mass flux due to the ventilation system

$\dot{m}_{vapour}$ is the additional mass flux due to the evaporation of water mist

$\dot{m}_{nozzle}$ is the additional mass flux due to 2nd phase propellant coming out of the misting nozzles.

In order to achieve mass balance inside the chamber it is necessary to ensure that $\dot{m}_{total}$ is balanced by an equal and opposite mass flux. This mass flux can be removed or added over the whole domain or over a particular zone. The source term for the pressure equation can be represented by :-

$$S_i = \frac{\dot{m}_{total}}{\sum_{i=1}^{zone} V_i} V_i \qquad (4.7.1.1.2)$$

Source terms are also required to ensure that the gas species mass fractions are correctly maintained.

$$S_{i,O_2} = X_{O_2} \frac{\dot{m}_{total}}{\sum_{i=1}^{zone} V_i} V_i \qquad (4.7.1.1.3a)$$

$$S_{i,CO_2} = X_{CO_2} \frac{\dot{m}_{total}}{\sum_{i=1}^{zone} V_i} V_i \qquad (4.7.1.1.3b)$$

$$S_i = \frac{\dot{m}_{total}}{\sum_{i=1}^{zone} V_i} V_i \qquad (4.7.1.1.3c)$$

Within the DLR compartment it is assumed that all the leakage is due to cracks in the door seal so the zone is distributed over this area.

4.7.2 Comparison of model with experiment

4.7.2.1 Introduction

The model was compared to the DLR T9 case (3 box fire only), the DLR T2 case (3 box fire with 5 nozzles active) (see Table 4-5) and the DLR T8 case with the Cerberus level 1 detection /activation system operational.

In the model it was assumed that the walls are perfectly insulating to both radiation and convected heat. This was assumed, as there is a lack of information about the walls and the walls were made of an insulating material; this was unlike the SINTEF and GMAv test chambers that possessed metal walls. The result of this assumption was that predicted temperatures will be much higher than anticipated. In the computer model it was further assumed that the 3 box heat release rate calculated from the SINTEF trials were representative of the burning boxes within the DLR test chamber.

Table 4-5 - Summary of DLR test cases

T9	3 box fire located at position P1
T8	3 box fire located at position P1 with the nozzles controlled by the detection activation model using level 1. Activation is based on smoke detection and a critical temperature of 130°C (403K).
T2	3 box fire located at position P1 with 5 nearest nozzles activated using a pulsed 15s on / 30s off pulsed sequence after a 90s preburn period

The simulations were conducted on the ¼ chamber. The position of the nozzles and thermocouples are shown below in Figure 4-42. The thermocouples were located just below the ceiling. The ¼ nozzle and the full nozzle were equivalent to 5 nozzles due to the symmetry of the problem. If the 3 nozzles illustrated in Figure 4-42 were active this is equivalent to nozzles N9, N10, N11, N12, N13, N14 and N15 being active in the full size chamber.

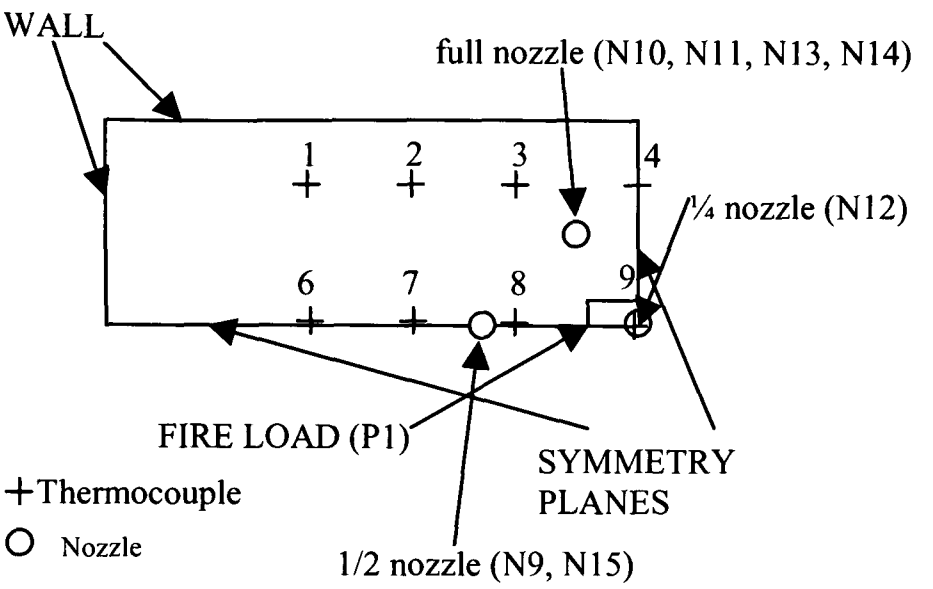


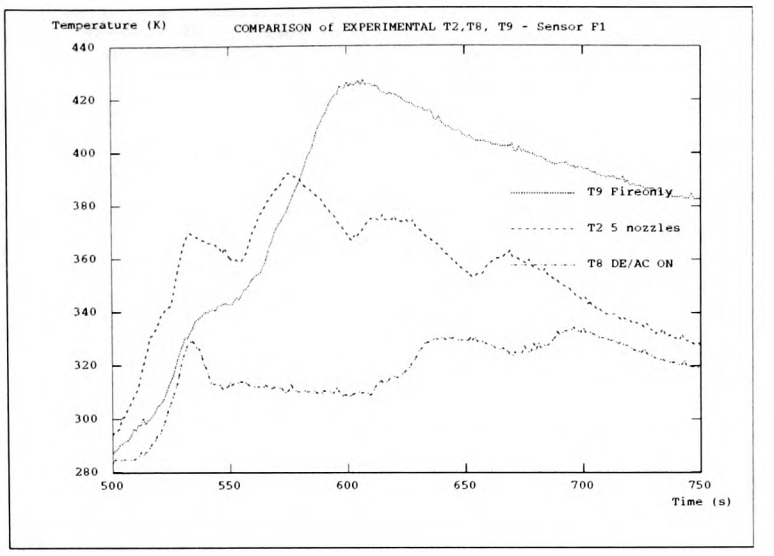
Figure 4-42 - Plan view of DLR 1/4 Chamber

The thermocouples selected for the comparison are thermocouples 1 and 8.

4.7.2.2 Results and discussion

Due to the nature of the simulations the start point was different to the experimental values as there is no need to simulate 500s of inactivity. It must be noted that the computational results do span the same time range.

4.7.2.2.1 Thermocouple 1 (F1)



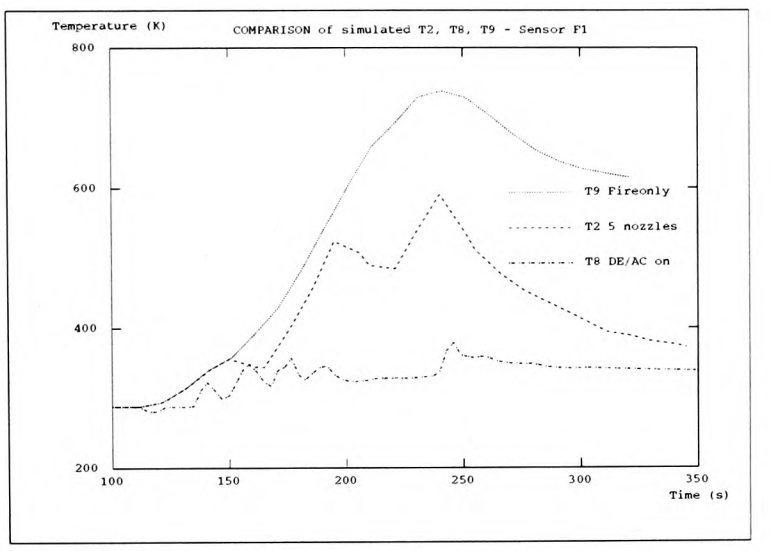
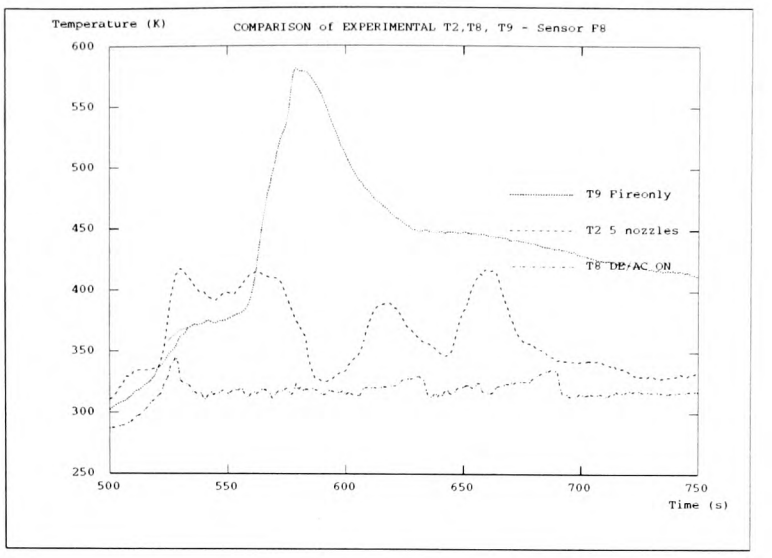


Figure 4-43a/b - (a) Experimental comparison of T2, T8 and T9 at thermocouple F1 and (b) same comparison with computed results at thermocouple F1

The figure on the left hand side of Figure 4-43 shows the experimental comparison between T2 and T9 and the figure on the right hand side shows the computational comparison between T2 and T9. From the above graphs it can be seen that the temperatures were far higher in the simulated fire scenario. This was expected due to the use of insulating wall conditions. It can be seen though that the same trends were observed in computed results with the pulsed nature of the nozzles coming through. In the T7 case it can be seen that significantly lower temperatures were achieved both in the experimental case and in the computed case compared to the T9 fire only case.

4.7.2.2.2 Thermocouple 8 (F8)



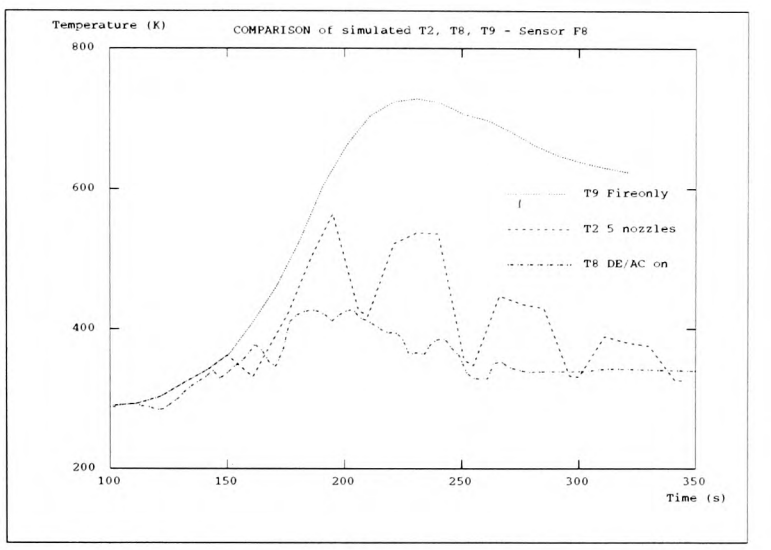


Figure 4-44a/b - (a) Experimental comparison of T2, T8 and T9 at thermocouple F8 and (b) same comparison with computed results at Thermocouple F1

In Figure 4-44 it can be seen that the simulated temperature at sensor 8 was much closer to the experimental than at the sensor 1. This was expected due to the closer proximity of thermocouple 8 to the fire. In the computational case, due to the insulating walls, there will be no heat loss as the hot layer moves across the ceiling. In the experimental case there will be heat losses as the hot layer moves across the ceiling away from the fire. Heat was continuously lost causing a reduction in layer temperature with displacement. Again it can be seen that there is a significant reduction in temperature when the detection / activation system is used.

Differences in the model and the experimental results have been highlighted previously [GGPM1997b, 1999] and in this chapter. These can be summarised as:-

- Lack of repeat tests to verify the reliability of the data.
- Simple switch mechanism to represent complex combustion process the suppression of the fire is poorly represented.

- Difficulties in establishing real heat/gas species sources for the various fires in the various configurations.
- Lack of data about boundary conditions

4.7.2.3 Detection / Activation sequence

The DE/AC sequence used in the model was the same as that used in the DLR type C cargo mock-up. This sequence was initially based on smoke detection and then used temperature measurements for subsequent misting control [Odi1999].

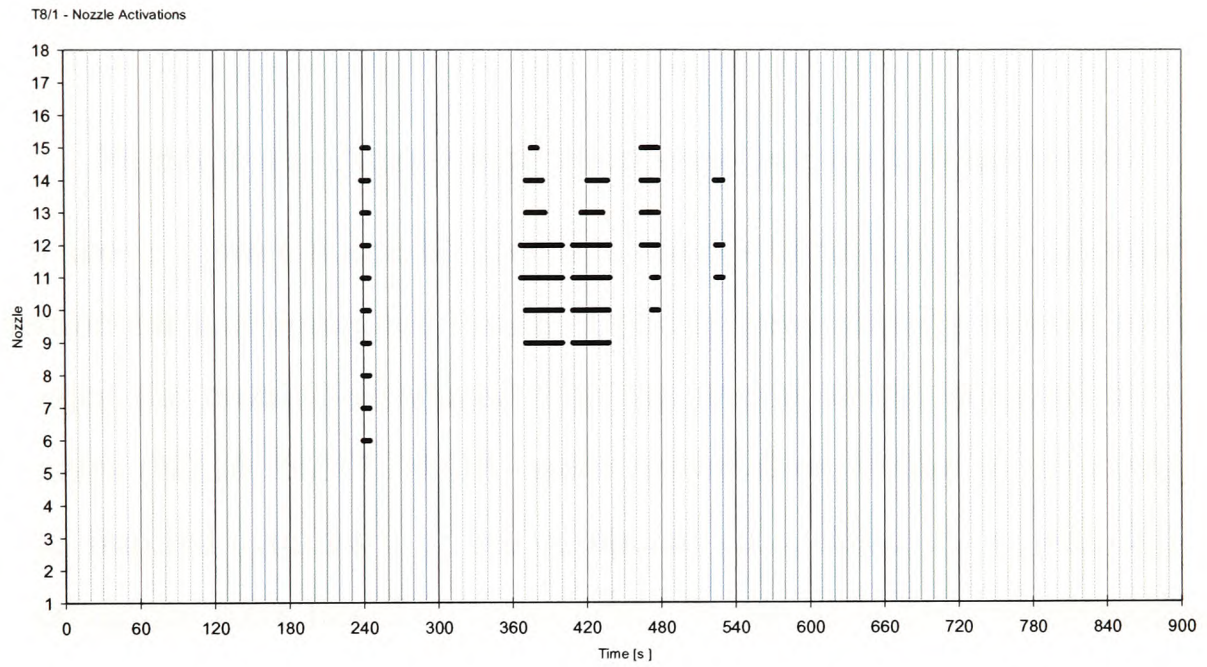


Figure 4-45 - Nozzle activation sequence for Experimental T8 (supplied by DLR)

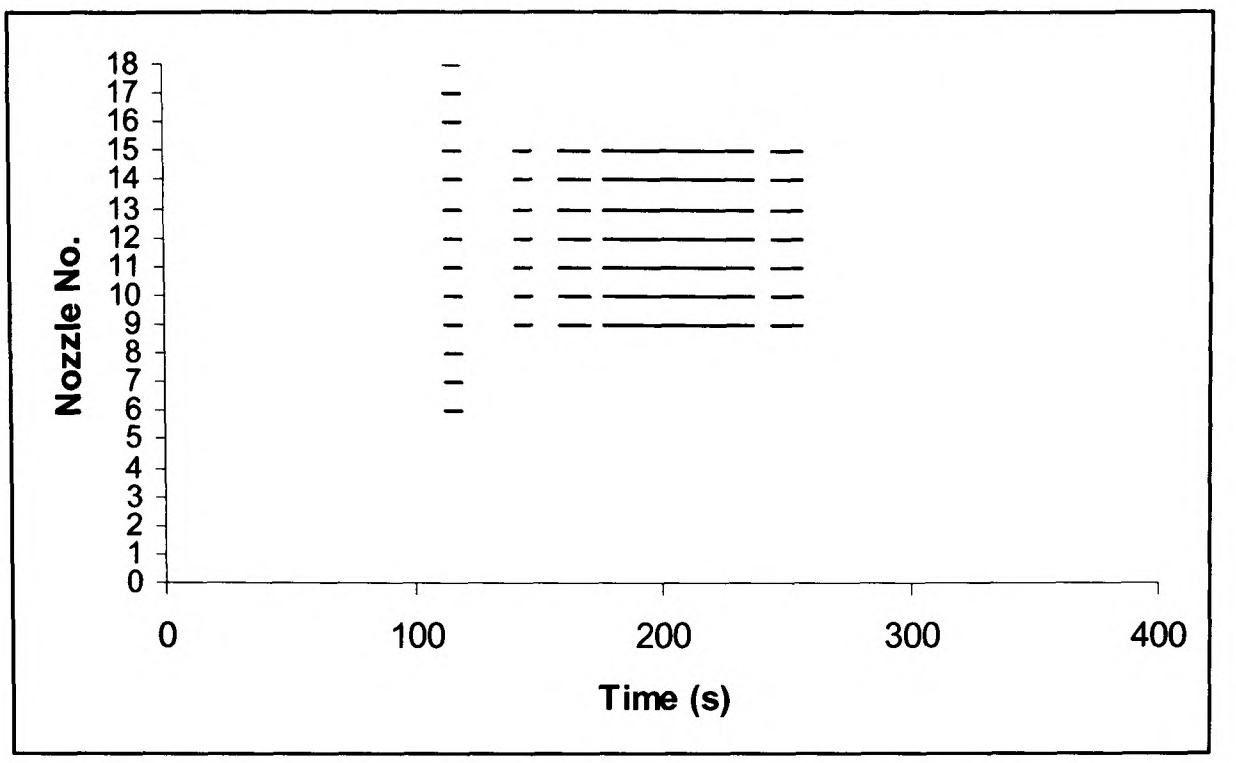


Figure 4-46 - Nozzle activation sequence for Computed T8

From the above figures it can be seen that similar behaviour was exhibited between the experimental and computed results. Both systems initially reacted to the smoke and activated the nozzles associated with the central smoke detector for 6 seconds. More nozzles (NZ16, NZ17, NZ18) were activated on smoke in the computed case due to the implied symmetry of the computed case. After the smoke activation all further activations were based on temperature. It can be seen by comparing the range of activation on both graphs that the nozzles were on for approximately the same duration. The main difference is the time between nozzle activation based on smoke and then temperature sensors. There are many possible reasons for this which includes the effect of water mist on the real temperature sensors which is not modelled, the initial fire reduction effect which is not modelled, and inaccurate or poor modelling of the smoke sensor. The same nozzles were activated in both cases although the computed case did observe symmetry. The model predicts that 7 nearest nozzles were sufficient to control the temperature within the compartment, this also proved to be the case experimentally.

4.7.2.4 Water usage for mist system

Table 4-6 - Comparison of water used by the computed and experimental T8 case.

Case	Water Used (litres)
Experimental T8	28
Computed T8	44

The amount of water used in the computed case was in reasonable agreement with the experimental value (see Table 4-6) considering all the approximations in the model and the lack of certainty associated with the experimental values. The experimental result represents the outcome from a single experimental trial. It should also be noted that the numerical fire conditions may be more severe than the experimental fire as the numerical fire is not suppressed (i.e. ramped down) on the way to extinguishment.

4.7.2.5 Runtime

The runtime of the above ¼ chamber took 130+ hours to run on a Dec alpha 433MHz. The runtime may be reduced by the following:

1) Not calculating gas mass fractions could reduce this time by a third but would prevent use of the extinguishment criterion.
2) Not using the radiation model although this may lead to inaccurate results for larger fire loads.
3) Advances in hardware (i.e. faster processing times).
4) Advances in algorithms (i.e. more efficient solution strategies)
5) Parallel Processing techniques to distribute the problem over many processors.

Due to the long runtime and delays in the finalised versions of the modules it was not possible to produce more computational trials.

4.7.2.6 Extinguishment

The highest extinguishment criterion for the SINTEF extinguishment relation (4.6.1) was used within the model to give the most optimistic time of extinguishment. The fire was extinguished towards the end of the simulation at 256s. The other limits indicated no extinguishment. There was no available experimental data to compare against.

4.8 Concluding remarks

A numerical model was developed to aid in the development and optimisation of water mist based fire suppression and extinguishment systems. The aim was to provide for industry a tool which can be used to help speed up, and reduce the costs

involved in, the design of such systems. The full model includes submodels to simulate the fire, radiation field, water mist, fire suppression, temperature sensors and the misting nozzle activation system.

The comparisons showed good qualitative agreement, the models accurately capturing the trends in the experimental results. Comparisons of predicted and measured temperatures also showed good quantitative agreement. In particular, the measured and predicted temperatures near the compartment ceilings showed excellent agreement. This is an important result as the model is targeted on the development of systems which involve ceiling based fire detection systems. Poorer qualitative agreement was noted for the prediction of CO and smoke. This is partly due to the introduction of considerable uncertainty in the release rates for these products.

The smoke concentration within the compartment has an influence on the predicted temperature field. The sources for the smoke were not well known however and the predictions of the smoke distribution showed poor agreement with the experimental results. It is likely therefore that this uncertainty in the smoke release rates was responsible for a large part of the variation seen between the predicted and measured temperatures, particularly in the lower part of the compartment.

The extinguishment model demonstrated in this thesis was a simple on-off criterion. This was based on the O_2 concentration and temperature of entrained air. When the criterion was deemed to be satisfied the fire model was abruptly shut down. Given all the approximations made in the other FIREDASS submodels, the extinguishment submodel provides a reasonable estimate of the extinguishment time although further research and testing is required. A model has been proposed that would ramp down the fire output on the way to extinguishment. However, this model requires more development and testing and this is left for later work.

The FIREDASS model is a useful tool for optimising fire suppression systems. Given the assumptions of the model, to be considered a practical tool, the FIREDASS model requires accurate release rate and boundary data and a sufficiently powerful computer. If these conditions are met, the model can be used to better target the testing necessary for development and approval of the detection and water mist system.

Apart from the successful use of the model, Industrial users, FSEs and regulators (CAA), raised concerns about the practicality of CFD based fire models with particular reference to FIREDASS. The time factor was obviously important with FSEs requiring answers in economic timeframes. It must be conceded that times of 130+ hours to run some of the simulations (section 4.7.2.5) does reduce the overall usability of the software. These long simulations took advantage of two-fold symmetry, if non-symmetric scenarios were to be simulated the runtime could have been 500+ hours. Indeed even relatively small cases (section 4.4.2.1 - section 4.4.2.3) required runtimes of 30+ hours. These times would reduce over time with advances in computing hardware. However FSEs will always want to run larger and larger problems in ever diminishing runtimes. Typically FSE are only interested in CFD based fire modelling techniques when zone models are inadequate, such as in the case of FIREDASS, and the problems therefore tend to be complex and require long runtimes. Economic methods of reducing the runtimes must be developed if CFD based fire models are to become practical fire engineering tools. A possible technology has been investigated in chapters 7 and 8 using parallel processing techniques on conventional office based PC equipment.

FSEs and regulators have also raised concerns about the confidence of the results produced by CFD codes in particular the lack of independent testing of the models. It should also be noted that there is significant uncertainty associated with the experimental results. In section 4.4.2 it was seen that the model provided reasonable predictions near the ceiling but overpredicted the temperature at lower levels. It can be seen that as the modelling gets more complex the differences between the model and experiment grow. Is this due to some weakness of the model, experimental data or the user? For such complex physical processes the model is an approximation but more research both theoretical and experimental is required to get an improved model. Possible improvements to the model would be more accurate combustion, pyrolysis, and suppression modelling; improved modelling the obstruction effect of the fuel source and improved modelling of boundary conditions. The experimental data was also questionable due to the lack of repeat tests, these issues have been described in section 2.3.1.3.1. In the above work a number of engineering assumptions were made by the author such as the values/functions to associate with combustion efficiency, the

wall emissivity and production rates of chemical species. Another modeller may have made different decisions and therefore obtained different results. An unbiased assessment procedure of CFD based fire modelling software is required to help FSEs gain confidence in the use of fire models and the limitations of the models they are using. Issues of CFD model assessment are addressed in chapters 5 and 6.

5 Fire Modelling Standards/Benchmark

There is a growing trend in fire safety to move away from old prescriptive safety codes towards performance based codes. With this growing trend, FSEs and regulatory authorities need to be confident that CFD based fire modelling codes are capable of producing good quality results. As found in the previous chapter there is clearly a need to establish confidence in CFD methods so that failure of prediction can be attributed to the correct cause whether it is due to limitations of the model or due to other factors such as the user (see section 2.3.2) or errors with experimental results. In order to achieve this, a set of standard/benchmark procedures has been defined [GGP2003a, GGP2003b] that are now described in this chapter. The UK home office Fire Research and Development Group funded this work as they have identified the need for such a procedure for assistance in evaluating CFD based fire models.

5.1 Introduction

The ultimate purpose of the standards/benchmarks proposed here is to aid the fire safety approvals authority e.g. fire brigade, local government authority, etc in assessing the appropriateness of using a particular model for a particular application. Currently there is no objective procedure that assists an approval authority in making such a judgement. The approval authority must simply rely on the reputation of the organisation seeking approval and the reputation of the software being used. In discussing this issue it must be clear that while these efforts are aimed at assisting the approval authorities, there are in fact three groups that are involved, the approvals authority, the general user population and the model developers. Ideally, the proposed standards/benchmark should be of benefit to all three groups. In proposing the standards/benchmark, it is not intended that meeting these requirements should be considered a SUFFICIENT condition in the acceptance process, but rather a NECESSARY condition. It is not a sufficient condition as the model may require specific verification and validation for specific applications. It is a necessary condition as failure to meet this condition indicates it is not suitable for the purposes of fire modelling. For example a CFD based fire modelling code may be able should be able to correctly model the well known Steckler room experiments [SQR1982]. It is a NECESSARY condition that the Steckler room can be modelled as it is unlikely

that it would be able to model any other room based fire configuration if it could not. It is not a SUFFICIENT condition as the Steckler room is a simple test case and other fire scenarios may require more advanced modelling not tested by the Steckler room case. Finally, the benchmarks are aimed at questions associated with the software, not the user of the software.

It is essential to set standards/benchmarks to assess both the Computational Fluid Dynamics (CFD) engine and the fire model component for each type of code. However, within the fire modelling community, testing of fire field models has usually completely ignored the underlying CFD engine and focussed on the fire model. Thus, when numerical fire predictions fail to provide good agreement with the benchmark standard, it is not certain if this is due to some underlying weakness in the basic CFD engine, the fire model or the manner in which the problem was set-up (i.e. questions of user expertise). Furthermore, the case that is being used as the benchmark/standard is usually overly complex or cannot be specified to the precise requirements of the modellers. All of this is often to the benefit of the code developer/user as it allows for a multitude of reasons (some may say excuses) to explain questionable agreement (for example see section 2.3.1.1.2).

Furthermore, what fire modelling testing that is undertaken is usually done in a non-systematic manner, performed by a single individual or group and is generally based around a single model. Thus it is not generally possible for other interested parties to exactly reproduce the presented results (i.e. verify the results) or to apply the same protocol to other models. This makes verification of the results very difficult if not impossible and the comparison of one model with another virtually impossible.

When discussing standards/benchmarks, there are essentially three groups of interested party, the approval authorities, the user groups and the software developer. While maintaining the highest level of safety standards is of general interest to all parties, each interest group has a specific reason for requiring a standard/benchmark. In order to maintain safety standards, the approvals authority must be satisfied that appropriate tools have been employed, the user wants to be assured that he is investing in technology that is suited to the intended task, while the developer would like to have a definable minimum target to achieve.

To satisfy the differing requirements of the approvals authority, user and software developer populations, any suite of benchmarks/standards must be both diagnostic and discriminating. Hence, the proposed suite of benchmarks/standards would ideally exercise each of the components of the fire field model i.e. CFD engine and fire model. This means that standards based simply around instrumented room fire tests are insufficient. This would for example require benchmarks/standards for simple recirculating flows, buoyant flows, turbulent flows, radiative flows, etc. Furthermore, in addition to the quality of the numerical results, details of the computer and compiler used to perform the simulations and the associated CPU time expended in performing the calculations could be provided. While not of particular interest to the approval authorities, this will be of interest to the user community.

Ideally, the proposed benchmarks/standards will evolve into a measure of quality, indicating that the fire model has reached a minimum standard of performance. This does not necessarily mean that the software may be used for any fire application (i.e. it is NOT a SUFFICIENT condition); however it would eliminate from consideration those software products that have not demonstrated that they can attain the standard (i.e. it is a NECESSARY condition).

5.2 THE SOFTWARE PRODUCTS (SP)

To demonstrate the concept of the above procedure a variety of software products needed to be tested to test the applicability to general CFD and fire modelling CFD software products. The concept had to demonstrated with a sufficient number of SPs but also had to be manageable within the time constraint.

Several developers of well known fire field models currently used in the UK were approached to participate in this exercise namely, the developers of JASMINE [ref needed], SOFIE [Rub1997], CFX [CFX1997], PHOENICS [RST1983] and SMARTFIRE [EGP+1999, GKP+1999]. Three code developers agreed to participate in the first phase of the exercise. They were:

The general purpose CFD codes,

CFX 4.2 [CFX] and
PHOENICS 3.1 [RST1983]

and the specific fire field model,
SMARTFIRE v2.01 b389D[GKP+1999].

These versions of the code used were the latest available versions of the code at the University of Greenwich at the commencement of the work.

5.3 BENCHMARK PROCEDURES

The benchmarks are divided into two categories, basic CFD and fire. Two types of simulation were performed by each SP being subjected to the benchmarks; these are known as phase 1 and phase 2 simulations. The nature of the phase 1 simulation has been rigidly defined and includes the mesh specification, physics to be activated, algorithms to be employed and results to be generated (see Appendix B and C). Also the same constants will be used by all the codes, when the turbulence model and combustion model are used instead of their default settings. Where possible, the specification of phase 1 simulations has been such that all of the SP participating in the trial will be able to achieve the specification. By using this very rigid specification it was hoped that user bias would be reduced, and hopefully eliminated, from this phase of testing. It is acknowledged that this process does not necessarily produce optimal results for all of the SPs.

The phase 1 simulations will be completed before proceeding to attempt the phase 2 simulations. The phase 2 simulations will be free format in nature, allowing the participants to repeat the simulation using whatever specification they desire. Phase 2 simulations were intended to allow the participants to demonstrate the full capabilities of their SP. However, phase 2 simulations will only be allowed to utilise features that are available within their software product i.e. additional code or external routines are generally not permitted.

Each phase 1 simulation was performed at least once. The participants were also requested to run at least two of the 10 phase 1 simulations using their SP. Participants

were free to choose which two simulations to run, however they must include at least one from the CFD category and one from the fire category. Participants were of course free to (and were indeed encouraged to) run all 10 of the phase 1 simulations. It was however imperative that the participants do not inform FSEG which of the phase 1 simulation they intend to run. The purpose of repeating the simulations is to ensure that the author/FSEG had not fabricated results or incorrectly run the code.

On completing the phase 1 simulations participants were invited to undertake their phase 2 simulations. All participants have to complete a similar pro-forma as those supplied for the phase 1 simulations. This was necessary, as the author/FSEG would have repeated the phase 2 simulations in order to independently verify the results.

5.3.1 THE BENCHMARK CASES

A variety of test cases were required, these needed to be relatively easy to set up given the amount of time required to specify the problem for each software product. This eliminated the use of BFC type problems due to the difficulty in specifying them in each of the different SPs. As a first attempt at defining the benchmarks, 10 cases were considered, these involved five CFD cases and five fire cases. All of the phase 1 fire type simulations were defined with relatively coarse meshes in order to keep computation times to reasonable levels. Participants were free to refine meshes when undertaking the phase 2 simulations.

The cases are defined as follows:

5.3.1.1 CFD Cases:

2000/1/1 Two dimensional turbulent flow over a backward facing step.
2000/1/2 Turbulent flow along a long duct.
2000/1/3 Symmetry boundary condition.
2000/1/4 Turbulent buoyancy flow in a cavity.
2000/1/5 Radiation in a three-dimensional cavity.

5.3.1.2 Fire Cases:

2000/2/1 Steckler Room (heat source).

2000/2/2 Steckler Room (combustion model).

2000/2/3 Fire in a completely open compartment with lid (heat source).

2000/2/4 CIB W14 fire (combustion model).

2000/2/5 Large fire (combustion model)

Full details concerning the specification of the phase 1 simulations may be found in Appendix B and C.

5.4 CFD cases

5.4.1 2000-1-1 – Backward Facing Step

This test is a standard CFD test case used by a number of CFD code developers. Its primary purpose was to test the turbulence model used by the CFD code. Comparative values have been taken at 0.285m downstream of the inlet and at the outlet. Predictions of the location of the stagnation point were compared with experimental data [KKJ1980].

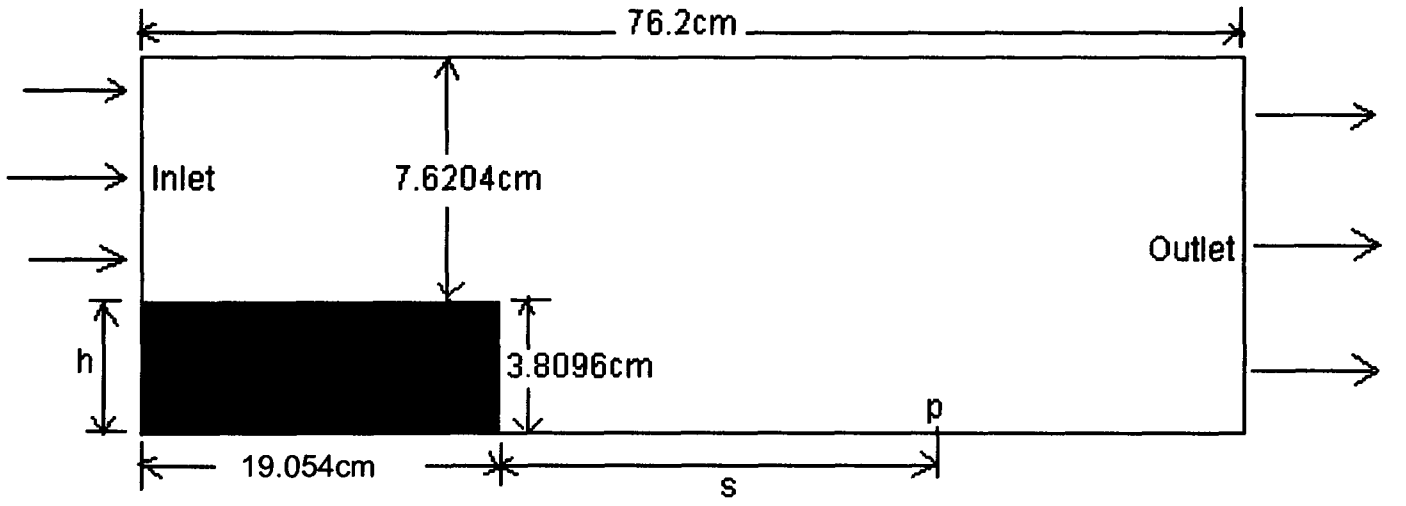


Figure 5-1 - Backward facing step configuration

The flow is incompressible, fully turbulent and isothermal. The fluid has a density of 1.0 kg/m^3 and a laminar viscosity of 1.101E-5 kg/ms. The geometry of the case is illustrated in Figure 5-1.

The upper and lower surfaces are walls and there is a solid obstruction below the inlet. The fluid enters the chamber at 13.0 m/s.

See Appendix B.1 for further setup details.

5.4.2 2000-1-2 Heat transfer in a long thin duct

This test is a standard CFD test case used by a number of CFD code developers. Its primary purpose is to test the turbulence model in conjunction with turbulent heat transfer. Predictions of the velocity and enthalpy profile at the outlet are cross compared between the codes.

The geometry of the case is depicted in Figure 5-2. The flow is non-buoyant, fully turbulent, incompressible with heat transfer but with no radiation. Flow enters the inlet at 50m/s with an enthalpy of 50 J/Kg. The wall has a fixed enthalpy value of 1 J/Kg. The fluid density is 1.0 kg/m^3, the conductivity is 0.07179 W/mK, the density is 1.0 kg/m^3, laminar viscosity is 5e-5 kg/ms, specific heat is 1005 J/kgK

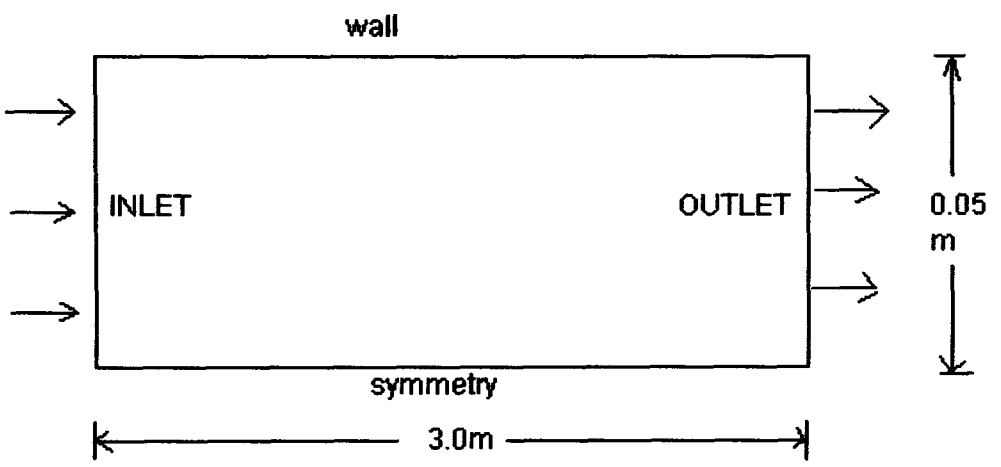


Figure 5-2 - Turbulent long duct flow configuration

See Appendix B.2 for further setup details.

5.4.3 2000-1-3 Symmetry

This test is a relatively simple CFD test case. Its primary purpose is to test if the symmetry function works correctly for turbulent isothermal flow situations. Model predictions for the symmetric case are compared with and without the symmetry function in operation. The predictions from the SPs are also cross compared.

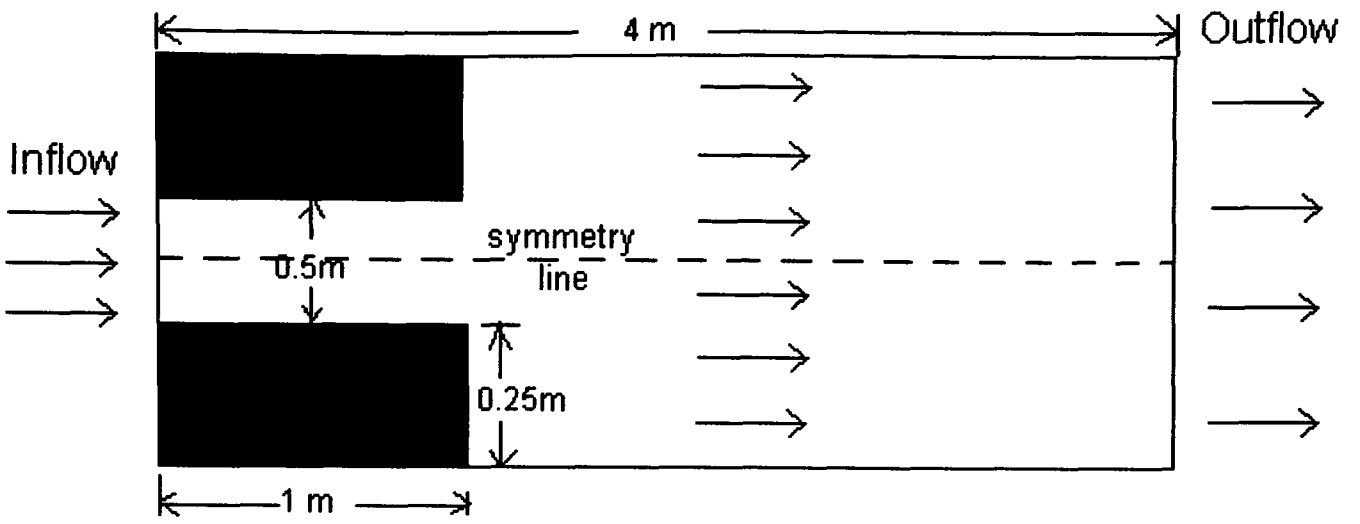


Figure 5-3 - Expanding duct with symmetry line indicated

The case involves flow expansion from a small duct into a larger duct. The configuration is shown in Figure 5-3 above. The case was simulated using the whole flow domain and then repeated using a symmetry boundary condition along the central axis. Two tests must be conducted one using the full domain and the other using a half domain with a symmetry plane. The results from these two tests should agree with one another. The flow enters the domain at 1.0m/s.

See Appendix B.3 for further setup details.

5.4.4 2000-1-4 Buoyant turbulent flow

This test is a standard CFD test case used by a number of CFD code developers. Its primary purpose is to test the turbulence model, turbulent heat transfer and buoyancy model. Predictions of a number of parameters were made and cross compared. Model predictions were also compared with experimental results [CKZ1986].

The geometry used for this case is depicted in Figure 5-4 below.

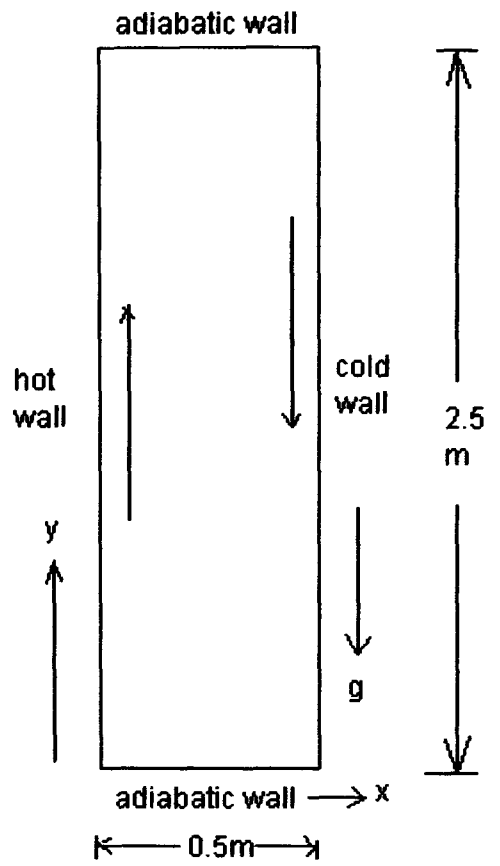


Figure 5-4 – Configuration for buoyancy flow in a duct

The flow is fully turbulent, buoyant and fully compressible but with no radiative heat transfer. The hot wall is at a temperature of 353K and the cold wall is at 307.2K. The other walls are adiabatic. The acceleration due to gravity (g) is $-9.81 m/s^2$. The fluid has the following properties:

conductivity is 2.852158e-02 (W/mK)
density is 1.071 (kg/m^3) determined by ideal gas law as fully compressible.
specific heat is 1.008e+03 (J/kgK)
laminar viscosity is 2.0383e-05 (kg/ms)
thermal expansion is 3.029385e-03 (1/K).

See Appendix B.4 for further setup details.

5.4.5 2000-1-5 Radiation in a 3D cavity.

The primary purpose of this test case was to test the radiation model used by the SPs. Model predictions are cross compared and also compared with theoretical predictions derived from detailed zone methods [Lar1983, Fiv1988].

The geometry used for this test case consists of a three dimensional unit cube (1m x 1m x 1m) cavity with three walls with planes x=1, y=0 and z =0 set to a unit emissive power and the three other walls set to zero emissive power. All the walls are considered radiatively black have unit emissivity and the fluid has a unit absorption coefficient. Scattering is neglected. No fluid flow is considered

See Appendix B.5 for further setup details.

5.5 Fire cases

In the phase-1 testing regime all the codes are used in a simplistic manner with all the walls assumed to be adiabatic and perfectly reflecting of radiation. Furthermore, all the model are made to behave as similarly as possible but using the same parameters for combustion models, radiation models and turbulence models.

5.5.1 2000-2-1 & 2000-2-2 – Steckler fire case

The two fire cases are exactly the same apart from the second case uses a combustion model instead of a volumetric heat source. This test is a standard fire model test case used by a number of field and zone model developers. Its primary purpose is to test the fire models predictive capability in predicting temperature and flow distributions in a small compartment subjected to a steady non-spreading fire. Predictions of several parameters were made and cross compared. Model predictions were also compared with experimental results [SQR1982].

The non-spreading fire was created using a centrally located (position A in Figure 5-5) 62.9kW methane burner with a diameter of 0.3m. The experiments were conducted by Steckler et al. in a compartment measuring 2.8m × 2.8m in plane and 2.18m in height (see Figure 5-5) with a doorway centrally located in one of the walls measuring 0.74m wide by 1.83m high. The walls and ceiling were 0.1m thick and they were covered with a ceramic fibre insulation board to establish near steady state conditions within 30 minutes.

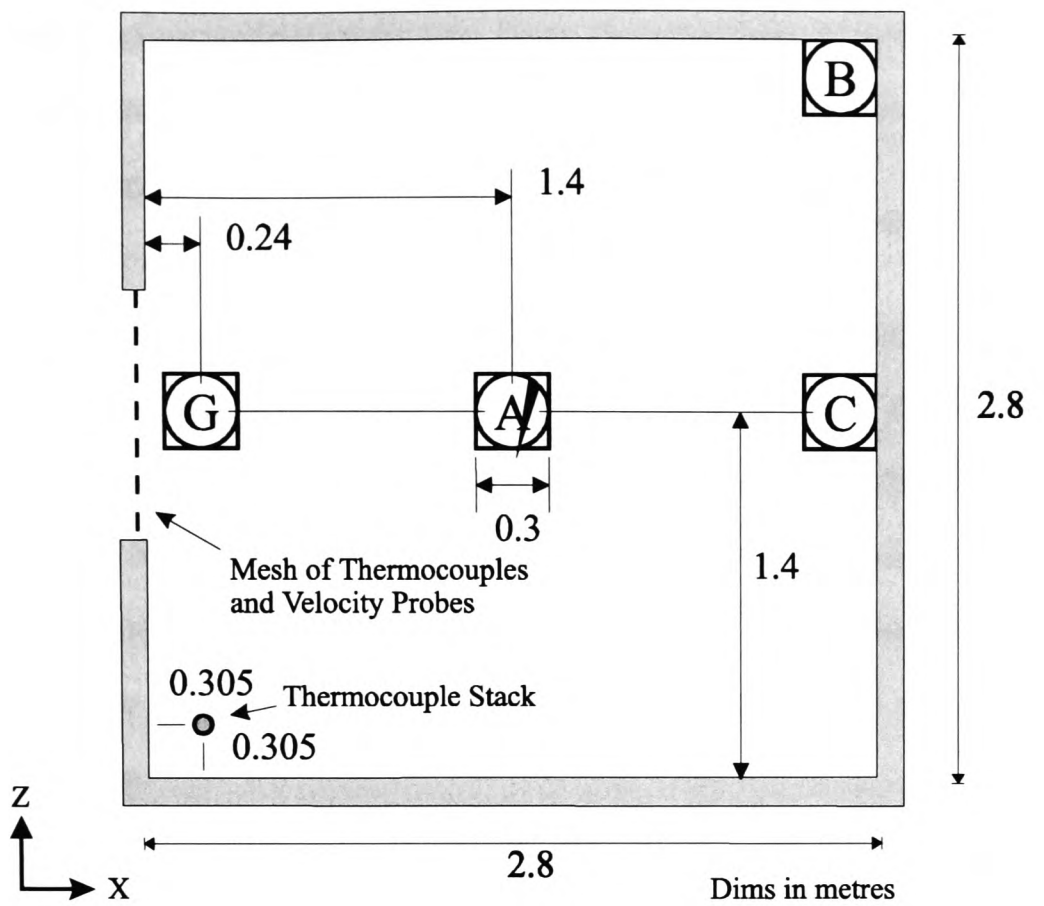


Figure 5-5 – Configuration of Steckler room

The door measures 0.74m wide and 1.83m high and is centrally located in one of the walls. Within the models, the walls are all assumed to be adiabatic and perfect radiative reflectors. The case is run for 200s of simulated time using 200 time steps of 1s at which point steady state conditions are achieved in the simulation.

This case has been modelled using 2 methods: -

1) Using a simple volumetric heat source (2000-2-1)
2) Using a combustion model (2000-2-2)

In PHOENICS and SMARTFIRE a six-flux radiation model is used, while in CFX the discrete transfer model is used with a single ray in the co-ordinate direction to emulate the behaviour of a six-flux radiation model.

See Appendix C.1 and C.2 for further setup details.

5.5.2 2000-2-3 – Open Fire with Lid case

This test is an artificial fire test case. There are no experimental results for comparison purposes. Its primary purpose is to test the fire models predictive

capability in predicting temperature and flow distributions in a small well ventilated compartment subjected to a non-spreading fire. Predictions of several parameters are made and cross compared.

This fire case utilises a volumetric heat source. The compartment is completely open apart from a solid ceiling (see Figure 5-6). The fire is located on the floor at the centre of the building. The prescribed fire volume is 1m x 1m x 1m. The fire power is defined as $H = 0.188t^2(kW)$ (i.e. t squared fire and t is measured in seconds). The compartment is 5m(wide) × 5m(long) × 3m(high). The ceiling is adiabatic. The ambient temperature is 303.75K. The case was run for 110s of simulated time using 110 time steps of 1s.

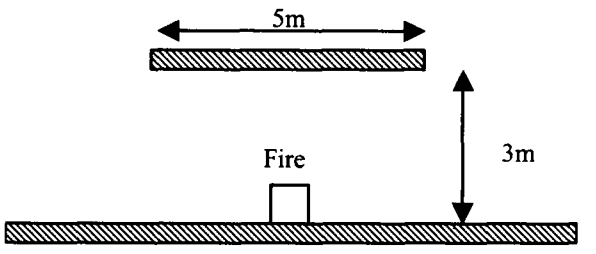


Figure 5-6 - Configuration of open fire with lid

See Appendix C.3 for further setup details.

5.5.3 2000-2-4 – CIB W14 case

This case arises from the CIB round robin tests of which subscenario B1 is the case of interest [HK1998]. The fire compartment measured 14.4 m × 7.2 m in plan and 3.53 m in height and contained a doorway of dimensions 2.97 m × 2.13 m. The walls of the compartment were made of aerated concrete blocks (with siporex mortar) with thickness 0.3 m and the following material properties: specific heat 1.05 kJ/kg.K, thermal conductivity 0.12 W/m.K and density 500 kg/m^3. The initial air temperature was measured as 20.0 °C.

The fire was located on the floor in the centre of the room. The fire fuel consisted of softwood (Pinea ecelsa) timber cribs nailed into 40mm x 40mm battens. The crib measured 2.4m in length, 2.4 m in width and 1.4 m in height.

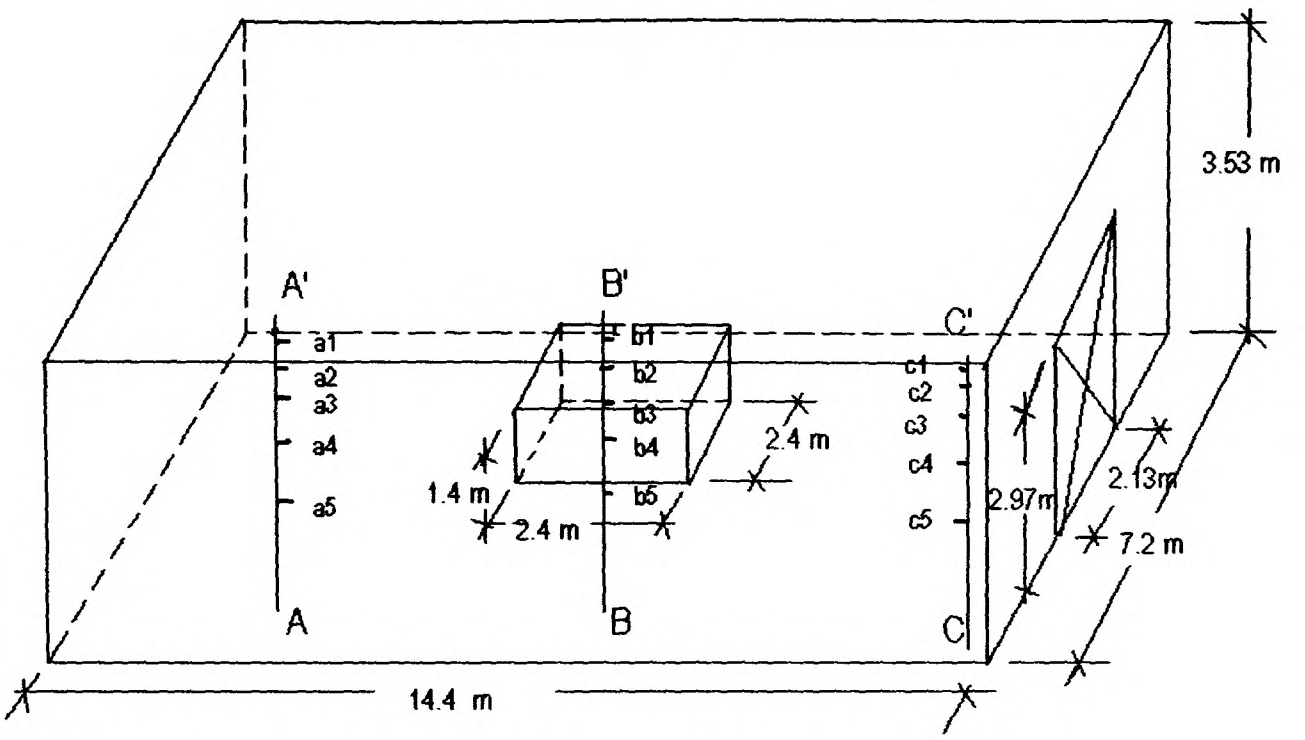


Figure 5-7 – Depiction of fire compartment geometry showing location of fire source.

The heat release rate ($\dot{Q}$) is given by the following calculation (5.5.3.1): -

$$\dot{Q} = \chi \cdot \Delta H_c \cdot \dot{m} \qquad (5.5.3.1)$$

The efficiency factor (χ) and heat of combustion (ΔH_c) were given as χ =0.7 and ΔH_c is 17.8 MJ/kg for burning wood with a 10% moisture content and the mass loss rate ($\dot{m}$) (kg/s) for the wood crib is presented in the table below. A maximum heat release rate of approximately 11 MW was produced. It is assumed that the fuel molecule is $CH_{1.7}O_{0.83}$.

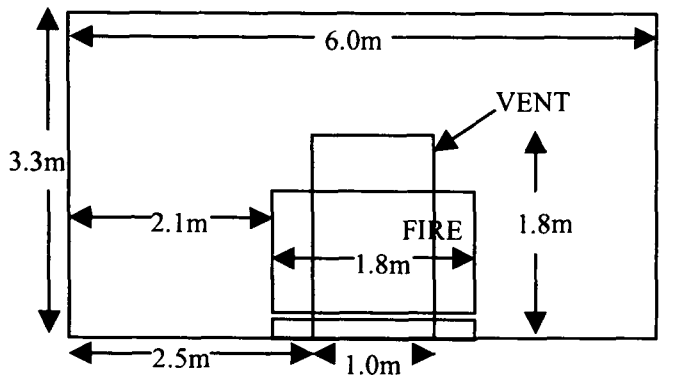
Time (s)	0	60	120	180	240	300	360	420	480	540	600
Mass loss rate(kg/s)	0	0.005	0.004	0.009	0.013	0.014	0.019	0.033	0.052	0.08	0.207

The case was run assuming that all the walls were adiabatic and were completely reflecting (emissivity = 0.0). The case was run for 600s of simulated time using 120 time steps of 5s.

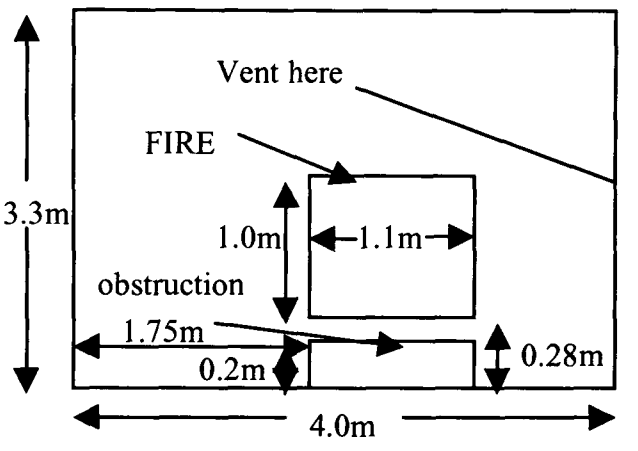
See Appendix C.4 for further setup details

5.5.4 2000-2-5 – LPC007 case

This test case arises from a fire test conducted by the Loss Prevention Council (LPC) [GAC1997]. The test is a burning wood crib within an enclosure with a single opening. The test compartment is illustrated below and had a floor area of 6m x 4m and a 3.3m high ceiling. The compartment contained a doorway (vent) measuring 1.0m x 1.8m located on the rear 6m x 3.3m wall. The walls and ceiling of the compartment were made of fire resistant board (Asbestos) which were 0.1m thick. The floor was made of concrete.



y-z view



x-y view

The heat release rate ($\dot{Q}$) is given by (5.5.3.1). The efficiency factor (χ) and heat of combustion (ΔH_c) were given as χ=0.7 and ΔH_c is 17.8 MJ/kg for burning wood with a 10% moisture content and the mass loss rate ($\dot{m}$) (kg/s) for the wood crib is presented in Table 5-1. It is assumed that the fuel molecule is $CH_{1.7}O_{0.83}$.

Table 5-1: Mass Loss rate for LPC fire test case.

Time(s)	0	150	450	460	1650
$\dot{m}$ (kg/s)	0	0.01835	0.18636	0.1978	0.1978

See Appendix C.5 for further setup details

6 Fire Modelling Standards/Benchmark results

This chapter details the results and outcomes from applying the phase-1 regimes to the PHOENICS, CFX and SMARTFIRE SPs and the phase-2 testing regime to the SMARTFIRE SP.

The CFD and fire cases were designed to test the basic features of the SPs to ensure that these functioned correctly. In Phase-1, the testing was designed to ensure that the codes were set up as similarly as possible. This included using the same computational mesh in all cases and the physics switched on in all cases consisted of the lowest common denominator between the SPs. While this was the aim of this part of the testing process, some differences existed between the various SPs. The most obvious differences between the SPs was that PHOENICS used a staggered velocity mesh whereas SMARTFIRE and CFX used a co-located velocity mesh by means of Rhie and Chow interpolation [RC1983] and while SMARTFIRE and PHOENICS made use of a six-flux radiation model, CFX used a more sophisticated discrete transfer model.

Details of the numerical set-ups for the phase 1 CFD and fire cases can be found in Appendix B and C. The phase 2 set-ups are described in Appendix D.

6.1 Phase 1 testing regime

6.1.1 CFD cases

In this section the results generated for the CFD cases are presented. In the first four cases radiation was either relevant to the situation or made no significant contribution to the simulation and so was not modelled.

6.1.1.1 2000-1-1 – Backward Facing Step

This test is a standard CFD test case used by a number of CFD code developers. Its primary purpose was to test the turbulence model used by the CFD code. Results from the SPs were cross compared and predictions from the SPs were compared with experimental data. Comparative values have been taken at 0.285m downstream of the

inlet and at the outlet. Predictions of the location of the stagnation point were compared with experimental data [KKJ1980].

The flow is incompressible, fully turbulent and isothermal. The fluid has a density of 1.0 kg/m^3 and a laminar viscosity of 1.101E-5 kg/ms. The geometry of the case is illustrated in Figure 6-1.

The upper and lower surfaces are walls and there is a solid obstruction below the inlet. The fluid enters the chamber at 13.0 m/s.

See Appendix B.1 for further setup details.

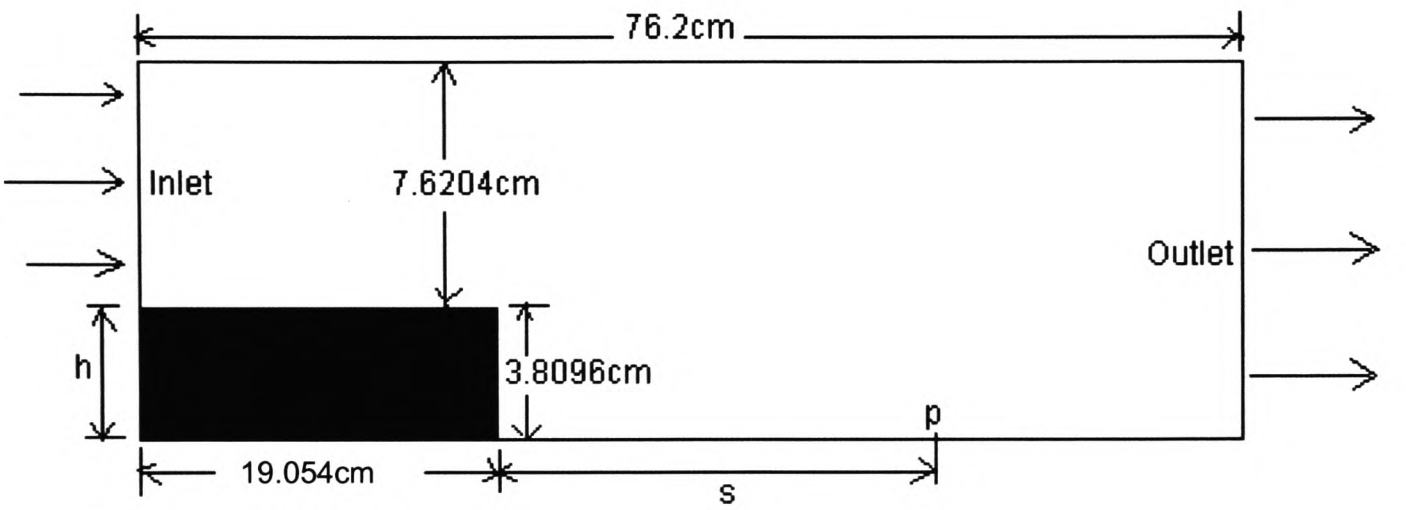


Figure 6-1 - Backward facing step configuration

Results

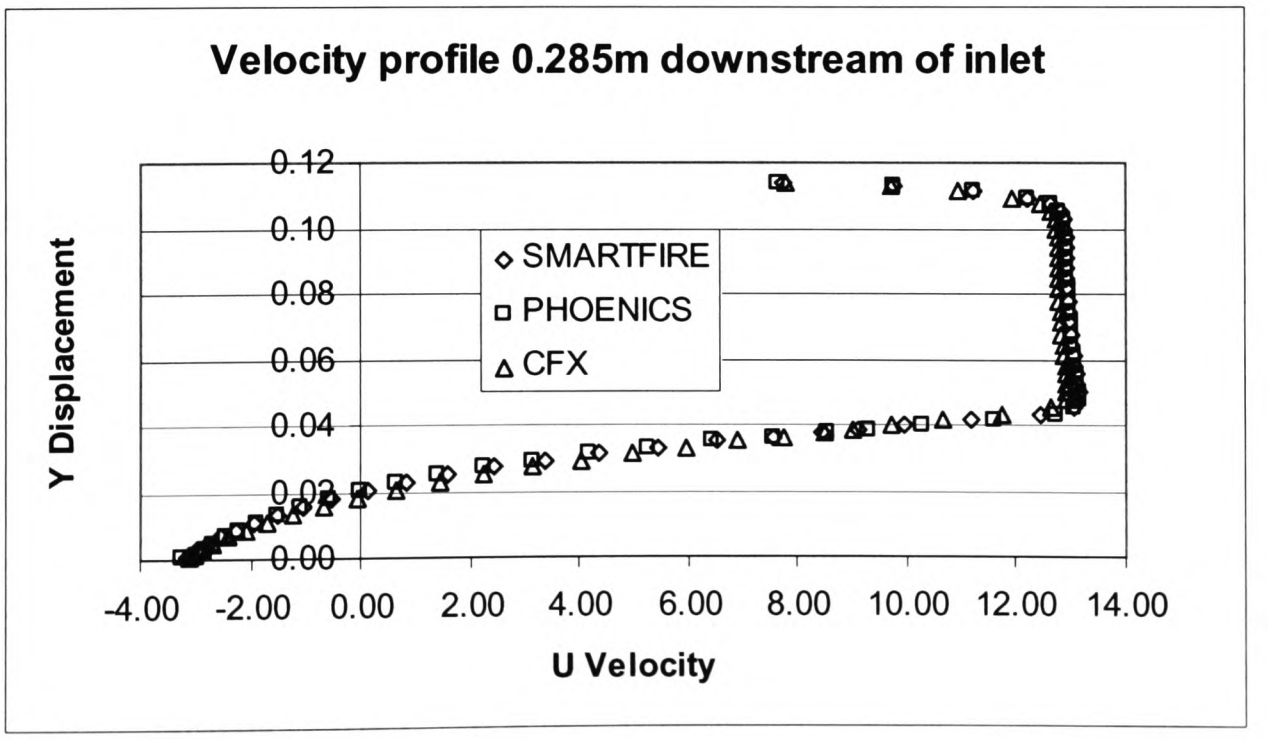


Figure 6-2 - Velocity profile 0.285m downstream of inlet for 2000-1-1

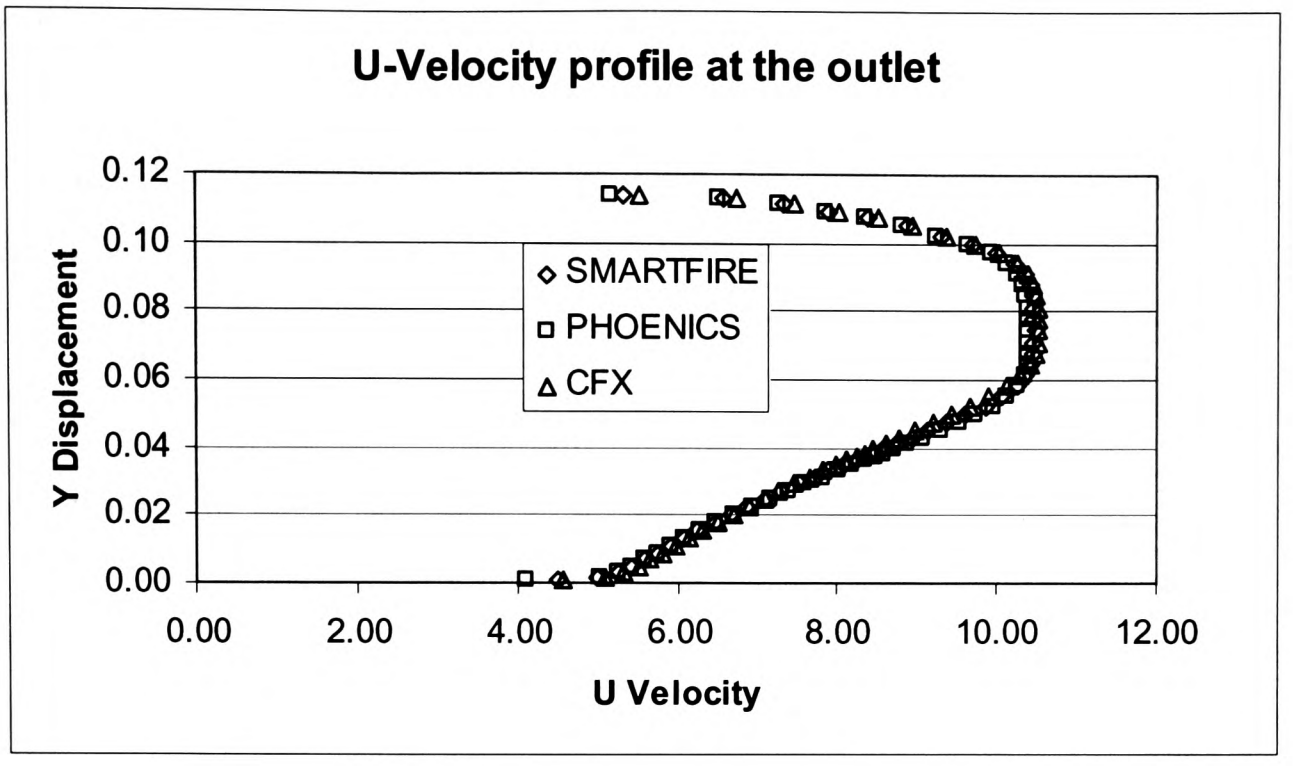


Figure 6-3 - Velocity profile at the outlet for 2000-1-1

In Figure 6-2 and Figure 6-3 it can be seen that there is extremely good agreement between the three SPs.

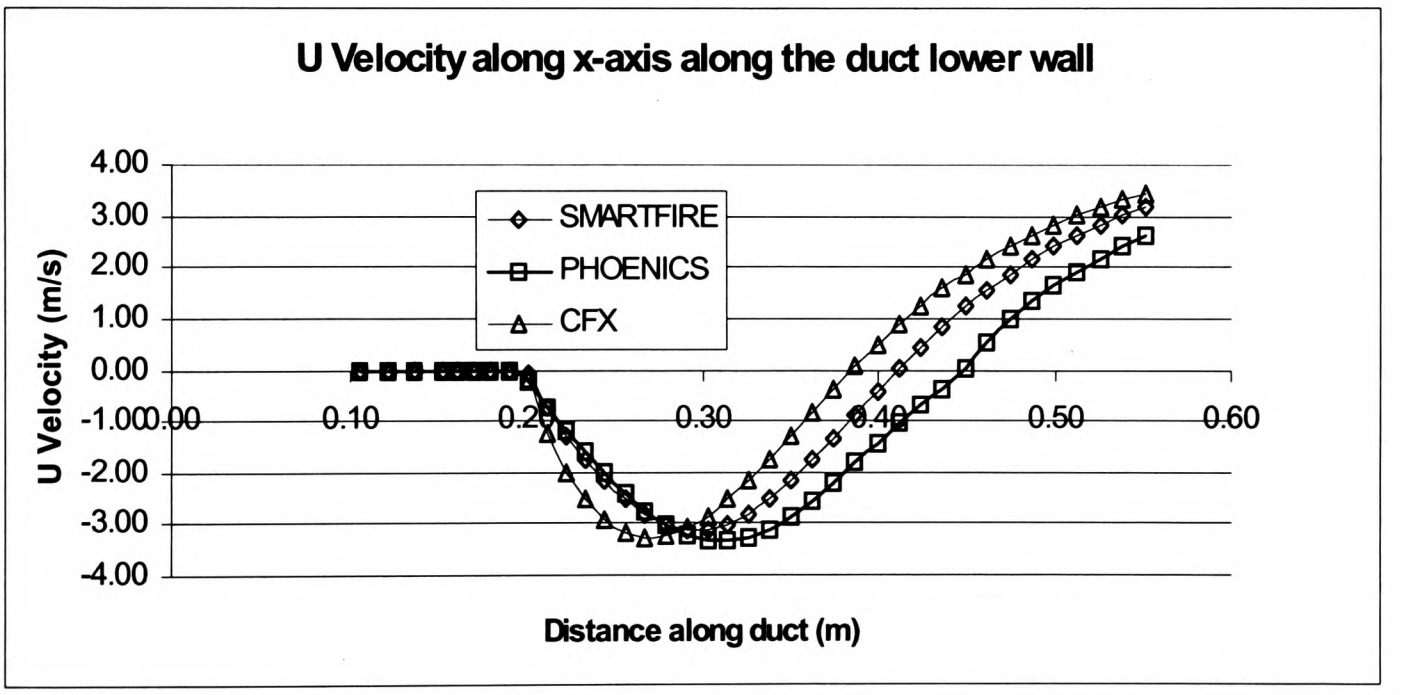


Figure 6-4 - U Velocity along the duct lower wall

Table 6-1 - Comparison of stagnation point for the CFD codes

Stagnation point	x	S	S/h (where h=.0381)
SMARTFIRE	0.412	0.2217	**5.82**
PHOENICS	0.449	0.2587	**6.79**
CFX	0.387	0.1967	**5.16**

The stagnation point is the point where the recirculation due to the step ends along the lower duct wall (point p in Figure 5-1). There was some variation in the predicted stagnation point ratio (S/h) (see Figure 6-4 and Table 6-1), the experimental value of 7.2 was most closely matched by PHOENICS followed by SMARTFIRE with CFX being the furthest away from the experimental value. However these values were obtained using each code's standard k-e turbulence model and improved results may be expected with enhanced turbulence models.

6.1.1.2 2000-1-2 Heat transfer in a long thin duct

This test is a standard CFD test case used by a number of CFD code developers. Its primary purpose was to test the turbulence model in conjunction with turbulent heat transfer. Predictions of the velocity and enthalpy profile at the outlet were cross compared.

The geometry of the case is depicted in Figure 6-5. The flow is non-buoyant, fully turbulent, incompressible with heat transfer but with no radiation. Flow enters the inlet at 50m/s with an enthalpy of 50 J/Kg. The wall has a fixed enthalpy value of 1 J/Kg. The fluid density is 1.0 kg/m^3, the conductivity is 0.07179 W/mK, the density is 1.0 kg/m^3, laminar viscosity is 5e-5 kg/ms, specific heat is 1005 J/kgK

See Appendix B.2 for further setup details.

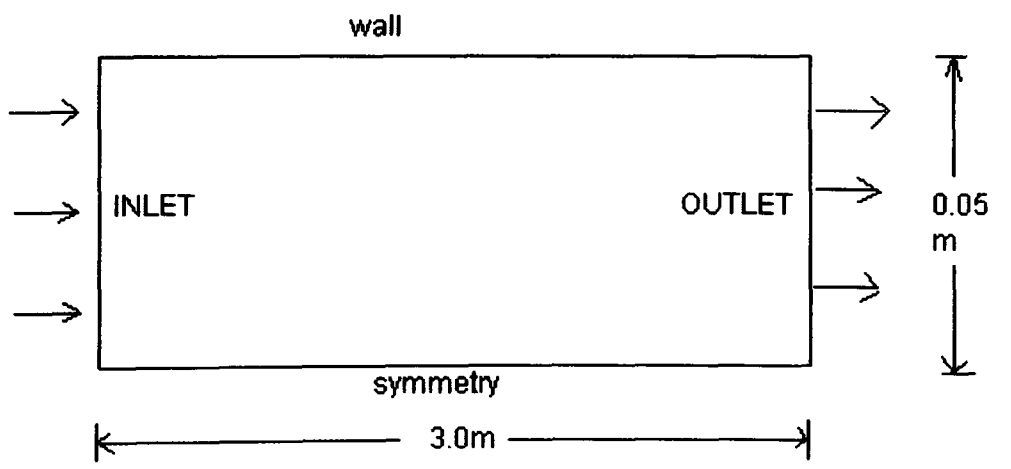


Figure 6-5 - Turbulent long duct flow configuration

Results

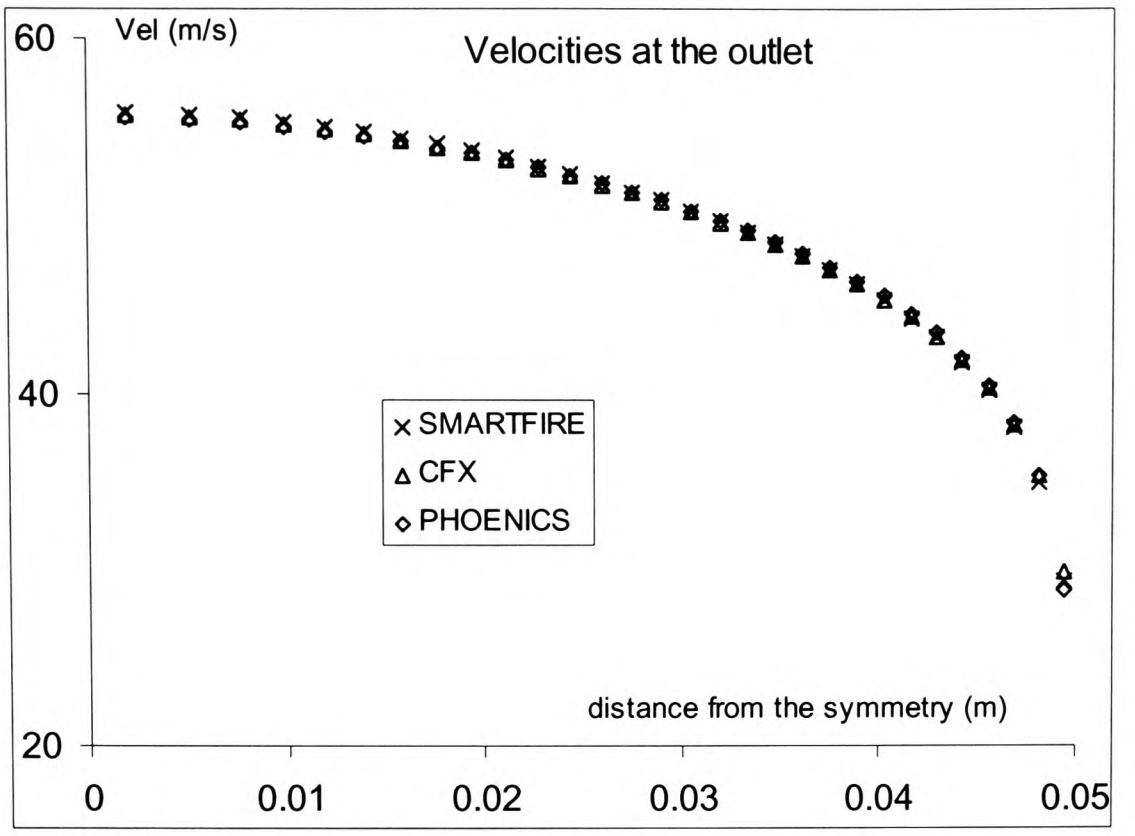


Figure 6-6 - Velocity profile at outlet for 2000-1-2

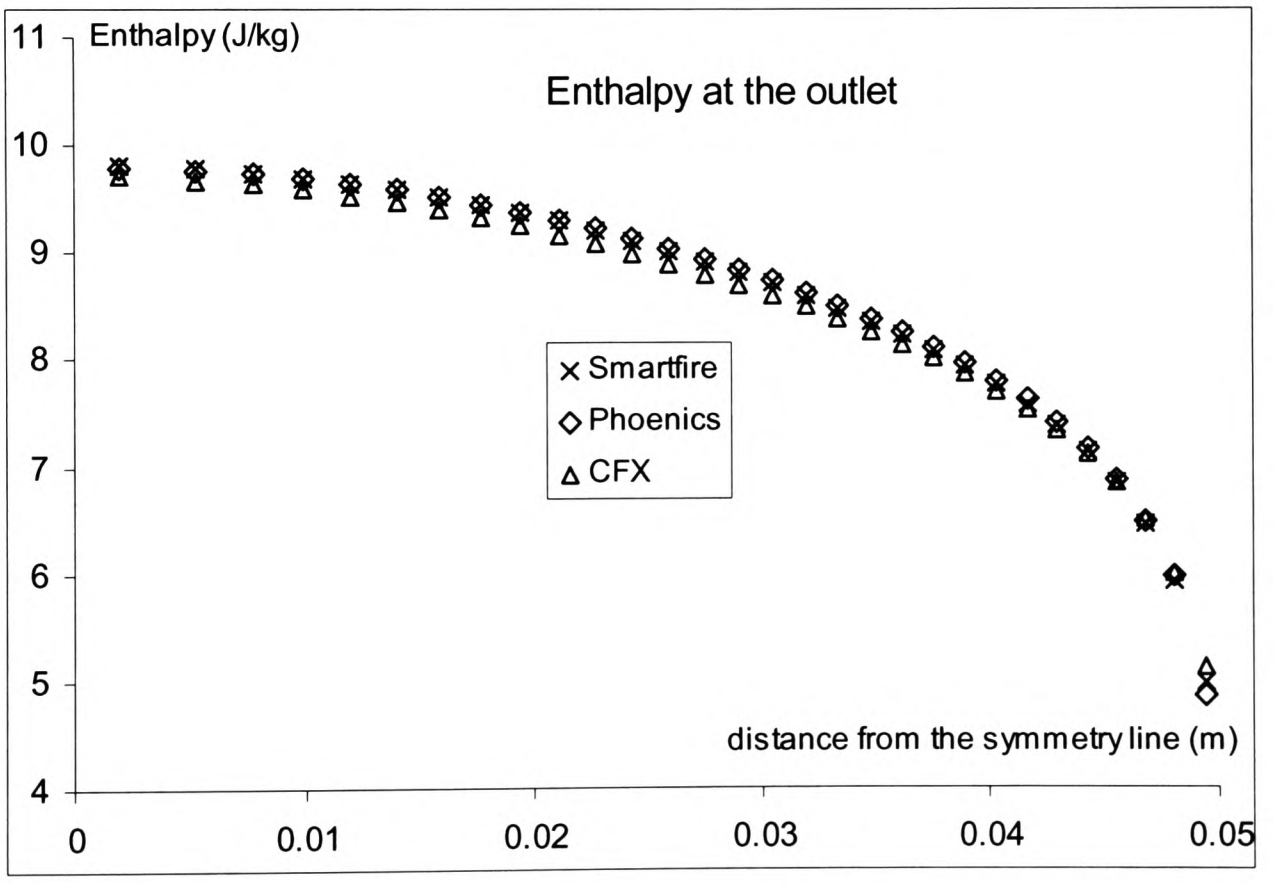


Figure 6-7 - Enthalpy profile at the outlet for 2000-1-2

Depicted in Figure 6-6 is the velocity profile generated by the three SPs at the outlet, while depicted in Figure 6-7 is the enthalpy profile at the outlet. As can be seen from these figures, there was extremely good agreement across the SPs.

6.1.1.3 2000-1-3 Symmetry

This test is a relatively simple CFD test case. Its primary purpose is to test if the symmetry function works correctly for turbulent isothermal flow situations. Model predictions for the symmetric case were compared with and without the symmetry function in operation. The predictions from the SPs were also cross compared.

The case involves flow expansion from a small duct into a larger duct. The configuration is shown in Figure 6-8 below. The case was simulated using the whole flow domain and then repeated using a symmetry boundary condition along the central axis.

See Appendix B.3 for further setup details.

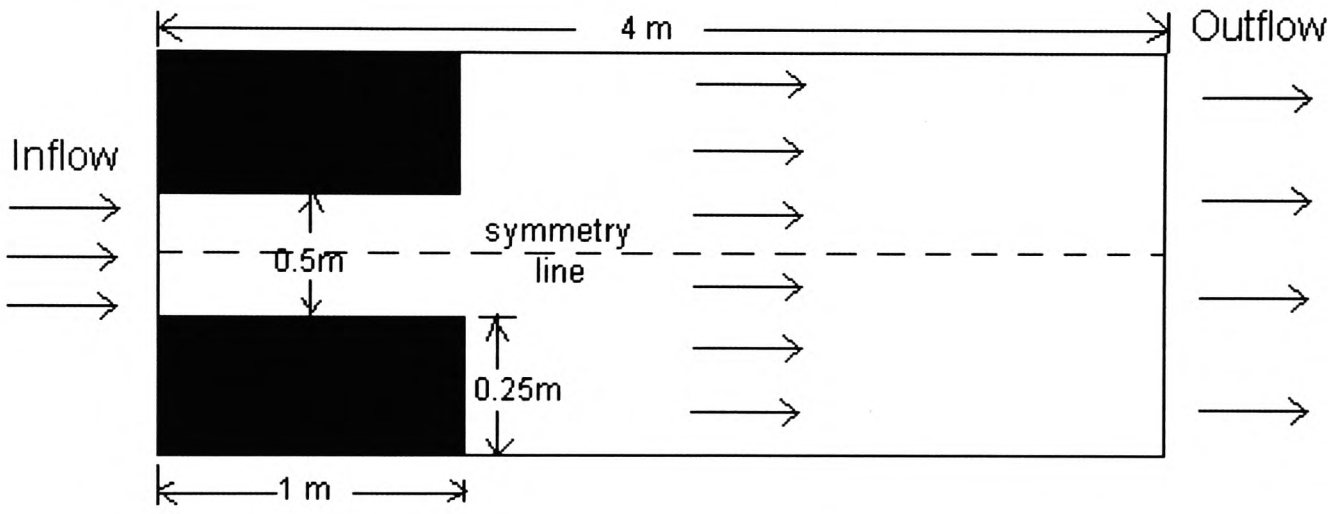


Figure 6-8 - Expanding duct with symmetry line indicated

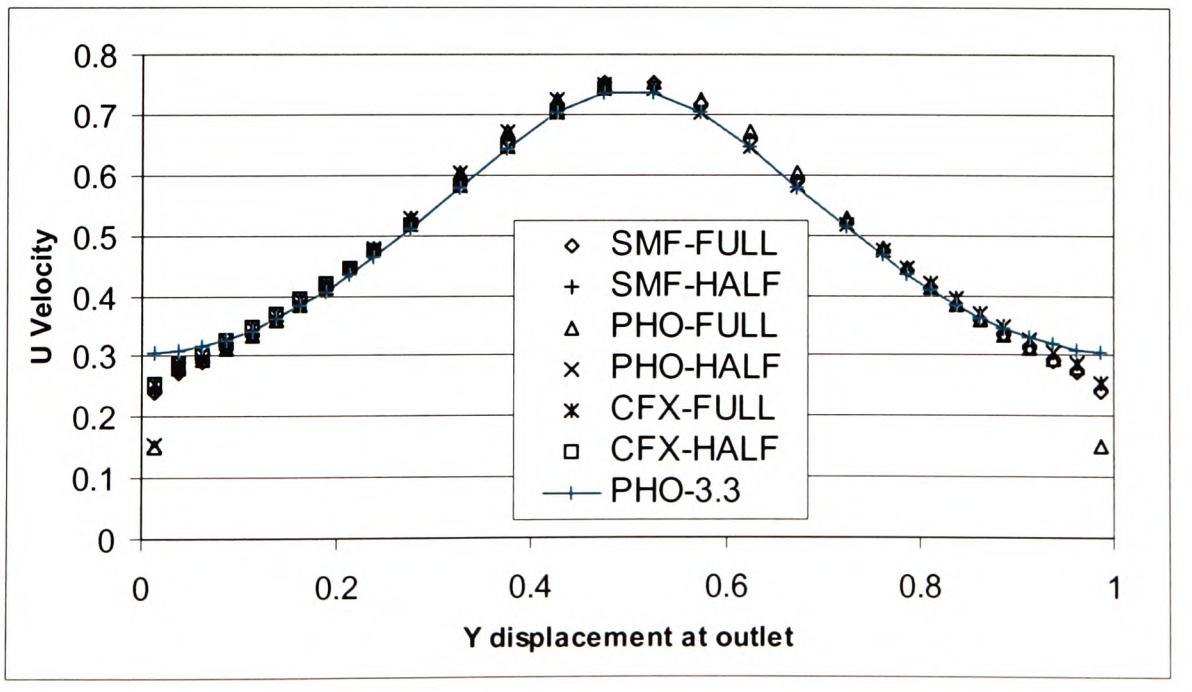


Figure 6-9 - U Velocity profile at the outlet for 2000-1-3

All the U-velocity profiles for all the codes using both the full and half geometry are encapsulated in Figure 6-9. Within the SPs the same results for the U velocity at the outlet have been produced for the symmetry (half) and full geometry versions of the test case. Across the codes the results were also close to one another although the PHOENICS generated nearest wall velocity is significantly different to that of SMARTFIRE and CFX. CHAM – the developers of PHOENICS – repeated the above test case using a more recent version of PHOENICS, i.e. PHOENICS V3.3 and found that the velocity at the wall was increased compared to the result generated using V3.1. The result is now slightly faster than that produced by CFX and SMARTFIRE but is more inline with the general trends.

6.1.1.4 2000-1-4 Buoyant turbulent flow

This test is a standard CFD test case used by a number of CFD code developers. Its primary purpose was to test the turbulence model, turbulent heat transfer and buoyancy model. Predictions of a number of parameters were made and cross compared. Model predictions were also compared with experimental results [CKZ1986].

The geometry used for this case is depicted in Figure 6-10 below.

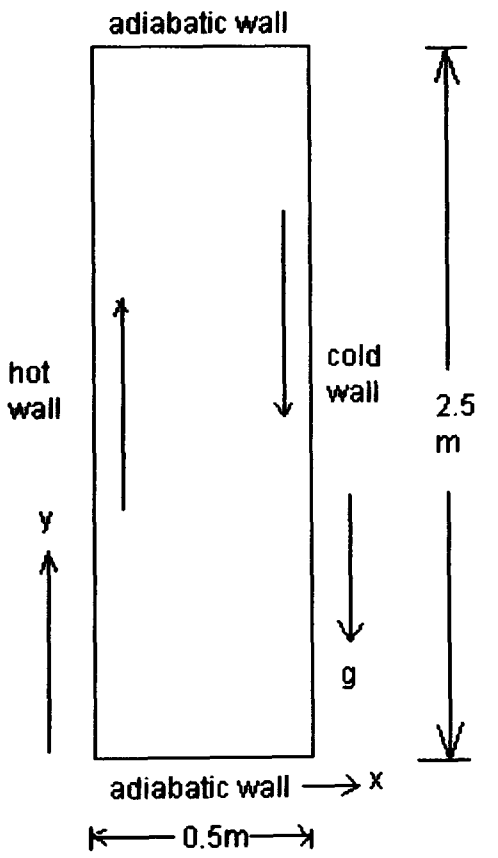


Figure 6-10 – Configuration for buoyancy flow in a duct

The flow is fully turbulent, buoyant and fully compressible but with no radiative heat transfer. The hot wall is at a temperature of 353K and the cold wall is at 307.2K. The other walls are adiabatic. The acceleration due to gravity (g) is $-9.81 m/s^2$. The fluid has the following properties:

conductivity is 2.852158e-02 (W/mK)
density is 1.071 (kg/m^3) determined by ideal gas law as fully compressible.
specific heat is 1.008e+03 (J/kgK)
laminar viscosity is 2.0383e-05 (kg/ms)
thermal expansion is 3.029385e-03 (1/K).

See Appendix B.4 for further setup details.

Model predictions are presented for the following:

The v-velocity profile at y/H = 0.5
The normalised temperature profile at y/H = 0.5 and x/L = 0.5
where $T_{normalised} = (T_{actual} - T_{cold})/(T_{hot} - T_{cold})$
The turbulent fluctuations, $\sqrt{k}$, at y/H = 0.5
The turbulent viscosity scaled with the laminar viscosity at y/H = 0.5.

In the above, L is full length across the x direction of the duct (0.5m) and H is the full height of the duct in the y direction (2.5m).

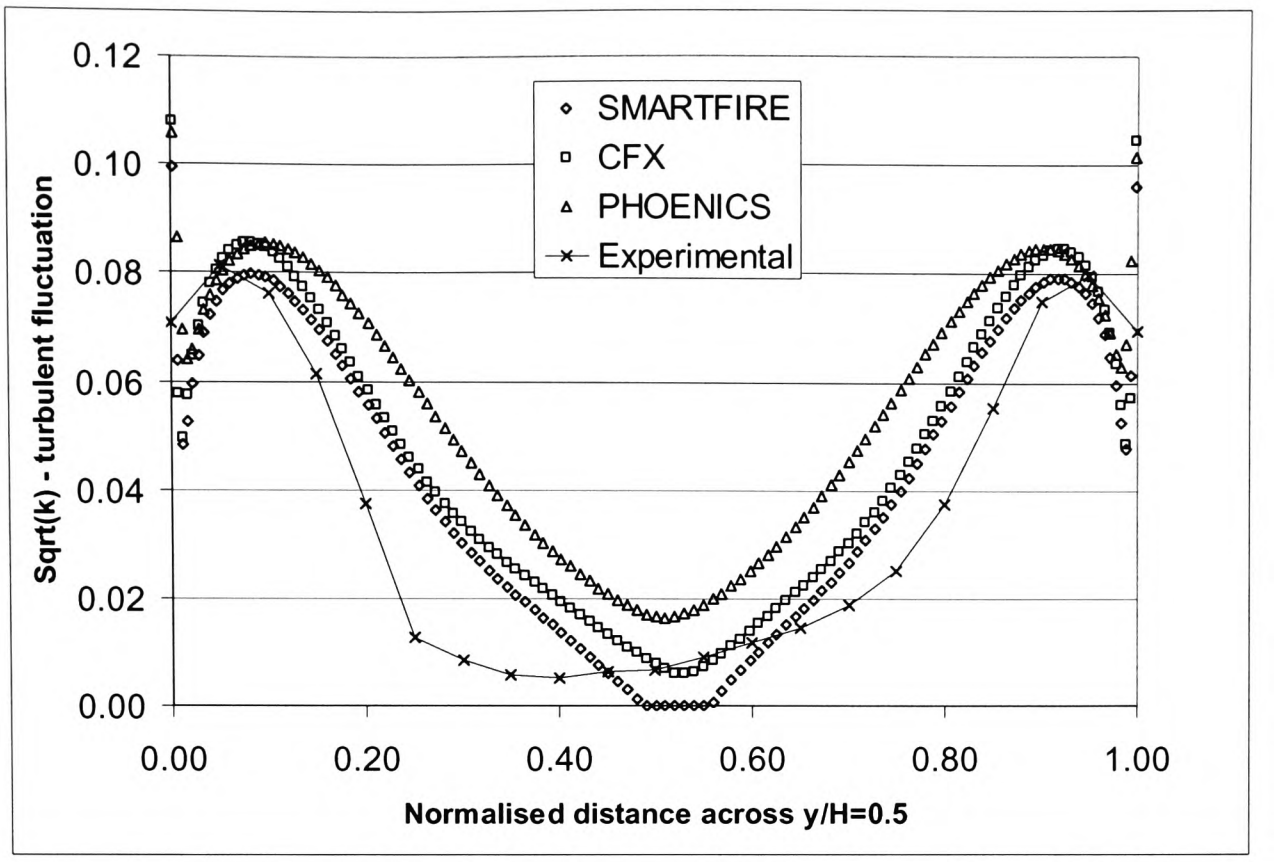


Figure 6-11 - Turbulent fluctuations across y/H=0.5 for 2000-1-4

Depicted in Figure 6-11 are the turbulent fluctuations at y/H=0.5 predicted by the SPs and the experimental results. All the codes were in reasonable agreement with one another although SMARTFIRE had a noticeable point where no turbulent fluctuations existed. All the SPs results were in good agreement with the experimental data. All the models exhibit high values close to the walls that are not reflected in the experimental result, this is due to a shortcoming that exists in the standard high-Re k-ε model.

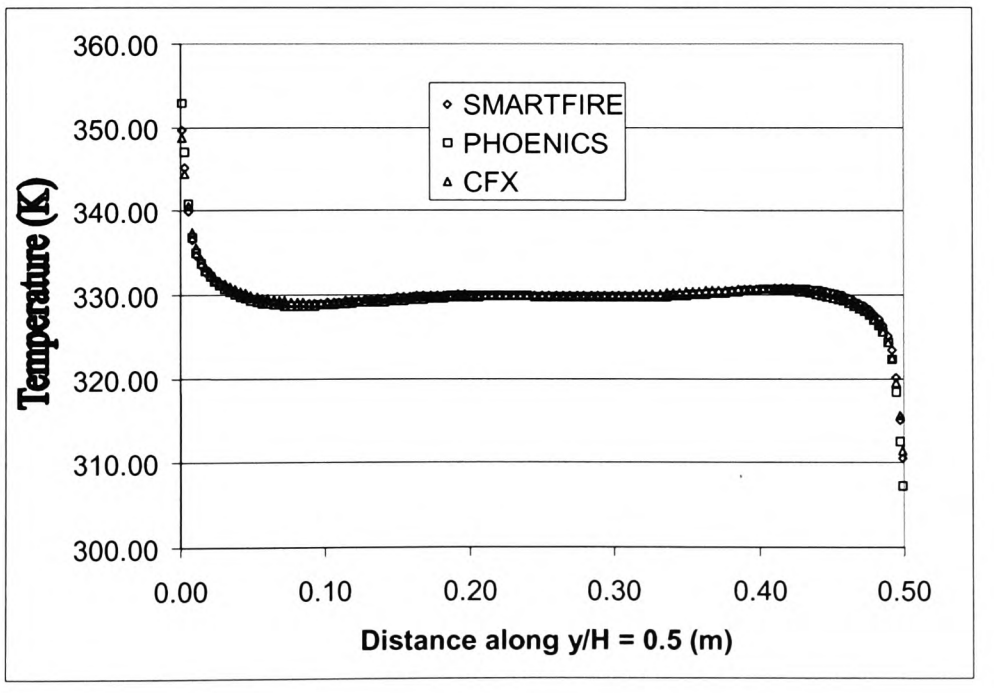


Figure 6-12 – Temperature variation along the y/H = 0.5 axis.

Depicted in Figure 6-12 are the temperature predictions along y/H = 0.5. As can be seen there was excellent agreement between the SP for the temperature variation across the x-axis.

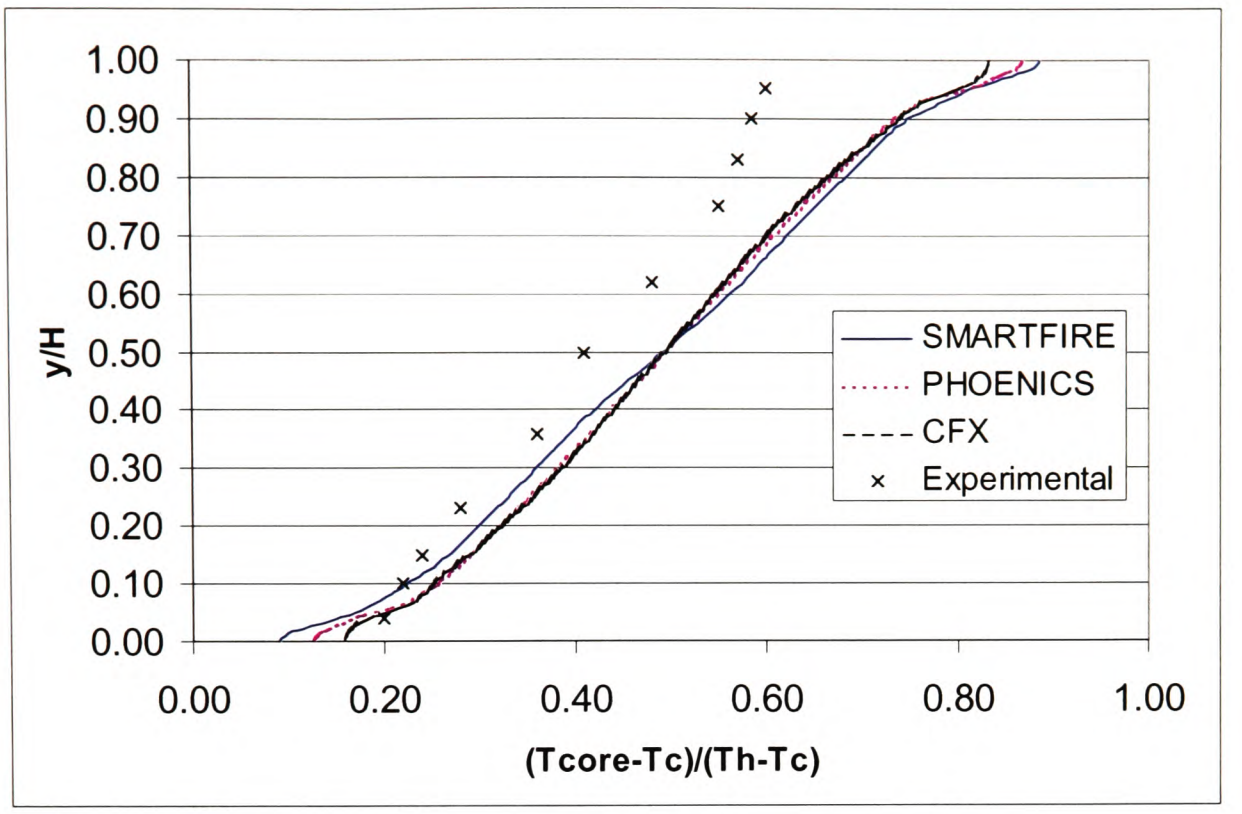


Figure 6-13 Temperature variation along the x/L = 0.5 axis

Depicted in Figure 6-13 are the temperature predictions along x/L = 0.5 predictions produced by the SPs and the experimental results. The SPs were in excellent agreement with each other although diverge from the experimental results at the higher end of the temperature differential. This difference was probably due to the three dimensional nature of the real problem and the heat losses which would occur on the top and bottom surfaces which have been assummed to be adiabatic in the modelling.

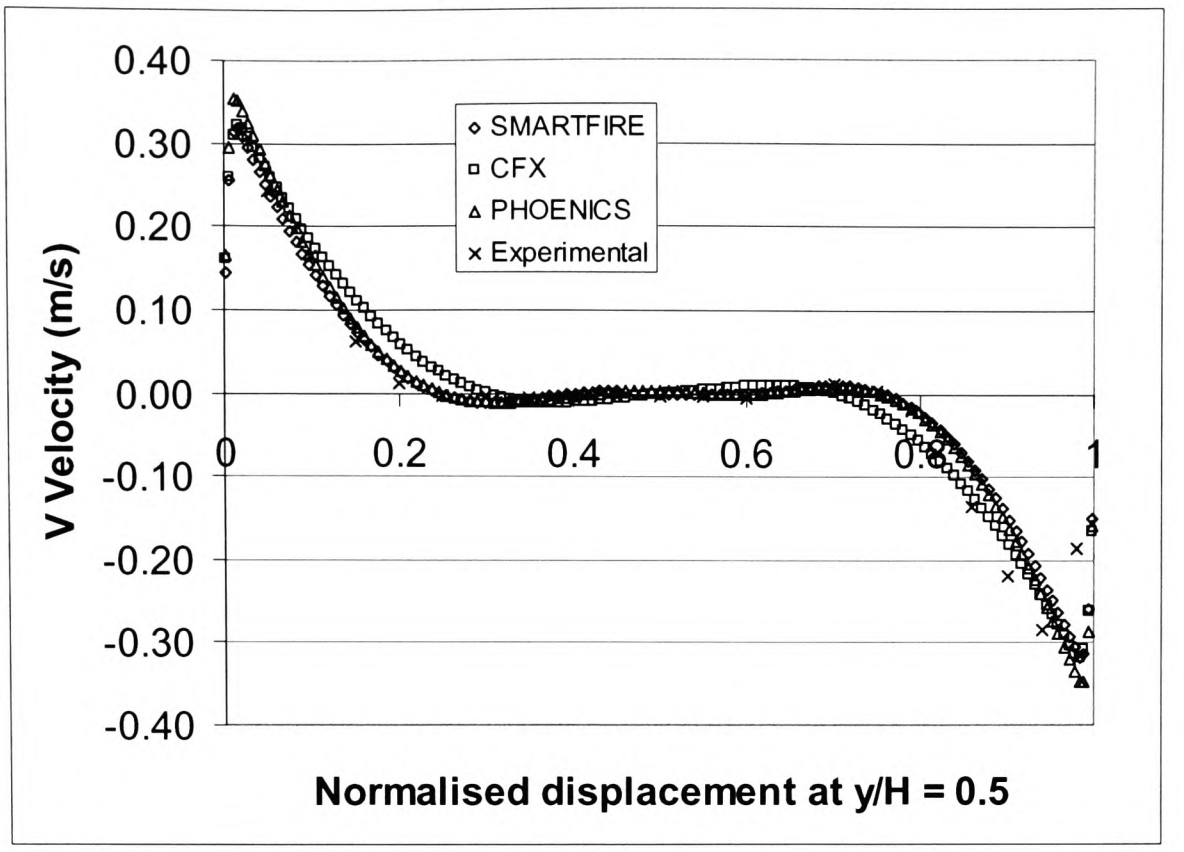


Figure 6-14 – Variation of V-Velocity along y/H = 0.5

Depicted in Figure 6-14 are the V velocity predictions and experimental results at y/H = 0.5. As can be seen, SMARTFIRE and PHOENICS are in reasonable agreement with one another. Both SPs produce slightly different results to CFX. It can be seen that the experimental values are closer to the SMARTFIRE and PHOENICS results between 0.0 – 0.5 and the experimental values are closer to CFX between 0.5 – 1.0.

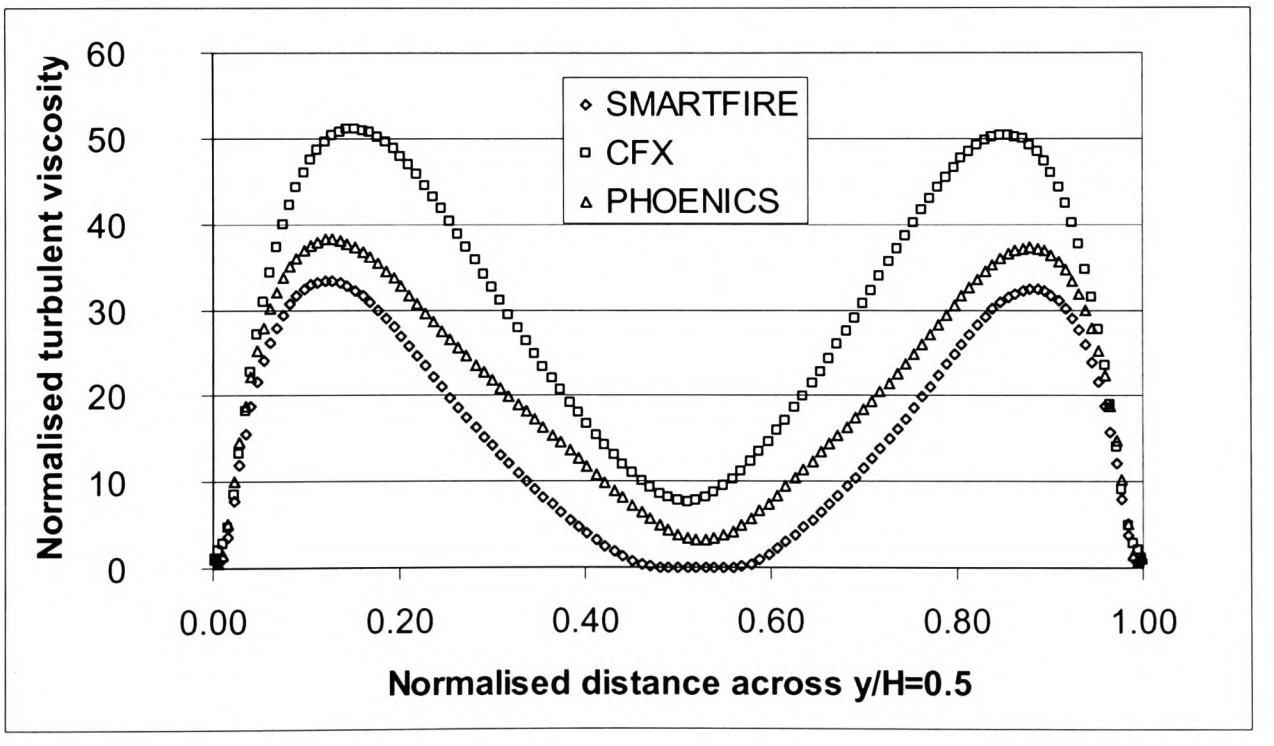


Figure 6-15 – Variation of normalised turbulent viscosity along y/H = 0.5.

Depicted in Figure 6-15 are the predictions for the normalised turbulent viscosity across y/H=0.5. As can be seen, SMARTFIRE and PHOENICS predictions were in

reasonable agreement with each other while CFX predicted a far greater turbulent viscosity. There were no experimental results for this parameter and so it is difficult to conclude which set of predictions are correct. Also note that SMARTFIRE predicts that the normalised turbulent viscosity goes to zero at the centre.

From Figure 6-11 to Figure 6-14 it can be seen that there was reasonable agreement between the codes and experimental data. While some differences exist between the codes, these were not considered to be significant.

6.1.1.5 2000-1-5 Radiation in a 3D cavity

The primary purpose of this test case was to test the radiation model used by the SPs. Model predictions were cross compared and also compared with theoretical predictions derived from detailed zone methods [Lar1983, Fiv1988].

The geometry used for this test case consists of a three dimensional unit cube (1m x 1m x 1m) cavity with three walls with planes x=1, y=0 and z =0 set to a unit emissive power and the three other walls set to zero emissive power. All the walls are considered radiatively black have unit emissivity and the fluid has a unit absorption coefficient. Scattering is neglected. No fluid flow is considered

For the CFX cases it was not possible to generate a radiation grid with the same number of cells as CFD cells. In order to generate an approximately equivalent model to that of SMARTFIRE and PHOENICS a CFD grid with 4 times as many cells in each of the co-ordinate directions was generated. This allowed the creation of a radiation grid with the same number of cells as used by the other codes. This should produce approximately the same effect, as the radiation cells that contain the medium will have the same temperature as the CFD cells as energy is only transported radiatively. This is seen in the stepped profiles from the CFX cases. The CFX cases were run in two configurations, the first using a single ray to emulate the behaviour of the six flux models of SMARTFIRE and PHOENICS, and using 12 rays which is the default option for the CFX radiation model.

See Appendix B.5 for further setup details.

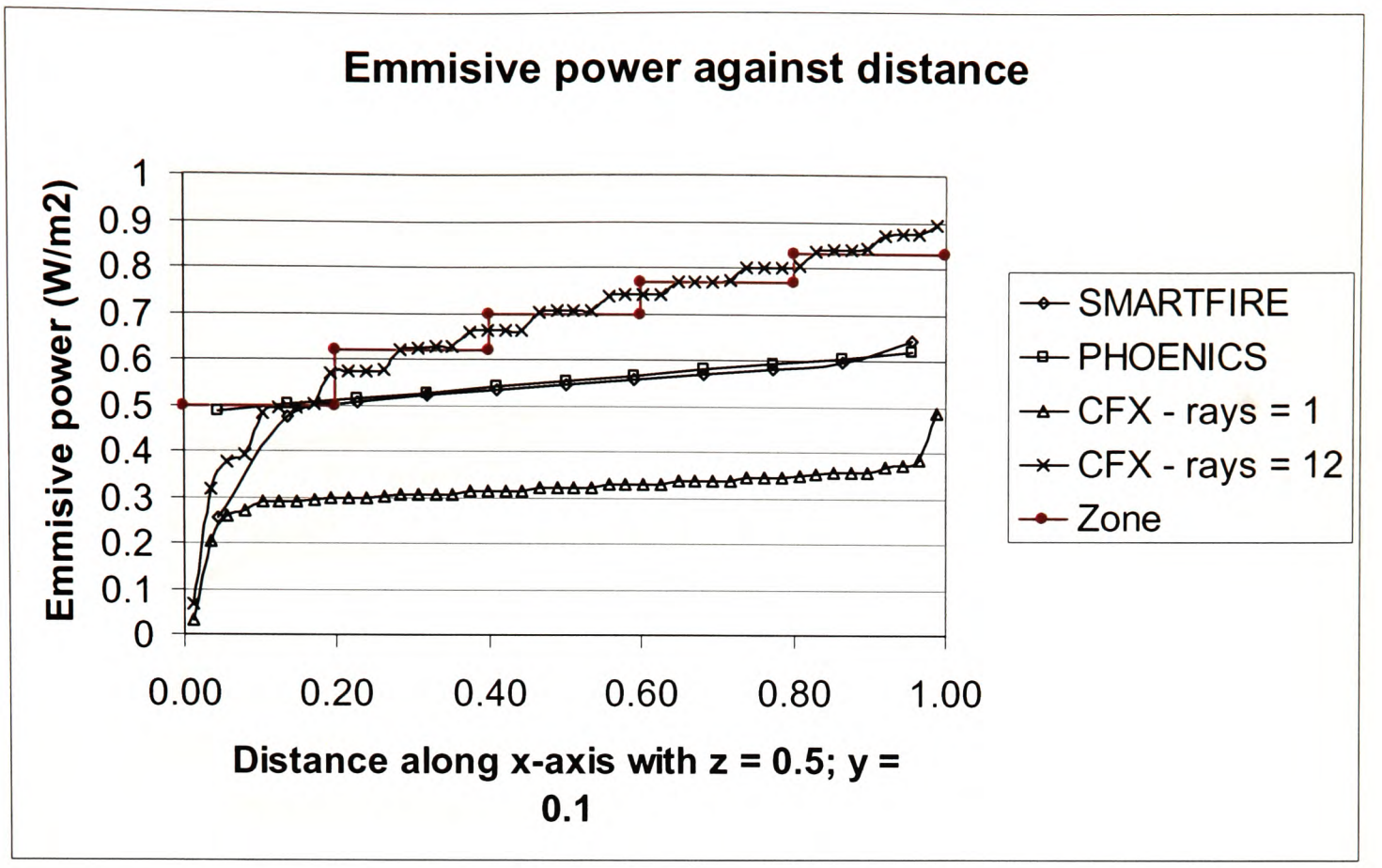


Figure 6-16 - Emissive power against distance along x-axis for z = 0.5; y = 0.1

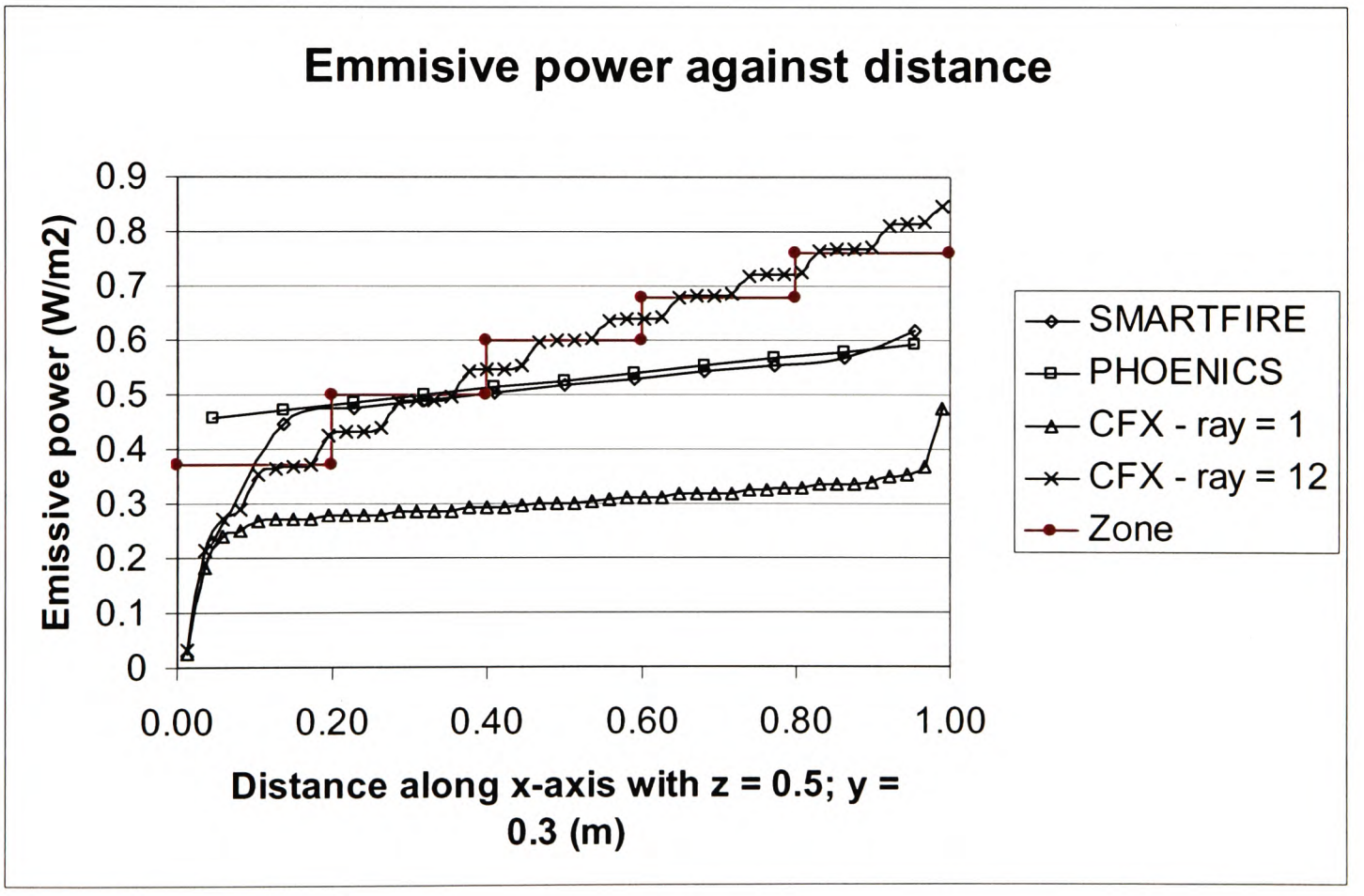


Figure 6-17 - Emissive power against distance along x-axis for z = 0.5; y = 0.3

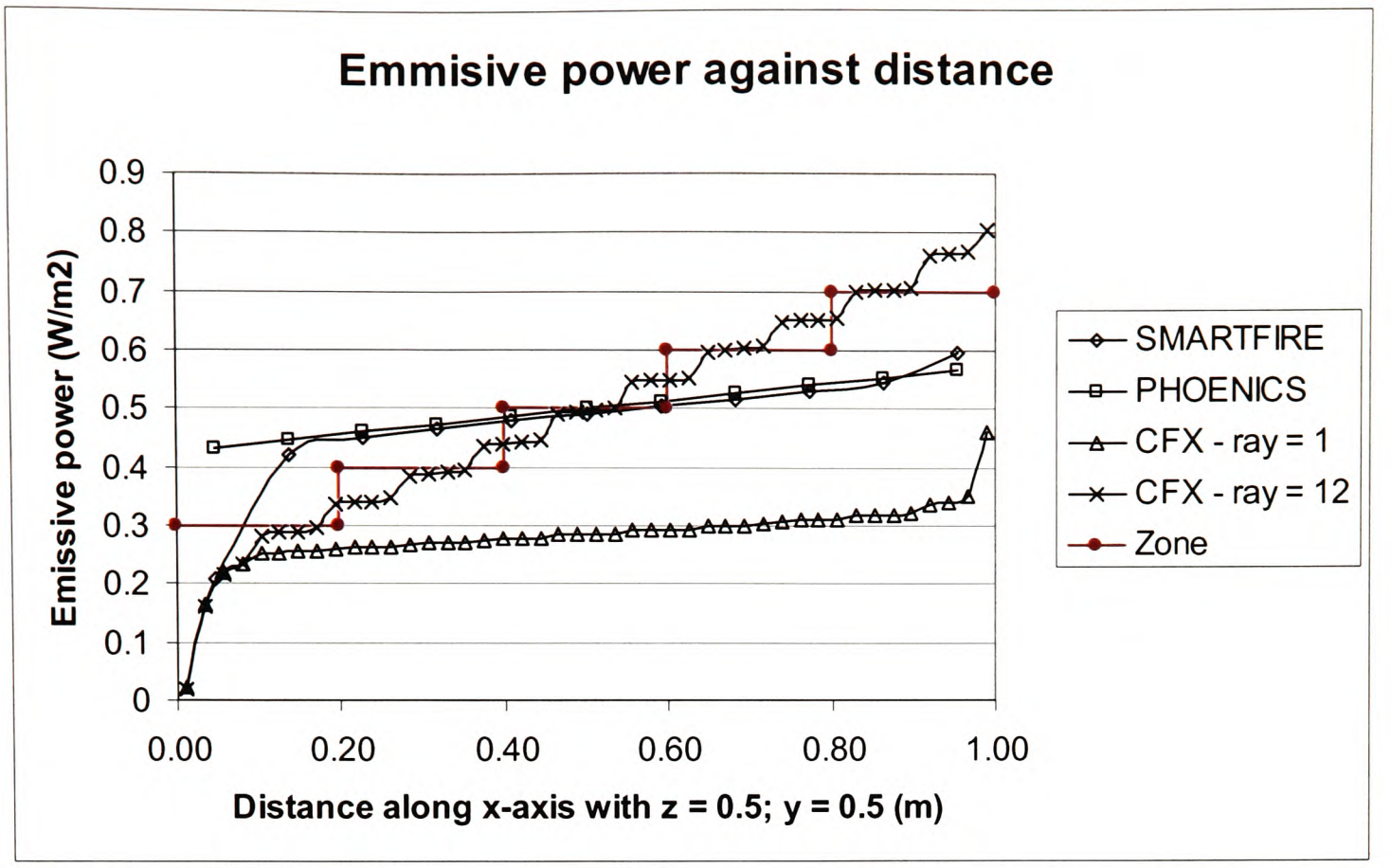


Figure 6-18 - Emissive power against distance along x-axis with z = 0.5; y = 0.5

In the above figures (Figure 6-16, Figure 6-17 and Figure 6-18) it can be seen that the 12-ray CFX radiation model produced a very good approximation to the theoretical zone emissive power. The six-flux model used by PHOENICS and SMARTFIRE – while producing similar results to one another - only provided a crude approximation to the theoretical emissive power.

It should be noted that the one-ray CFX radiation model is not mathematically equivalent to the six-flux model, because of the manner in which direction is discretised. In fact, as the results demonstrate, it is cruder than the six-flux model. Users of the CFX code are generally advised not to use this radiation model with a single ray. The default setting for this model has 12 rays specified. The difference between the 12 ray model and the six flux model is not surprising as the resolution of the radiation field is expected to be much better when using 12 rays as opposed to one ray. Furthermore, the six-flux model relies on a high degree of scattering to distribute the radiation and no scattering is present in this case. It should be further noted that the six-flux model is not intended for applications where the accuracy of the heat flux at a solid surface is a crucial component of the calculations, such as situations involving flame spread over solid surfaces or when structural interaction with the fire is being predicted. It is intended for applications where the dominant factor is the

radiative heat loss from the flame. This is commonly the situation when representing non-spreading fires.

6.1.2 Fire cases

In this section the results generated by the author are presented.

6.1.2.1 2000-2-1 & 2000-2-2 – Steckler fire case

This test is a standard fire model test case used by a number of field and zone model developers. Its primary purpose was to test the fire models predictive capability in predicting temperature and flow distributions in a small compartment subjected to a steady non-spreading fire. Predictions of several parameters were made and cross compared. Model predictions were also compared with experimental results [SQR1982].

The non-spreading fire was created using a centrally located (position A in Figure 6-19) 62.9kW methane burner with a diameter of 0.3m. The experiments were conducted by Steckler et al. in a compartment measuring 2.8m × 2.8m in plane and 2.18m in height (see Figure 6-19) with a doorway centrally located in one of the walls measuring 0.74m wide by 1.83m high. The walls and ceiling were 0.1m thick and they were covered with a ceramic fibre insulation board to establish near steady state conditions within 30 minutes.

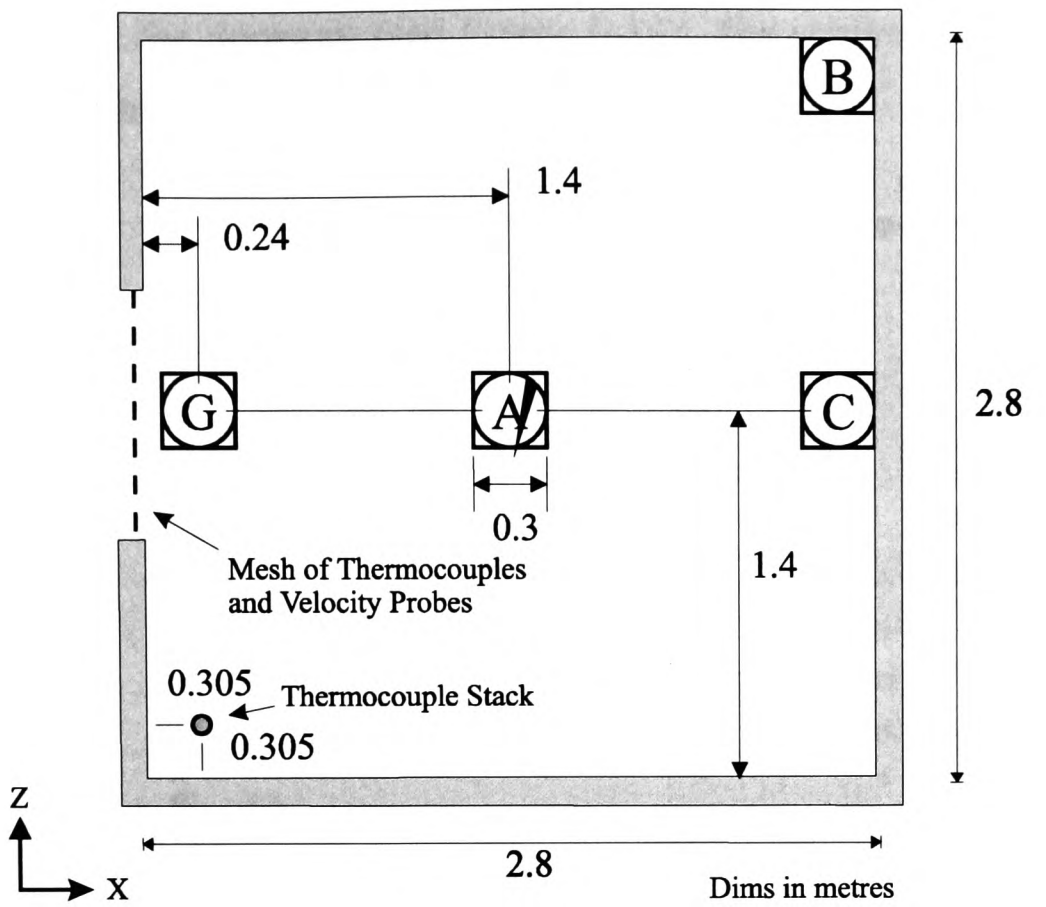


Figure 6-19 – Configuration of Steckler room

The door measures 0.74m wide and 1.83m high and is centrally located in one of the walls. Within the models, the walls are all assumed to be adiabatic and perfect radiative reflectors. The case is run for 200s of simulated time using 200 time steps of 1s at which point steady state conditions are achieved in the simulation.

This case has been modelled using 2 methods: -

3) Using a simple volumetric heat source (2000-2-1)
4) Using a combustion model (2000-2-2)

In PHOENICS and SMARTFIRE a six-flux radiation model is used, while in CFX the discrete transfer model is used with a single ray in the co-ordinate direction to emulate the behaviour of a six-flux radiation model.

See Appendix C.1 and C.2 for further setup details.

Comparisons between the SPs using both a simple heat release model and a combustion model are presented below (Figure 6-20 - Figure 6-22). The comparisons were made at two different locations; corner thermocouple stack located in one of the near corners to the doorway and a thermocouple and velocity measuring stack

centrally located in the doorway (see Figure 6-19). The results presented are after 200s of simulated time at which point the results are steady state.

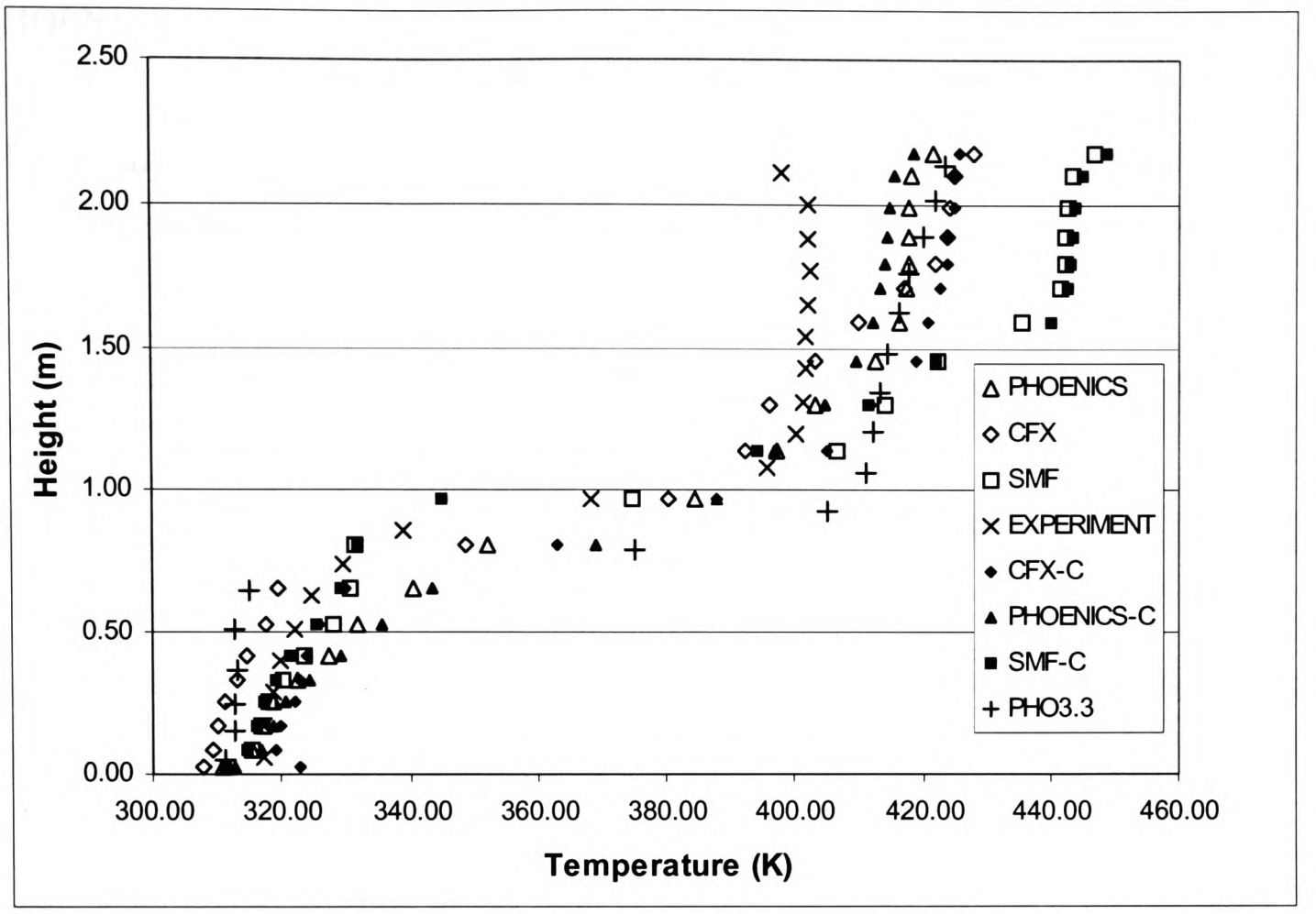


Figure 6-20 - Corner Stack temperatures produced using heat source model and combustion model.

Depicted in Figure 6-20 is the corner stack temperature profile generated by the SPs using the volumetric heat source model and the combustion model along with the experimental results. The temperature profile for the volumetric heat source model provided by CHAM using PHOENICS V3.3 is also supplied. In viewing these results it must be remembered that the walls have been treated as adiabatic. As a result it is expected that the upper layer temperatures will be in excess of the measured temperatures.

Table 6-2 - Approximate upper heat layer temperature for Steckler's room (A74) using Heat Source model (H) and Combustion model (C).

	Exp	PHO-H	PHO-C	PHO3.3	CFX-H	CFX-C	SMF-H	SMF-C
Temp (K)	401	412	414	420	423	424	442	443

From Figure 6-20 and Table 6-2 it can be seen that all three SPs over predict the upper layer temperatures. It is interesting to note that the combustion models do not improve the prediction of the upper layer temperature. It is also interesting to note that all three

SPs produce different estimates of the upper layer temperature, with SMARTFIRE predicting the hottest and furthest removed from the experimental value and PHOENICS predicting the coolest temperature and closest to the experimental value.

The location of the hot layer can be estimated by determining where uniform temperatures are established in the upper layer. These values are presented in Table 6-3.

Table 6-3 - Approximate upper heat layer height, using uniform temperatures, for Steckler's room (A74) using Heat Source model (H) and Combustion model (C).

	Exp	PHO-H	PHO-C	PHO-3.3	CFX-H	CFX-C	SMF-H	SMF-C
Height (m)	1.25	1.5	1.5	1.05	1.75	1.5	1.6	1.6

As the hot layer is not sharply defined, an alternative definition for the height of the thermal interface can be defined as the height with the largest spatial (vertical) temperature gradient. These values are presented in Table 6-4.

Table 6-4 - - Approximate upper heat layer height, using largest temperature gradient, for Steckler's room (A74) using Heat Source model (H) and Combustion model (C).

	Exp	PHO-H	PHO-C	PHO-3.3	CFX-H	CFX-C	SMF-H	SMF-C
Height (m)	0.97	0.9	0.77	0.75	0.85	0.81	0.97	1.05

The alternative definition gives a closer comparison for all the models with the experimental values. The greatest difference between experiment and model prediction is 0.2m for the PHO-3.3 case using the alternative definition. This was the closest comparison using the original definition of hot layer height. The first definition suffers from the fact that the models tend to smear the interface and the temperature is still increasing within the hot layer so the exact location of the hot layer was open to interpretation. The alternative definition does not suffer from this open ended interpretation.

Both CFX models capture the temperature trend below 1m reasonably well. Above 1m the CFX heat source model does not capture the upper layer trend very well although the temperature predictions were not unreasonable given the adiabatic nature of the simulations compared to that of the experiment. The CFX combustion model produces a much better trend above 1m compared to the CFX heat source model.

Both SMARTFIRE models produce very similar results to one another. The trend below 1m is well captured although above 1m the temperature is hotter than the experiment and that predicted by both PHOENICS and CFX.

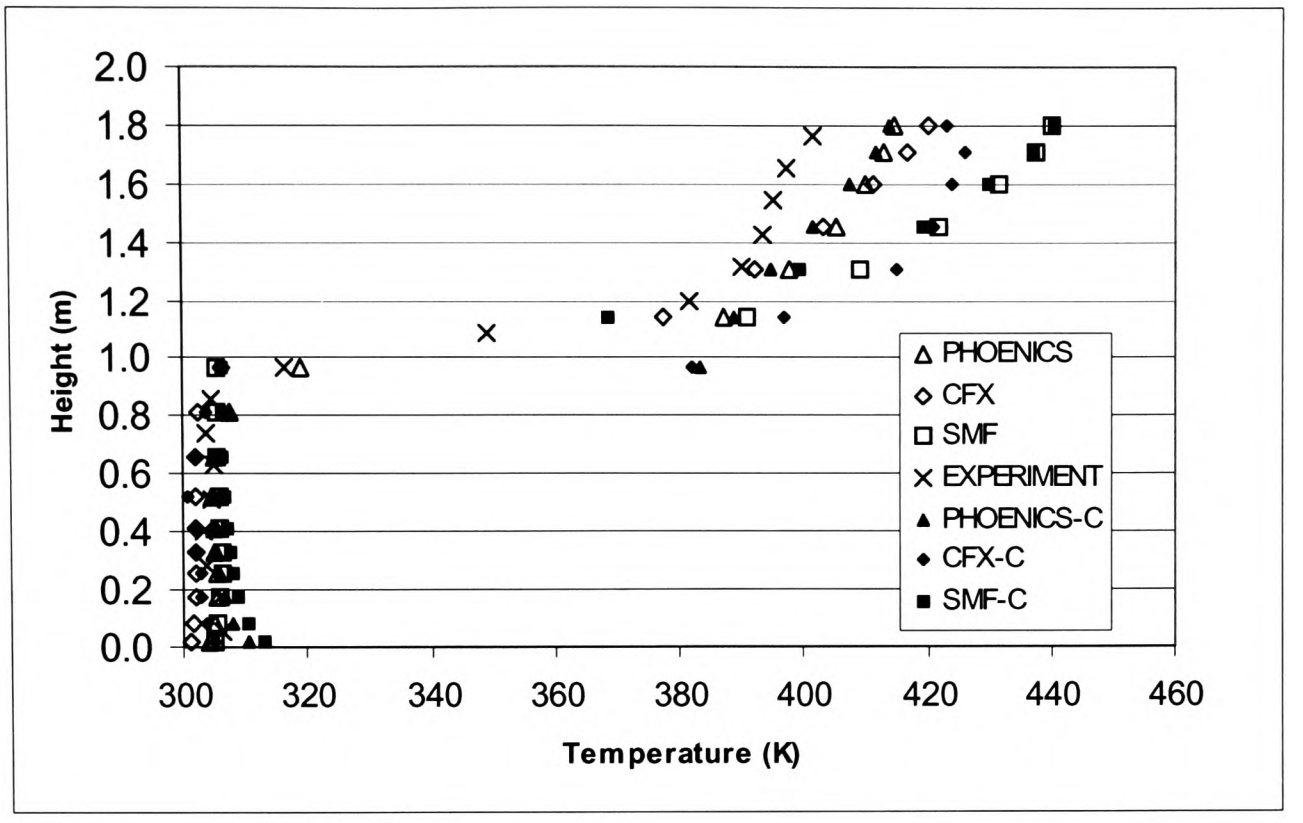


Figure 6-21 - Comparison of doorway temperatures for Steckler room

Depicted in Figure 6-21 is the doorway centre vertical temperature profile generated by the SPs using the volumetric heat source model and the combustion model along with the experimental results. In viewing these results it must be remembered that the walls have been treated as adiabatic. As a result it is expected that the upper layer temperatures and the resulting temperatures of the hot vented gases will be in excess of the measured temperatures.

From Figure 6-21 it can be seen that – as with the previous case - all three SPs over predicted the temperature of the hot gases being vented out of the compartment. Once again, it is interesting to note that the combustion models do not improve the prediction of the hot vented gas temperature. It is also interesting to note that all three SPs produced different estimates of the vented hot gas temperature, with SMARTFIRE predicting the hotest and furtherest removed from the experimental value and PHOENICS predicting the coolest temperature and closest to the experimental value.

Depicted in Figure 6-22 is the doorway centre horizontal velocity profile generated by the SPs using the volumetric heat source model and the combustion model along with the experimental results. All the SPs appear capable of generating an excellent prediction of the velocity profile. Below the neutral plane SMARTFIRE and PHOENICS appear to best reproduce the velocity profile while above the neutral plane, high up in the door, PHOENICS appears to best reproduce the velocity profile.

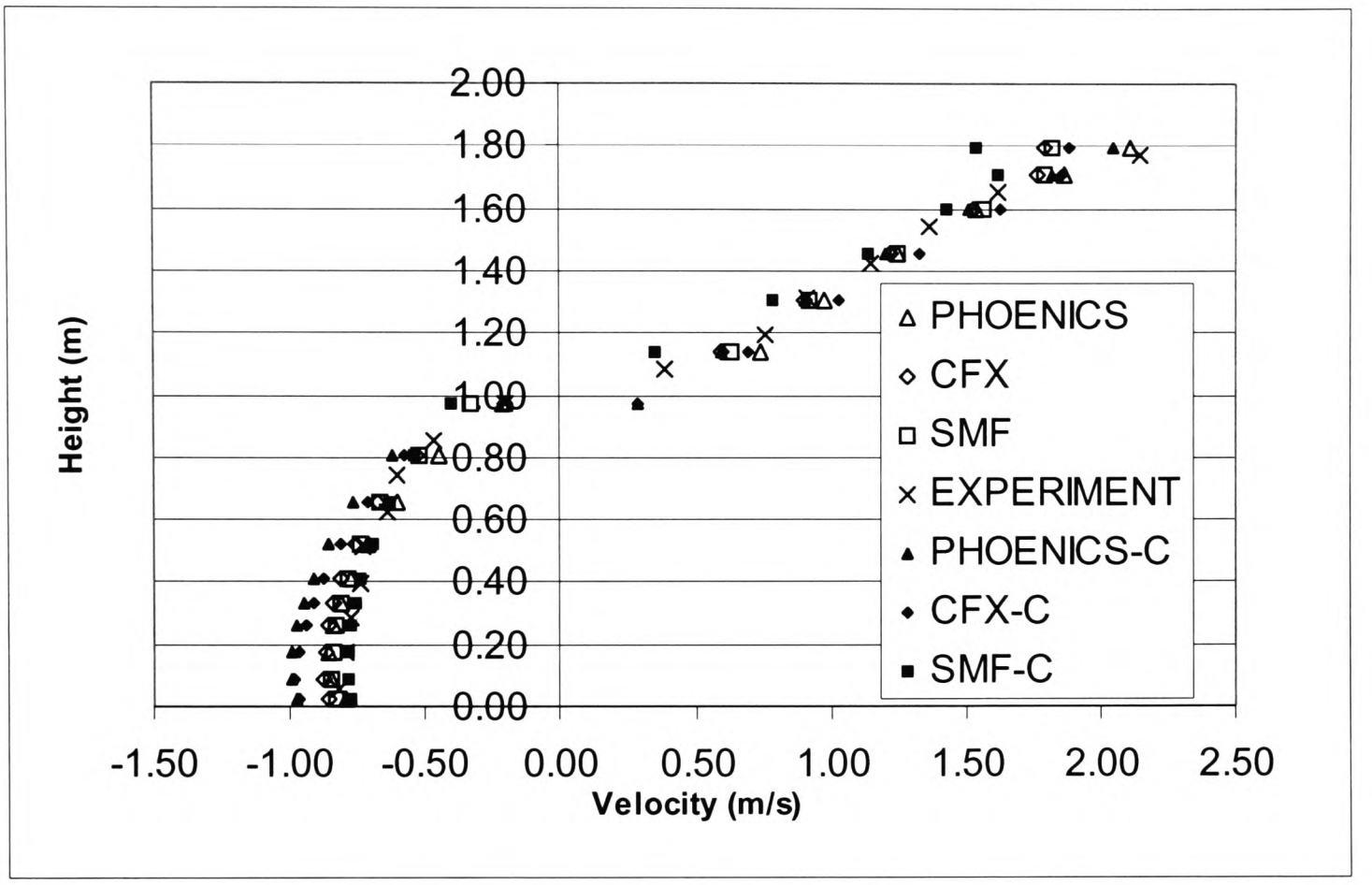


Figure 6-22 - Comparison of doorway velocity profiles for Steckler room

Depicted in Figure 6-23, Figure 6-24 and Figure 6-25 are temperature contour plots along the centre of the compartment for PHOENICS, CFX and SMARTFIRE respectively produced using the volumetric heat source model. From these temperature maps it can be seen that all the SPs produce similar trends. Most notable is the plume leaning away from the doorway. The temperature contours range from 320K to 500K and are separated by 20K increments. It can be seen that the lowest temperatures was produced by PHOENICS and the highest temperature was produced by SMARTFIRE, as would have been expected from the thermocouple stack comparisons (Figure 6-20 and Figure 6-21). The CFX and PHOENICS plume lean over by approximately the same amount with SMARTFIRE leaning over slightly less.

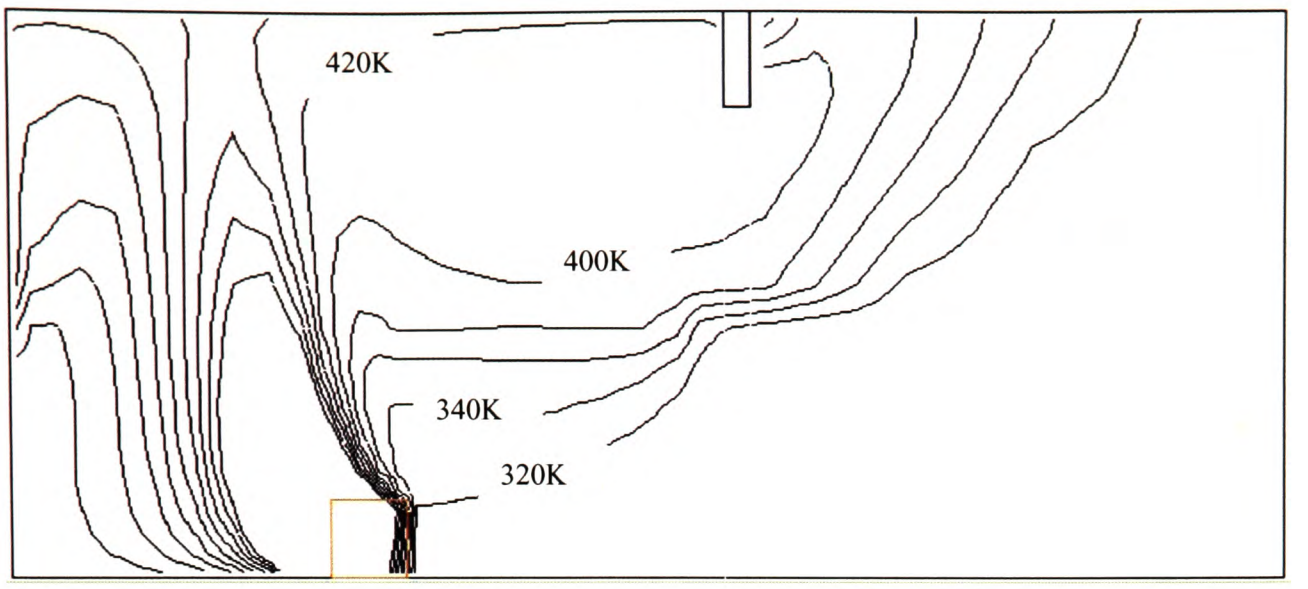


Figure 6-23 - Temperature contour plot produced by PHOENICS using the heat source model

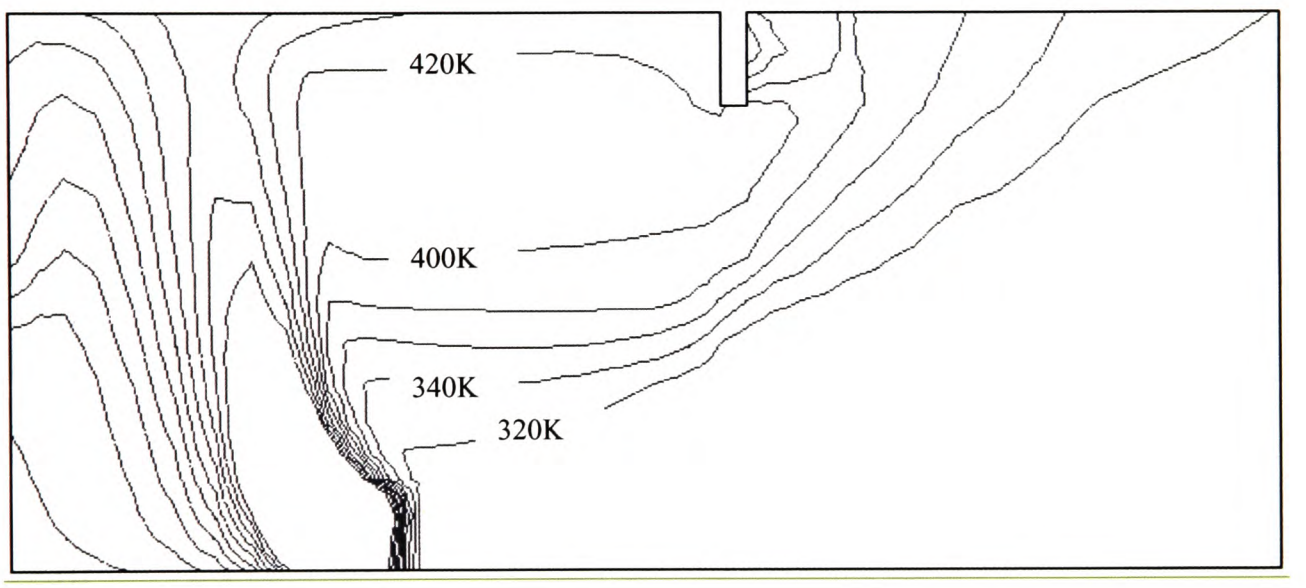


Figure 6-24 - Temperature contour plot produced by CFX using the heat source model

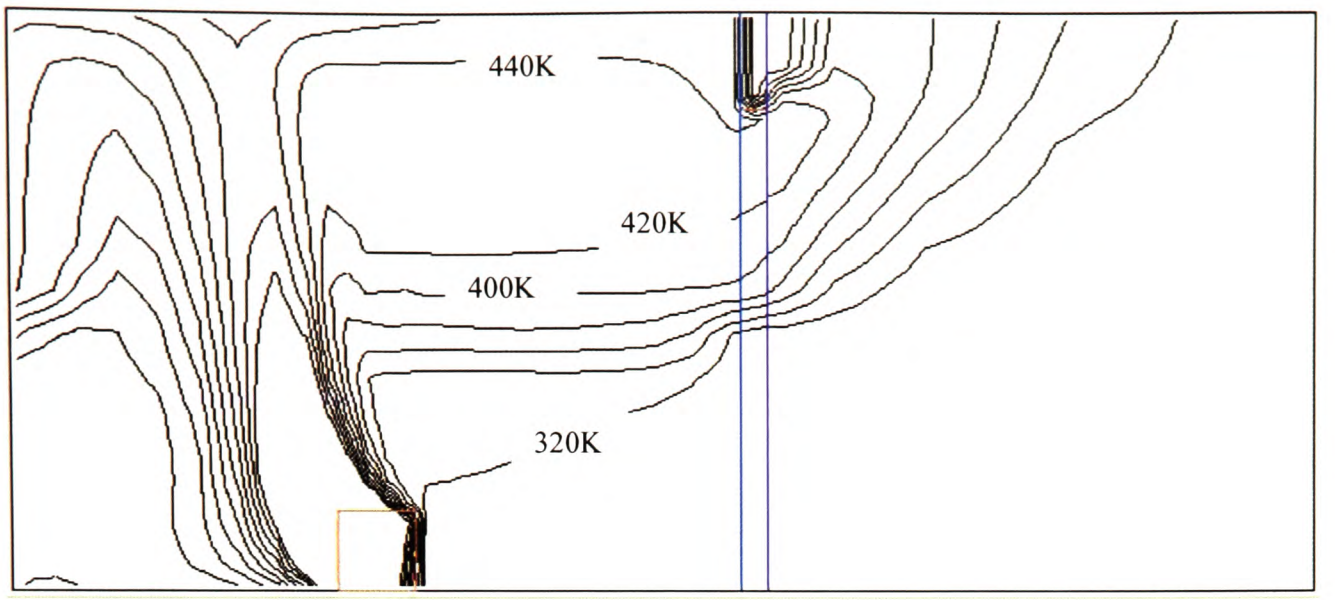


Figure 6-25 - Temperature contour plot produced by SMARTFIRE using the heat source model

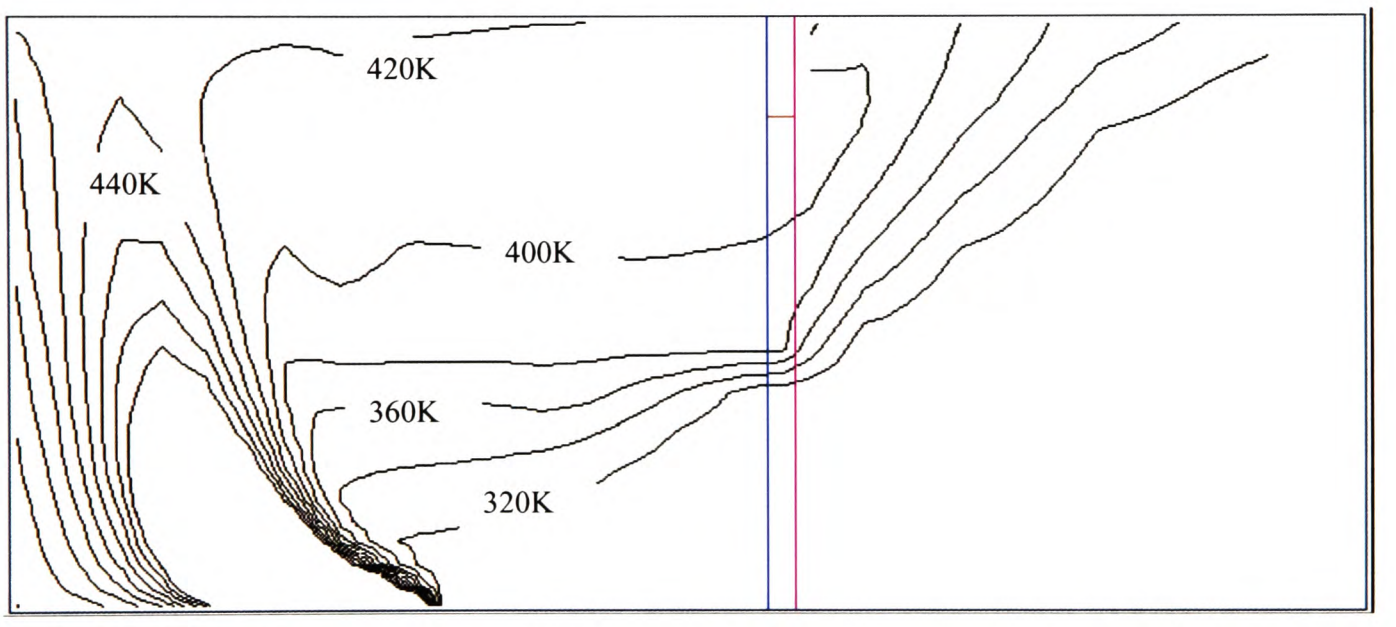


Figure 6-26 - Temperature contour plot produced by PHOENICS using the combustion model

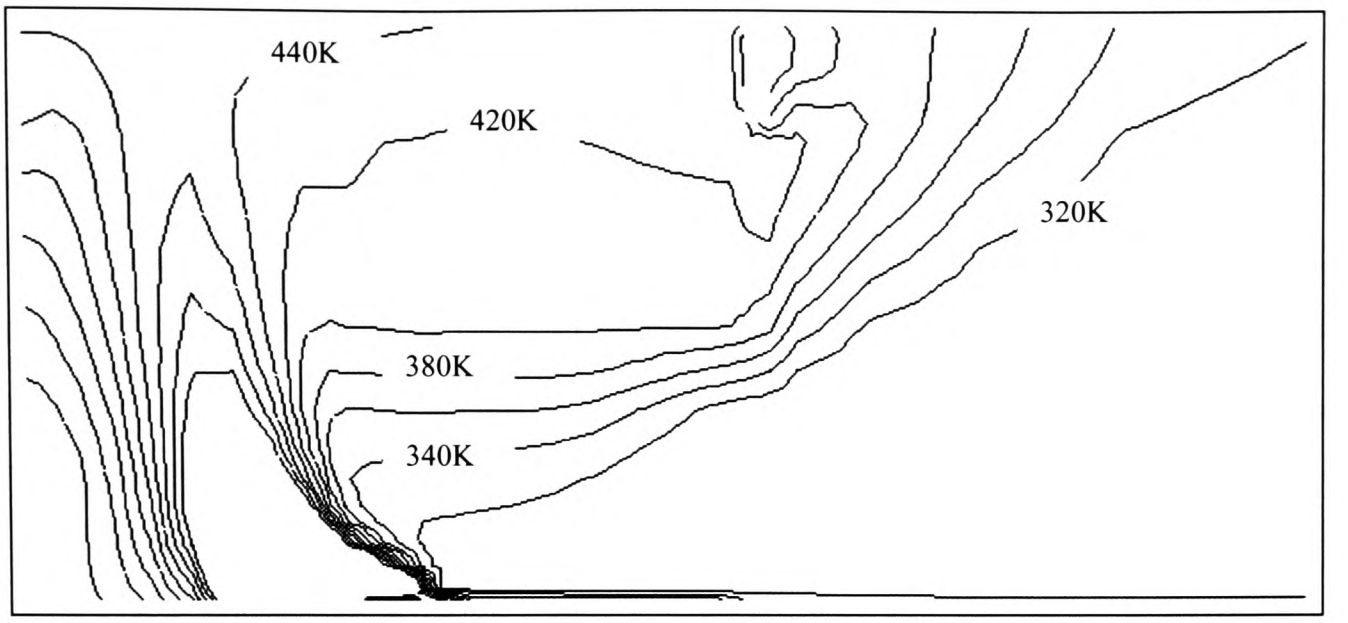


Figure 6-27 - Temperature contour plot produced by CFX using the combustion model

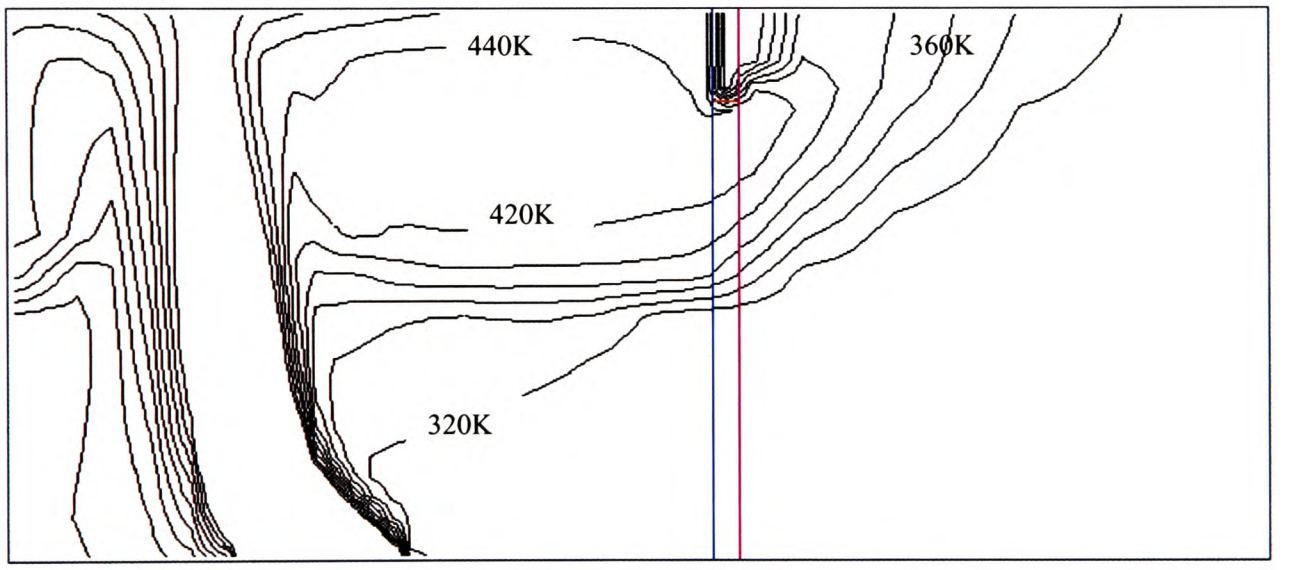


Figure 6-28 - Temperature contour plot produced by SMARTFIRE using the combustion

Depicted in Figure 6-26, Figure 6-27 and Figure 6-28 are temperature contour plots along the centre of the compartment for PHOENICS, CFX and SMARTFIRE respectively generated using the combustion model. All three SPs depict similar types of behaviour with a similar temperature distribution throughout the compartment. However, the behaviour of the plume appears to be noticeably different in the three cases. The greatest lean towards the rear wall was found in the PHOENICS case followed by the CFX case with SMARTFIRE producing the most up-right plume. This may indicate that more air is being entrained into the PHOENICS plume

reducing the overall temperature prediction in the compartment. CFX shows a heating of the floor under the incoming cool air, by radiation from the hot gases.

Comparing the heat source model (Figure 6-23, Figure 6-24 and Figure 6-25) and the combustion model (Figure 6-26, Figure 6-27 and Figure 6-28) it is apparent that the artificial volume created to release the heat source partly defines the nature of the plume. This was most evident on the doorway side of the plume with the cubic nature of the source showing up in all the SPs used in heat source mode. This did not occur in the combustion model. This effect could be minimised by using a volume of reduced height however, this would reduce the volume over which to add the heat increasing the temperature of the flame resulting in too much heat being lost as radiation. This could also lead to problems in convergence. Despite this shortcoming in the heat source model it still produces good agreement with the experimental and combustion model results for the specified monitor locations.

In conclusion, although the SPs display some differences between one another the results are all reasonably close and self-consistent. It should also be noted that these results may be greatly improved using more sophisticated boundary conditions, grid refinement and other physical models that have not been used in the phase-1 exercise.

6.1.2.2 2000-2-3 – Open Fire with Lid case

This test is an artificial fire test case. There were no experimental results for comparison purposes. Its primary purpose was to test the fire models predictive capability in predicting temperature and flow distributions in a small well ventilated open compartment subjected to a non-spreading fire. Predictions of several parameters are made and cross compared.

This fire case utilises a volumetric heat source. The compartment is completely open apart from a solid ceiling (see Figure 6-29). The fire is located on the floor at the centre of the building. The prescribed fire volume is 1m x 1m x 1m. The fire power is defined as $H = 0.188t^2(kW)$ (i.e. t squared fire and t is measured in seconds). The compartment is 5m(wide) × 5m(long) × 3m(high). The ceiling is adiabatic. The

ambient temperature is 303.75K. The case was run for 110s of simulated time using 110 time steps of 1s.

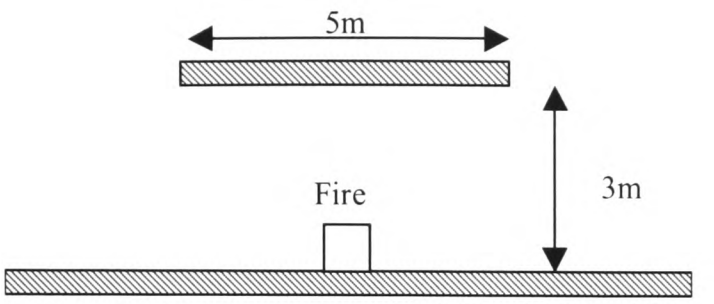


Figure 6-29 - Configuration of open fire with lid

See Appendix C.3 for further setup details.

The PHOENICS and SMARTFIRE simulations make use of the six-flux radiation model while in CFX the discrete transfer model is used with a single ray in the co-ordinate direction to emulate the behaviour of a six-flux radiation model.

All the results below show the temperature distribution at 110 seconds.

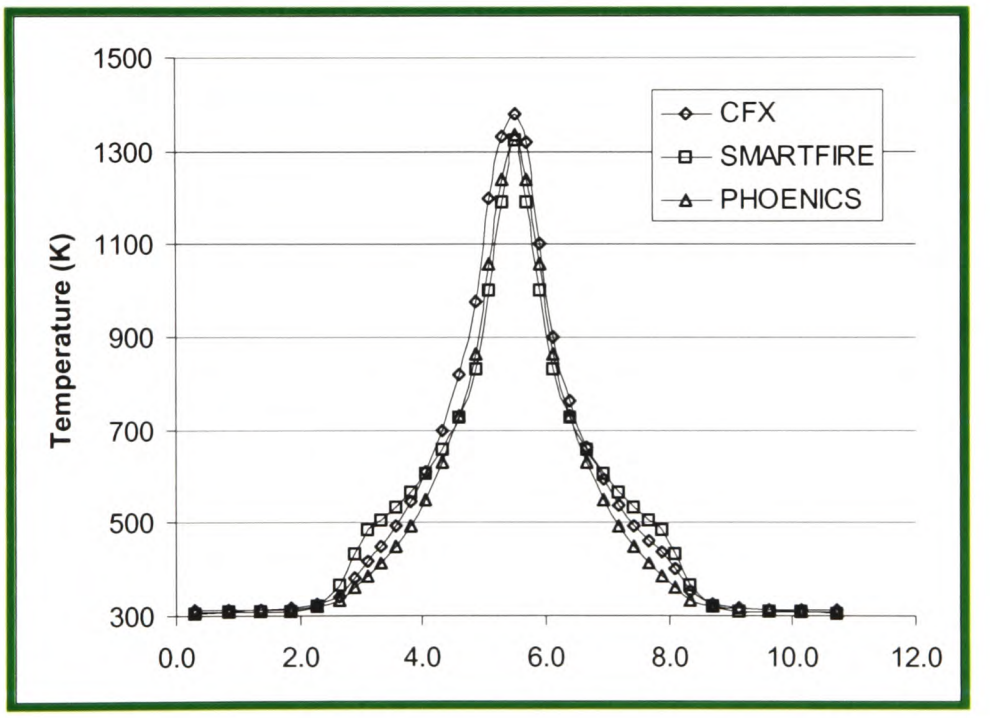


Figure 6-30 - Temperature profile 0.1m below ceiling along centrally located x - axis

Depicted in Figure 6-30 is the temperature distribution 0.1m below the ceiling along the centrally located x-axis. As can be seen, all three SP's produced temperature predictions that broadly agreed. There was a 5% difference in the maximum ceiling temperature predicted. CFX predicted the highest temperature at 1380K while SMARTFIRE predicted the lowest temperature at 1320K. There also appeared to be a

slight difference in the temperature profile at approximately 0.3m and 0.8m along the ceiling. At these locations, SMARTFIRE appears to predict slightly elevated temperatures. At the same locations, CFX appears to predict a much smaller increase in the temperature, but this is still greater than the temperature predicted by PHOENICS.

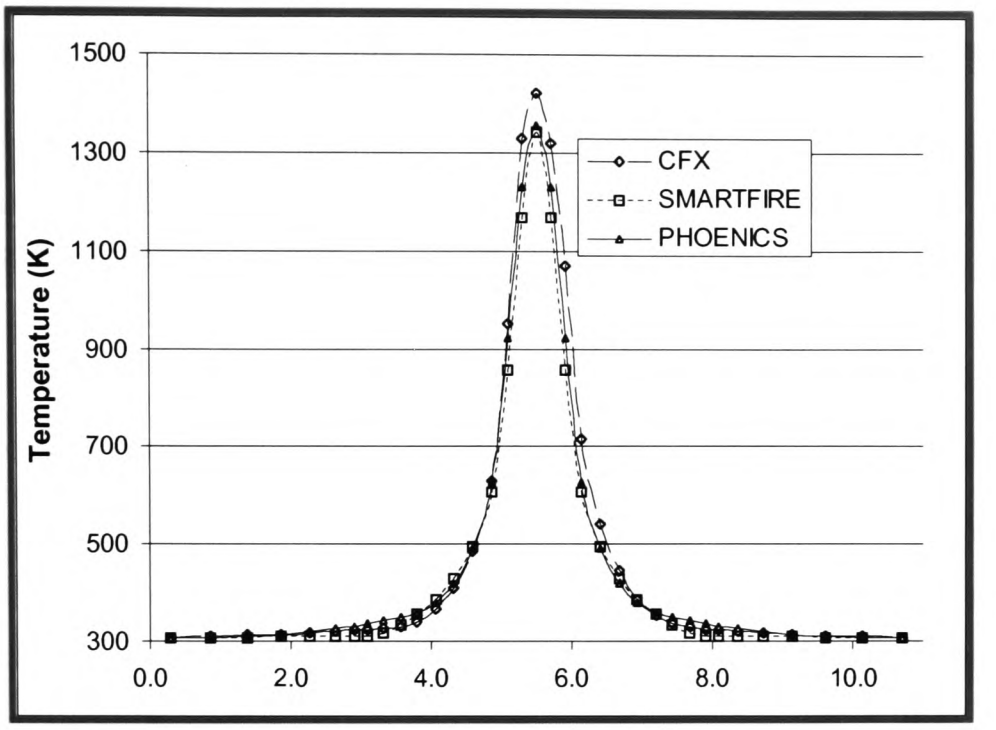


Figure 6-31 - Temperature profile 0.3m below ceiling along centrally located x - axis

From Figure 6-31 it can be seen that the temperature profiles of all the SPs are very similar.

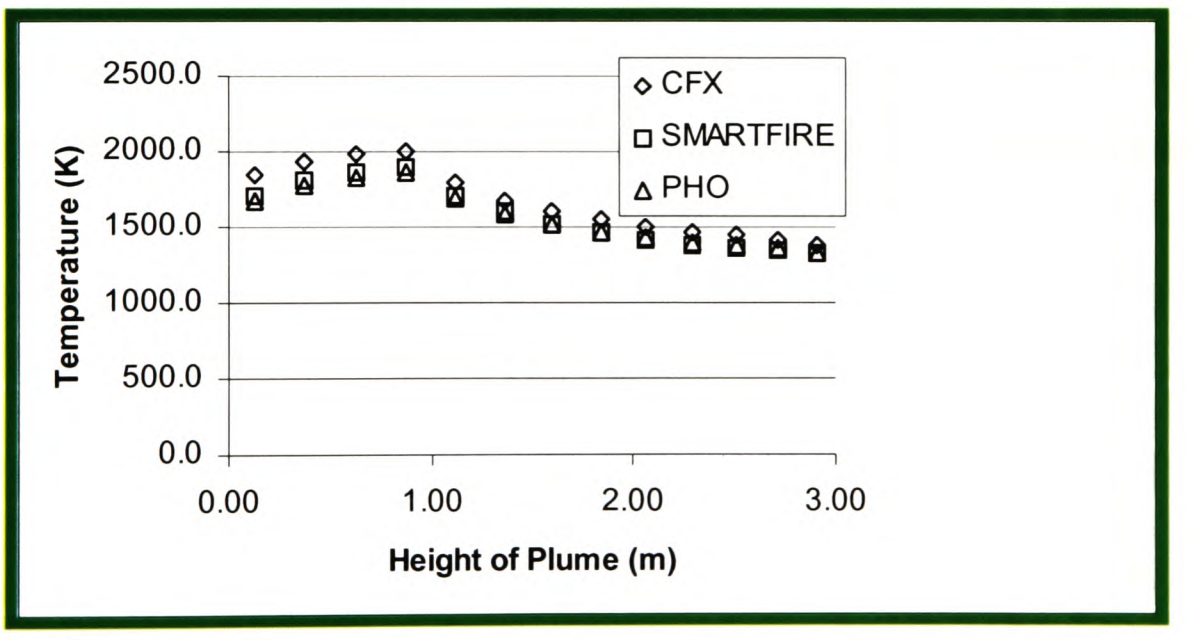


Figure 6-32 - Temperature profile through the centre of the fire plume

From Figure 6-32 it can be seen that there is good agreement between the SPs for the predicted plume variation with height.

Depicted in Figure 6-32 is the variation of plume temperature with height. As can be seen, all three SPs produced the same trends and variation. PHOENICS produced the cooler temperatures and CFX produced the hottest temperatures. The variation between the maximum temperatures is shown in Table 6-5. As can be seen there are no significant differences in the predicted peak temperatures.

Table 6-5 - Variation of peak temperature between SPs for 2000-2-3

	PHOENICS	SMARTFIRE	CFX
Maximum Temp	**1860K**	**1900K**	**2000K**
%difference with PHOENICS	-	**2.15%**	**7.53%**

Depicted in Figure 6-33, Figure 6-34 and Figure 6-35 are the temperature contours through the vertical central plane passing through the fire plume as predicted by CFX, PHOENICS and SMARTFIRE respectively. The temperature contours are separated by 200 degrees and range between 400K to 1600K. From these figures it is apparent that SMARTFIRE, PHOENICS and CFX produced similar profiles that resemble a fire plume impacting on a flat ceiling.

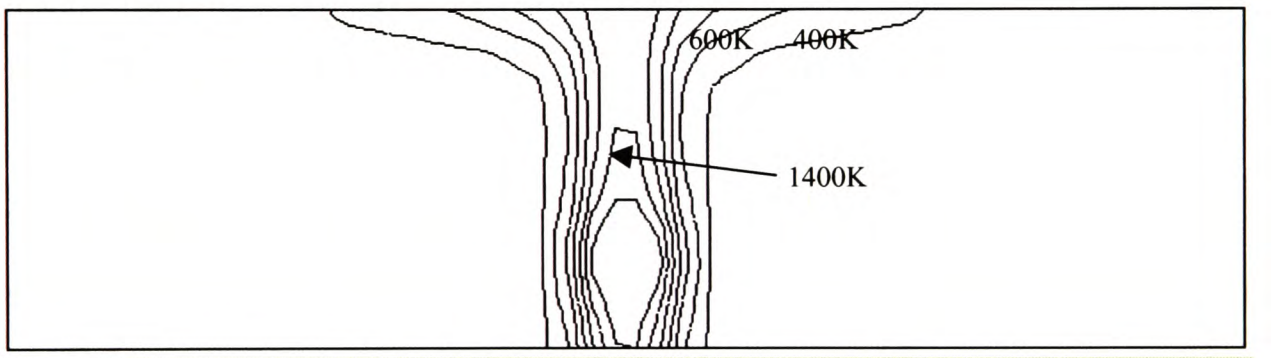


Figure 6-33 – CFX generated temperature contours through plume on central vertical plane

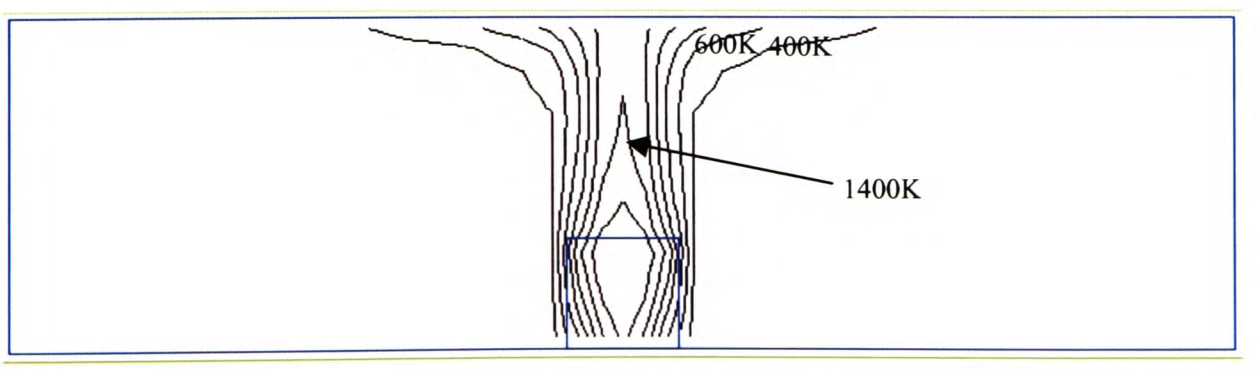


Figure 6-34 – PHOENICS generated temperature contours through plume on central vertical plane

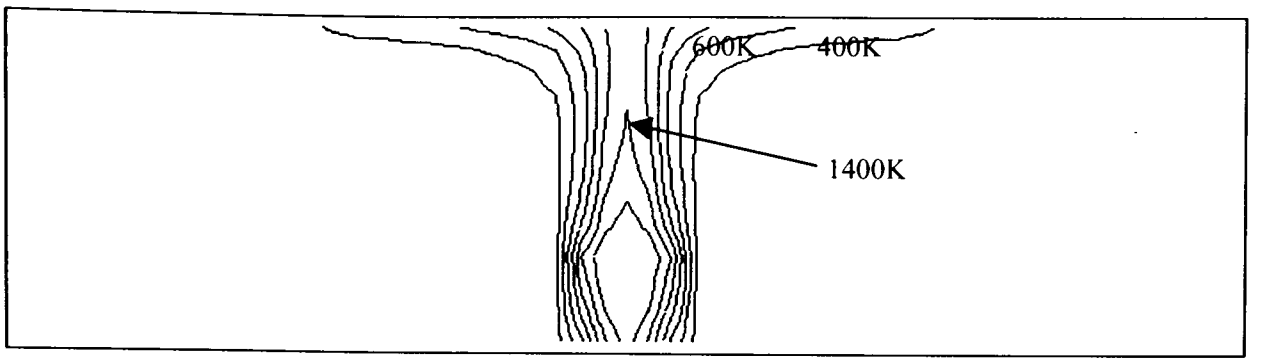


Figure 6-35 – SMARTFIRE generated temperature contours through plume on central vertical plane

6.1.2.3 2000-2-4 – CIB W14 case

This case arises from the CIB round robin tests of which subscenario B1 is the case of interest [HK1998]. The fire compartment measured 14.4 m × 7.2 m in plan and 3.53 m in height and contained a doorway of dimensions 2.97 m × 2.13 m. The walls of the compartment were made of aerated concrete blocks (with siporex mortar) with thickness 0.3 m and the following material properties: specific heat 1.05 kJ/kg.K, thermal conductivity 0.12 W/m.K and density 500 kg/m^3. The initial air temperature was measured as 20.0 °C.

The fire was located on the floor in the centre of the room. The fire fuel consisted of softwood (Pinea ecelsa) timber cribs nailed into 40mm x 40mm battens. The crib measured 2.4m in length, 2.4 m in width and 1.4 m in height.

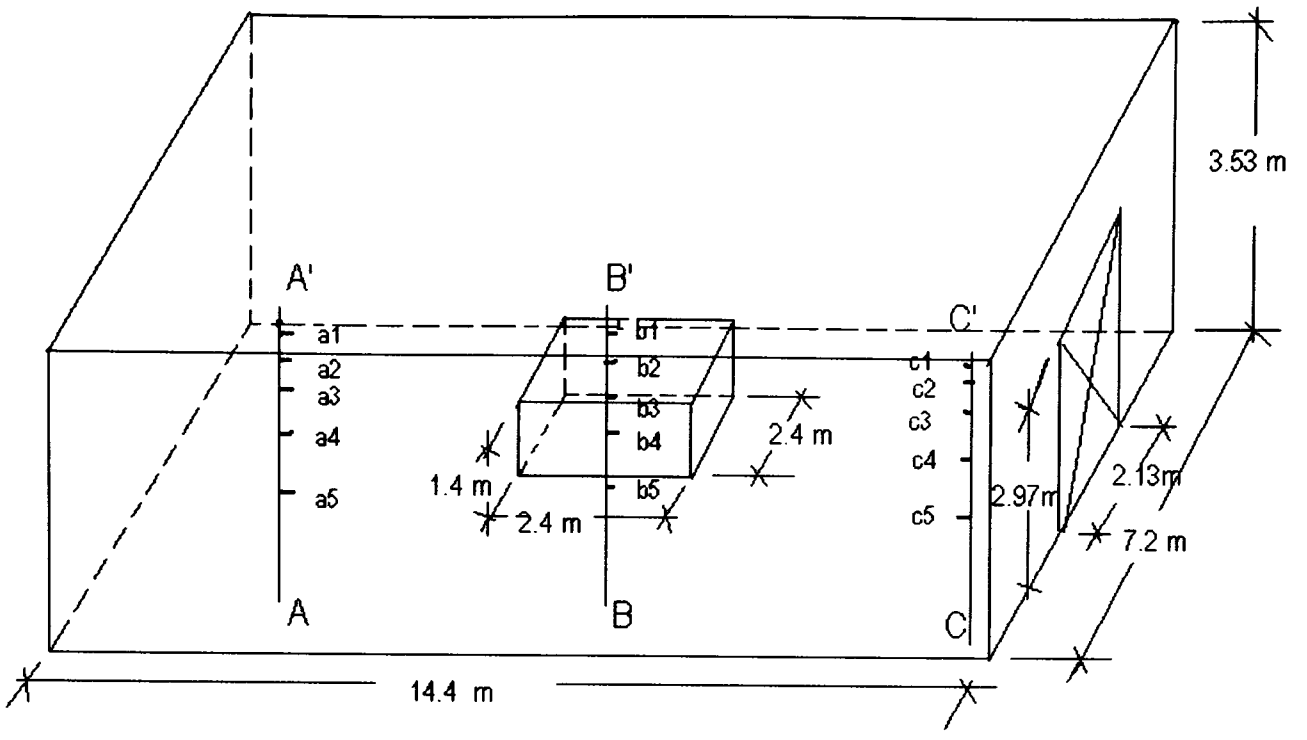


Figure 6-36 – Depiction of fire compartment geometry showing location of fire source

The heat release rate ($\dot{Q}$) is given by (5.5.3.1). The efficiency factor (χ) and heat of combustion (ΔH_c) were given as χ=0.7 and ΔH_c is 17.8 MJ/kg for burning wood with a 10% moisture content and the mass loss rate ($\dot{m}$) (kg/s) for the wood crib is presented in the table below. A maximum heat release rate of approximately 11 MW was produced. It is assumed that the fuel molecule is $CH_{1.7}O_{0.83}$.

Time (s)	0	60	120	180	240	300	360	420	480	540	600
Mass loss rate(kg/s)	0	0.005	0.004	0.009	0.013	0.014	0.019	0.033	0.052	0.08	0.207

The case was run assuming that all the walls were adiabatic and were completely reflecting (emissivity = 0.0). The case was run for 600s of simulated time using 120 time steps of 5s.

See Appendix C.4 for further setup details

Comparisions between predicted and measured temperatures at termocouple stacks A, B and C are illustrated in Figure 6-37 - Figure 6-44. The locations of the thermocouple can be seen in Figure 5-7.

It can be seen that in all cases the temperature was overestimated by all the CFD codes. The differences may be due to errors in the experimentally determined mass loss rate. Cross comparing the SPs indicates that all were in reasonable agreement

with one another. Generally CFX produced the hottest results and PHOENICS the coolest for this test case.

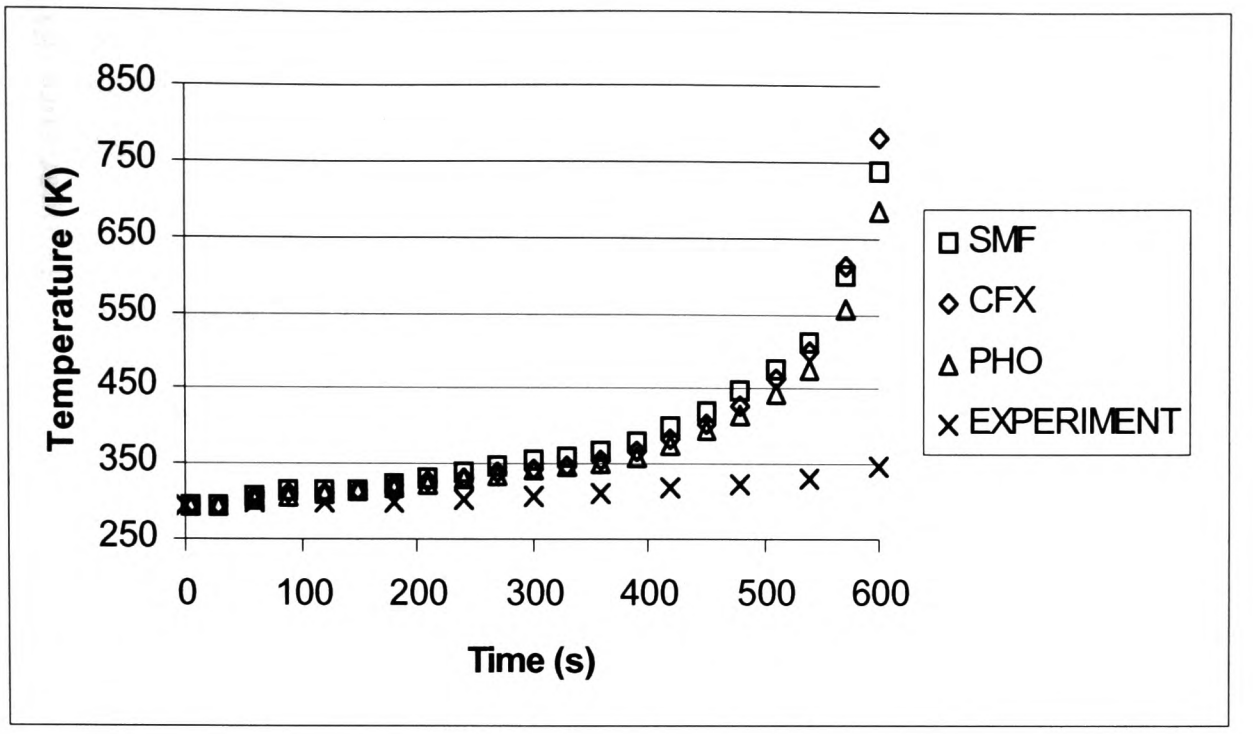


Figure 6-37 – Temperature history for Ta(1)

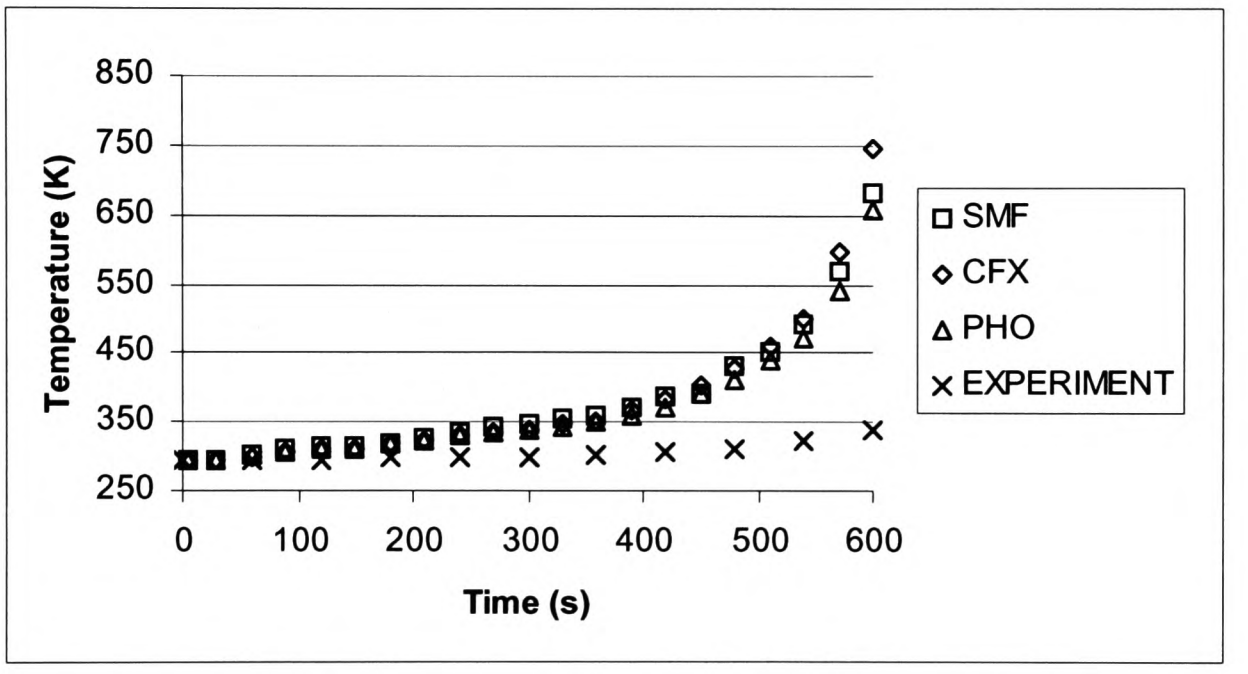


Figure 6-38 - Temperature history for Ta(3)

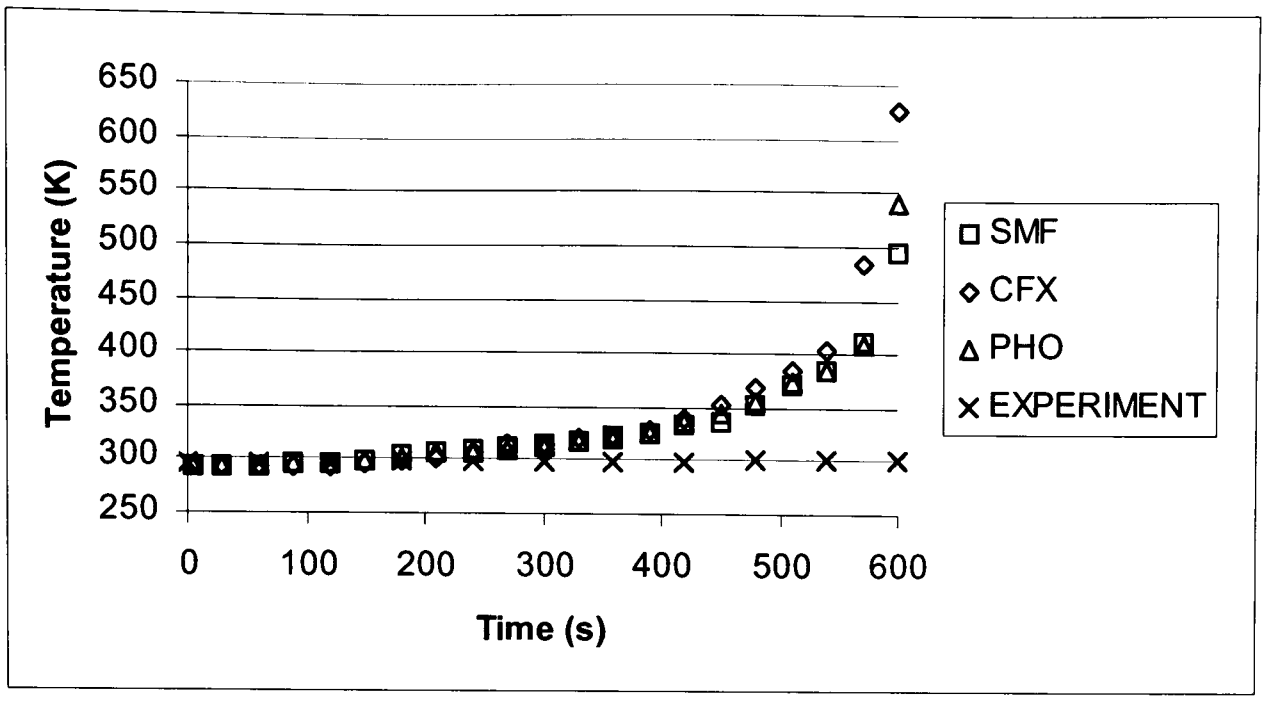


Figure 6-39 - Temperature history for Ta(5)

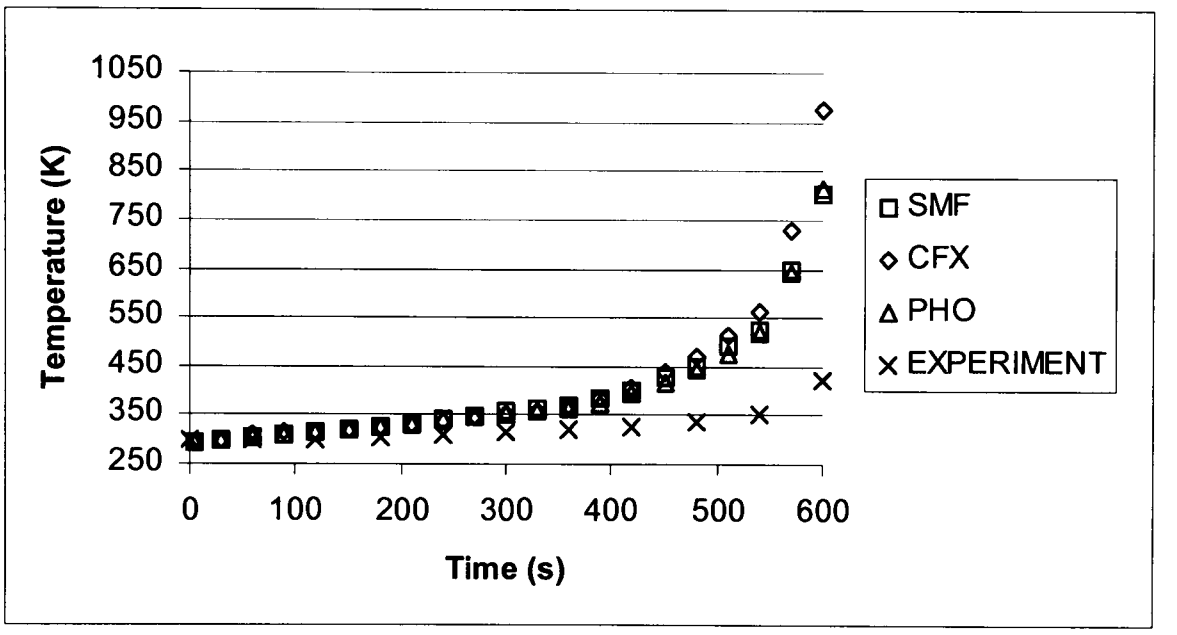


Figure 6-40 - Temperature history for Tb(1)

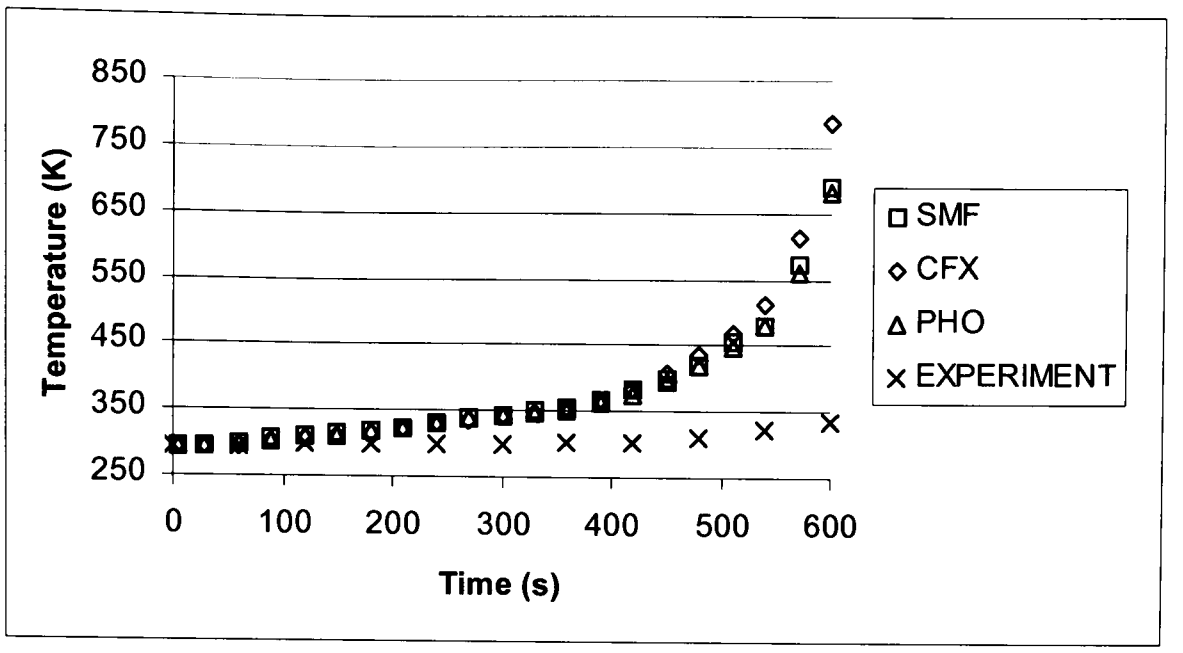


Figure 6-41 - Temperature history for Tb(3)

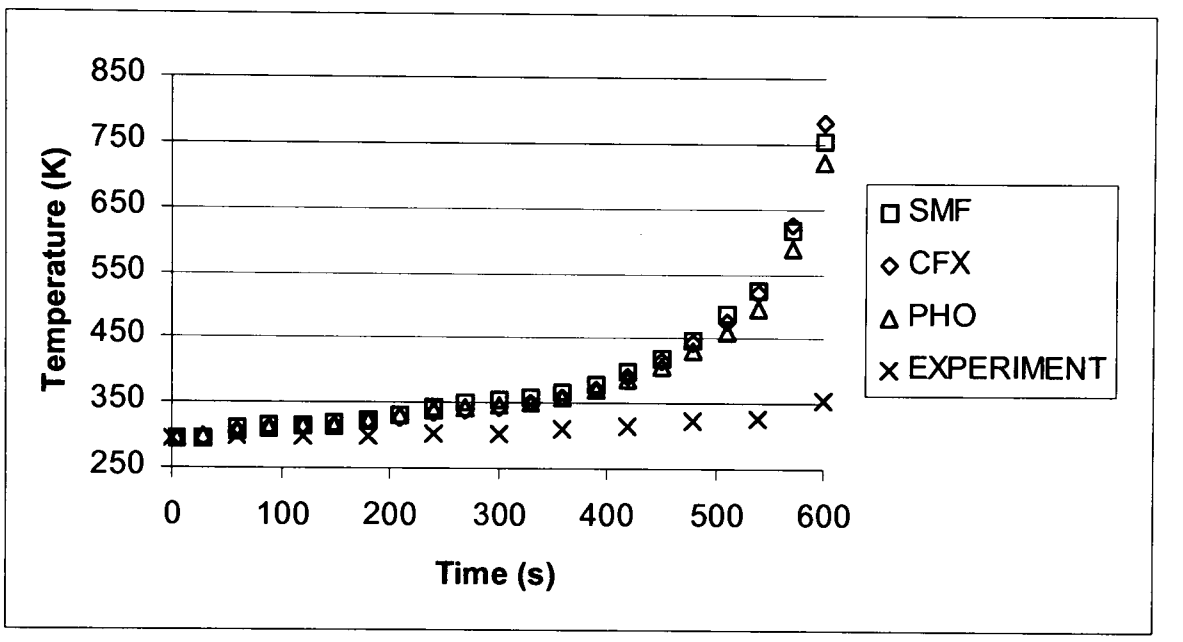


Figure 6-42 - Temperature history for Tc(1)

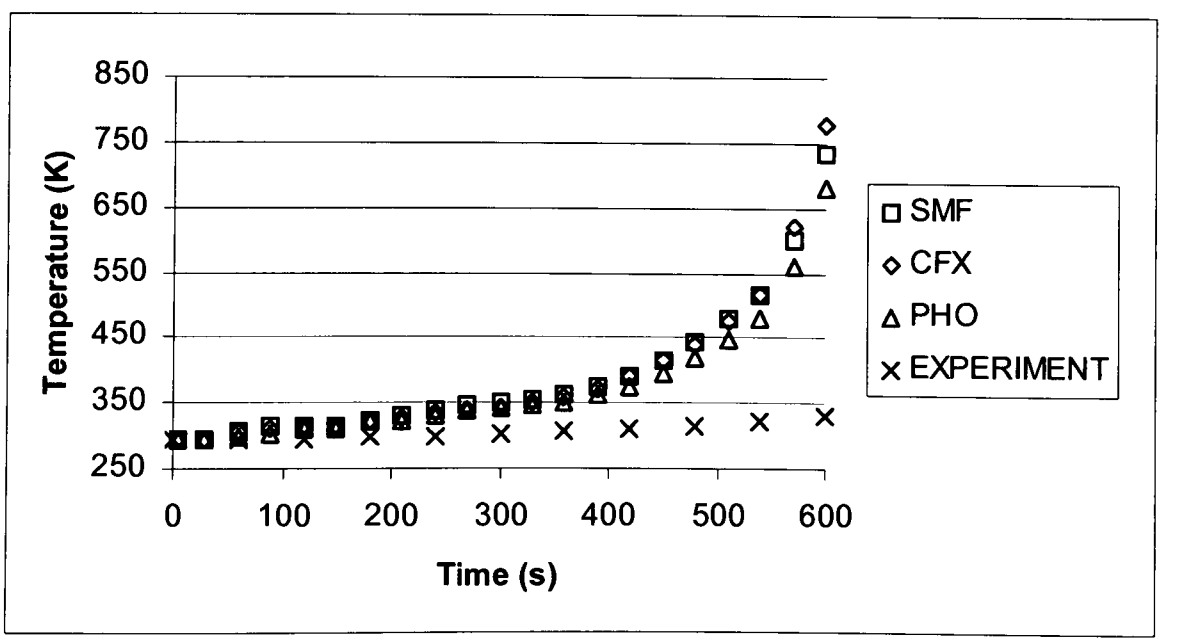


Figure 6-43 - Temperature history for Tc(3)

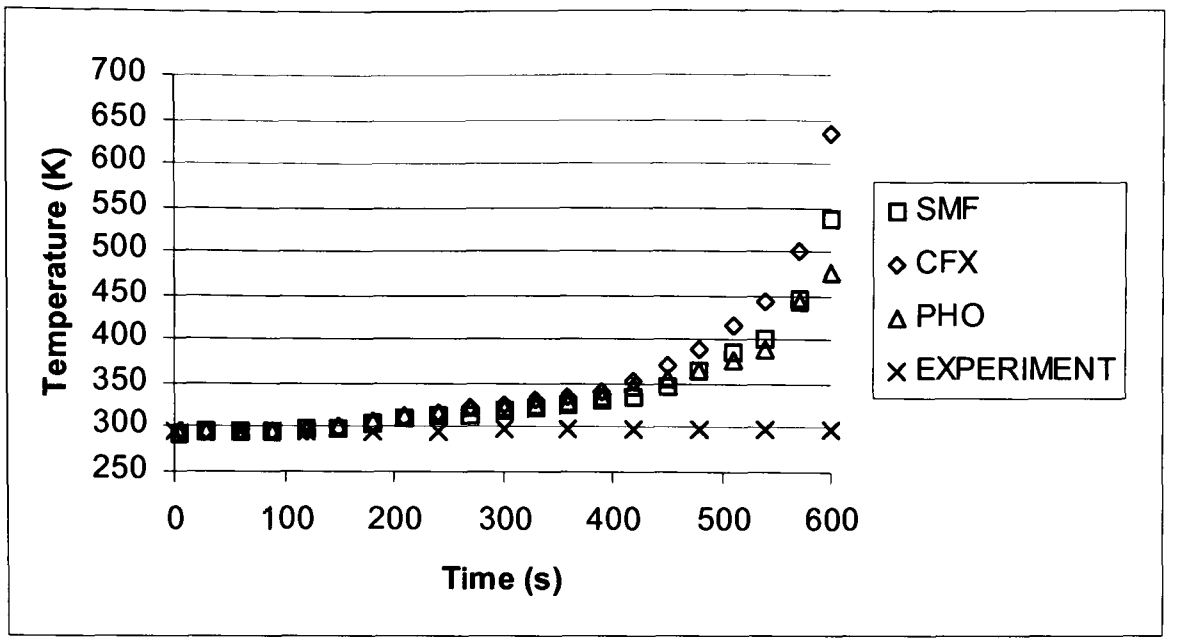


Figure 6-44 - Temperature history for Tc(5)

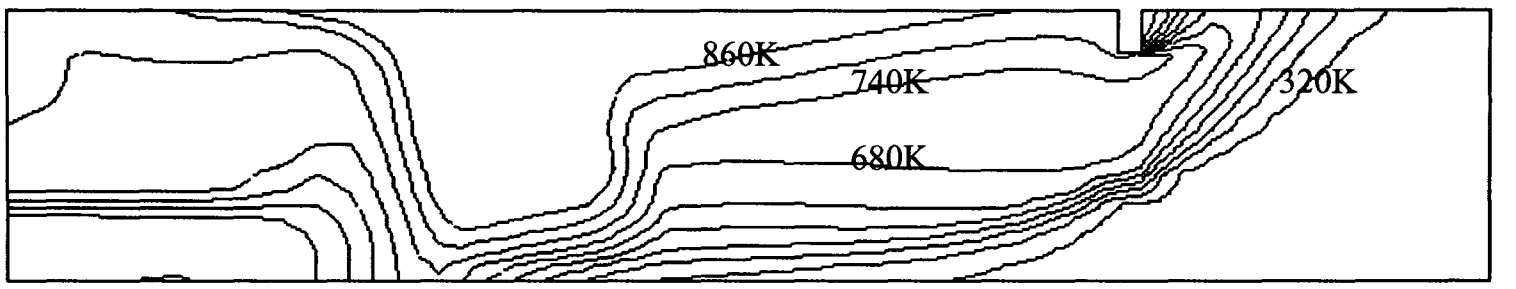


Figure 6-45 – CFX predicted temperature contour s through the vertical central plane.

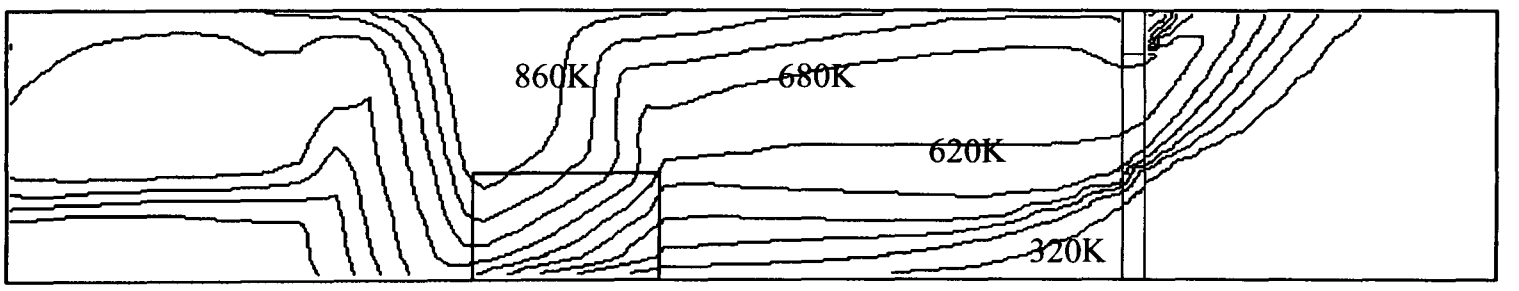


Figure 6-46 – PHOENICS predicted temperature contour s through the vertical central plane.

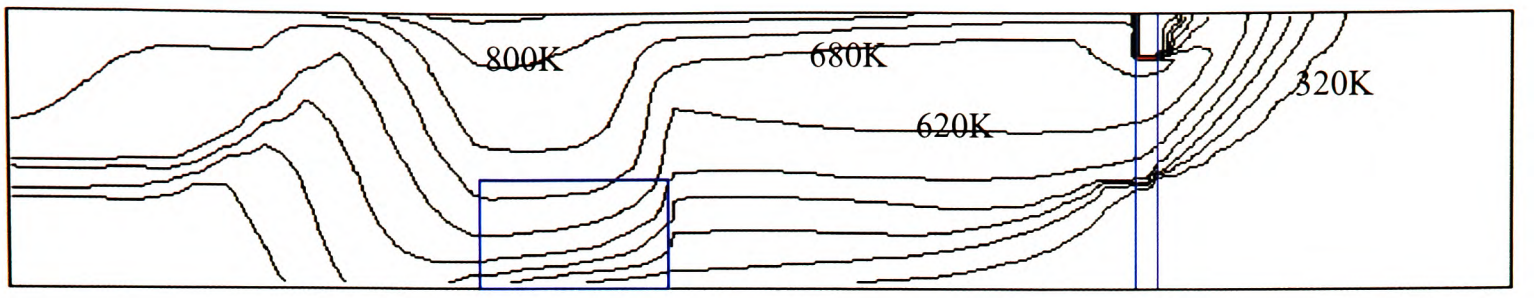


Figure 6-47 – SMARTFIRE predicted temperature contours through the vertical central plane.

Depicted in Figure 6-45 to Figure 6-47 are temperature contours along the centre vertical plane at 600 seconds into the simulation as generated by the three SPs. The temperature contours are separated by 60K and range from 320K to 860K. From these figures it is clear that the variation between the predictions made by the SPs were largest in the near field region above the fire source. In the far field region, temperatures were much closer together, particularly when comparing SMARTFIRE against PHOENICS. CFX produced slightly hotter results that may be due to the use of the non-standard usage of the CFX radiation model with one-ray. It was also expected that the results would be overpredicted due to the use of 1) adiabatic walls and 2) perfectly reflecting walls.

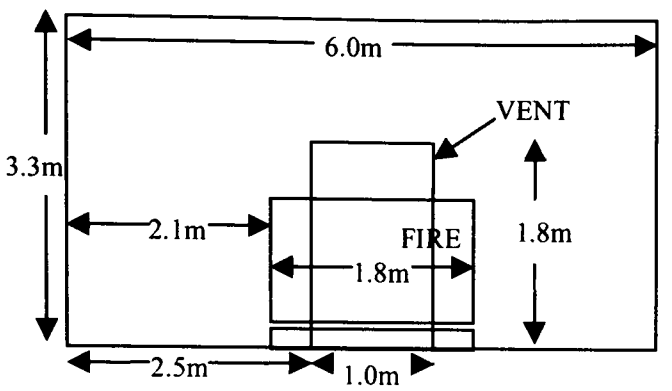
With all the SPs it can be seen that the plume leans away from the doorway.

Due to time constraints it was not possible to run the SPs for a longer simulation time. This would have proved useful as additional experimental data was available for comparison purposes. In addition, it would be interesting to compare the maximum temperatures predicted by the three SPs.

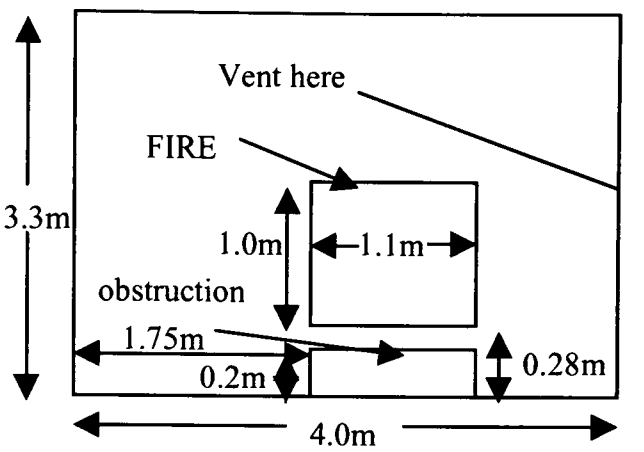
6.1.2.4 2000-2-5 – LPC007 case

This test case arises from a fire test conducted by the Loss Prevention Council (LPC) [GAC1997]. The test is a burning wood crib within an enclosure with a single opening. The test compartment is illustrated below and had a floor area of 6m x 4m and a 3.3m high ceiling. The compartment contained a doorway (vent) measuring 1.0m x 1.8m located on the rear 6m x 3.3m wall. The walls and ceiling of the

compartment were made of fire resistant board (Asbestos) which were 0.1m thick. The floor was made of concrete.



y-z view



x-y view

The heat release rate ($\dot{Q}$) is given by (5.5.3.1). The efficiency factor (χ) and heat of combustion (ΔH_c) were given as χ =0.7 and ΔH_c is 17.8 MJ/kg for burning wood with a 10% moisture content and the mass loss rate ($\dot{m}$) (kg/s) for the wood crib is presented in Table 5-1. It is assumed that the fuel molecule is $CH_{1.7}O_{0.83}$.

Table 6-6: Mass Loss rate for LPC fire test case.

Time(s)	0	150	450	460	1650
$\dot{m}$ (kg/s)	0	0.01835	0.18636	0.1978	0.1978

See Appendix C.5 for further setup details

The results for the plume thermocouple and room corner thermocouple stack for the first 300s are shown in Figure 6-48 and Figure 6-49. The lower (L) and higher (H) values refer to measurements at 1.5m and 3.0m above the ground respectively. The corner thermocouple stack is located at 0.57m away form the side wall and 0.5m away from the front wall containing the vent. The plume temperature measurements were taken at 3.0m away from the side wall and 2.392m away form the back wall of the compartment.

This test case proved problematic due to the high temperatures involved and the limited compartment ventilation. From the experimental data, the compartment achieves a flashover between 150 and 450 seconds. Numerically, all the codes were predicting very high temperatures within the first 300 seconds. The temperatures

predicted indicate that by 300 seconds the compartment had reached flashover conditions.

After 300 seconds, it was not possible to achieve well-converged solutions for any of the SPs and so all the simulations were terminated at this point. The simplistic and artificial nature of the boundary conditions used in this case was thought to contribute to the premature development of the flashover and the poor convergence characteristics. The walls were treated as adiabatic and radiatively reflective which resulted in large amounts of heat being retained within the compartment.

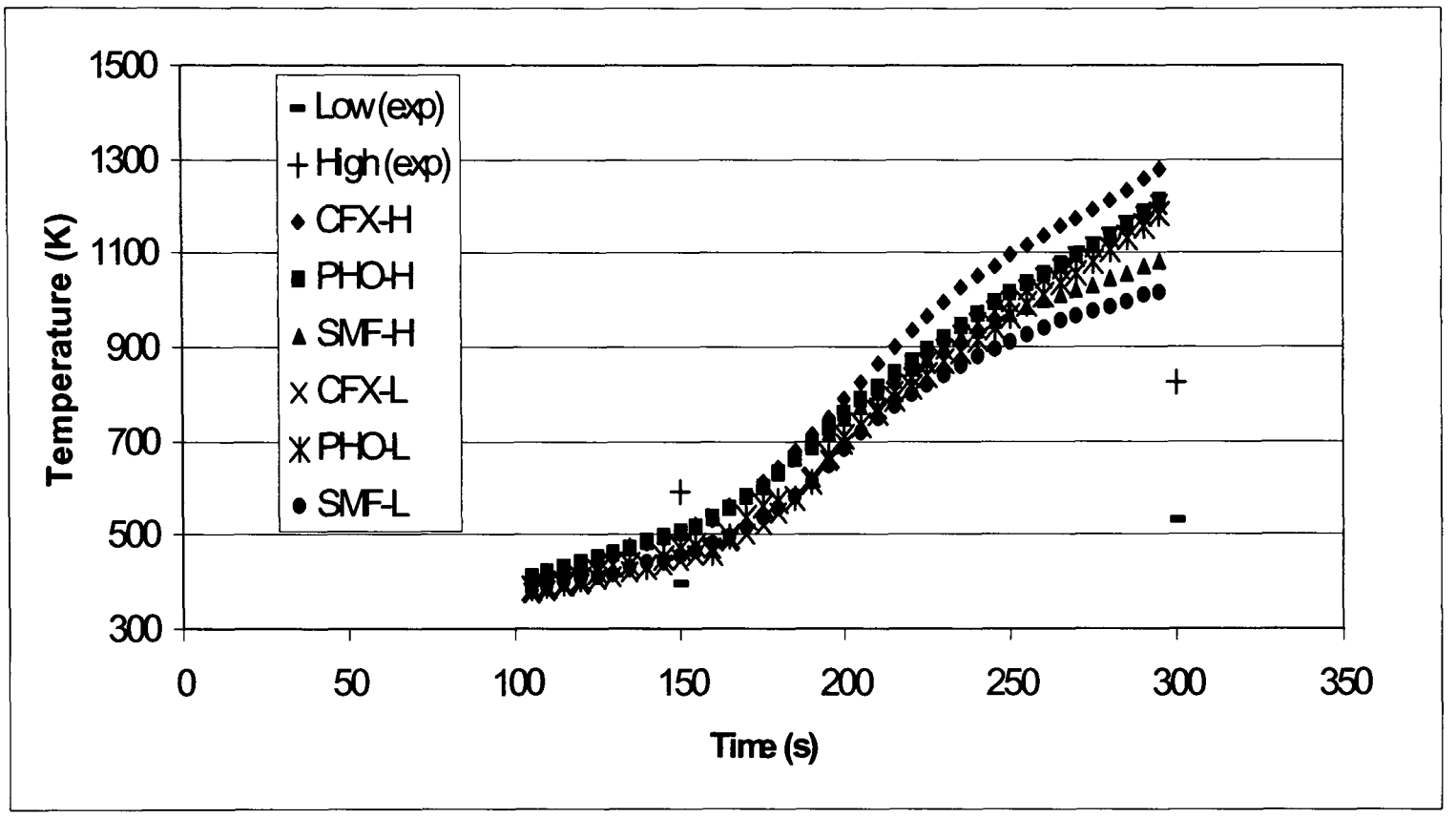


Figure 6-48: Predicted and measured Corner Stack Temperatures at 1.5m (L) and 3.0m (H) above the floor for the LPC test case.

Up to approximately 200 seconds there was good agreement between all the SPs for both the corner stack and plume predictions. At 150 seconds, the SPs appear to under predict the higher temperatures and over predict the lower temperatures. The predicted level of stratification thus appears to be less than that suggested by the experimental results.

After approximately 200 seconds differences between the predictions generated by the various SPs begin to appear and all of the SPs seriously over predicted the experimental results. The plume temperatures were difficult to assess as the movement of the plume can have a significant effect on the value whether

experimental or predicted. One consistent feature produced by the SPs is that the lower predicted temperature is consistently hotter than the upper predicted temperature. This trend was not observed in the experimental results. One difference between the experimental setup and the simulations is that the burning wooden crib that would have caused an obstruction to the flow which is not modelled and may cause significant differences in the near field region of the fire. With all the SPs the hottest temperatures were at the lower point as the hot gases cool as they leave the plume. In the experiment it is possible that the combustion process is occurring higher in the compartment which could be attributable to the obstructing effect of the crib on the oxidant flow.

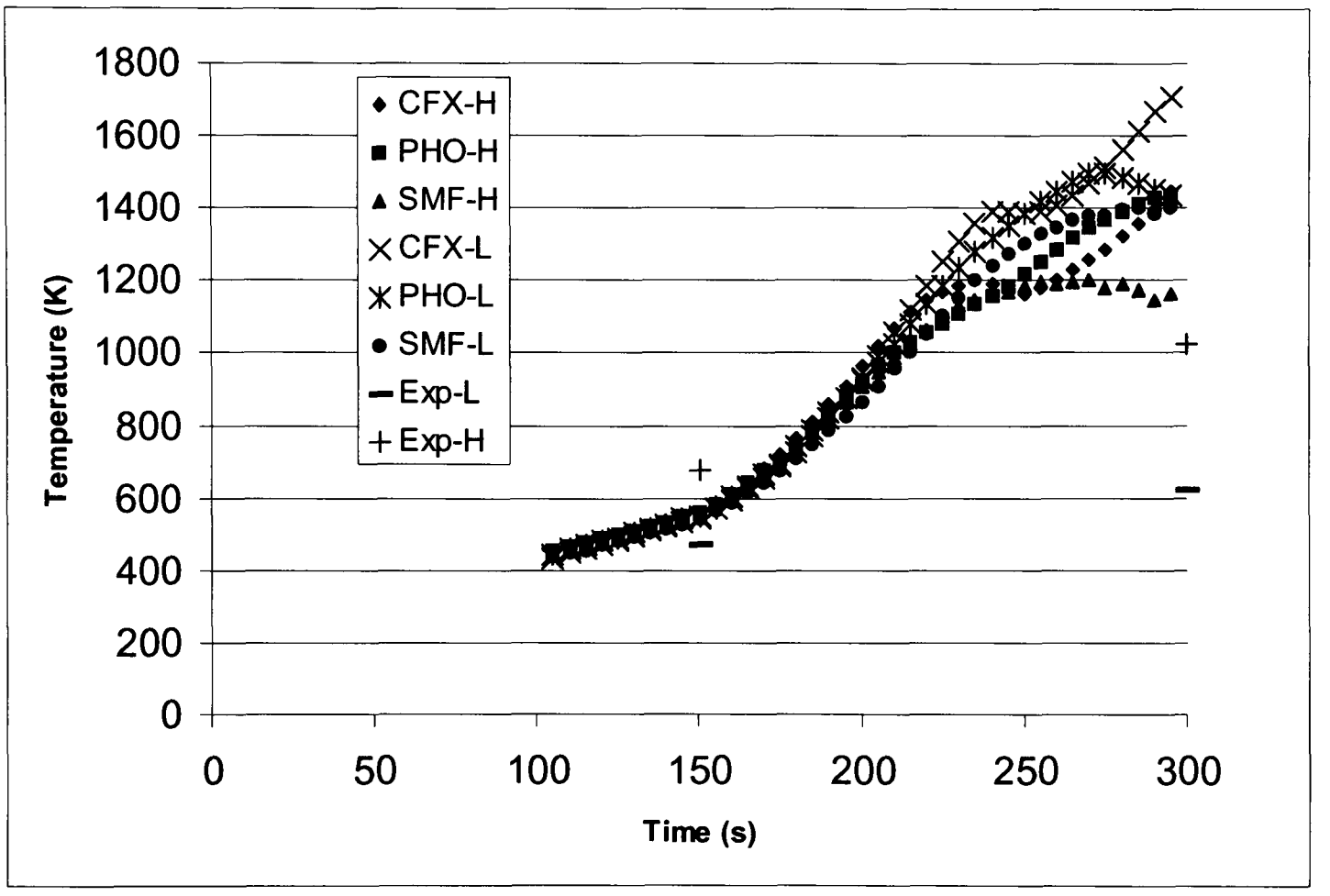


Figure 6-49: Predicted and measured Plume temperatures at 1.5m (L) and 3.0m (H) above the floor for the LPC test case.

The differences between the observed and predicted results may be due to the artificial nature of the boundary conditions used in this benchmark case. This will be examined further in the Phase 2 analysis.

In the figures below it can be seen that SMARTFIRE (Figure 6-50) and CFX (Figure 6-51) produce plumes that lean towards the window. However the PHOENICS (Figure 6-52) plume leans over to a much lesser degree. From these figures it is clear

that the temperature measurement at the lower level for the SPs was greater than the temperature measurement at the higher level. From these figures it can be further seen that CFX had the hottest plume followed by PHOENICS with SMARTFIRE being the coolest. It is possible that the use of the one-ray radiation model in the CFX simulation contributed to the higher temperature prediction.

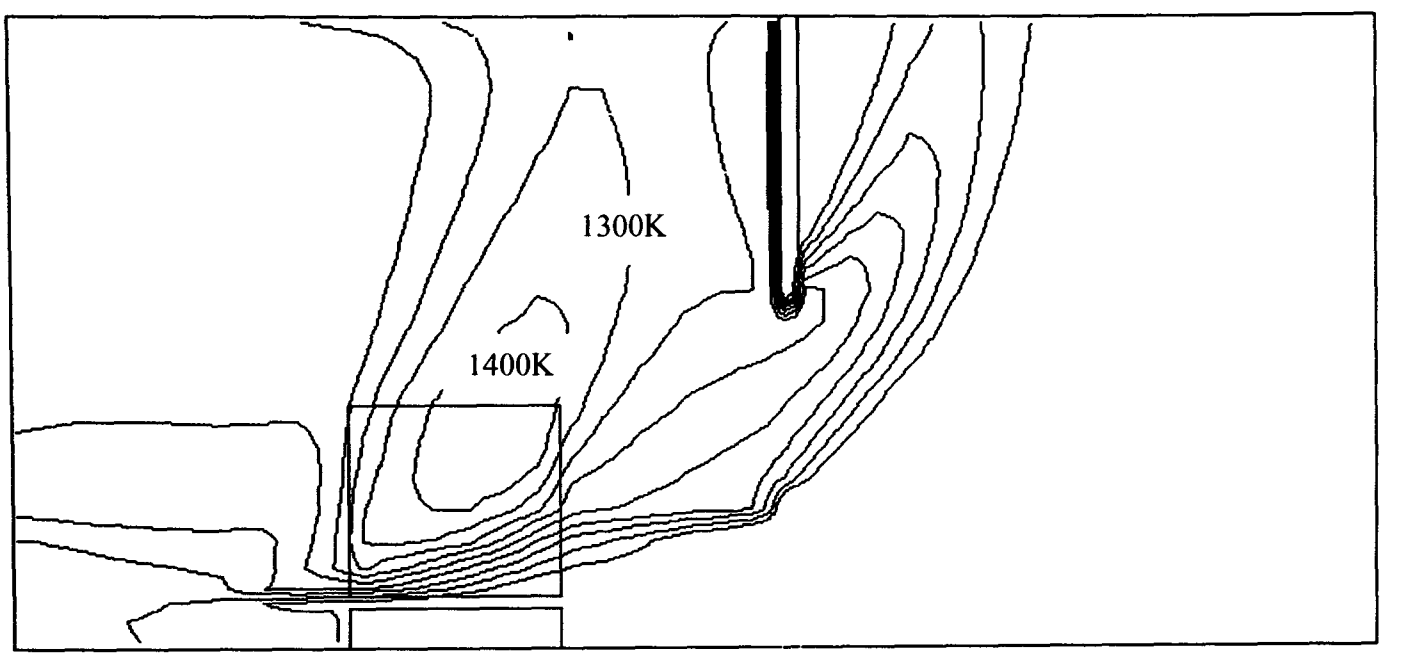


Figure 6-50 - SMARTFIRE plume at 300s

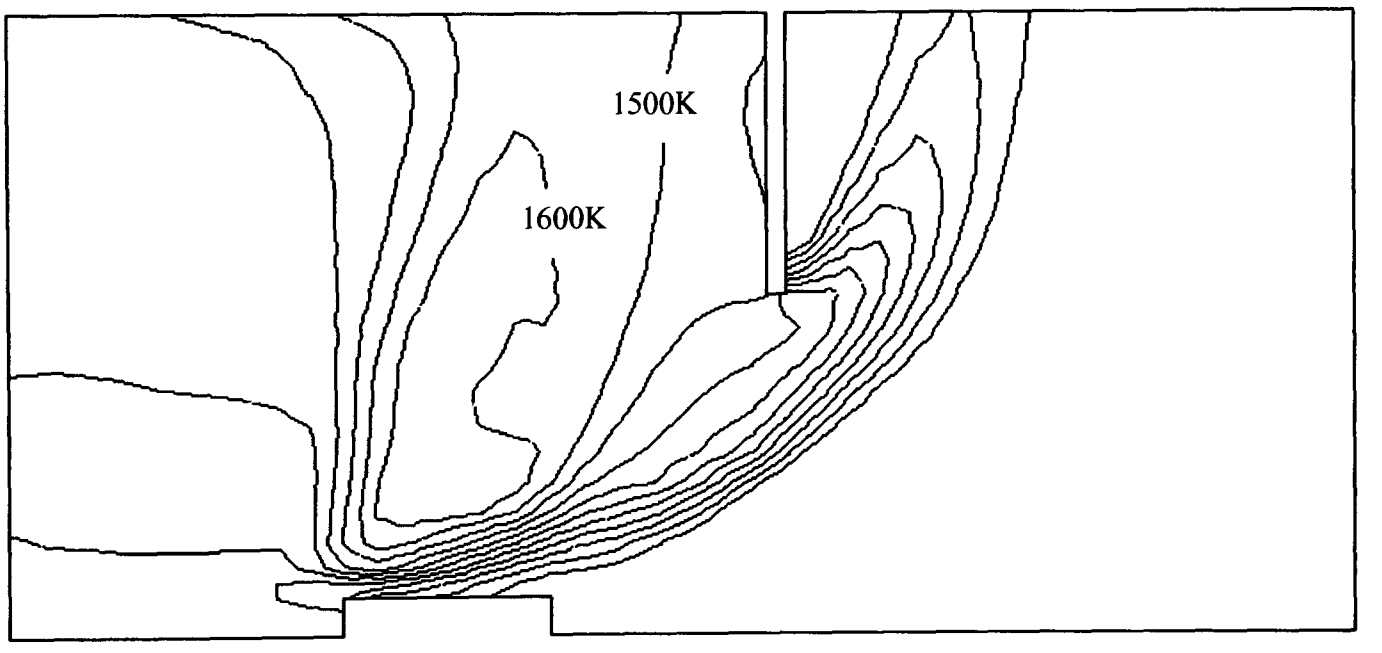


Figure 6-51 - CFX Plume at 300s

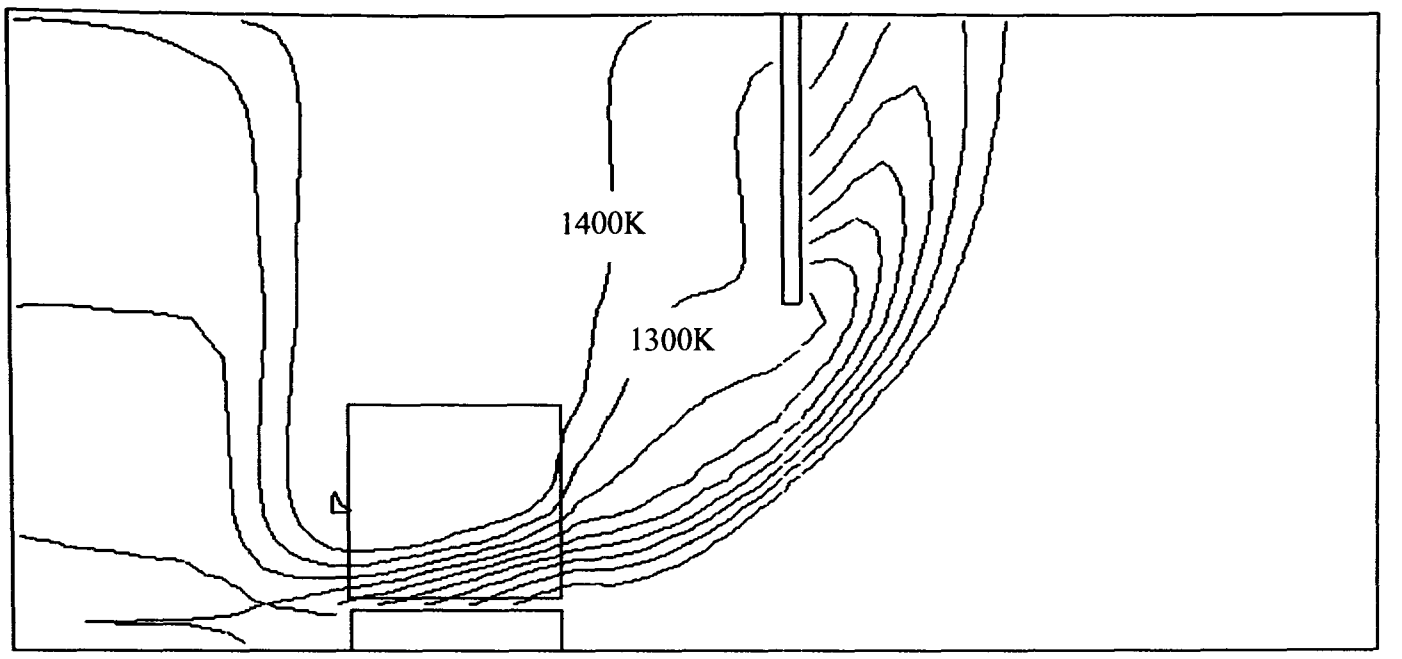


Figure 6-52 - PHOENICS plume at 300s

6.1.3 GENERAL DISCUSSION of phase 1

In studying the outcome of the Phase 1 test cases, it was clear that when identical physics was activated, identical computational meshes used and similar convergence criteria applied, all of the software products tested were capable of generating similar results. This is an important observation and suggests – that within the limitations of the tests undertaken – that the three codes had a similar basic capability and were capable of achieving a similar basic predictive standard.

The results from the CFD test cases were consistent with the view that the basic underlying physics implemented within the codes are similar and are capable of producing similar representations of the physical phenomena modelled. In addition, where experimental results or theoretical solutions were available, the software products produced reasonable agreement with those results. No doubt, it could be argued that improved agreement could be achieved if the spatial mesh and time stepping or better physical models and boundary conditions were used. This may be demonstrated in the Phase 2 simulations.

The one area that showed relatively poor agreement with theoretical results concerned the radiation model performance. The six-flux radiation model used by SMARTFIRE and PHOENICS produced very similar results however, they displayed significant differences to the theoretical results. While the six-flux model appears capable of representing the average trends within the compartment, it does not produce an

accurate representation of local conditions. The CFX radiation model when used with a single ray (the closest approximation to the six-flux model possible but not mathematically equivalent) displays a more significant weakness and severely under predicts the emissive power in the cavity. It should however be noted that the producers of CFX do not recommend that the discrete transfer radiation model be used with so few rays. The radiation model used by CFX is inherently a more sophisticated model then the six-flux model and is capable of utilising more rays.

It should be recalled that the purpose of the Phase 1 test cases was to compare the performance of the various codes when similar physics capabilities were utilised in all three codes. It should however be noted here that when 12 rays are used in the CFX radiation model, it produces very good agreement with the theoretical results. It is clear from these results that users should be aware of the limitations of the six-flux model when performing fire simulations. Situations that are strongly radiation driven, such as the prediction of flame spread over solid surfaces, or structural response to fire should be treated with care. When using the six-flux model, it is possible that target surfaces would not be preheated by radiation to the extent that would otherwise occur, thereby slowing the flame spread process or unreliably predicting structural response.

The fire cases were intended to provide a more challenging series of tests. Unlike the simple CFD test cases, the fire cases make use of a range of CFD capability. Furthermore, they focus attention on the software's capability within the specific domain of interest i.e. fire modelling.

The first two fire cases consisted of the small non-spreading fire within the small ventilated compartment modelled using heat source and gaseous combustion model. For these cases, all the software products appear to produce a good representation of the measured temperature distribution within the compartment and velocity profile within the doorway. Furthermore, there are insignificant differences between the temperatures predicted by heat source model and gaseous combustion model. However, all the software products appear to slightly over predict the hot layer temperature. This over prediction is likely to be due to the simple specification of the conditions required in phase 1. No doubt, it could be argued that improved agreement

could be achieved if more sophisticated physics were used in the simulations. This may be demonstrated in the Phase 2 simulations. It should however be pointed out that the fire in this case is quite small and so radiative heat transfer does not play a significant role in this situation.

The third fire case consisted of a fire – represented by a prescribed heat release rate – centrally located in the open compartment. While there were no experimental results for comparison purposes, it was clear that all three software products produced near identical results.

The forth fire case consisted of a large fire in a medium sized compartment which was well ventilated. The fire was modelled using a prescribed mass release rate in conjunction with a gaseous combustion model. Here again all three software products produced good agreement when compared with each other. However, towards the end of the simulation period, there was a significant difference between the predicted and measured temperatures. This is thought to be due to problems with the experimentally determined heat release rates. Had time permitted, it would have been interesting to continue the numerical predictions for a longer period of time to compare the maximum temperatures produced by the various codes.

The fifth fire case consisted of a large fire in a small sized compartment which was under ventilated. The fire was modelled using a gaseous combustion model. Here again all three software products produced good agreement when compared with each other in the early phases of the fire development. However, towards the end of the simulation period, there was a significant difference between the predicted and measured temperatures and between the predictions produced by the various software products. This is thought to be primarily due to the simplicity of the boundary conditions imposed on the calculations resulting in very high temperatures being generated within the compartment. It is also worth noting that all the simulations had to be prematurely stopped due to convergence difficulties. This test case will be examined further in Phase 2 using more representative boundary conditions.

The results from the fire cases support the conclusions drawn from the CFD test cases. While there are minor differences between the results produced by each of the

software products; on the whole they produce – for practical engineering considerations – identical results.

The completion of Phase 1 has highlighted several areas in which improvements can be made to both the procedures used and the test cases examined.

It is suggested that once the test case has been specified at a high level by the BTG, the test case input files should be set up by each of the participating SP developers. These should then be checked by the BTG to ensure that they conform to the standards of the benchmark. In this way, the test case input files would be optimised for the particular SP within the guidelines set down by the BTG. While the representatives of the BTG that conducted the assessments (i.e. FSEG) may have expertise in all of the SPs utilised in this study, it is unlikely that they will have sufficient expertise in all of the products likely to be tested. While this places pressure on the participating software producer to generate the input files, if the benchmarking procedure becomes a recognised standard, code vendors will be prepared to participate at this level.

In addition, once a version of a SP is entered into the benchmarking process, all the test cases must be run with that version. If another release version of the SP is produced, this to will need to go through the benchmark process in its entirety. However, a mix and match process in which different versions of a code are used in order to improve the level of agreement should not be permitted.

With regard to the benchmark cases utilised in the current procedure, several improvements can be suggested for the fire cases.

Fire case 2000-2-4 was run for 10 minutes of simulation time. Although all the SPs exhibit the same growing trend and similar temperatures it would be useful to run the case for a longer time period. This could be compared with the experimental results in order to determine the differences between maximum predicted and maximum measured temperatures.

Fire case 2000-2-5 proved difficult to obtain converged predictions due to the artificial nature of the boundary conditions utilised in Phase 1. This case was also complicated as flashover occurs and the fire becomes ventilation controlled. While it is necessary in Phase 1 to select a set of "simple" boundary conditions that can be represented by most SPs, another choice of boundary conditions would be appropriate. It is possible to run this case with a fixed wall temperature with unit emissivity. It must be noted that these boundary conditions are just as unrealistic as the adiabatic boundary conditions used in Phase 1. However, this would have the effect of artificially removing a large amount of heat from the compartment and may allow the simulation to run for longer. It was important that the cases were modelled using the original phase-1 specification blindly for all the SPs to prove the concept of using the rigid definition. It was found that all the models failed at a similar point in the simulation. It is likely that this case would need to be at least modified in any future exercise to obtain more meaningful results.

6.2 Phase 2 results

This section contains the results from the Phase-2 testing regime. The CFD and fire cases were designed to test the basic features of the SP to ensure that these functioned correctly. Only results from the SMARTFIRE SP are presented as this work was performed by the author and furthermore the other SPs did not participate in this phase due to commercial constraints.

In Phase-1, testing was designed to ensure that the codes were set up as similarly as possible. This included using the same computational mesh and physical models. In Phase-2, the participants were free to optimise the set-up of each of the test cases. This meant that the mesh could be refined and more sophisticated physics routines that were available within the codes could be activated. In addition, participants were free to select which of the test cases they wished to repeat. However, all software set-ups were reported so that they could be repeated. Details of the numerical set-ups for the phase 2 CFD and fire cases can be found in Appendix D.

6.2.1 CFD cases

The CFD cases were intended to test the fundamental physical modelling capabilities of the SPs.

On the whole, all the SPs performed well on the Phase 1 CFD test cases. The only case that showed room for considerable improvement was the radiation test case (2000-1-5). This case was attempted by SMARTFIRE as part of phase-2 of the assessment.

6.2.1.1 SMARTFIRE: 2000-1-5 Radiation in a 3D cavity.

6.2.1.1.1 Introduction

The primary purpose of this test case was to test the radiation model used by the SP. Model predictions were cross compared and also compared with theoretical predictions derived from detailed zone methods [Lar1983, Fiv1988].

The geometry used for this test case consists of a three dimensional unit cube (1m x 1m x 1m) cavity with three walls with planes x=1, y=0 and z =0 set to a unit emissive power and the three other walls set to zero emissive power. All the walls were considered radiatively black with unit emissivity and the fluid had a unit absorption coefficient. Scattering was neglected. No fluid flow was considered.

6.2.1.1.2 Phase 2 Model Configuration

From the results for phase 1 it was apparent that although the six-flux radiation model could produce good results for certain fire cases i.e. non-spreading fires, it was inadequate for other fire applications such as those involving fire spread. While the six-flux model appears capable of representing the average trends within a compartment, it does not produce an accurate representation of local conditions. As part of the phase 2 simulations, the SMARTFIRE multi-ray radiation model was tested to see if this would provide the user with better predictions. Whilst this model was implemented within the current release version of SMARTFIRE it has not been made available to general users via the GUI but can be activated via the INF file.

The multi-ray radiation model [Jia1999] is more advanced than the six-flux radiation model as the user may specify as many ray directions as is wished, allowing for a

model with a more realistic radiation distribution. The multi-ray model may also be used on unstructured meshes. A drawback of the method is that each ray direction requires a linear solver and as many as 24 ray directions may be required to produce a good radiation distribution.

The multi-ray radiation model was run using the following configurations;

1) 6-rays – which is equivalent to the 6-flux model with the rays directed in the co-ordinate directions
2) 24-rays
3) 48-rays

The rays were weighted and spread over 4π steradians so that the overall radiation distribution was conserved; the rules for doing this are described by Lathrop and Carlson [LC1965, Jia1999]. Further set up details are available in appendices D1-D3.

Phase 2 Results

The SMARTFIRE results were compared at three locations with the above setups (1-3), the theoretical zone model result and the SMARTFIRE six-flux results generated in phase 1.

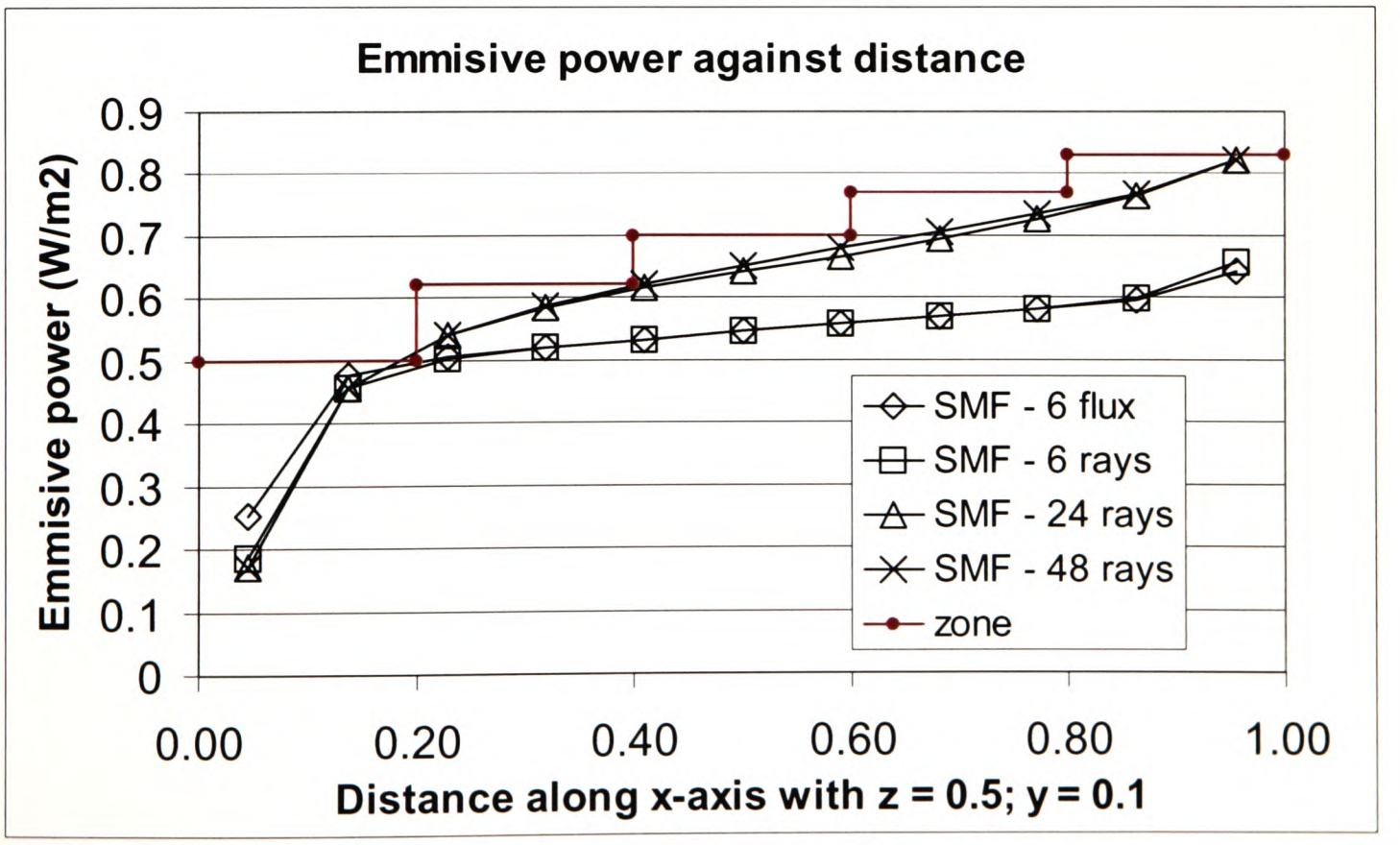


Figure 6-53 – SMARTFIRE generated emissive power against distance along x-axis for z = 0.5; y = 0.1 using six-flux and multi-ray radiation models

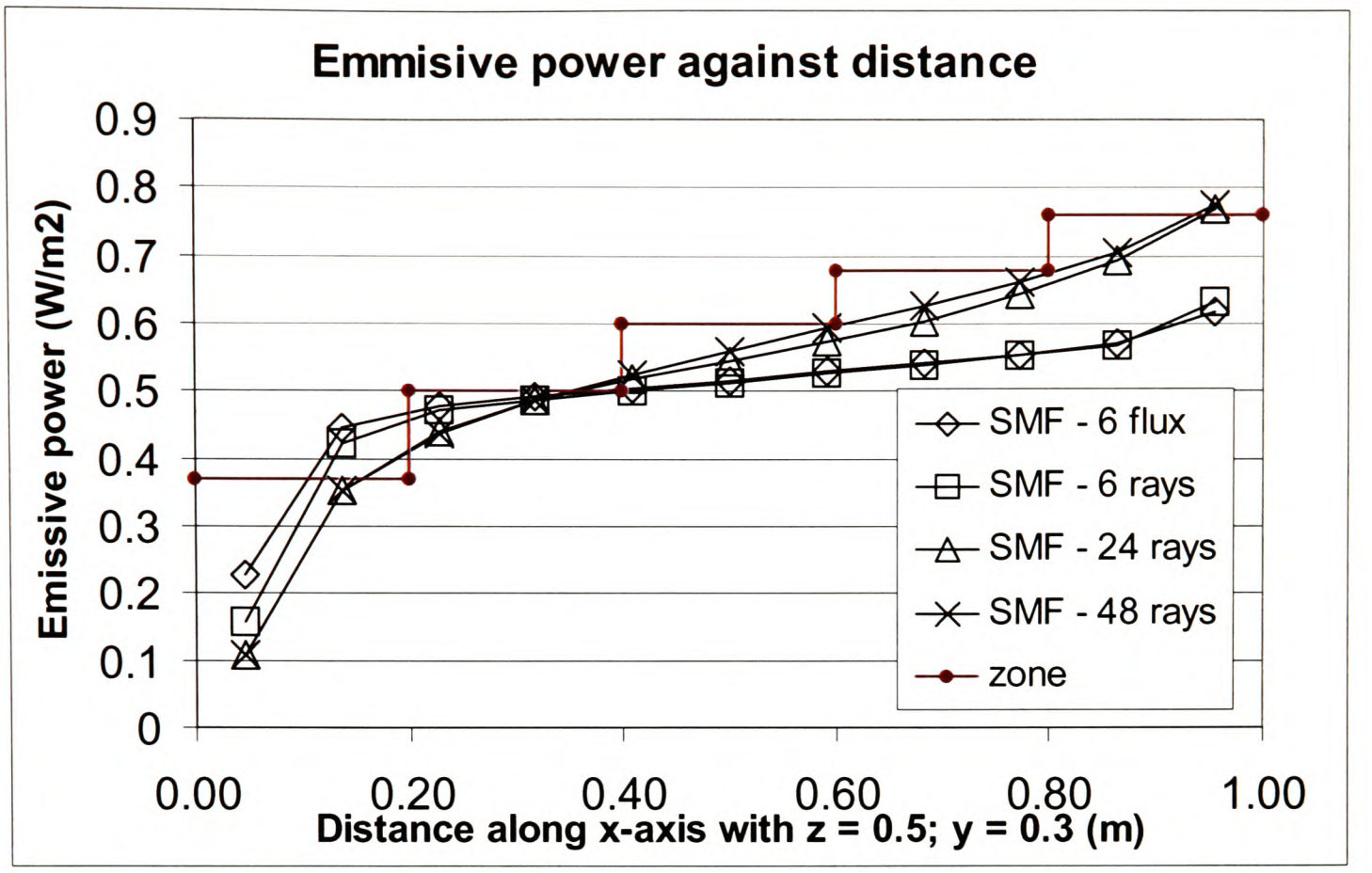


Figure 6-54 – SMARTFIRE generated emissive power against distance along x-axis for z = 0.5; y = 0.3 using six-flux and multiray radiation models

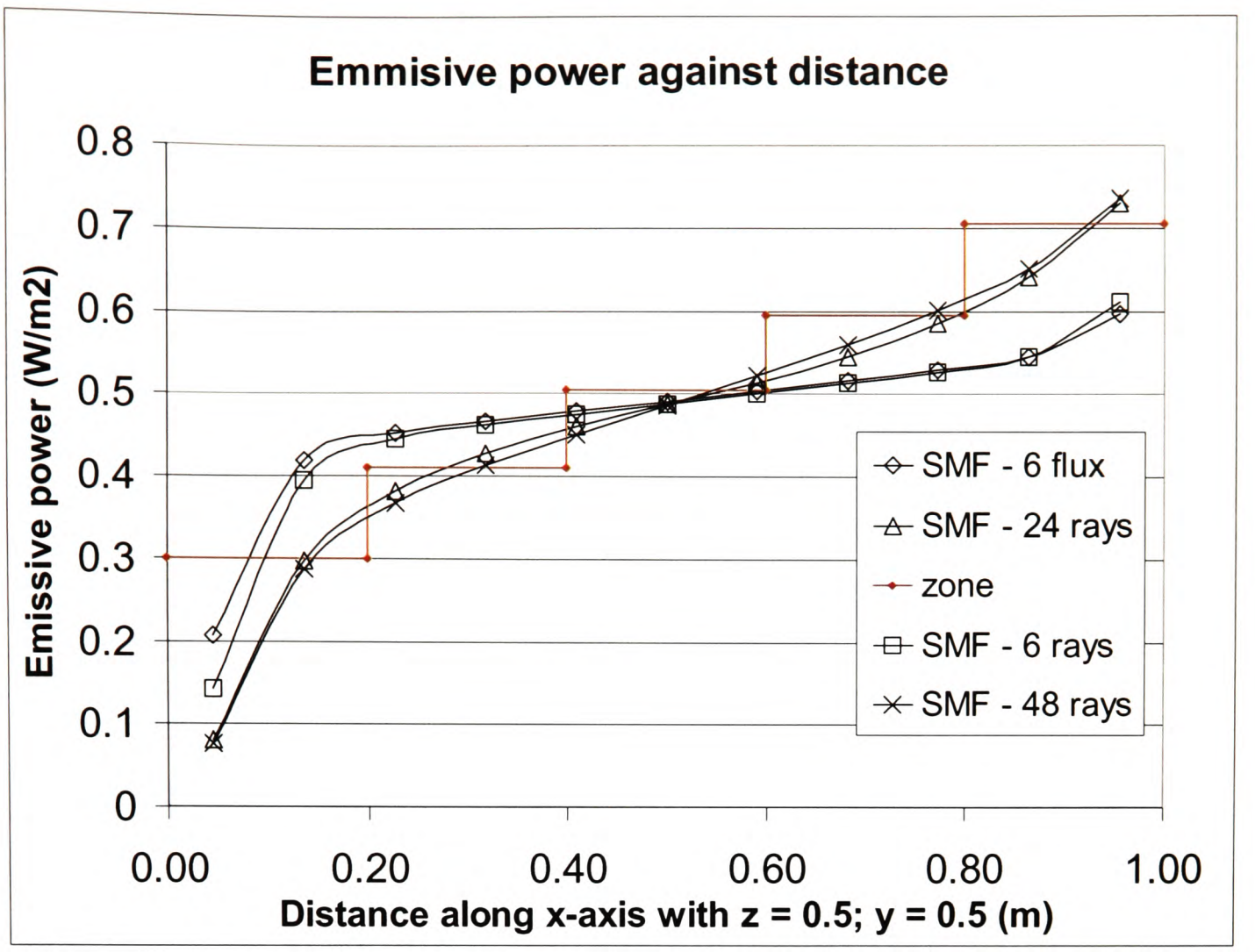


Figure 6-55 – SMARTFIRE generated emissive power against distance along x-axis with z = 0.5; y = 0.5 using six-flux and multi-ray radiation models

In Figure 6-53 to Figure 6-55 it can be seen that when the multi-ray model was configured with six rays it produced near identical results to the SMARTFIRE six-flux radiation model. As found in the phase 1 (section 6.1.1.5), these results only approximate the radiation distribution within the cavity. When the multi-ray model was configured with 24 and 48 rays, the model produced results much closer to the theoretical results. Furthermore, the results generated using 48 rays show only a marginal improvement over the results generated using 24 rays.

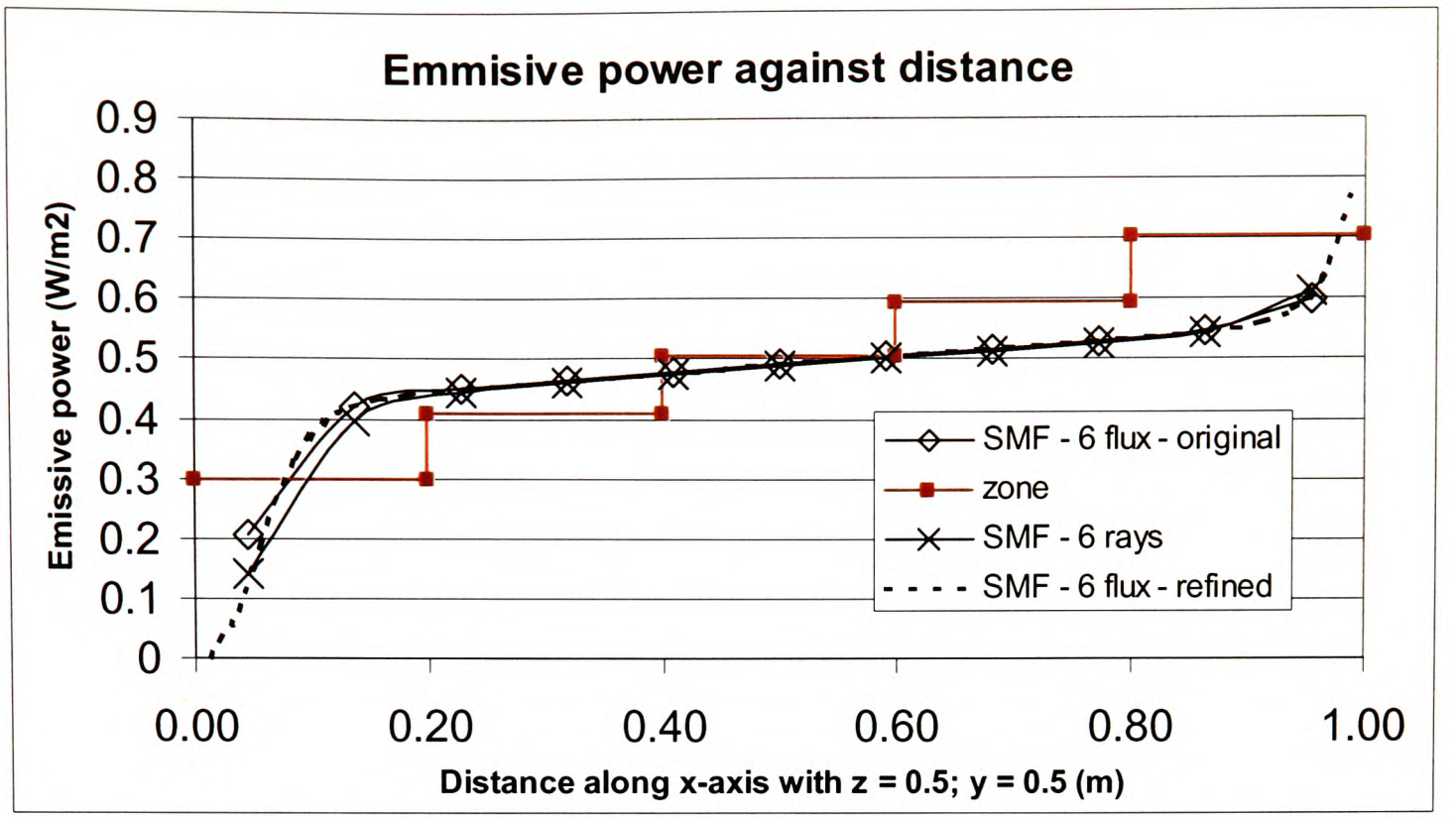


Figure 6-56 - Refined mesh for SMARTFIRE six-flux radiation model

The six-flux model was also tested using a much-refined mesh in order to investigate if mesh refinement would improve the predictive capability of the model. The mesh was refined to 41 x 41 x 41 and the case re-run with just the six-flux radiation model. Further set up details can be found in appendix D4. The results for this case are depicted in Figure 6-56. As can be seen, refining the mesh did not improve the results for the six-flux model. Furthermore the results were very similar to those found on the original mesh.

The results produced by the SMARTFIRE multi-ray model were compared with the results generated by the CFX discrete transfer radiation model (see section 6.1.1.5). In comparing the two results it is important to note the difference between the two models. The ray definition used by SMARTFIRE was different to that used by CFX. In SMARTFIRE, the specified number of rays was the number of rays emanating from a nodal point. However in CFX, the number of rays was the number of rays leaving each cell surface. Therefore CFX with 1 ray was approximately equivalent to SMARTFIRE with 6 rays and CFX with 12 rays was approximately equivalent to SMARTFIRE with 72 rays. Also, while the CFX radiation model makes use of a different computational mesh to that used in the flow calculations, the SMARTFIRE multi-ray model used the same computational mesh for both radiation and flow calculations.

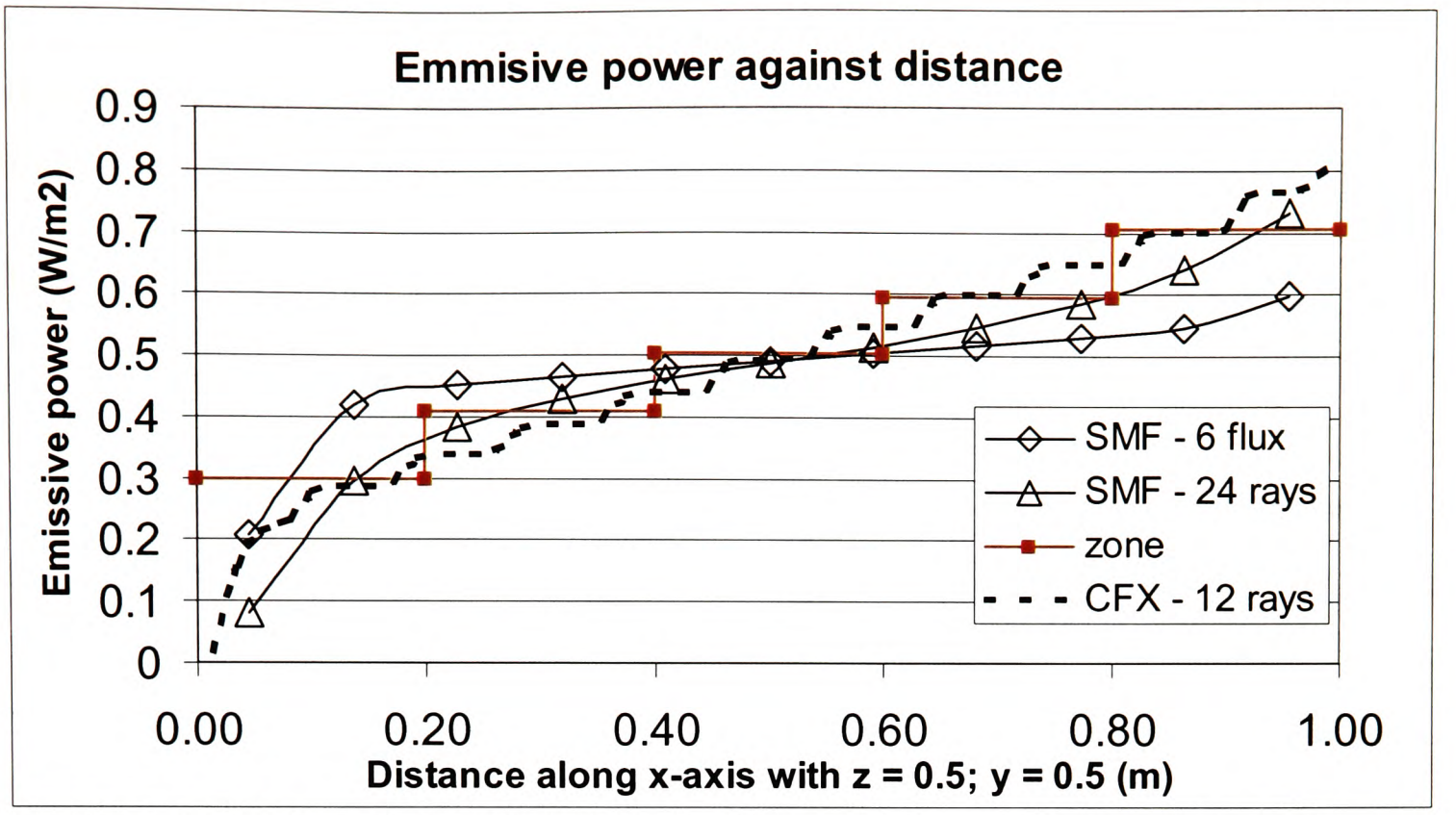


Figure 6-57 - Comparison of SMARTFIRE radiation models with CFX radiation model

As can be seen from Figure 6-57, both the CFX and SMARTFIRE radiation models produce a good comparison with the theoretical zone model result. The stepping noted in the CFX results was a consequence of the differences between the CFD mesh and the radiation mesh.

In conclusion, the SMARTFIRE multi-ray radiation model produced better agreement with the theoretical results than the standard six-flux model. It was also apparent that the quality of the results were dependent on the number (and direction) of rays used. When using six rays directed along the co-ordinate axes, the multi-ray model produced identical results to the SMARTFIRE six-flux model. For this particular problem, using 24 rays produced similar results to 48 rays but at much reduced computational cost, approximately half the time in this instance.

While the multi-ray radiation model was computationally more expensive – hence less desirable - than the six-flux model, in situations where radiation plays a key role such as in the modelling of spreading fire, it is essential.

6.2.2 Fire cases

Several of the fire test cases completed in Phase 1 showed room for improvement. All of the SPs tested could have improved their predictions through the use of a combination of refined meshes, activation of more sophisticated sub-models and the

use of more realistic boundary conditions. Two fire cases were submitted for Phase 2, these were the Steckler fire case (2000-2-1) and the LPC007 fire case (2000-2-5). Both of these cases were attempted by the author using the SMARTFIRE SP.

6.2.2.1 SMARTFIRE: 2000-2-1 – Steckler fire case

6.2.2.1.1 Introduction

This test case (2000-2-1) was simulated with the prescribed heat release rate model. The volumetric heat source was chosen as there was little difference between the results generated by the combustion model and the volumetric heat source model in Phase-1 (see section 6.1.2.1). The Steckler case is a standard fire model test case that has been used by a number of field and zone model developers. Its primary purpose was to test the fire models predictive capability in predicting temperature and flow distributions in a small compartment subjected to a steady non-spreading fire. Predictions of several parameters were made and cross-compared. Model predictions were also compared with experimental results [SQR1982].

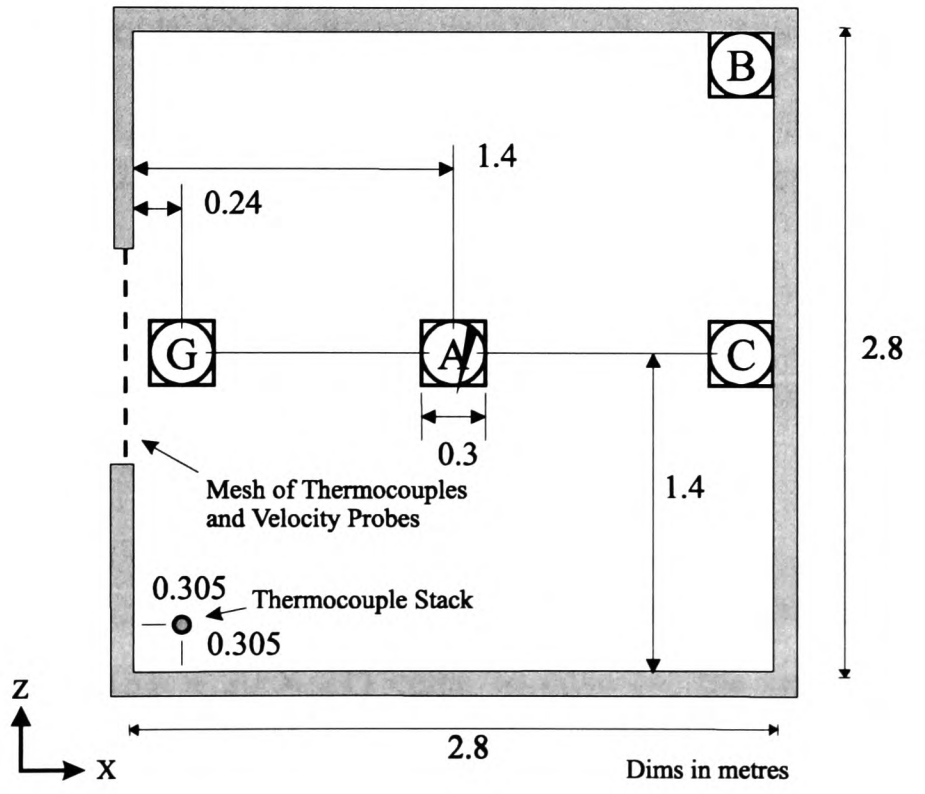


Figure 6-58 – Configuration of Steckler room

The non-spreading fire was created using a centrally located (position A in Figure 5-5) 62.9kW methane burner with a diameter of 0.3m. The experiments were conducted by Steckler et al [SQR1982] in a compartment measuring 2.8m × 2.8m in plane and 2.18m in height (see Figure 5-5) with a doorway centrally located in one of the walls measuring 0.74m wide by 1.83m high. The walls and ceiling were 0.1m

thick and they were covered with a ceramic fibre insulation board to establish near steady state conditions within 30 minutes. The door measured 0.74m wide and 1.83m high and was centrally located in one of the walls.

6.2.2.1.2 Phase 2 Model Configuration

From the Phase 1 results it was apparent that all the SPs over-predicted the temperatures generated by the fire. This was expected as all the SPs assumed that the walls were adiabatic and perfect radiative reflectors. In the second phase of the validation process SMARTFIRE was used in a variety of configurations that included the use of the multi-ray radiation model, more realistic physical properties and better specified boundary conditions. Finally a more refined mesh was used.

The simulation results presented can be summarised as follows:-

1) Phase 1 results for 2000-2-1 (using a simple volumetric heat release rate model).
2) As (1) with improved physical properties and improved boundary conditions.
3) As (2) with the multi-ray radiation model with 24 rays replacing the six-flux radiation model.
4) As (2) with refined mesh and taking advantage of symmetry so that only half the domain is simulated.

All the cases were run for 200 seconds of simulated time using 200 time steps of 1 second at which point steady state conditions are achieved.

In cases (1), (2) and (3) the same computational mesh was used; this mesh was composed of 13,020 ($31 \times 20 \times 21$) cells. In case (4) the computational mesh was 49,980 ($49 \times 34 \times 30$) cells; it must also be remembered that only half the domain is modelled as symmetry was used which produces an equivalent cell budget of 99,960 ($49 \times 43 \times 60$) cells.

In set up (1) it was assumed that the walls were adiabatic and perfectly reflecting (emissivity = 0). It was also assumed that the absorption coefficient of the air and gas mixture had a constant value of 0.315.

In set ups (2), (3) and (4) it was assumed that all the walls were composed of heat conducting "common" bricks of 0.1m thickness which have the following material properties: specific heat 840 J/kg.K, thermal conductivity 0.69 W/m.K and density 1600 kg/m^3. The wall emissivity is assumed to be 0.8. The model used turbulent (log-law) momentum and heat transfer at the walls. The effect of radiation was also modelled at the wall. The modelling of the heat transfer at the wall can be expressed as:-

$$-\lambda_w \partial T/\partial n \,|_w = H_c(T_w - T_{gas}) + \varepsilon\sigma T_w^4 - \varepsilon \dot{Q}_r''$$

where λ_w was the conductivity of the wall material, T_w was the wall surface temperature, T_{gas} was the air temperature next to the wall, H_c was the convective heat transfer coefficient, ε was the wall emissivity and $\dot{Q}_r''$ was the radiative heat flux at the wall surface.

The absorption coefficient's dependence on temperature was modelled as a set of three piecewise linear curves [EJG+2002]. Below 323K the absorption coefficient has a constant value of 0.01. Above 700K an empirical correlation for methane, based on the work of Hubbard and Tien [HT1978] from fire plume measurements, was used. Between these two temperatures it was assumed that the absorption coefficient varies linearly between the two correlations.

$$a = 0.01, \text{ if } T < 323\text{K};$$
$$a = 0.01 + (3.49/377)(T-323), \text{ if } 323\text{K} <= T < 700\text{K};$$
$$a = 3.5 + (3.5/700)(T-700), \text{ if } T > 700\text{K}.$$

Further set up details can be found in Appendices D5-D7.

6.2.2.1.3 Results

Comparisons between the above set ups are presented below (Figure 6-20 - Figure 6-22). The comparisons were made at two different locations; corner thermocouple stack located in one of the near corners to the doorway and a thermocouple and velocity measuring stack centrally located in the doorway (see Figure 5-5). The results

presented were after 200s of simulated time at which point the results were steady state.

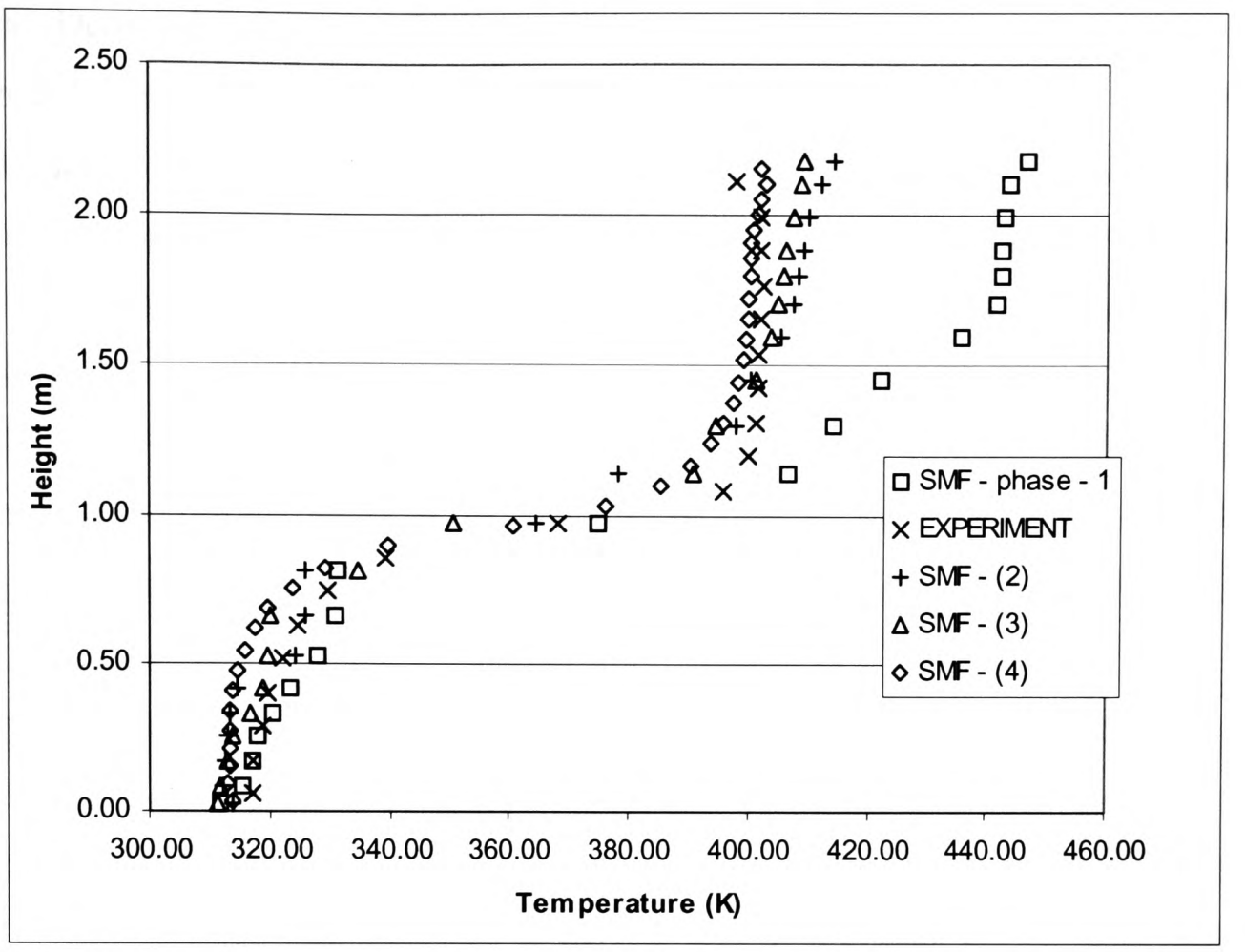


Figure 6-59 - Corner Stack temperatures produced using the various set ups for SMARTFIRE.

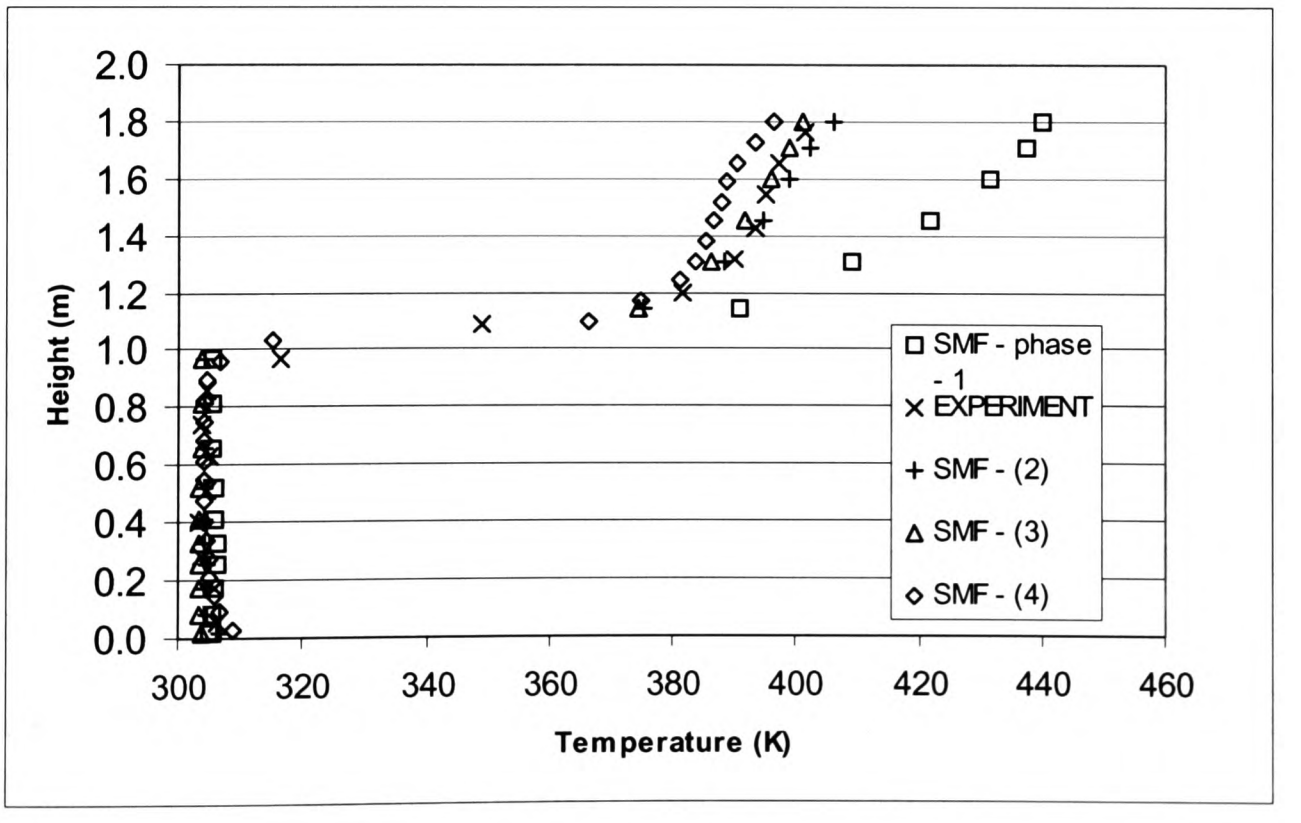


Figure 6-60 - Comparison of doorway temperatures for Steckler room

Depicted in Figure 6-20 is the corner stack temperature profile generated by SMARTFIRE using the four model configurations along with the experimental results. Depicted in Figure 6-21 is the doorway centre temperatures. As can be seen, all three simulations produced a much better reproduction of the temperature distribution, within the compartment, than the original Phase 1 predictions. Table 6-2 shows the model estimated upper layer temperature using the values given in Figure 6-20. Here again one notes that the refined models produced a much better representation of the upper layer temperature.

The overprediction produced in the Phase-1 simulations has been greatly reduced by each of the measures. Improving the physical properties and the wall boundary conditions produced the most significant improvement in the results. This has brought the upper layer temperatures closer to the observed values. Further improving the representation of the radiation distribution has not lead to a further significant improvement. This was expected as the temperatures were rather low and so the heat transfer via radiation was expected to be low and in this case the six-flux radiation model suffices. Finally, refining the mesh (i.e. case 4) leads to only a minor improvement in the model predictions compared with that obtained by using improved material properties and wall boundary conditions (i.e. case 2).

Table 6-7 - Approximate upper heat layer temperature for Steckler's room (A74) using the four SMARTFIRE configurations

	Exp	SMF-(1)	SMF-(2)	SMF-(3)	SMF-(4)
Temp (K)	401	442	408	406	400

The location of the hot layer can be estimated by determining where uniform temperatures are established in the upper layer. The height for the hot layers are detailed in Table 6-8. These represent the height of the bottom of the hot layer from the floor.

Table 6-8 - Approximate height of the hot layer f for Steckler's room (A74) using the four SMARTFIRE configurations

	Exp	SMF-(1)	SMF-(2)	SMF-(3)	SMF-(4)
Height (m)	1.25	1.6	1.35	1.5	1.4

As the hot layer is not sharply defined, an alternative definition for the height of the thermal interface can be defined as the height with the largest spatial (vertical) temperature gradient. These values are presented in Table 6-9.

Table 6-9 - Approximate height of the hot layer f for Steckler's room (A74) using the four SMARTFIRE configurations (alternative definition)

	Exp	SMF-(1)	SMF-(2)	SMF-(3)	SMF-(4)
Height (m)	0.97	1.0	0.97	1.0	0.97

The alternative definition gives a closer comparison for all the models with the experimental values and the predicted heights are basically the same within the resolution of the mesh. The first definition suffers from the fact that the models tend to smear the interface and the temperature is still increasing within the hot layer so the exact location of the hot layer was open to interpretation. The alternative definition does not suffer from this open ended interpretation and was also used by Steckler [SQR1982].

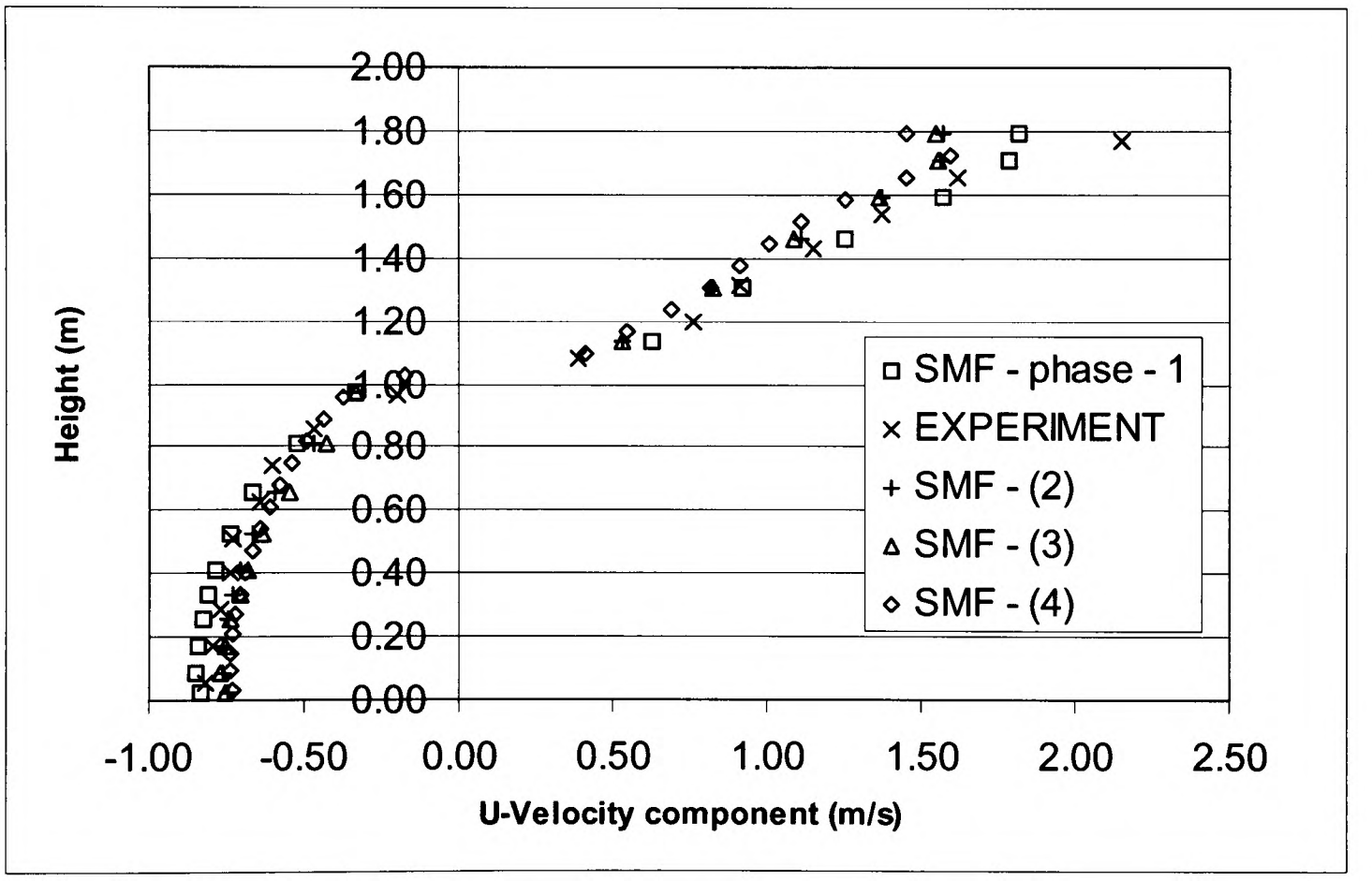


Figure 6-61 - Comparison of doorway velocity profiles for Steckler room

Depicted in Figure 6-22 is the horizontal velocity distribution along the centre vertical axis of the doorway. As can be seen all model configurations produced a very good representation of the velocity distribution. Each of the model configurations predicted

that the neutral plane height would be slightly above 1m, which was approximately the value obtained from the experiment.

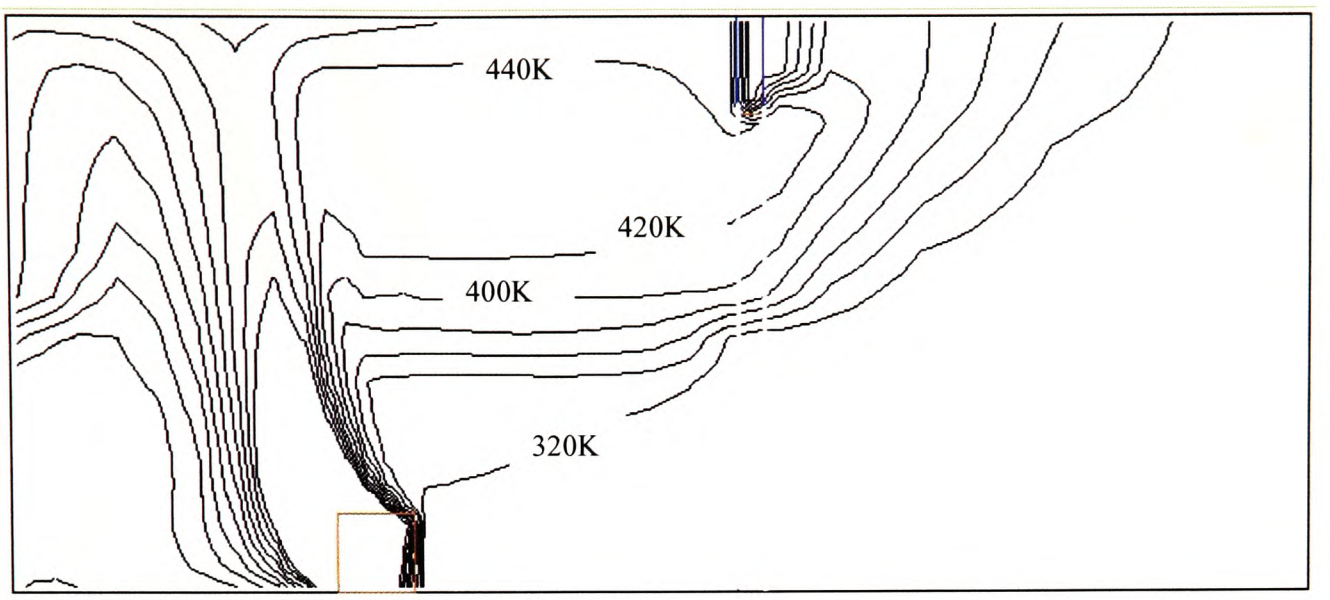


Figure 6-62 - Temperature contour plot produced by SMARTFIRE using the heat source model with phase-1 conditions

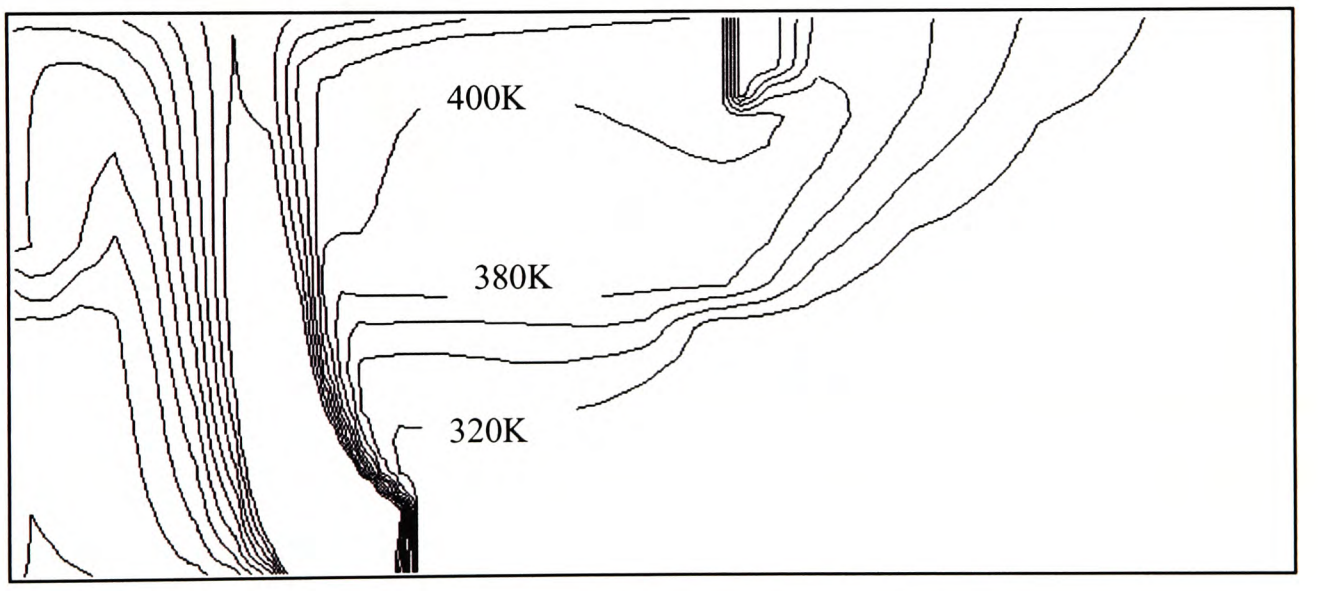


Figure 6-63 - Temperature contour plot for Case 2

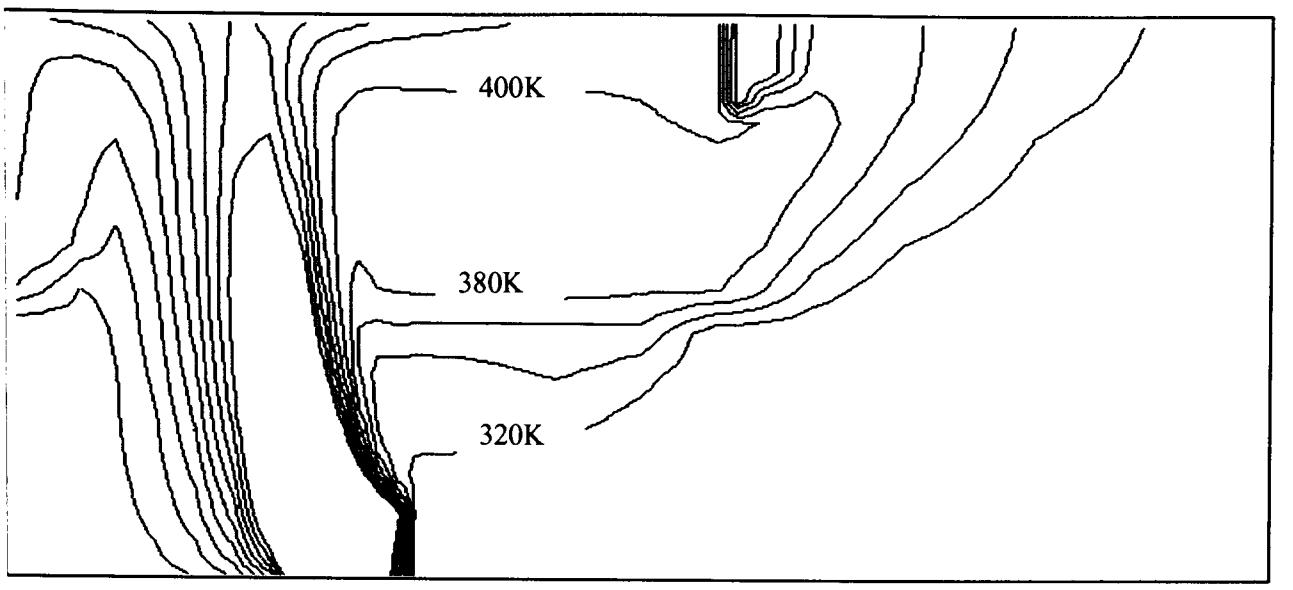


Figure 6-64 -- Temperature contour plot for Case 3

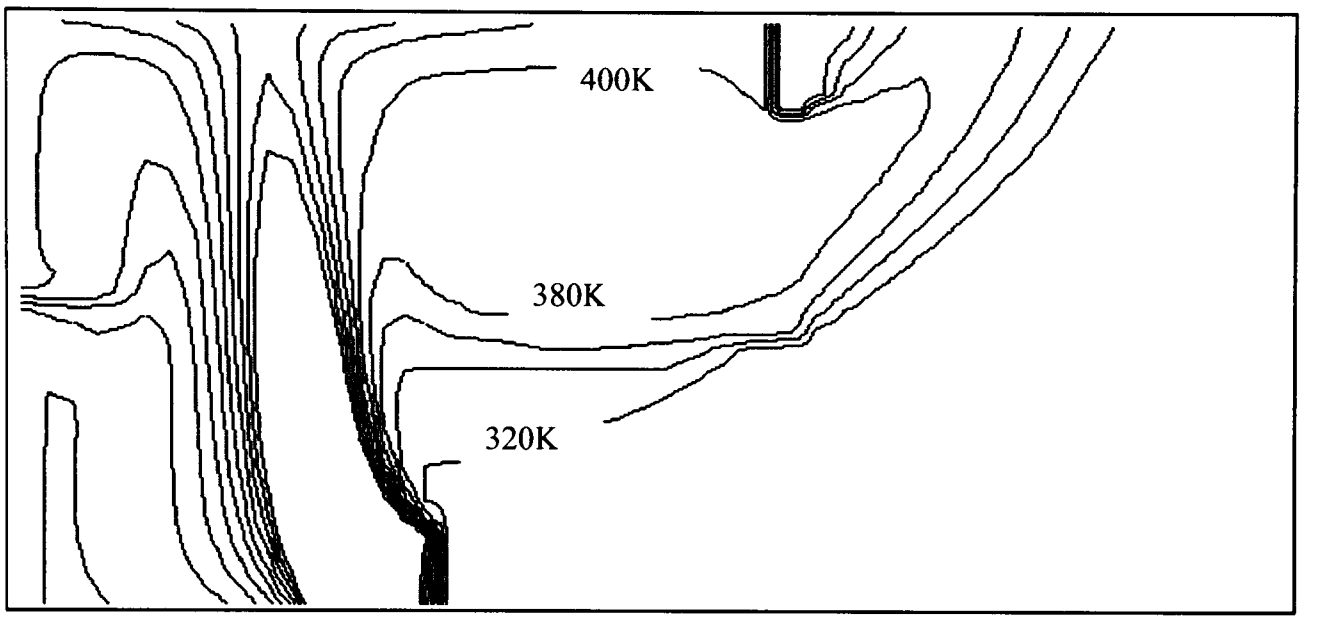


Figure 6-65 - - Temperature contour plot for Case 4

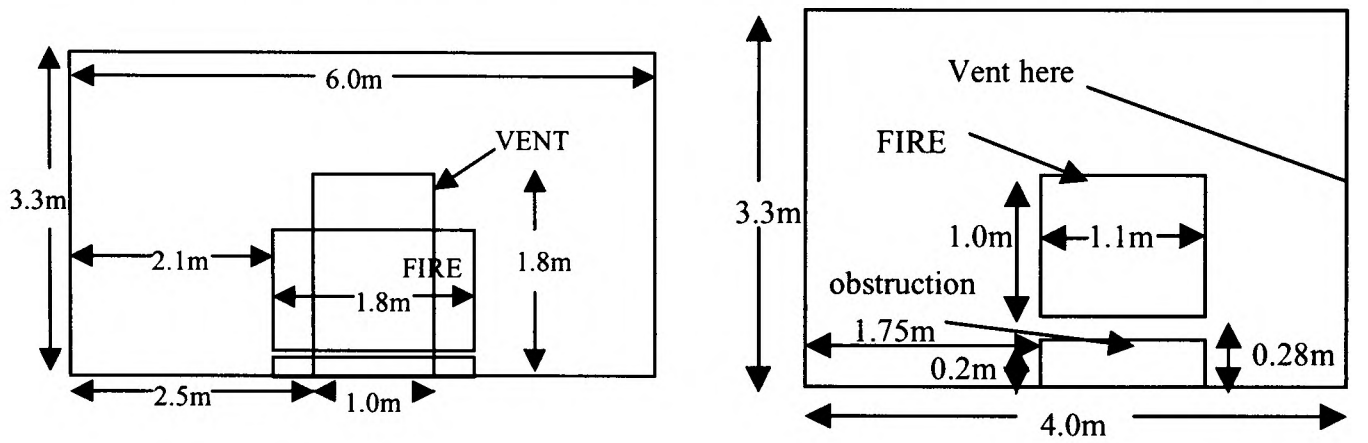
Depicted in Figure 6-25 to Figure 6-65 are temperature contour plots along the centre of the compartment for each of the four model configurations. As can be seen, each of the refined model configurations produced similar temperature distributions

From the above results it was clear that the Steckler room predications were improved by the introduction of the improved boundary conditions and material properties. For this particular problem there was no real advantage in using the multi-ray radiation model over the six flux model. Furthermore, once the improved boundary conditions and material properties were used, grid refinement does not further significantly improve the quality of the predictions.

6.2.2.2 SMARTFIRE: 2000-2-5 – LPC007 case

6.2.2.2.1 Introduction

This test case arises from a fire test conducted by the Loss Prevention Council (LPC) [GAC1997]. The test is a burning wood crib within an enclosure with a single opening. The test compartment is illustrated below and had a floor area of 6m x 4m and a 3.3m high ceiling. The compartment contained a doorway (vent) measuring 1.0m x 1.8m located on the rear 6m x 3.3m wall. The walls and ceiling of the compartment were made of fire resistant board (Asbestos) which were 0.1m thick. The floor was made of concrete. A steel obstruction measuring 1.1 x 1.8 and 0.2 m high was located on the floor below the fire. The corner thermocouple stack was located at 0.57m away from the side wall and 0.5m away from the front wall containing the vent. The plume temperature measurements were taken at 3.0m away from the side wall and 2.392m away from the back wall of the compartment.



y-z view x-y view

The heat release rate ($\dot{Q}$) is given by the following calculation (see equation 1).

$$\dot{Q} = \chi \cdot \Delta H_c \cdot \dot{m} \tag{1}$$

The efficiency factor (χ) and heat of combustion (ΔH_c) were given as χ=0.7 and ΔH_c is 17.8 MJ/kg for burning wood with a 10% moisture content and the mass loss rate ($\dot{m}$) (kg/s) for the wood crib is presented in the table below. It was assumed that the fuel molecule is $CH_{1.7}O_{0.83}$. The mass loss rate is given in Table 6-10 below.

Table 6-10 - Fuel mass loss rate used in test case 2000-2-5

Time(s)	0	150	450	460	1650
$\dot{m}$ (kg/s)	0	0.01835	0.18636	0.1978	0.1978

See Appendix D8 and D9 for further set-up details.

6.2.2.2.2 Phase 2 Model Configuration

The basic configuration of the model was the same as Phase 1. Phase 1 included the use of the combustion model. In Phase 1 this case proved difficult to converge and the numerical predictions were prematurely terminated. As a result, only part of the experimental data was utilised. It was felt that the difficulty in Phase 1 was caused in part by the artificial nature of the boundary conditions i.e. the use of adiabatic boundary conditions and the use of perfectly reflecting walls (emissivity = 0). The absorption coefficient was assumed to be a constant of 0.315.

In the Phase 2 simulations the wall boundary conditions were more accurately modelled, better physical properties were used and the multi-ray radiation model was used. In total two additional simulations were performed.

The same mesh was used for all the cases, the mesh had a cell budget of 26,040 (31 x 24 x 35). The case was run using 180 x 5-second time steps to give an overall simulation time of 900s.

The physical properties detailed in Table 6-11 were used for both the phase-2 simulations. The Phase-1 model used the same properties for air but all the solids were assumed non-conducting.

Table 6-11: Material properties used in test case 2000-2-5

Mat. Name	Density	Viscosity	Conductivity	Specific heat
Air	Ideal Gas (molecular weight = 29.35)	1.798E-05 + turbulent value	0.02622	1007.0
Asbestos	577	1E+10	0.15	1050.0
Concrete	2300	1E+10	1.4	880.0
Steel	7850	1E+10	45.8	460.0

The first case involved the following configuration:

The boundary conditions were modelled more accurately using heat-conducting walls that took into account the physical properties of the wall (asbestos). The properties of the floor (concrete) and the steel obstruction were also taken into account.

The wall emissivity was assumed to be 0.8. The model uses turbulent (log-law) momentum and heat transfer at the walls. The effect of radiation was also modelled at the wall. The model of the heat transfer at the wall can be expressed as:-

$$-\lambda_w \partial T/\partial n \,|_w = H_c(T_w - T_{gas}) + \varepsilon\sigma T_w^{\,4} - \varepsilon \dot{Q}_r''$$

where λ_w is the conductivity of the wall material, T_w is the wall surface temperature, T_{gas} is the air temperature next to the wall, H_c is the convective heat transfer coefficient, ε is the wall emissivity and $\dot{Q}_r''$ is the radiative heat flux at the wall surface. Currently there is no history term to account for the for the heat accumulation in the wall.

The SMARTFIRE gaseous combustion model was used. A volumetric mass loss source was used to represent the burning of the wood crib whose time dependent curve of mass loss rates was provided by the LPC report (see Table 6-10 above). The combustion efficiency was assumed to be 0.7. The heat of combustion used in the simulations was 17.8 MJ/kg.

A simple one-step global chemical reaction is adopted in the gaseous combustion model (see section 3.8).

The radiation absorption coefficient for both radiation models was assumed to take the following form:-

$a = 0.01$, if $T < 323$K;

$a = 0.01+0.305/377(T-323)$, if $323 <= T < 700$;

$a = 0.315+0.315/700(T-700)$, if $T > 700$.

The second case involved the following configuration:

As the first case but with the six-flux radiation model replaced with the multi-ray radiation model using 24 rays (identified as SMF-MR) (see section 3.9.2).

6.2.2.2.3 Phase 2 Results

The results for the plume thermocouple and room corner thermocouple stack are shown in Figure 6-66 and Figure 6-68. The lower (L) and higher (H) values refer to measurements at 1.5m and 3.0m above the ground respectively. In the graph keys SMF refers to case 1 (i.e. improved boundary conditions and material properties with six-flux model) and SMF-MR refers to case 2 (i.e. improved boundary conditions and material properties with multiray radiation model).

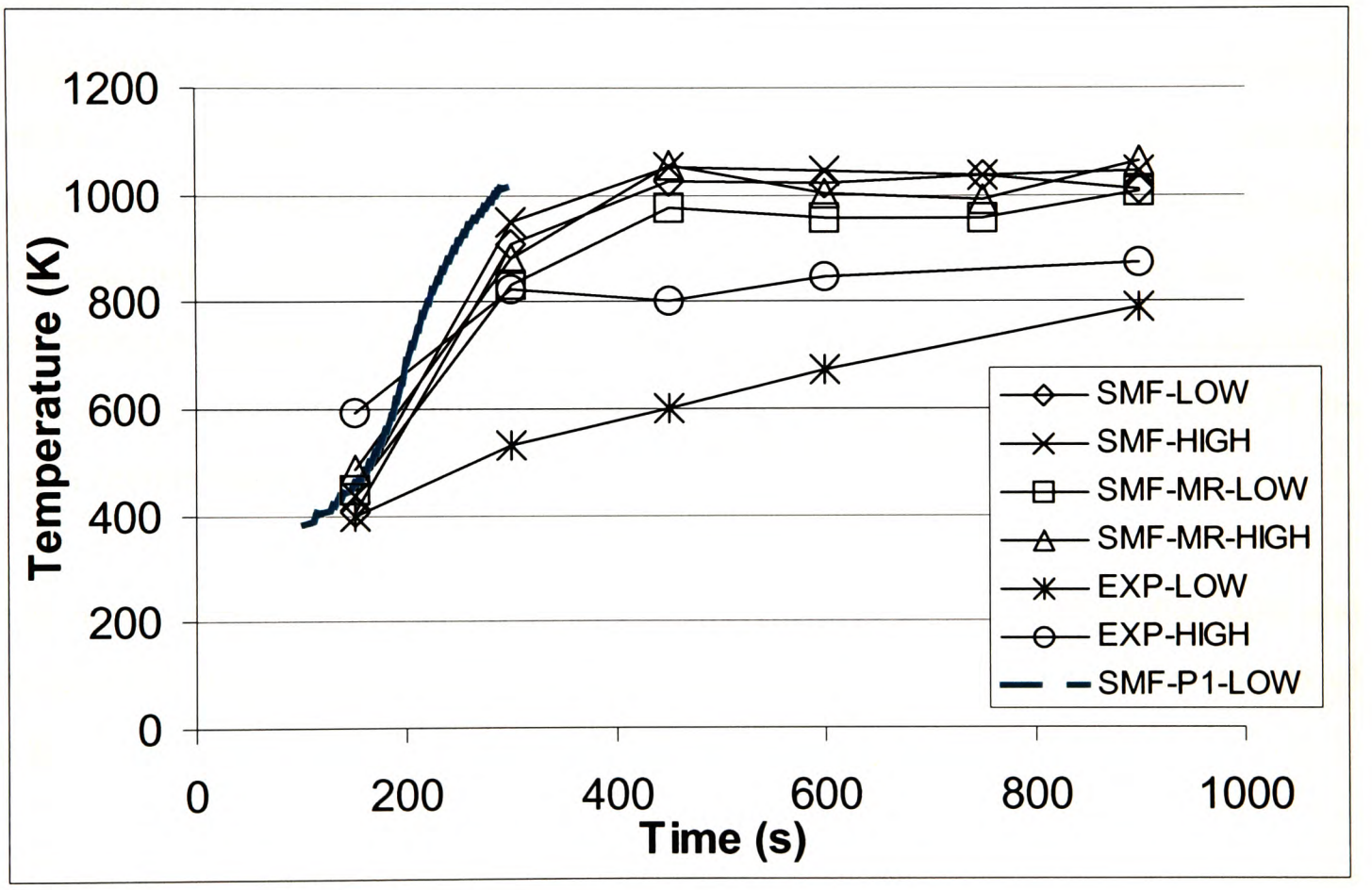


Figure 6-66 - Predicted and measured Corner Stack Temperatures at 1.5m (L) and 3.0m (H) above the floor for the LPC test case

With the improved Phase 2 model specifications, it was possible to simulate the entire duration of the experiment. The convergence problems noted in Phase 1 were removed. Furthermore, examination of the corner stack temperature predictions reveals that the improved boundary conditions and physical models reduced the temperature predictions compared to the phase-1 results bringing them closer in line

with the measured values. In addition, the incorrect behaviour noted in Phase 1 where the temperature predictions in the lower region of the room exceeded the temperatures in the higher region was corrected. However, the level of stratification observed in the predicted results was not as great as that observed in the experiment. The multi-ray model produced a slightly greater stratification between the upper and lower temperatures than that produced using the six-flux model.

Both of the Phase 2 cases overpredicted the experimental values with the results generated using the multi-ray model being slightly closer to the measured results. Differences between the multi-ray model and the experimental results for the high measuring location was as high as 30%, while for the low measuring location, the error was as high as 63% (see Figure 6-67). For the six-flux model, the maximum errors were 31% and 71% respectively. The experimental trends in the upper temperatures were reproduced well by the numerical predictions. These temperatures tended to increase until about 300 seconds into the fire and then remained approximately constant. The numerical predictions followed this trend but the peak was reached at approximately 425 seconds. The experimental trends in the lower temperatures showed a continual increase over the entire duration of the experiment. However, the numerical predictions for the lower temperatures followed those of the upper temperatures.

The noted overprediction could be due to inaccuracies in the experimental data and deficiencies in the model assumptions such as assuming a constant wall emissivity of 0.8.

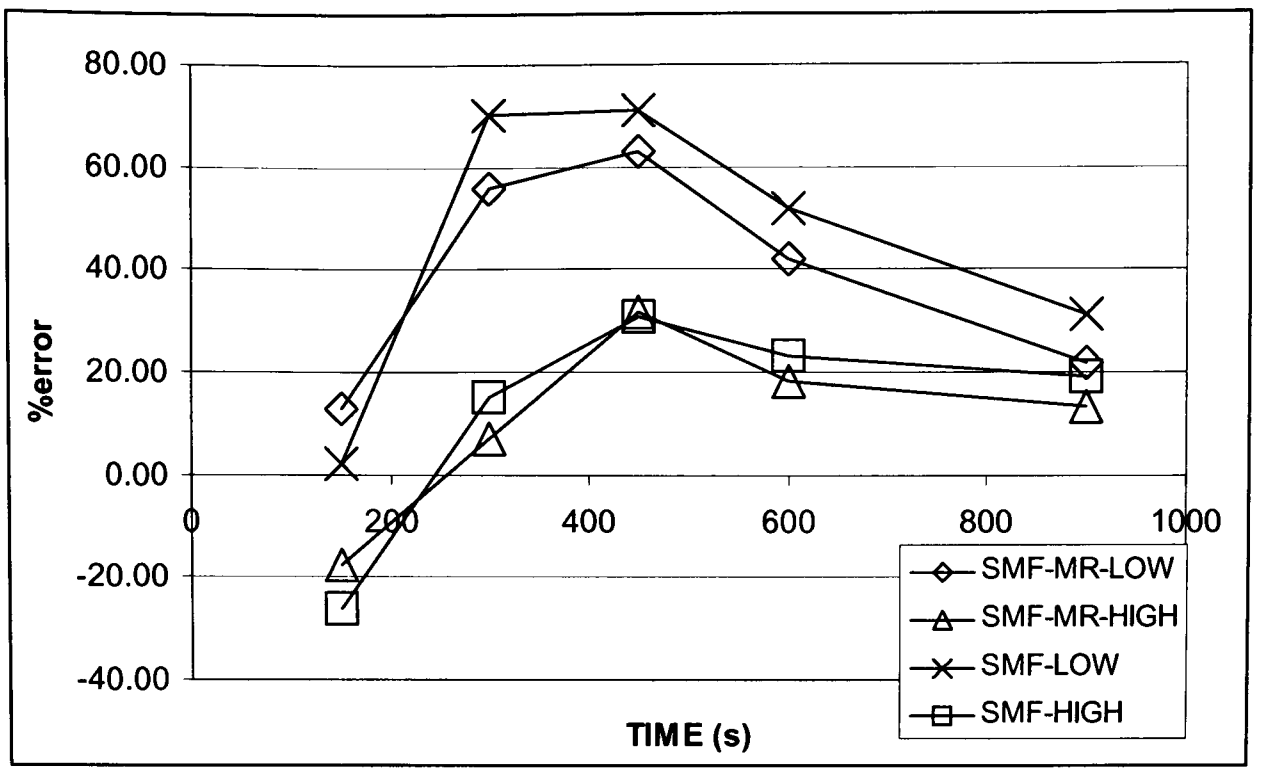


Figure 6-67 – Error (%) in the SMARTFIRE predictions using the six-flux model for the corner stack in test case 2000-2-5

The plume stack temperature predictions are depicted in Figure 6-68. They followed the general experimental trend of a peak followed by a dip (see Figure 6-68). The SMARTFIRE simulation demonstrated that this trend was mainly caused by the changes of the fire plume shape. After the initial phase of fire growth, the fire became quite large and the hot combustion products accumulated beneath the ceiling creating a gradually deepening hot layer. In conjunction with the fresh air being entrained into the compartment by the fire, the downward movement of the hot upper layer pushed the fire plume back so that it tilted away from the window towards the rear wall. Thus, the fire plume has shifted away from the central vertical line of the crib. Since this line – and hence the measuring devices - are not in the centre of the fire plume, the temperatures along it predicted and measured are reducing after they reach the peak value.

It was also notable that the experimental measurements indicate that the plume was hotter at the top than at the lower level that was suggested by the model predictions. This could be due the combustion behaviour with combustion occurring more in the upper layers of the compartment. It was noted in the experiment that some flaming combustion occurred outside the compartment. This may explain the higher temperatures numerically predicted within the compartment as all the combustion is assumed to have occurred within the compartment. Another source of error is the

presence of the wooden crib. This would act as an obstruction, which is dynamically changing throughout the combustion process. This obstruction effect is ignored in the numerical modelling as it is difficult to model this changing shape. The obstruction would have some effect on the airflow within the compartment particularly into the plume.

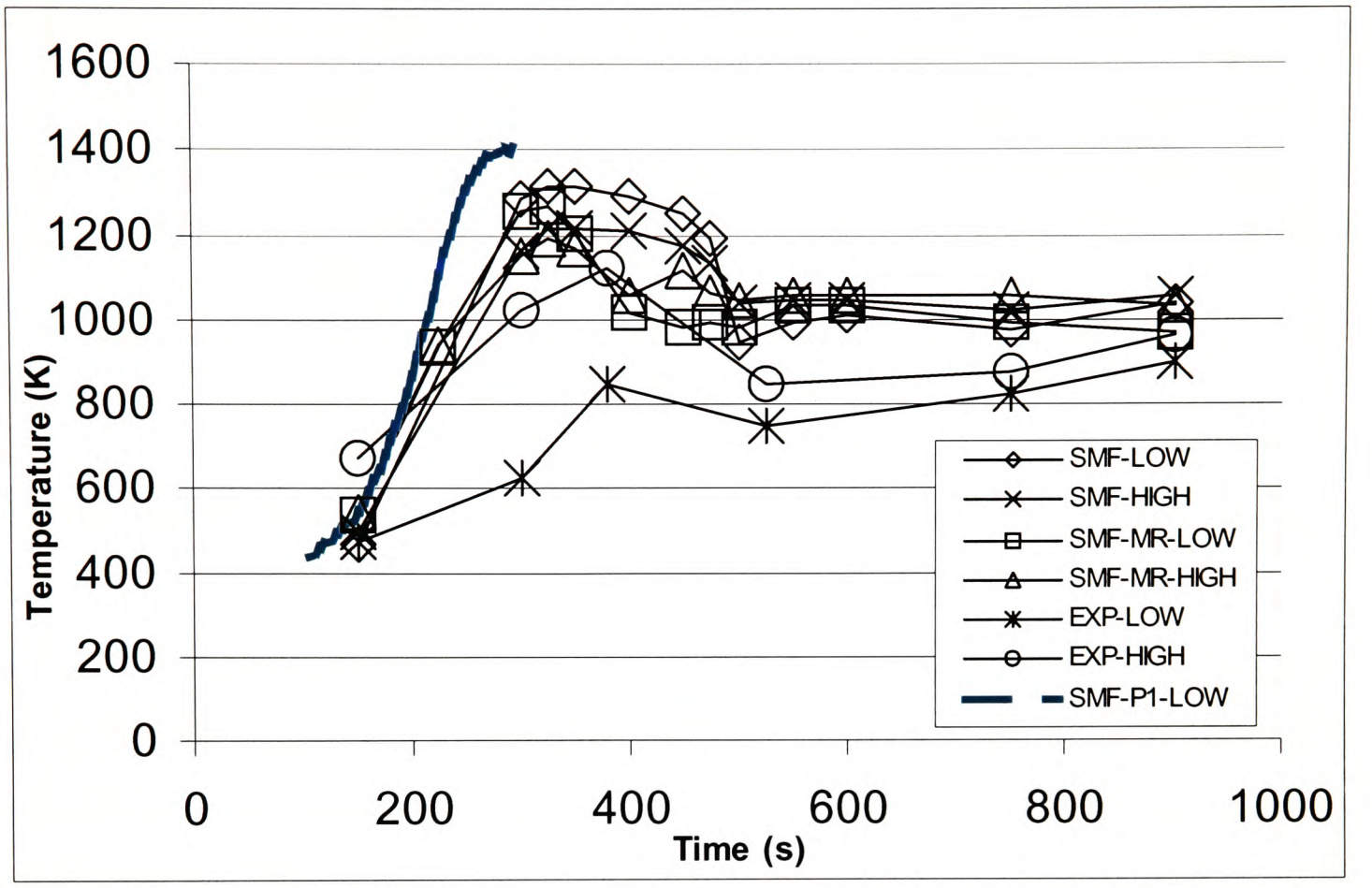


Figure 6-68 - Plume stack temperature predictions for SMARTFIRE for test case 2000-2-5

There was a distinct difference in the plume behaviour for the six-flux model and the multi-ray radiation model (24 rays). The peak in temperature lasts longer for the six-flux model but it is difficult to say which of the model predictions is better in this particular simulation.

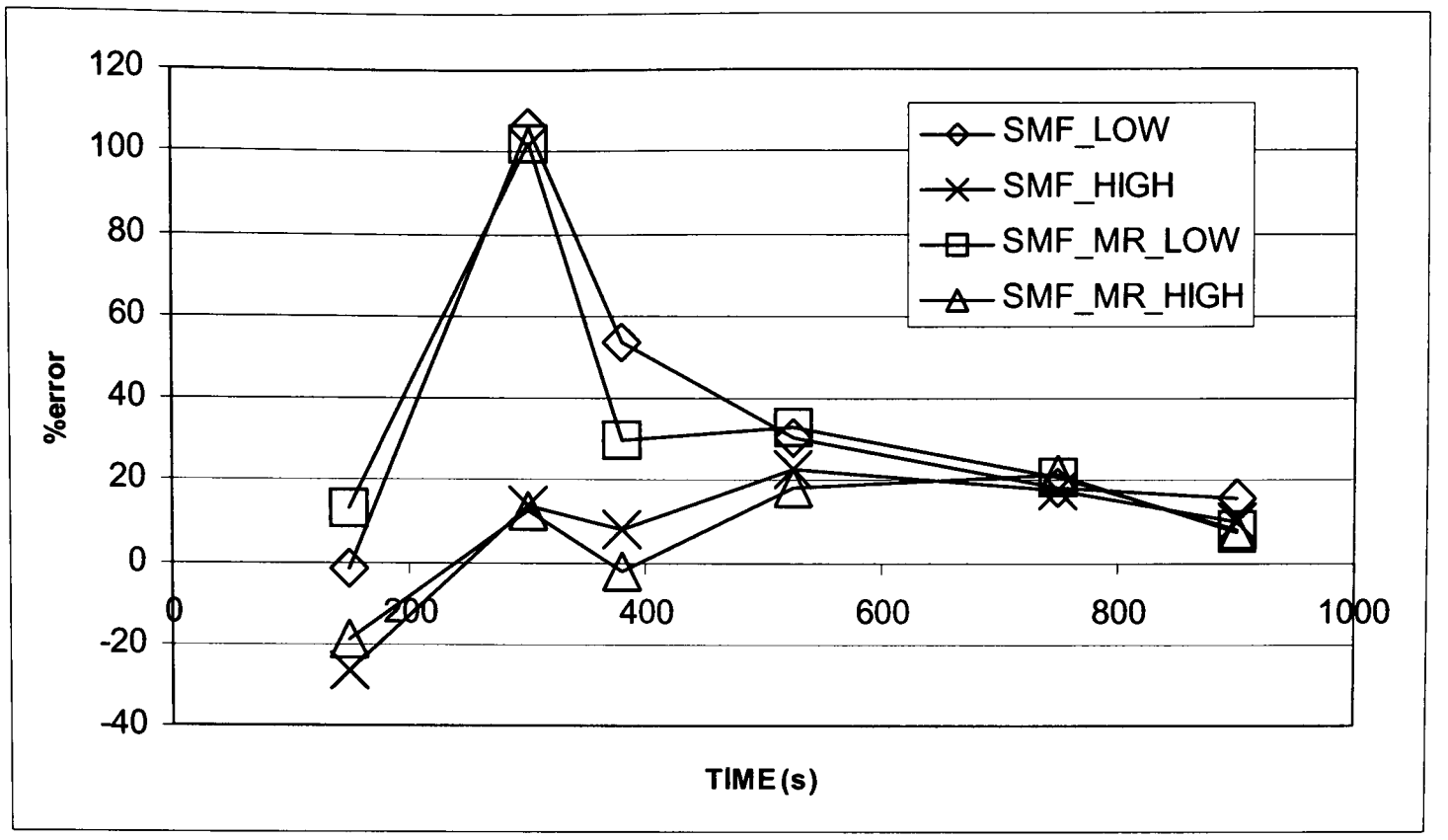


Figure 6-69 - Error (%) in the SMARTFIRE predictions using the multi-ray model for the corner stack in test case 2000-2-5

As with the corner measurements, both of the plume simulations overpredicted the experimental values with the results generated using the multi-ray model being slightly closer to the measured results. Maximum differences between the multi-ray model and the experimental results for the high measuring location were 21%, while for the low measuring location; the error is 100% (see Figure 6-69). For the six-flux model, the maximum errors were 27% and 106% respectively.

The improved boundary conditions and wall properties have greatly improved the quality of the model predictions. The numerical predictions followed the general experimental trends although experimental results were overpredicted. The discrete transfer radiation model has marginally improved the quality of the numerical predictions but at the cost of increased computational time.

6.2.3 Discussion of Phase 2 results

Predictions for the radiation test case (2000-1-5) using the SMARTFIRE multi-ray radiation model with 24 rays, showed considerable improvement over the results generated in Phase 1. The results from this simulation indicate the greater inherent accuracy that the multi-ray radiation model has over the simpler six-flux model. It is important to note that the greater degree of accuracy offered by the multi-ray model

may not manifest itself in producing more accurate fire predictions. Whether or not the multi-ray radiation model will make a significant difference in a fire simulation depends on the nature of the case being examined.

In the Phase 1 simulations, all the SPs predictions for the Steckler room fire case (2000-2-1) failed to accurately reproduce the measured temperatures, but successfully captured the overall trends. The results for Phase 2 showed that considerable improvement could be achieved by a more sophisticated treatment of the wall boundary conditions and more accurately representing the material properties. While further improvement could be achieved through the use of the multi-ray model and mesh refinement, these were insignificant in comparison.

In Phase 1, it was not possible to generate converged solutions of the LPC-007 case (i.e. 2000-2-5) beyond 300s. This was thought due to the nature of the boundary conditions selected for Phase 1. In Phase 2, with a more sophisticated treatment of the wall boundary conditions - which included a heat loss calculation - it was possible to generate converged solutions for the entire duration of the experiment. While errors in the numerical predictions persisted, the numerical predictions were able to reproduce most of the observed trends in the experimental results.

In studying the outcome of the Phase 2 test cases, it is clear that by activating sophisticated physical models, the SP tested was capable of generating improved predictions in all of the cases examined. While this may seem an intuitively obvious result, it is a necessary demonstration of the capability of the fire modelling tool that this can be done in a measurable and reproducible manner.

6.3 CONCLUDING REMARKS

Both phases of the testing programme were successfully completed. In studying the results generated it is important to note the following points:

1) The results generated and comments made only refer to the software actually used in the trials. This should not simply be taken to mean the product name but also the release number and version number of the software.

2) The Phase 1 results are not intended to represent mesh independent solutions. They are intended to represent converged solutions on "reasonable" meshes. In each test case, the same computational mesh is used by each software product. Phase 2 simulations can be used to explore simulations performed using finer meshes.
3) In Phase 2 of the study only 3 test cases were selected. These were the radiation test case (2000-1-5), the Steckler Room case (2000-2-1) and the LPC-007 (2000-2-5).
4) The Phase 1 results do not make use of the most sophisticated physics available in each of the software products. A base line set of characteristics has been set that allow a fair comparison between the codes. Where model predictions are compared with experimental data, these predictions can be improved through the use of more sophisticated physical sub-models. Phase 2 simulations were used to explore the benefits of using more sophisticated physics and improved modelling.
5) The series of trials undertaken in this work should not be considered to be definitive. They have been selected as a basis for exploring the potential of the benchmarking process. It is intended that additional tests should be added to the suite of test cases.

The results from the CFD test cases were consistent with the view that the basic underlying physics implemented within the codes would be similar and provide a good representation of reality. This should come as no surprise as all three software products purport to model fluid dynamics processes using similar techniques. However, from a regulatory viewpoint, it is reassuring to have an independent verification of this similarity. In addition, where experimental results or theoretical solutions were available, the software products have produced reasonable agreement with these results. No doubt, it could be argued that improved agreement could be achieved if the spatial mesh and time stepping are improved.

The results from the fire cases support the conclusions drawn from the CFD test cases. While there are minor differences between the results produced by each of the software products; on the whole they produce – for practical engineering considerations – identical results.

A significant – and somewhat reassuring - conclusion to draw from these results is that an engineer using the basic capabilities of any of the three software products tested would be likely to draw the same conclusions from the results generated irrespective of which product was used. From a regulators view, this is an important result as it suggests that the quality of the predictions produced are likely to be independent of the tool used – at least in situations where the basic capabilities of the software are used.

A second significant conclusion is that within the limits of the test cases examined in the phase 1 testing regime and taking into consideration experimental inconsistencies and errors, all three software products were capable of producing reasonable engineering approximations to the experimental data, both for the simple CFD and fire cases.

The use of the Phase 2 testing protocols allowed the more advanced features of the CFD code to be demonstrated by the code manufacturer. For the SMARTFIRE SP improvement over that obtained in phase 1 was demonstrated for three of the test cases.

Furthermore, the results should not be treated in isolation but taken within the context of the Phase 1 findings. A significant conclusion from the Phase 1 predictions was that within the limits of the Phase 1 testing regime and taking into consideration experimental inconsistencies and errors, all three SPs were capable of producing reasonable engineering approximations to the experimental data, both for the simple CFD and fire cases. With the completion of the Phase 2 testing, this statement is somewhat strengthened - at least for the SP tested in Phase 2.

The concept and testing protocols developed as part of this work have been shown to be a valuable tool in providing a verifiable method of benchmarking and gauging the basic and advanced capabilities of CFD based fire models on a level playing field. The phase 1 protocols have been demonstrated to show that the CFD codes can be compared in a fair manner with similar physics, numerics and meshing utilised throughout. The phase 2 protocols demonstrated that by utilising the advanced features of a particular SP, in this case SMARTFIRE; improved results beyond the

basic level of benchmarking achieved in phase 1 can be obtained. To further improve the capabilities of the approach, it is recommended that additional test cases in the two categories, theoretical and experimental, be developed and several of the fire cases should be refined.

In addition, a modification to the testing procedures is suggested that would reduce the burden and cost of performing the testing by the test organisation. While all of the test cases using all of the codes were run by a single organisation – in this case the author of this thesis – the code developers also were requested to run an independent selection of the test cases as specified. This was necessary to verify that the results produced in this report were a true and fair representation of the capabilities of the various software products under the specified test conditions. This has proven to be quite useful as it brings the developers into the benchmarking process and it eliminates issues concerning fairness and biased reporting of results. However, if this process is to become a mandatory requirement, the testing organisation will have a considerable amount of work to do if it is to run every software product and its various upgrades through each of the test cases. In order to reduce the cost of testing, it is suggested that the test organisation should only perform the random testing and require the software developers to run and submit all of the test cases.

With these protocols in place it is hoped that a FSE can gain more trust in CFD methods. Previous to the above exercise little had been done on formalising independent verification, validation and cross comparison of CFD software utilised for fire modelling purposes.

7 Parallel Implementation of SMARTFIRE

In this chapter, the parallelisation of the fire modelling software SMARTFIRE is described. The runtime of CFD based fire field models was identified as one of the significant issues associated with the effective use of fire field modelling for use in performance based fire safety regulations. This was demonstrated in chapter 4, where the FIREDASS model, while capable of modelling the phenomena under consideration, was impractical for routine design engineering applications due to the excessive runtime.

It was decided that the SMARTFIRE software would be the basis of the parallel investigation. Therefore a brief overview of the SMARTFIRE software is given with the CFD engine being identified as the key component for parallelisation.

In this investigation it will be determined whether the potential benefits of parallel processing can be realised within a FSE's office environment. This means that the parallel version of the fire simulation software should run on standard Windows NT based machines attached to a standard Local Area Network and requires minimal additional investment. The benefits of parallel processing are simply summarised as how much faster will a fire modelling problem be solved on n processors and how large a problem can be run on n processors. Furthermore the parallel strategy should work on a variety on possible PC combinations and should be able to operate within a FSE office environment without adversely affecting other computer users. The cost of using parallel processing must be kept to a minimum. From the point of view of the developer the source code must be easy to maintain

7.1 Overview of SMARTFIRE

The functionality and construction of the SMARTFIRE system will be briefly described here. More detailed descriptions can be found in the publications of Ewer [Ewe1999, Ewe2000] and the user manual [GKP+1999].

SMARTFIRE is an open architecture CFD environment written in C++ that is comprised of four major components: CFD numerical engine, Graphical User

Interfaces, Automated meshing tool and the Intelligent Control System. One of the main aims of the SMARTFIRE development group is to make fire field modelling more accessible to fire engineers with limited CFD experience. One of the ways this is achieved is by embedding expert knowledge into the CFD software.

The CFD engine in SMARTFIRE has many additional physics features that are required for fire field modelling [Gal1989, Cox1995]. These include a six-flux radiation [KGC1991] model, a multiple ray radiation model [RC1990], provision for heat transfer through walls, a volumetric heat release model or gaseous combustion model (using the eddy dissipation model) [LMR1997] to represent fires, smoke modelling and turbulence (using a two equation K-Epsilon closure with buoyancy modifications). The mathematics behind the CFD is described in detail in chapter 3.

The main Graphical User Interface (GUI) is used to specify the problem. Through this GUI the user sets the geometry, specifying the location of walls, wall materials, internal compartments and obstacles (such as desks, stairs or partitions). Also specified are the location of vents (along with any fans, inlets or outlets), the nature and location of the fire, the radiation model to be used and gaseous properties such as absorption coefficients. The GUI also provides access to the automated meshing tool and manual mesh editor as well as the CFD engine. Once the problem has been specified, the automated mesh generation tool is used to generate the control volume cells for the problem. It is important to note that embedded expertise is used to determine a mesh and cell budget that is appropriate for a *reasonable* solution to the problem.

The CFD engine is used to simulate the fluid dynamics of fire development by numerically solving a set of differential equations that describe the laws governing the physical processes inside the domain. The solution is found by performing a series of consecutive approximations, whose quality is measured by the residual error (defined as the magnitude of change between two adjacent approximations). The residual error is calculated separately for each solved variable and the time step is said to have "converged" if the residuals of all variables fall below a specified tolerance.

The interactive CFD code has its own unique multiple-windowed user interface. Unlike traditional fire field models, this allows the user to interact with the solution through *observation* of the developing solution (using graphs, visualisations and data explorers) and by allowing the user to make *adjustments* to control parameters. Such adjustments in traditional CFD codes generally involve terminating a simulation, editing input files and restarting.

The parallelisation process focuses on the CFD engine component of SMARTFIRE. This is the part of the software suite that requires the speed benefits that parallelisation may provide. Furthermore, the CFD engine was the only component that would benefit from parallelisation with the other tools being highly interactive in nature and performing minimal computation.

For this parallel investigation, both a shared memory (SM) and a distributed memory (DM) version of SMARTFIRE were developed. The DM version of SMARTFIRE was developed as a more conventional batch mode version with the interactivity temporarily removed for this implementation. The interactive component will be added in the future and the methods for creating an interactive parallel DM CFD code will be described in section 10.4. Removal of interactivity does not affect the purpose of this research which is assessing the potential benefits of utilising office based PCs for parallel fire modelling calculations. At this stage of the research there is little point in implementing the interactive aspect until it had been confirmed that the potential benefits of parallel processing warranted this additional software development. However, interactivity presented no problem for the SM approach and was therefore retained for that version.

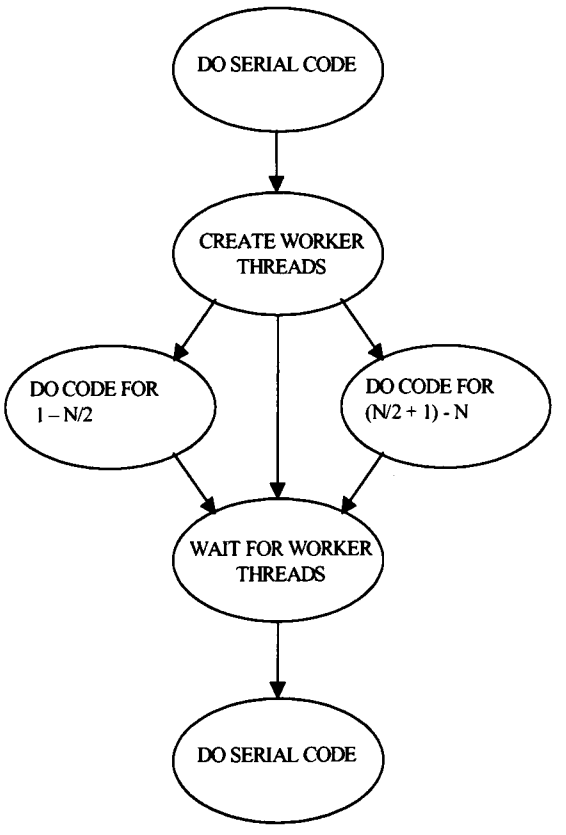
7.2 Shared Memory approach

Unfortunately the serial code version of SMARTFIRE as it stands can not simply be launched on a Shared Memory (SM) machine and automatically run in parallel. It was necessary to explicitly make the code utilise the multiprocessor environment. This was achieved by taking advantage of multithreading. A thread is an encapsulation of the flow of control in a program. Software engineers are used to writing single-threaded programs - that is, programs that only execute one path through their code at

a time. Multithreaded programs may have several threads running through different code paths simultaneously. In a typical process in which multiple threads exist, zero or more threads may actually be running at any one time. On a single processor machine the threads cannot be run simultaneously but must be time-sliced. On a multiple CPU machine however each CPU can run a thread at exactly the same time (i.e. in parallel).

A simple domain split was adopted for distributing the problem across the processors. Essentially on a dual processor system two worker threads were 'created' by the main thread. The first thread would deal with the cells 1 to NCELL/2 and the second deals with cells (NCELL/2 + 1) to NCELL (where NCELL is the total number of cells) while the main thread waits for these two threads to finish (see Figure 7-1 below).

A more generalised distribution of cells over the threads would be $((j-1)(NCELL/NP)+1)$ to $j(NCELL/NP)$ where j is the 'number' of the thread and NP is the total number of threads (CPUs available).




```
for ( i = 1; i <= 2; i++ ) {
   thread_index = i-1;
   thread_array[thread_index] =
         CreateThread(NULL, 0,
            ...
}
...
// The threads are running and we
//have to wait for
// them to finish
WaitForMultipleObjects(AJG_MAX_THREA
DS, thread_array, TRUE, INFINITE);
```

Figure 7-1 - Flow diagram for multithreaded code with code fragment for main thread

This method was applied to every routine that looped over the total number of cells in the domain. The advantage of this system was that it allowed the routines to be

changed and tested one at a time and the program could be run in a part-parallelised mode that made development relatively easy. The parallelism was localised within these loops.

Approximately 2000 lines of code were written to create the multithreaded version. Approximately 50 loops required modification. These modifications were only applied to loops within the solution algorithm; no modifications were necessary for the input, output or interactive parts of SMARTFIRE.

```
If   (   variable[PRESSURE_CORR].get_solver_class()   !=
GROUP_SOLVER ){
   for ( cell_num = 1; cell_num <= domain.num_of_cells;
cell_num++ ){
      cell[cell_num]->calc_pressure_grads();
   }
} else {
```

```
If   (   variable[PRESSURE_CORR].get_solver_class()   !=
GROUP_SOLVER ){
   do_with_simple_threads(pressure_gradients_thread);
 } else {
```

Figure 7-2 – Equivalent code fragments for serial (left) and multithreaded [SM] (right) approaches

In Figure 7-2 above the left hand box represents the original code and the right hand box represents the modified multithreaded code. The do_with_simple_threads function in Figure 7-2 is essentially the code in Figure 7-3. The parameter presssure_gradient_thread refers to another function which performs the same function as the original serial code (see left hand box of Figure 7-2). In the do_with_simple_threads creates a thread with this code and passes the start and end points for the loop dependent on the number of processors.

```
Unsigned long WINAPI pressure_gradients_thread(smf_thread_parameters*
smf_thread_param) {
   Int_Type cell_num, start, end;

   Start = smf_thread_param->start;
   End = smf_thread_param->finish;

   For (cell_num = start; cell_num <= end; cell_num++){

      Cell[cell_num]->calc_pressure_grads();

   }
   return 0;
}
```

Figure 7-3 – pressure_gradient_thread function code fragment

Results using the multithreaded version of SMARTFIRE are given in section 8.4.4.1

7.3 Distributed Memory approach

The Distributed Memory (DM) approach is probably the most beneficial approach to take as SM PCs are relatively rare and generally of small scale with dual processor

machines the most popular, quad processor machines hardly seen and 8-processor machines virtually non-existent. Therefore, the possible advantages to be derived from the SM approach are limited by the availability of present PC technology.

However, single processor PCs connected via a Local Area Network (LAN) are commonly found and can constitute a DM system. It should be further noted that the program written for a DM system would also run on a SM machine removing the need for two separate parallel versions. As with the SM case it is not simply the case that a serial code can be placed on the DM architecture and will then run in parallel. The DM formulation was more difficult to program than the multithreaded SM formulation as there were issues concerning problem space distribution over the processors (see section 7.4.2), communication of values between the processors (see section 7.4.4) and I/O issues (see section 7.4.11 and section 7.4.12). Each processor now runs an entire copy of the code so parallelisation is across the whole code and not just localised to the loops as in the SM case.

The code was written using the Single Program Multiple Data (SPMD) paradigm. This means that only one executable is used and a copy of the executable is launched on each processor which operates on its own part of the problem domain. Typically there is one process that handles the I/O and problem distribution, this is the master process (process 0). The other processes are exactly the same as one another and only differ slightly from the master, as they perform no I/O, and these processes are called slave processes. To differentiate between the master and the slave processes some conditional statements are added to the code such as: -

```
If (my_mpi_id == 0) {
```

Every process has its own my_mpi_id; this is 0 for the master process.

This strategy generally has been used on dedicated parallel hardware in the past as there would only have been 1 processor node that could handle I/O. Although this is not the case for general PCs on a LAN this methodology was followed to allow for the possibility of porting the software to dedicated parallel hardware. Figure 7-4 – Figure 7-6 illustrate the relationship between the master and slave processors.

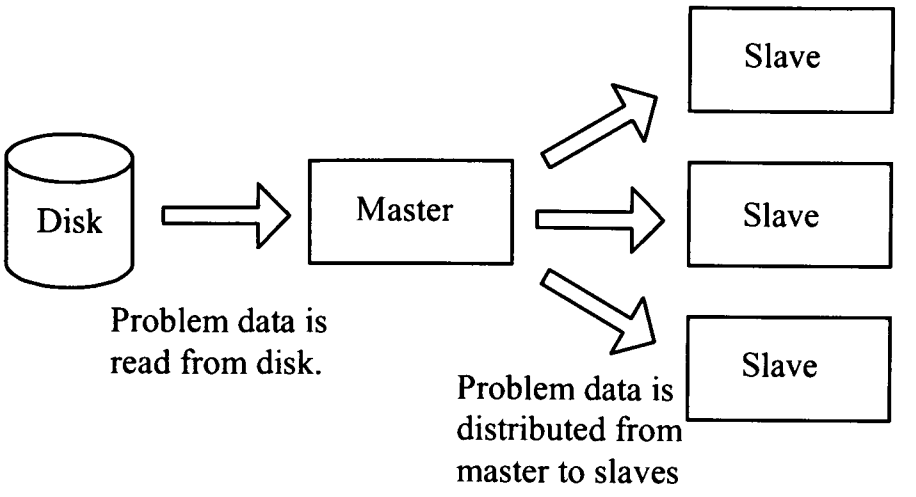


Figure 7-4 – Initial problem space distribution by master process to slave process

Initially the problem must be entered (normally via disk storage) into the master process (see Figure 7-4). The master process then appropriately subdivides the problem space and transmits this data to the relevant slave process.

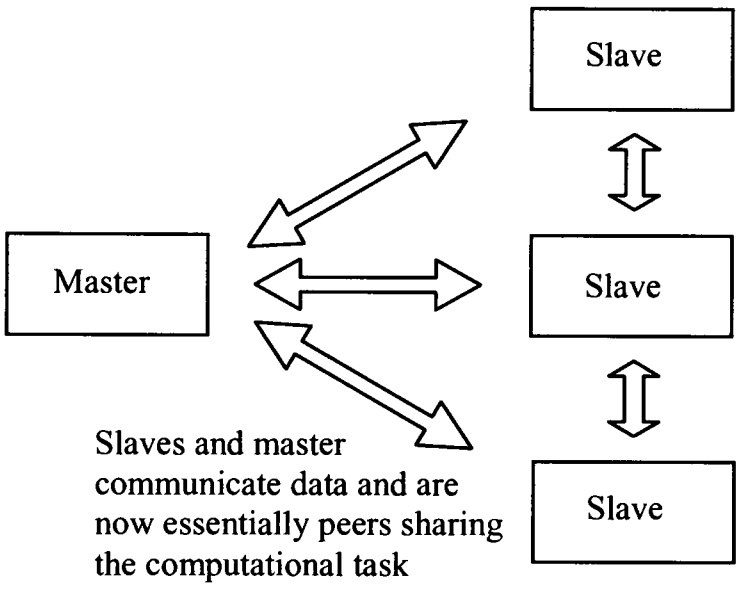


Figure 7-5 – Master and slave processes running as peers.

In the computation phase the slaves and master are now peers working co-operatively to solve the problem (see Figure 7-5). The slaves can directly communicate with each other without interaction of the master. These communications are necessary due to the coupled nature of the solution strategy.

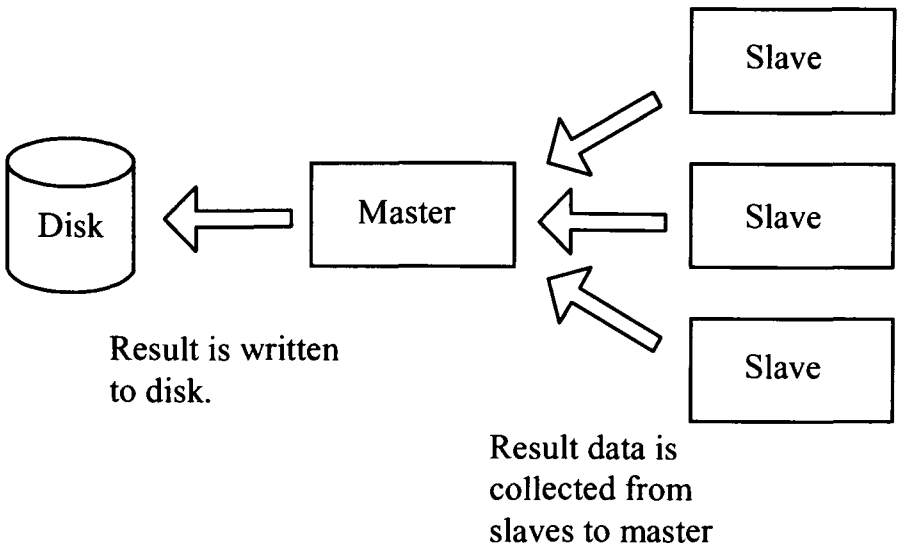


Figure 7-6 – Data is collected from the slave processes by the master process for saving results.

Once the computation has finished the results are gathered from the slave processes by the master and then transmitted to the master to the output deck, normally disk storage (see Figure 7-6).

7.4 DM Parallelising SMARTFIRE

The DM version of SMARTFIRE was designed to fulfil the following criteria in order to make the parallel version of SMARTFIRE as useful as possible for a FSE: -

- There should be no difference between the input or output files for DM SMARTFIRE and serial SMARTFIRE. This allows the problem to be designed and visualised using the serial version of SMARTFIRE.
- The possible speedup is as good as possible for the number of CPUs involved. Ideally a 5-processor machine will run the same problem 5 times faster than a single processor machine.
- The problem should scale well in memory; ideally a 5-processor machine should be able handle a problem 5 times as large as a single PC.
- Minimal additional investment required (time and money i.e. minimal additional hardware) to effectively run the software.
- The software must run effectively on both homogeneous and heterogeneous networked machines.

A requirement of the code developer is that a single source code unifies both the parallel and serial source code which facilitates easier code management. Source code consists of the high level programming statements that are created by a programmer with a text editor. The source code could use a series of compiler directives and conditional statements to differentiate between serial, master and slave code.

7.4.1 Types of computer networks

There are essentially two types of network that would be available to a FSE. These are homogeneous and heterogeneous networks.

A *homogeneous* network of computers is composed of a number of 'identical' machines. In the work presented in this thesis identical means that the CPUs in each computer are identical. Homogenous networks are commonly found in computer laboratories where the computers are normally upgraded or purchased simultaneously.

A *heterogeneous* network of computers is composed of a number of 'non-identical' machines. In the work presented in this thesis non-identical means that the CPU in at least one computer is different to the other computers that are used in that network. Heterogeneous networks are commonly found in office type environments when computers are bought, upgraded etc depending on the requirements of a particular user on an *ad hoc* basis. A more general definition could also include computers running different operating systems such as a mixture of UNIX, Linux and Windows. This usage is beyond the scope of this thesis with only Windows NT based machines considered. A homogeneous network can be considered to be a special case of the more general heterogeneous network.

Clearly, the parallel implementation must be capable of operating effectively on both homogeneous and heterogeneous networks to be usable in a typical FSE office environment.

7.4.2 Load balance

Load balance is the division of the computational workload over the participating processing units. In the ideal situation the best load balance is achieved when all the

processor are working at their highest workload i.e. no idle time. The time taken for the overall parallel computation is dictated by the time taken by the slowest process. In Figure 7-7 below 40 work units are distributed amongst 4 processors. A work unit takes 1 unit of time to be calculated assuming a homogeneous network of processors

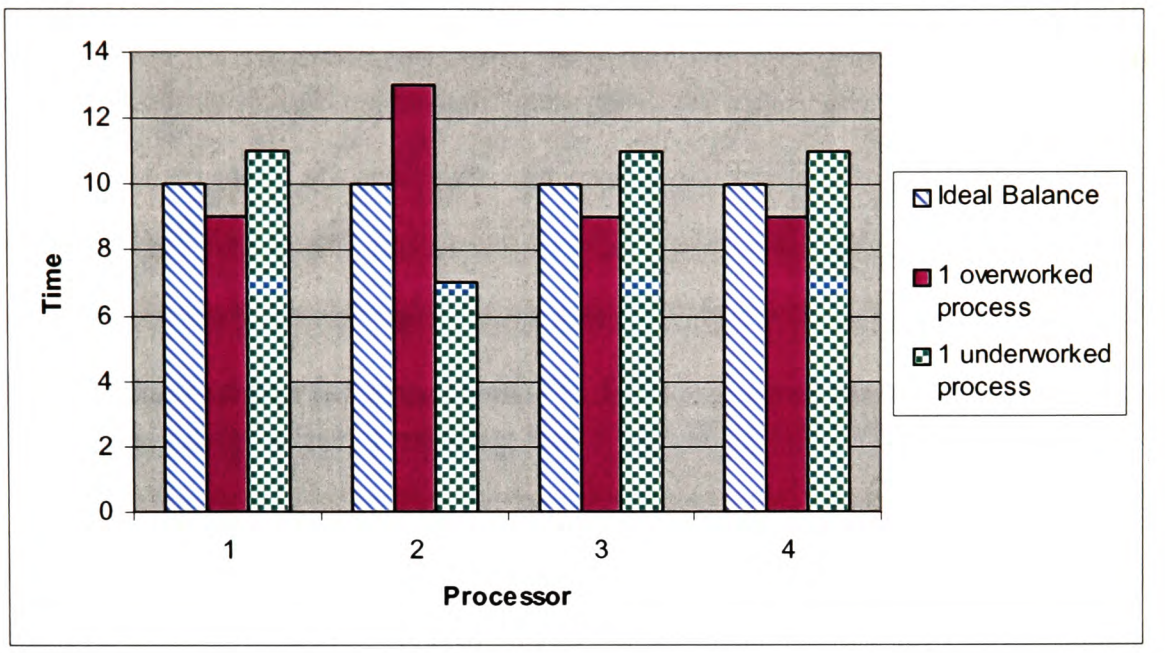


Figure 7-7 – Time taken to run a parallel problem dependent on load balance

It is important to ensure that that no one process has significantly more work than the average workload. This will cause all the other processes to idle whilst waiting for that process and cause an increase in overall runtime. It is not so much of a problem when one process has significantly less work than the average as the other processor will not be held up by that idle process. This is illustrated in Figure 7-7 with three possible workload distributions. The overall runtime is dictated by the height of the largest column for a particular load distribution.

For a system of homogeneous processors the most efficient load balance is generally easy to define by assigning a sub-domain of size NCELL / NP where NCELL is the total number of computational cells in the whole domain and NP is the number of processors, to each processor.

For the heterogeneous case where the processors will have varying performance characteristics the most efficient load balance becomes more difficult to assess. It could be naively assumed that the processor speed stated is directly related to the time taken to execute the SMARTFIRE code. From comparison of various machines this is not the case.

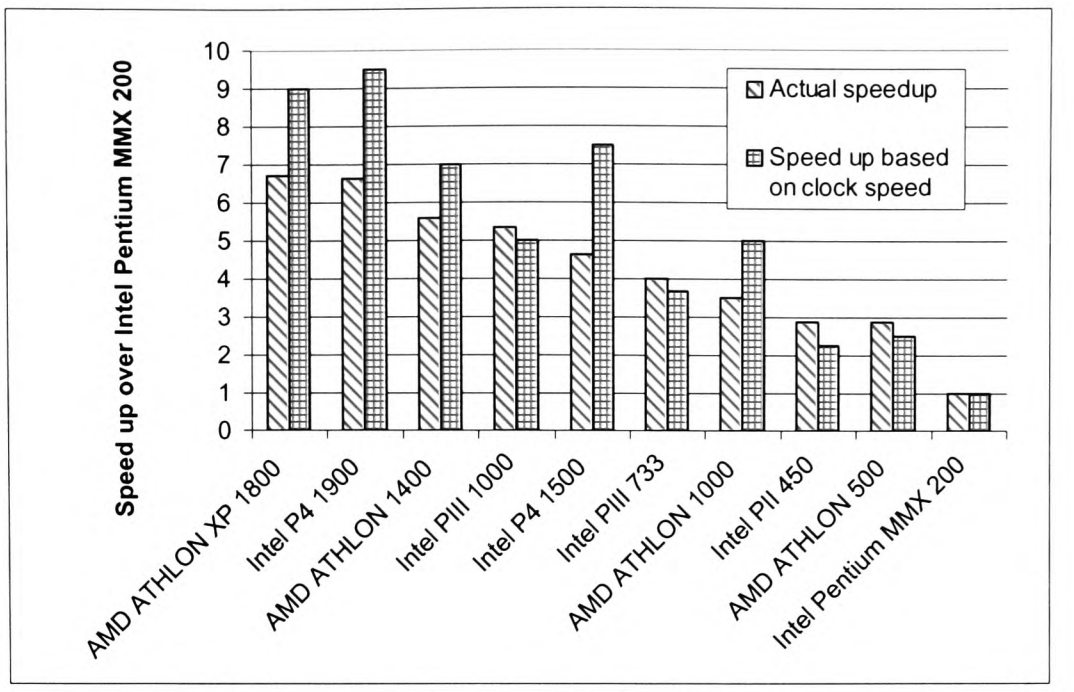


Figure 7-8 – Relative performance of PC CPUs based on clock speed and actual speedup when running the SMARTFIRE CFD code.

In Figure 7-8 it can be seen that the anticipated CPU 'clock speed' speed up generally does not match up to actual performance with a standard SMARTFIRE test case (LPC-007, see section 5.5.4).

A further complication is that different parts of the code execute at different speeds that may not be proportional to the speeds taken on other processors. This is due to the complex nature of modern day CPUs with various clock speeds and sizes of internal and external caches, the processor may be operating at the speed of the memory bus, L1-cache, L2-cache or the raw processor speed. The variation of MFLOPS with matrix size for the LINPAK benchmark case, a standard CPU benchmarking case, is illustrated in Figure 7-9 [Was2001] for a variety of CPUs.

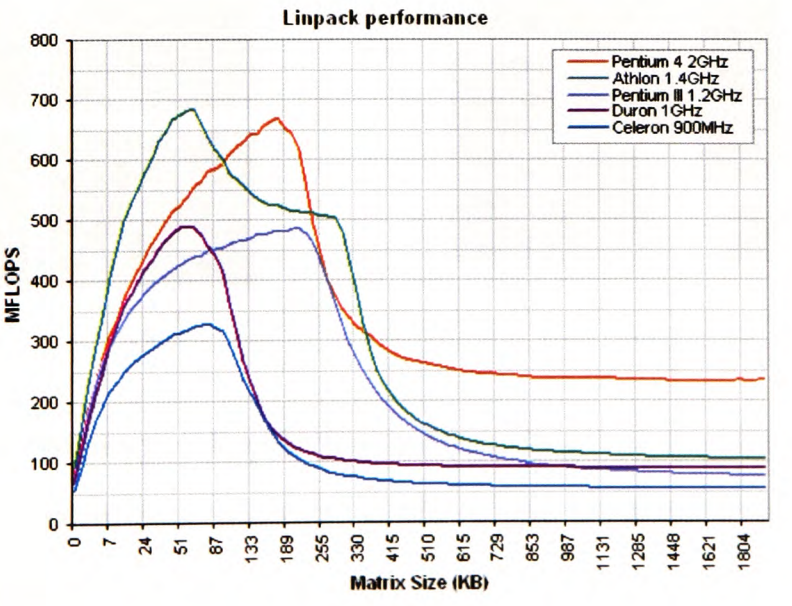


Figure 7-9 – Variation of MFLOPS against matrix size for five different CPUs

This performance variation could lead to difficulty in establishing an efficient load balance for a heterogeneous network. For a heterogeneous network of machines it is crucial that the actual speed of each processor is used to develop a good load balance and this would vary on a case by case basis.

Figure 7-9 also demonstrates the potential for super-linear speedups, a speedup that is greater than theoretically anticipated from the serial runtime of the code. As each processor gets a smaller amount of work to do then the MFLOPS rating of an individual processor increases. Whether such a super-linear speedup is possible is also dependent on other factors, particularly inter-processor communication behaviour and how the software is implemented.

7.4.3 Mesh Partition

The domain is subdivided into a number of sub-domains that each processor works on. This partition can be performed in a number of ways as illustrated in Figure 7-10 below with the decomposition of a simple 2D domain into 4 sub-domains.

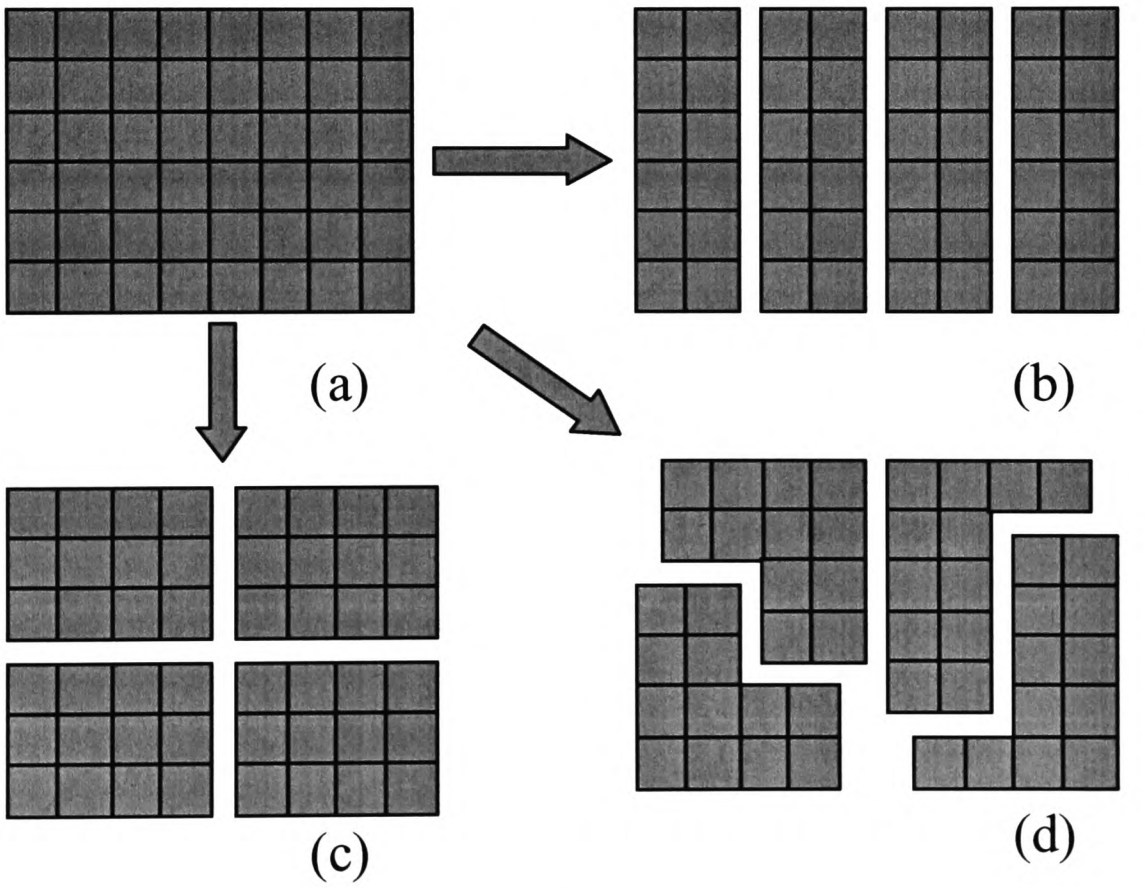


Figure 7-10 – Different ways of splitting a domain into equally sized sub-domains
Each of the domain decompositions (b, c, d) of the original domain (a) contain the same number of cells in each sub-domain but are quite different in shape. These

decompositions are just three examples and many other examples could be generated for this simple domain.

7.4.4 Halo Cells

Referring back to section 3.3.2 in the original discretisation process the value ϕ in a cell is related to the values of ϕ in the neighbouring cells. This is expressed by the matrix equation $\underline{\underline{A}}\underline{\phi} = \underline{b}$ (see section 3.2). However now that the domain has been divided between the different processors some of these neighbour values are no longer available at the partition. To remedy this, halo cells are added to the sub-domains to hold these values. The values in the halo cells are supplied, via message passing, by the neighbouring processor that does directly calculate those values in the computational cell. The (9 x 5) domain illustrated in Figure 7-11 is decomposed into three sub-domains, illustrated in Figure 7-12, each measuring 3 x 5 without halo cells. With the halo cells added (indicated with diagonal shading), the outer sub-domains measure 4 x 5, while the inner sub-domain measures 5 x 5. The arrows indicate the message passing of the data from the calculated cells of one process to the halo cell of the neighbouring process.

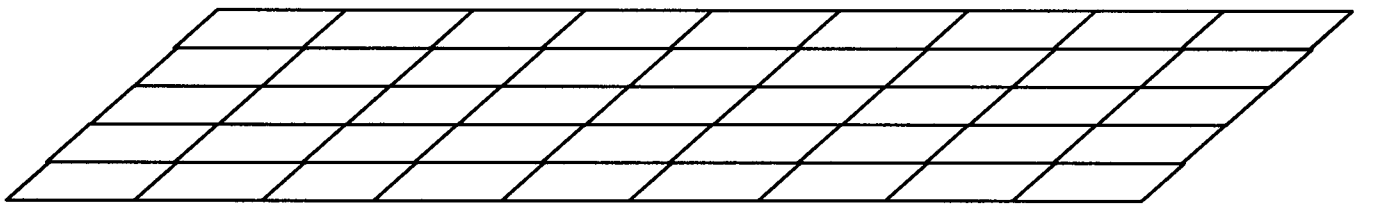


Figure 7-11 – 9 x 5 domain

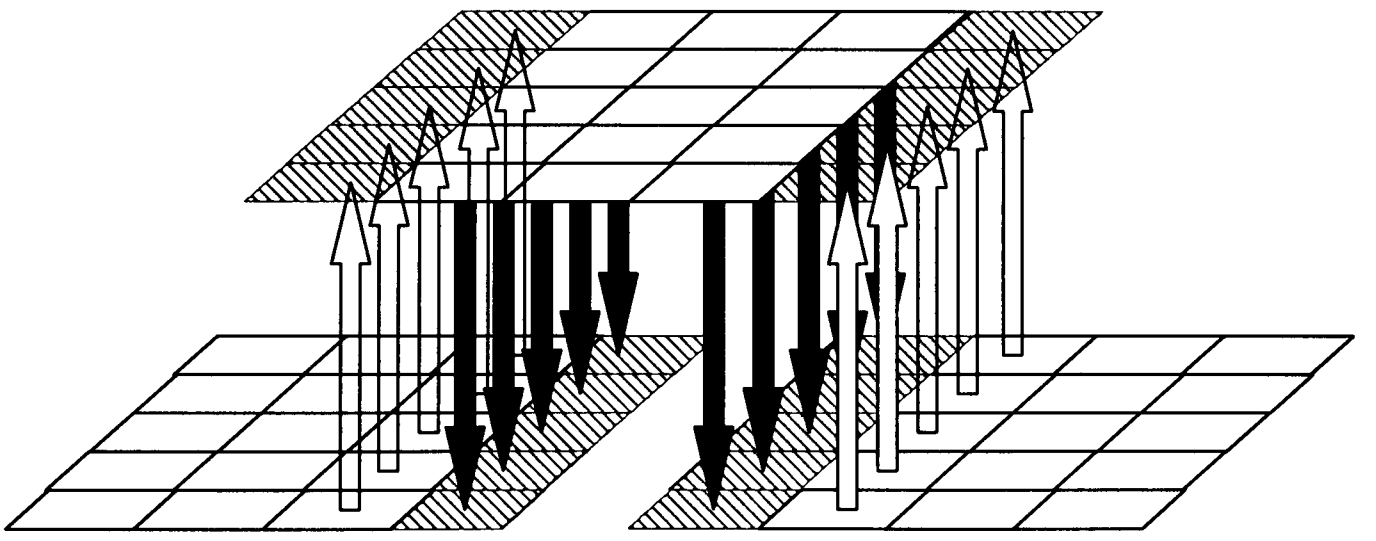


Figure 7-12 – Large domain subdivided with halo cells added for communication at the domain partitions

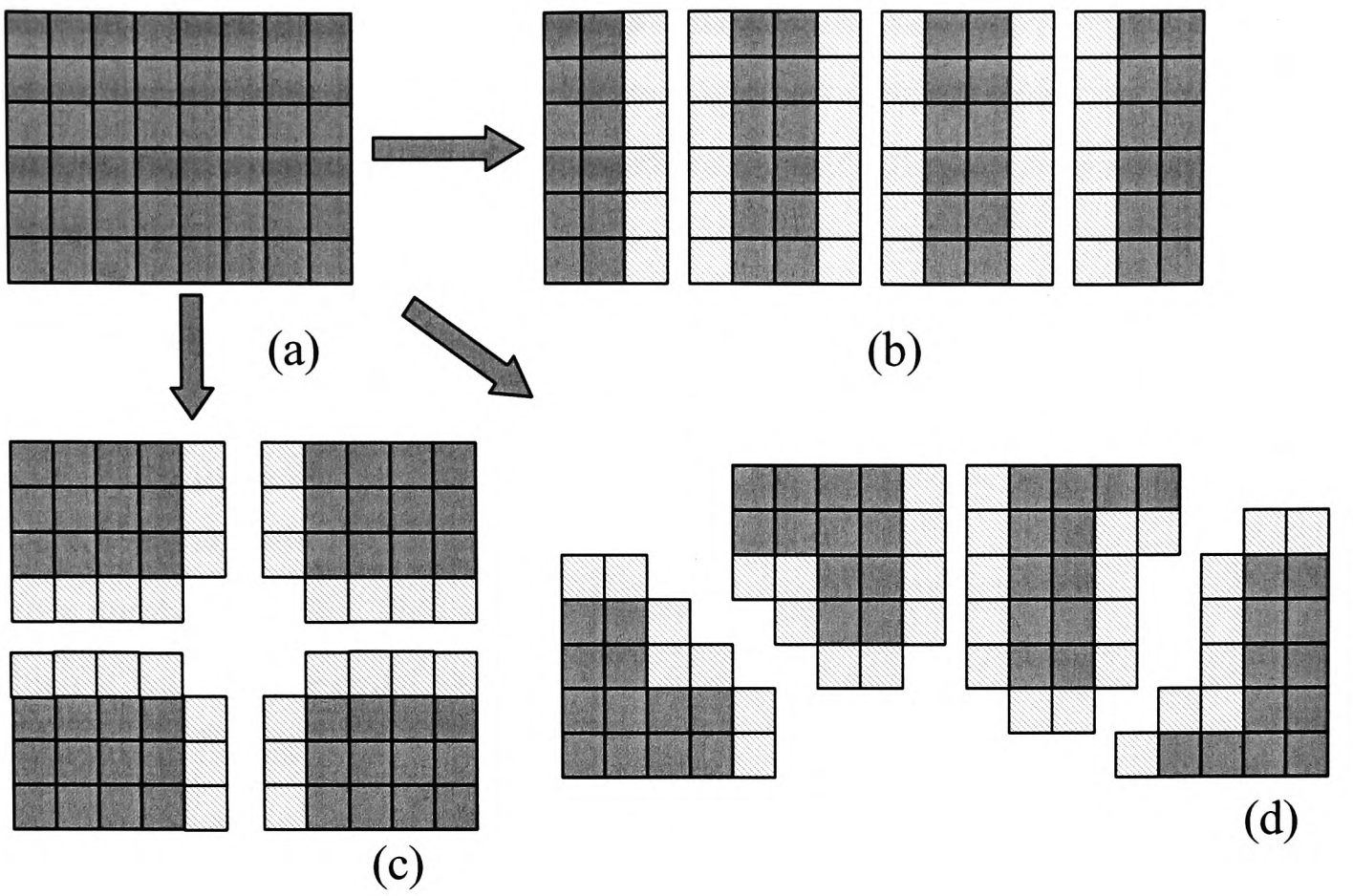


Figure 7-13 – Halo cells added to the partitions illustrated in Figure 7-10

The shape of the sub-domain is a crucial factor in determining the amount of communication required. This can be seen that in Figure 7-13 where the shapes used in Figure 7-10 now have the halo cells added. It can be seen that (c) has the least number of halo cells and only required communication with two neighbouring sub-domains. Domain decomposition (d) has the most halo cells and one of the sub-domains needs to communicate with three neighbouring sub-domains. The two important factors in communication are the size of the data communication, i.e. the number of halo cells, and the number of other processors, i.e. the number of adjacent sub-domains, which are communicated with by a processor. The time taken to communicate with another process is composed of two parts; the first part is the latency which is the time taken to initiate a communication, the second part is the bandwidth which determines how fast a data stream will be transmitted once this communication has been initiated. If a parallel system has a high communication latency and a high bandwidth this would suggest that minimising the number of messages broadcast is required for the shortest communication exchange. Conversely, if the parallel system has a relatively low latency and a low bandwidth would suggest minimising the size of the messages sent. The minimisation of the communication process, and hence calculating the shape of a sub-domain, can be complex and has been the subject of much research [Fox1988, RVD1993, Far1988, Wal1995,

KK1999]. These minimisation methods will not be considered further here as the domain decomposition used in this thesis was on a relatively small scale and the sub-domains were created using the straightforward method described in the following section.

7.4.5 One Dimensional Decomposition

In the work presented in this thesis a simple 1D partitioning scheme was generally used for the domain decomposition. This effectively divides the number of domains into a number of slabs. This scheme has been primarily selected due to the ease with which it was implemented. An example of this decomposition, without halo cells, is illustrated in Figure 7-14.

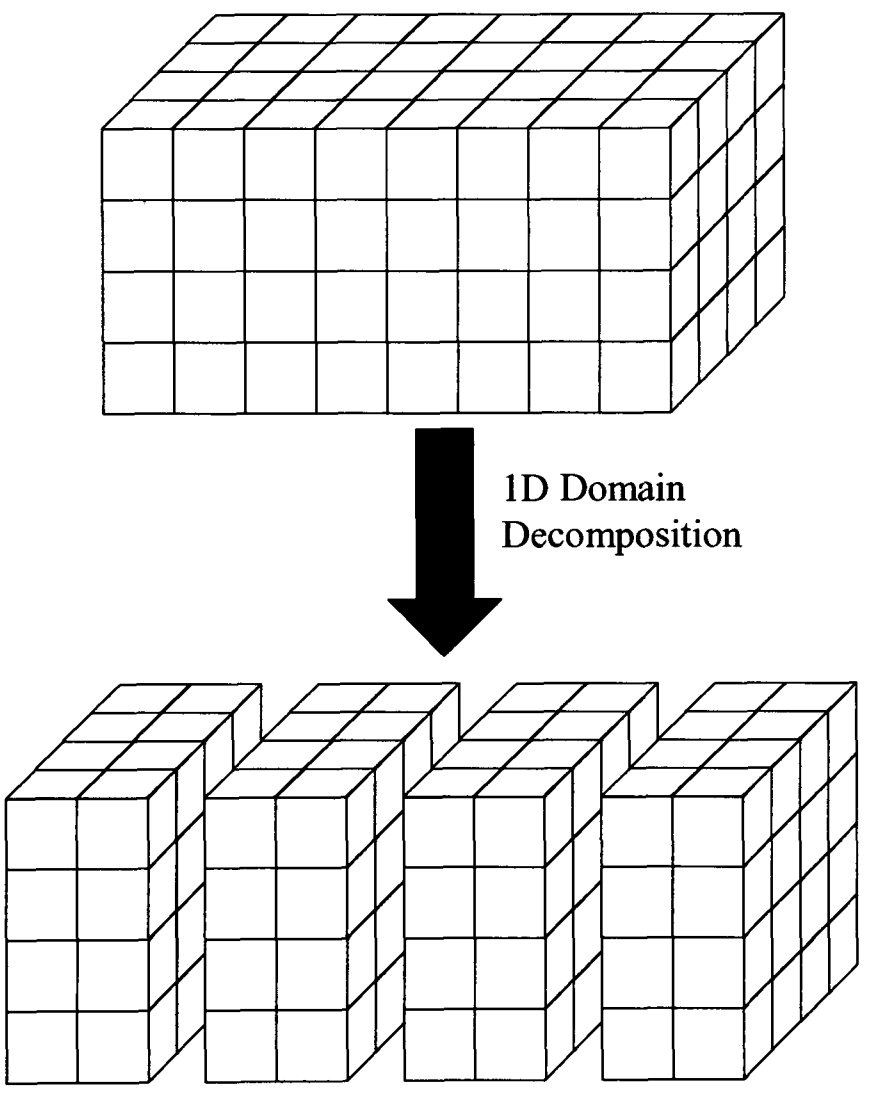


Figure 7-14 – Simple example of 1D decomposition scheme

The ease of implementation was due to the grid generation method used in the SMARTFIRE mesh generation program. This allows each sub-domain j to be easily defined as all the cells numbered from $((j-1)(NCELL/NP)+1)$ to $j(NCELL/NP)$ where NCELL is the total number of cells and NP is the number of processors

available. For the heterogeneous case the load balancing was performed using the following formula for the start (7.4.5.2) and end (7.4.5.3) points of the sub-domain.

$$n_j = \frac{\kappa_j}{\sum_{i=1}^{K} \kappa_i} \times NCELL \qquad (7.4.5.1)$$

$$start_j = \left(\sum_{i=1}^{j-1} n_j \right) + 1 \qquad (7.4.5.2)$$

$$end_j = start_j + n_j - 1 \qquad (7.4.5.3)$$

where κ_j is the relative performance of processor j compared to the fastest processor used in the computation and n_j is the number of cells assigned to the processor j.

The potential use of improved partitioning schemes will be discussed in the further work section.

7.4.6 Sub-domain renumbering

In Figure 7-15 below an example mesh has been divided into three sub-domains. At the edge of each sub-domain there are the halo cells. Each of these sub-domains is self-contained and the processor needs to store all the cell points, cells faces and cell values including the halo cells which make up the sub-domain.

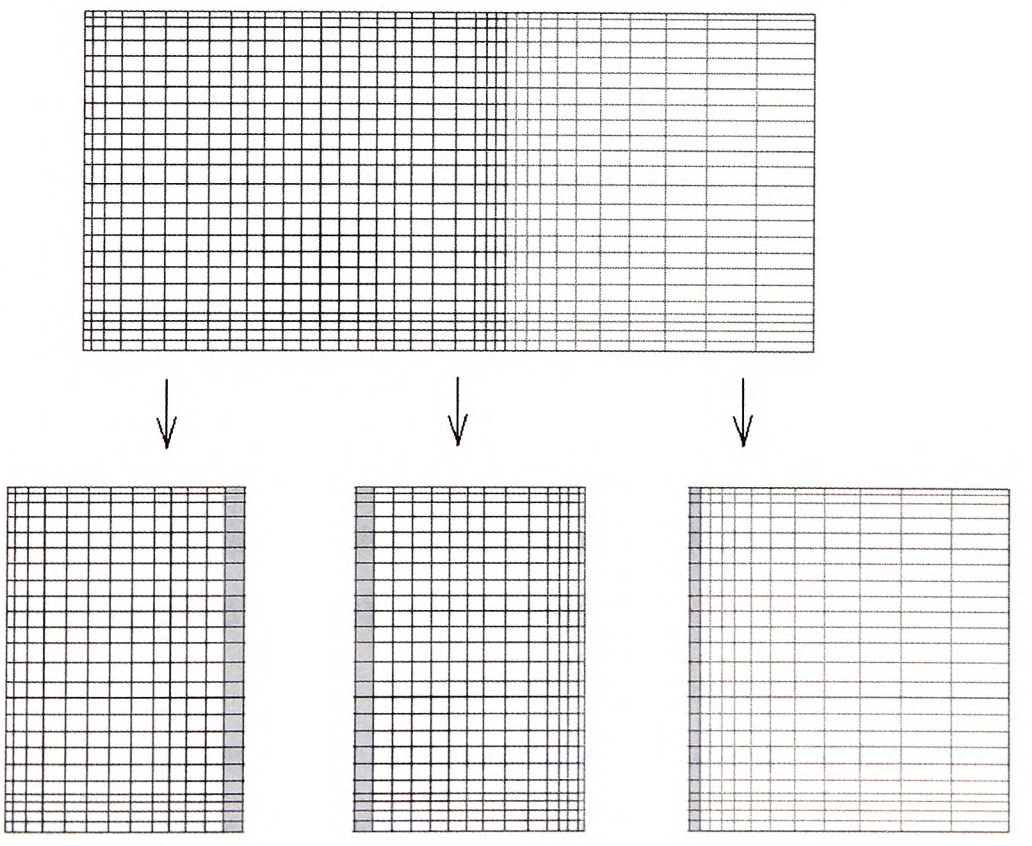


Figure 7-15 - Mesh partitioned into three parts with halo elements

Now that the domain decomposition has been decided each of the sub-domains were renumbered from the original global numbering so that the cells were contiguously numbered within the sub-domain from 1 to $N_{sub\text{-}domain}$. This is necessary so that the serial solution algorithms do not require modification to include additional pointers to the cell location. Renumbering is also required to avoid wasteful sparsely filled data arrays. The renumbering scheme is similar to that used by McManus [McM1996] and is detailed below:-

1. The number of cells, halo cells (sometimes referred to as 'outer halo' cells) and 'inner halo' cells are determined. Inner halo cells are the cells within the computational sub-domain that are mapped to the halo cells of an adjacent sub-domain.
2. A local2global array is created to index the global cell number, the number used in the original domain, to the sub-domain local cell number.
3. The global domain is looped over searching for 'inner halo' cells. When these are found the local_cell_num counter is incremented, local2global[local_cell_num] is set to the global cell number. The number of halos found between processors is noted.
4. The global domain is looped over searching for cells within the sub-domain that are have not already been found by step 3. When these are found the local_cell_num counter is incremented, local2global[local_cell_num] is set to the global cell number.
5. The global domain is looped over searching for halo cells. When these are found the local_cell_num counter is incremented, local2global[local_cell_num] is set to the global cell number.
6. Using the local2global array the cells and their adjacencies can be renumbered for the local domain.
7. The local2global array is transmitted to the appropriate slave process.
8. This process is repeated for each sub-domain.

The geometrical description of the sub-domain also consists of faces and points these are renumbered in a similar fashion to that of the cells.

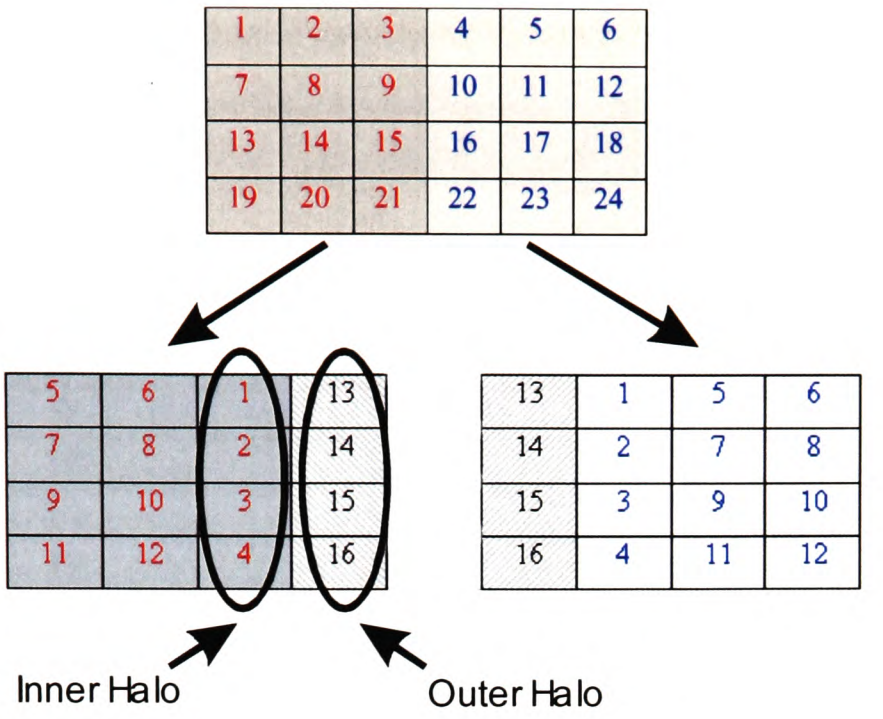


Figure 7-16 - Sub-domain partition and renumbering including halo cells

In Figure 7-16 above the result of the above algorithm on the cell numbering is illustrated. The global domain at the top of Figure 7-16 is divided into the two sub-domains below it. The first (left) sub-domain consists of global cells (1, 2, 3, 7, 8, 9, 13, 14, 15, 19, 20, 21) with global cells (4, 10, 16, 22) forming the outer halo. The second (right) sub-domain consists of global cells (4, 5, 6, 10, 11, 12, 16, 17, 18, 22, 23, 24) with global cells (3, 9, 15, 21) forming the outer halo. Each of these sub-domains is renumbered according to the scheme outlined above. Each step of the renumbering scheme will be demonstrated below for both the sub-domains. The renumbering scheme is performed by the master process.

1. The number of halo cells is 4, the number of inner halo cells is 4 and the number of computed cells is 12.
2. The local2global_cell array is created with a size of [computed cells + halo cells] 16.
3. The scan for inner halo cells leads to the first four elements of the local2global_cell being found:-

Local Cell Number	Global Cell Number 1st sub-domain	Global Cell Number 2nd sub-domain
1	3	4
2	9	10
3	15	16
4	21	22

4. The scan for the other cells in the sub-domain extends the array as follows:-

Local Cell Number	Global Cell Number 1st sub-domain	Global Cell Number 2nd sub-domain
5	1	5
6	2	6
7	7	11
8	8	12
9	13	17
10	14	18
11	19	23
12	20	24

5. The scan for the (outer) halo cells now extends the array:-

Local Cell Number	Global Cell Number 1st sub-domain	Global Cell Number 2nd sub-domain
13	4	3
14	10	9
15	16	15
16	22	21

6. The cells and their table of adjacencies can be renumbered using the local2global_cell array. For example local cell 1 is equivalent to global cell 3. It can be seen that global cell 3 is adjacent to global cells 2, 4 and 9 (see Figure 7-16). Using these global values in the local2global_cell array it can be seen, from the tables above, that local cell 1 is therefore adjacent to local cells 6, 13, 2.

Once the local2global arrays have been generated for each sub-domain a local2global_master_array is created, on the master process, by appending the local2global arrays from all the sub-domains, excluding the outer halo cells, together in order of the sub-domains. From the local2global_master_array array a global2local_master_array is formed. Using the above example sub-domain results in (local2global_array on the left and global2local_array to the right):-

Local2global master no.	Global Cell Number
1	3
2	9
3	15
4	21
5	1
6	2
7	7
8	8
9	13
10	14
11	19
12	20
13	4
14	10
15	16
16	22
17	5
18	6
19	11
20	12
21	17
22	18
23	23
24	24

Global Cell Number	Local2global master no.
1	5
2	6
3	1
4	13
5	17
6	18
7	7
8	8
9	2
10	14
11	19
12	20
13	9
14	10
15	3
16	15
17	21
18	22
19	11
20	12
21	4
22	16
23	23
24	24

The global2local_master_array is used for the purposes of I/O that is described in section 7.4.11.

7.4.7 Halo communication

The halo communication scheme is illustrated in Figure 7-17. The data in the halo is required for calculation of the values within the sub-domain. The values in the halos are obtained by a communication with the neighbouring process. In Figure 7-16 the outer halos the cells are stored contiguously due to the renumbering scheme. The inner halos are not necessarily stored contiguously and it would be impossible in some cases to achieve this (e.g. Figure 7-18). The inner halos would however be in the same order as the order in the outer halo of the neighbouring sub-domain. The inner halos are sorted using redirection pointers and sent to the outer halo of a neighbouring process, due to the consistent renumbering scheme used throughout no re-ordering is necessary at the receiving process.

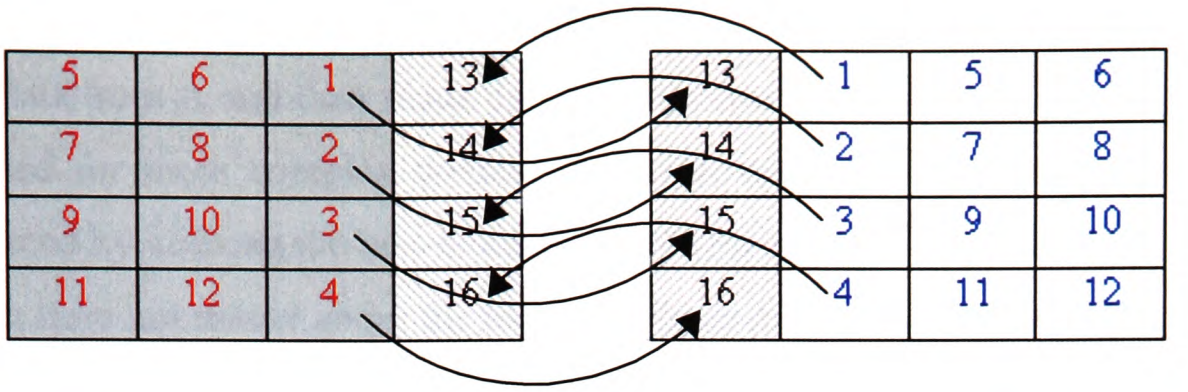


Figure 7-17 - Halo communication update scheme

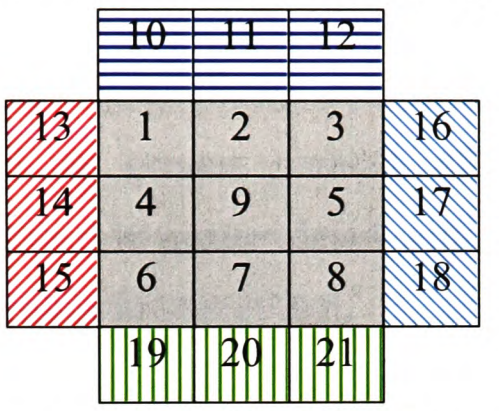


Figure 7-18 – Sub domain with many halos

It is not always necessary to communicate the values to the halos. For example setting a variable to a fixed value over the whole sub-domain does not require any communication. Values can sometimes be inferred from data already on the halo. For example it is possible to calculate the density of the halo cell using the temperature already communicated from the neighbouring sub-domain. Whether it is faster to communicate the value or calculate the value is going to be dependent on the speed of the necessary calculation against the speed of the communication. On a PC attached to a LAN calculation will be far quicker than communication so this has been implemented throughout the software.

7.4.8 Communication Deadlock

If blocking communications were used care would be required to ensure that the processes were properly synchronised so that no deadlock occurs. This is simply illustrated by the following example. If processor A sends some data to processor B it cannot continue code execution until B receives the data. At the same time processor B sends data to processor A and cannot continue code execution until A receives the data. Both processors are now caught in a deadlock as they are both sending data and are unable to receive the data from the other process to allow the other process to continue. This particular case is easily resolved by ensuring that processor A sends

data to B, then receives the data from processor B. At the same time processor B receives data from A and then sends data to processor A. This situation can get more complicated for more complex domain partitions but the entire problem has been circumvented by utilising the non-blocking communication modes of MPI. Using this protocol it does not matter about the order in which the sends and receives are posted by the processors. A blocking call is required afterwards to ensure that the sends and receives have been completed. After this blocking call the communicated data can be safely used. The use of non-blocking communication modes also allows asynchronous communications to be used. This allows computations and calculation to be overlapped and may lead to greater parallel efficiency. This asynchronous communication will be described in greater detail with reference to parallelisation of the numerical solvers used within SMARTFIRE.

7.4.9 Algorithmic considerations

The solution algorithm consists of solving equations and calculating required physical properties. For some of these physical properties it was necessary to communicate the values, whilst for others they may be communicated or calculated. For each physical equation there were essentially two phases, calculation of the matrix coefficients and then solving the equation using the matrix coefficients previously calculated. In both of these phases it was necessary to communicate via the halo cells.

The following figure shows a simple 8 cell domain with its associated matrix equation. The matrix equation is $\underline{\underline{A}}\underline{\phi} = \underline{b}$ (see section 3.2) where A is an 8 x 8 sparse matrix and ϕ, the solved variable, and b are vectors of length 8.

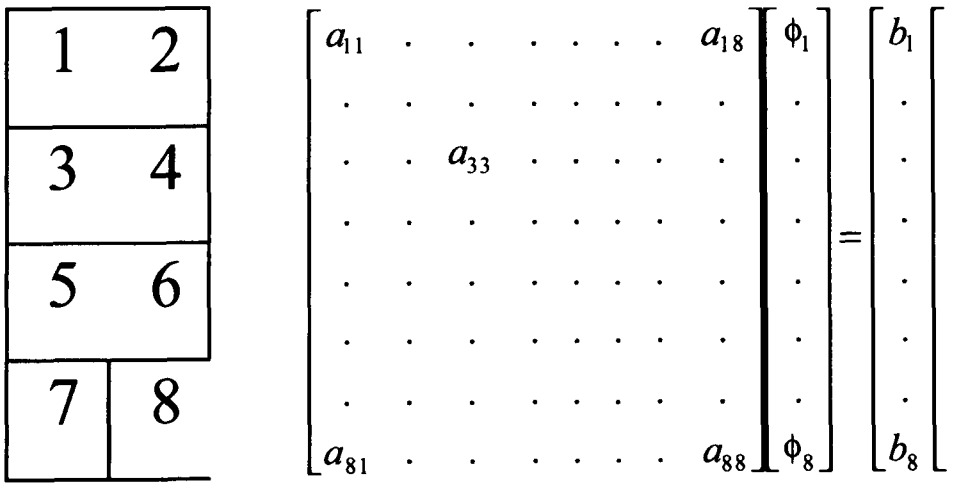


Figure 7-19 – Simple eight element problem space with matrix

When the problem in Figure 7-19 is decomposed the matrix equations for each of the two sub-domains becomes 4 x 6 for A, the b vector has a length of 4 and the ϕ vector a length of 6.

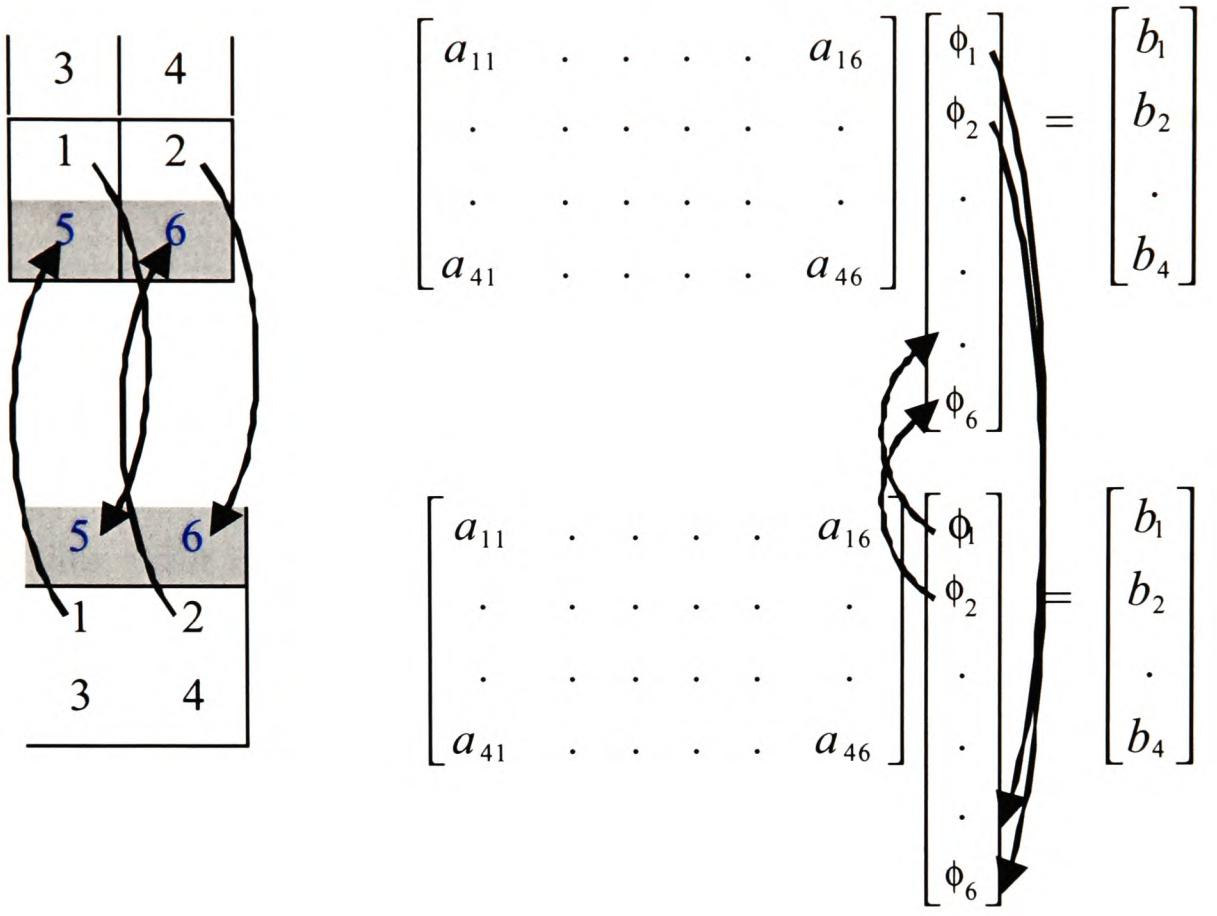


Figure 7-20 - Simple eight element problem decomposed into two smaller domains.

Figure 7-20 also illustrates the relationship between the physical decomposition of the domain and the decomposition of the mathematical matrices. The two sub-domains are coupled together by their halos. From the matrices it can be seen that a set of solution equations only exist for ϕ_1 to ϕ_4. ϕ_5 and ϕ_6 are the halo values; these have to be communicated from the adjacent sub-domain.

As mentioned in section 3.13 the matrices are solved using iterative methods, namely the JOR and SOR solvers. The value of ϕ_p is calculated using *known* values of ϕ_k, $\underline{b}$ and $\underline{\underline{A}}$. In the case of the JOR solver this is expressed as:-

$$\phi_p^{i+1} = (1-\alpha)\phi_p^i + \frac{\alpha}{A_{pp}}\left(b_p - \sum_{k \neq p} A_{pk}\phi_k^i\right) \qquad (7.4.9.1)$$

In the serial case the above iterative formula is used with the following algorithm.

1) loop over the cells p = 1 to NCELL applying (7.4.9.1) to calculate ϕ_p.
2) repeat 1) until the prescribed number of iterations have been performed or convergence has been achieved (i.e. $\left|\phi_p^{i+1} - \phi_p^i\right| < tolerance \quad \forall p$).

In the parallel case this above algorithm was modified slightly and utilised asynchronous communication. Asynchronous communications allows computation and communication to be overlapped which should reduce/hide the time taken to perform communications which would be apparent in blocking communication. Workstation clusters have a far higher communication cost than other forms of more specialised DM parallel machines so asynchronous communication is even more important. The communication needs to be tested for completion at some future point, only at this point can the data communicated be safely used and the computation between the communication initiation and completion must be independent of the data communicated.

1) loop over the cells p = 1 to $N_{inner\_halo\_cells}$ to calculate ϕ_p. This uses the values from the halos which have been updated in the previous ith iteration.
2) Communicate the inner halo cells to the outer halo cells of the adjoining sub-domains. These values will not be accessed until step 1 in the *next* iteration.
3) Loop over the cells p = $(N_{inner\_halo\_cell} + 1) - (N_{sub\text{-}domain\_cells})$ for the rest of the sub-domain.
4) Ensure the communication initiated in step 2 has completed and wait, if necessary, until it has.
5) Repeat 1) until the prescribed number of iterations has been performed or convergence is achieved.

The above scheme was similarly implemented for the SOR solver.

7.4.10 Effect of decomposition on the solution scheme.

The decomposition had no effect on the calculated properties (e.g. pressure gradients), matrix coefficients and the source terms. The JOR solver was likewise unaffected by the decomposition process. This is because the halo values are updated before the end of the previous JOR inner iteration thus ensuring behaviour consistent with the serial

scheme. The new value of ϕ_p^{i+1} is only dependent on the previous neighbouring values of ϕ_k^i. It should be noted that the cell ordering should make no difference to the value of ϕ_p^{i+1} obtained due to the calculation being dependent on values from the previous iteration and independent of the new values ϕ_k^{i+1}. However, there may be some discrepancies due to machine precision but this should be a minor effect.

The parallelisation process affected the SOR scheme slightly. This is due to the halo always containing values calculated at the last iteration as opposed to the newest possible value that may be required in order to be completely consistent with the serial scheme. For the SOR solver the cell ordering will affect the intermediate calculated values although the converged values should be the same. However, this would also be the case for the serial algorithm with a differing arbitrary mesh ordering. This small difference has little effect in practice and dispenses with the need for possible costly communication that would be required to ensure a completely consistent algorithm. Other values that may be affected are the mass error and solver residuals. This is due to machine precision that could also occur due to a different summation order (i.e. different mesh ordering) and is likely to be a very minor effect.

The original serial version of SMARTFIRE's SOR solver :-

```
for (num_of_iters = 1; num_of_iters <= max_iterations; num_of_iters++){

   used_iterations += 1;
   for (cell_num = 1; cell_num <= domain.num_of_cells; cell_num++){
      preval = local_cell[cell_num];
      curval = source[cell_num];
      l_s_coeff = s_coeff[cell_num];
      l_c_index = c_index[cell_num];
      for (indx = 2; indx <= c_index[cell_num][1]; indx++){

          curval    -=    l_s_coeff[indx]    *    cell[l_c_index[indx]]->access(NEWEST,
var_point);

      }

      curval /= l_s_coeff[1];
      if ( do_solver_relax ){
         curval = preval + variable[var_point].get_solver_relaxation() * (curval -
preval);
      }
      cell[cell_num]->access(NEWEST, var_point) = curval;
   }
}
```

This scheme is an accurate interpretation of the SOR solver. The cell object (double underlined) was used to store all the properties and values of a particular cell and was

derived from an object-orientated analysis of the problem [Ewe2000]. For the parallel version it was more convenient to store the values in a 1D array of floating point numbers. This allowed the standard MPI commands to be used for communicating the values to other processes. Also at this point of the calculation only one value from the cell is being changed and it would be wasteful of communication to send the whole object. The first phase of parallelism was to convert from the use of objects to a 1D array. This change was local so that the cell object concept was maintained throughout the rest of the code. This is illustrated in the following fragment of code, where the amendments to the code are in ***Bold Italics***:-

```
for (cell_num = 1; cell_num <= domain.num_of_cells; cell_num++){
      local_cell[cell_num] = cell[l_c_index[indx]]->access(NEWEST, var_point);
}
for (num_of_iters = 1; num_of_iters <= max_iterations; num_of_iters++){

   used_iterations += 1;
   for (cell_num = 1; cell_num <= domain.num_of_cells; cell_num++){
      preval = local_cell[cell_num];
      curval = source[cell_num];
      l_s_coeff = s_coeff[cell_num];
      l_c_index = c_index[cell_num];
      for (indx = 2; indx <= c_index[cell_num][1]; indx++){

         curval -= l_s_coeff[indx] * local_cell[l_c_index[indx]];

      }

      curval /= l_s_coeff[1];
      if ( do_solver_relax ){
         curval = preval + variable[var_point].get_solver_relaxation() * (curval -
preval);
      }
      local_cell[cell_num] = curval;
   }
}
for (cell_num = 1; cell_num <= domain.num_of_cells; cell_num++){
   cell[l_c_index[indx]]->access(NEWEST, var_point) = local_cell[cell_num];
}
```

The 1D local_cell array is now used to store the values which were previously held in the object cell. In the code fragment the variable of interest is initially copied into local_cell. This replaced all references to the cell object within the SOR/JOR algorithms. The values were copied back into cell object after the solver iterations are completed.

This transformation also significantly increased the performance of the serial code. This improvement was probably due to the removal of the object de-referencing of the cell object within the iterative process and improved use of the cache (higher cache hits) due to the use of 1D arrays. Typically on a 733MHz Pentium III processor this yielded a speed up of 1.4 – 1.5 over the original serial code for standard problems like the Steckler room (section 5.5.1) and LPC case (section 5.5.4). On a 400MHz

Pentium II processor speedups of about 1.3 were obtained. The speedup is highly case and processor dependent but does appear to give some improvement in all cases. This amendment was integrated into the serial code (from SMARTFIRE v2.01 b369D). All parallel speedup comparisons were performed using this enhanced serial code.

The next phase of the parallelising of the code was to split the code into two parts for the asynchronous communication as described previously and the addition of the message passing communication calls. The amendments to the code are shown in ***bold Italics***.

```
for (num_of_iters = 1; num_of_iters <= max_iterations; num_of_iters++){

   used_iterations += 1;
   for (cell_num = 1; cell_num <= domain.num_of_inner_halo_cells; cell_num++){
      preval = local_cell[cell_num];
      curval = source[cell_num];
      l_s_coeff = s_coeff[cell_num];
      l_c_index = c_index[cell_num];
      for (indx = 2; indx <= c_index[cell_num][1]; indx++){

         curval -= l_s_coeff[indx] * local_cell[l_c_index[indx]];

      }

      curval /= l_s_coeff[1];
      if ( do_solver_relax ){
         curval = preval + variable[var_point].get_solver_relaxation() * (curval -
preval);
      }
      local_cell[cell_num] = curval;
   }
   async_swap_halo_sol();
   for (cell_num = domain.num_of_inner_halo_cells+1; cell_num <=
domain.num_of_cells_ph; cell_num++){
      …
      … // same as above loop
      local_cell[cell_num] = curval;
   }
   async_sync_halo_sol();
```

The communications were initiated using ***async_swap_halo_sol()***, this sends the overlap data to the neighbouring sub-domains and initiates the non-blocking receive of data from the neighbouring sub-domains. The ***async_sync_halo_sol()*** ensures that the processes were blocked until the non-blocking send/receives have been completed.

The matrix coefficients were calculated using a similar scheme where the 1 to domain.num_of_inner_halo_cells cells were calculated, these values would then be communicated to the halos of the adjacent sub-domains while the rest of the values were calculated for the sub-domain.

7.4.11 Effect of Parallel Processing on input

The input for the SMARTFIRE CFD engine is generally composed of a case specification script file, a restart file and a geometry file. The case specification script file consists of information such as the relaxations, solver iterations, values associated with boundary conditions, and miscellaneous physical data. The restart data file contains data necessary to start a simulation from a previously saved simulation and acts as a results file as well. This data file contains all the variables that define the state of the system such as pressure, temperature, velocity, etc at a particular instance in time at all the discrete points of the computational grid. A restart file is not always necessary, and simple initial conditions (e.g. known ambient conditions) can be specified in the case specification file. The geometry file contains information about the computational grid, the points that make up the mesh and the relationships between the points, faces, cells and the references to volumetric and facial boundary conditions specified in the case specification script file.

7.4.11.1 Case Specification File

The case specification script is a text file designed to be read and modified by the user if necessary. The parser extracts one line of text from the case specification script at a time and then interprets this line and continues this process until the end of the case specification file is reached. Essentially the parser works as follows:-

1) Open case specification script
2) Read_line (copy line from case specification script to string
3) Interpret string
4) Read_line
5) Interpret string

..

n-1) End of file reached

n) Close case specification file.

In the DM version of SMARTFIRE the master and slave run the same parser as the serial code. There are two differences between the master/serial code and the slave code. The first difference is that only the master/serial process opens and closes the

file; the second difference lies in how the line of text is obtained. On the master process the line is directly read from the case specification script, this line is then broadcast to all the slaves. On the slave processes the string is received from the master process. This is achieved by modifying the read_line routine to include a MPI_Bcast call:-

```
void read_line( ifstream& from_file, char text_line[]){
// Line reading routine
   ..
   if ( not_slave_process() ) {
      // Only do this section of code if master or serial process
      // Read in the contents of the line from command script
      // stripping out unnecessary white space
      ..
      strcpy(text_line, (buff+start));
   }
   if ( control.use_mpi ) { // only do this in parallel implementation
      MPI_Bcast(text_line, MAX_LINE_LENGTH, MPI_CHAR, 0, MPI_COMM_WORLD);
      // This broadcasts the text_line to the slave processes from the
      // master process.
   }
   return;
}
```

The advantage of this method of extracting the command script information is that no additional work is required to maintain the parser than the work that has already been done on the serial code; the code is identical on serial, master and slave processes.

7.4.11.2 Restart Data File

This file is stored according to the original serial global cell numbering and needs to be distributed to the partitioned sub-domains on the slave processes. The restart file is formatted so that all the cell values of pressure are stored contiguously followed by the cell values of temperature and so on. The algorithm is as follows:-

1) On the master process the cell values for a particular variable are read from the restart file into a temporary storage area, the global domain variable storage area.

```
For ( lcell = 1; lcell <= tot_global_cells; lcell++ ) {
   Read_value_from_restart(value);
   Work_space[1][global2local_cell_master_array[lcell]] = value;
}
```

The global2local_cell_master_array index ensures that the values are distributed in the same order as the local numbering schemes of the sub-

domains. The Work_space[1], on the master process, is dimensioned to the number of global cells.

2) The variables in the global temporary storage are transmitted to the appropriate slave processes using the following algorithms:-

On the Master process

Receive the number of cells in the sub-domain from the slave process.

Send the Work_space array elements offset to (offset + sub-domain size) to the slave process, where the offset for the jth domain is $\sum_{i=2}^{j} sub\_domain\_size_{i-1}$. For the first domain the offset is zero.

Repeat this step until all the slaves have received the data.

On the Slave Processes

Send out the number of cells in the sub-domain to the master process.

Receive the restart data from the master process. This data will be correctly mapped to the sub-domain ordering.

3) The above steps are repeated for each variable.

7.4.11.3 Geometry File

Initially the full adjacency information is read onto the master process. From this data the sub-domains and halos are created and renumbered as described previously. The code then allocates the required amount of memory required for each sub-domain on each computer. The geometry data is then read in, by the master process, partitioned and sent to the appropriate slave process in a similar manner to the restart data.

7.4.12 Effect of Parallel Processing on Output

The problem and therefore the results are distributed across a number of computers and need to be collected together and saved to a result file. This result file must be ordered in the same way as the original serial global domain, to ensure compatibility with the serial code. The result file is also the restart file. This process is essentially the compliment of the input of the restart data.

1) The variable values on the slaves are transmitted to the master processes using the following algorithms:-

On the Master process

Receive the number of cells in the sub-domain from the slave process.

Receive the Work_space array elements offset to (offset + sub-domain size) from the slave process, where the offset for the jth domain is $\sum_{i=2}^{j} sub\_domain\_size_{i-1}$. For the first domain the offset is zero.

Repeat this step until the master has received the data from all the slaves.

On the Slave processes

Send out the number of cells in the sub-domain to the master process.

Send the restart data from the master process. This data will be correctly mapped to the sub-domain ordering.

2) On the master process the values in the temporary workspace are written to the restart file for a particular variable using the code below:-

```
For ( lcell = 1; lcell <= tot_global_cells; lcell++ ) {
   Value = work_space[1][global2local_cell_master_array[lcell];
   Write_value_to_file(Value);
}
```

The global2local_cell_master_array index ensures that the values are written to the restart file in the same order as the global numbering scheme. The Work_space[1], on the master process, is dimensioned to the number of global cells.

3) The above steps are repeated for each variable.

7.4.13 Program Amendment

The serial code required amendment to produce the DM version. Approximately 4000 lines of new code were required. About 200 lines of code were needed for the parallel communication necessary during the computational phase of the code. About 500 lines of code were required for the renumbering, load balancing and memory allocation processes. The rest of the amendment was primarily concerned with the I/O process associated with the geometry file, the restart file and other output files.

These amendments have been added in such a way as to allow the same source code to be used for both the serial and parallel code. Looking at the code for the modified read_line routine (see section 7.4.11.1) the same code works in parallel and serial with the parallel parts only activated if the program is running in parallel. Referring back to the SOR solver code (section 7.4.10) the ***domain.num_of_inner_halo_cells*** is set to zero in the serial mode. This means that the first loop is effectively ignored. The parallel communications calls would be immediately returned, i.e. effectively ignored, and the parallel SOR code is reduced back to the serial SOR behaviour. For the I/O associated with the geometry, restart and results files much of the code used in parallel and serial is different and requires software switching between the two types although the amendments are made at the lowest subroutine level possible, i.e. the parallel changes are not apparent higher up the subroutine calling tree.

7.5 Concluding remarks

Two possible methods of parallelisation that maybe suitable for use by a FSE within an office environment have been discussed. These were the shared memory (SM) and the distributed memory (DM) approach. The use of multithreading on a multiprocessor PC (SM) is currently limited by the scale of multiprocessor motherboards currently available. This approach therefore offers little potential for large scale speedup. The DM architecture is more useful as most engineering offices consist of many networked PCs. The domain decomposition strategy was selected for parallel processing as the other possible strategies are impractical. In this strategy the problem is parallelised by sub-dividing the problem space between the processors and each processor runs the same code (SPMD) with I/O and problem distribution and renumbering carried out by the master process.

The renumbering of the sub-domains has been discussed and along with the I/O issues constituted the majority of the programming effort required. With the renumbering performed only minimal changes were required for the rest of the serial algorithm. These changes were essentially to do with halo communications between the sub-domains. The parallel implementation utilised 1D arrays to simplify and speedup the parallel processing. As a consequence of the use of 1D array for the solvers it was

found that a significant speed-up was achieved in the serial version over the previous serial version. This amendment was therefore included in both the parallel version and future serial versions of the code.

The code was written in such a way to ensure that the source code could be used to generate both serial and parallel versions of SMARTFIRE to ease code maintenance. It should be noted that all the models and algorithms including combustion and all radiation models have been included in the parallel version of the code.

In the next chapter the processing performance of the parallel implementation of SMARTFIRE will be examined.

8 Parallel Processing Performance of DM - SMARTFIRE

In this chapter the actual performance of the DM-SMARTFIRE parallel code will be assessed. Initially a feasibility study was carried out to ensure that the effort of parallelising SMARTFIRE was worthwhile. The performance of DM-SMARTFIRE on both homogeneous and heterogeneous networks of PCs was examined. The effects of running DM-SMARTFIRE upon the activities of other users within an office environment were examined and a solution to the problems caused to the FSE and other computer users was implemented and tested. The theoretical memory usage for a network of homogeneous PCs will be examined. The likely impact on parallel performance of future developments in computer hardware will also be discussed.

8.1 Performance Measures

The most obvious parallel performance measurement for a FSE is the time taken by the parallel network of computers compared to that of a solitary serial PC. However, sometimes such measures are not possible such as when the problem can be fitted onto a network of PCs but not onto the single serial machine. When measuring performance it is common practice for timings of CFD codes to simply measure the time taken to perform the computation and ignore the time spent in I/O, such as reading the problem specification input files and the writing of the final result files. This is entirely reasonable as these I/O processes are essentially a one off investment and does not increase with additional iterations being performed on the problem.

Speedup (S_n) can be defined as the ratio of the runtime on a single processor t_1 compared to the runtime on the parallel 'machine' with n processors t_n for a particular problem.

$$S_n = \frac{t_1}{t_n} \tag{8.1.1}$$

In the ideal case the speedup will equal to the number of processors assuming a homogenous processor network is used. In the case of a heterogeneous network of processors, a network composed of differing processors, the ideal speedup would be $\sum_{i=1}^{n} \kappa_i$ where κ_i is the performance of processor i compared to the single processor

used for the speedup comparison. This performance κ_i is the time taken to run the serial problem on the comparative processor divided by the time taken to run the same problem on processor i. This ideal speedup is virtually impossible to achieve for a CFD code with performance dropping off with increasing numbers of processors. For a fixed problem, as the number of processors increases, the compute time on each processor decreases while the communication time remains constant or increases slightly [Joh1992]. It is this reduction in the computation to calculation ratio that accounts for most of the drop in parallel performance with increasing numbers of processors. This communication accounts for part of the parallel code that is inherently serial; this fraction of the code f_s ($0 < f_s < 1$) places a restriction on the possible speed up S_{max} and is known as Amdahl's law [Amd1967].

$$S_n^{\max} = \frac{t_1}{\frac{(1-fs)t_1}{n} + f_s t_1} \tag{8.1.2}$$

as $n \to \infty$

$$S_\infty^{\max} = \frac{1}{f_s} \tag{8.1.3}$$

Amdahl's law is illustrated in Figure 8-1 below for the case when the serial fraction f_s is 0.05. This leads to a theoretical maximum speedup of $S_\infty^{\max} = 20$.

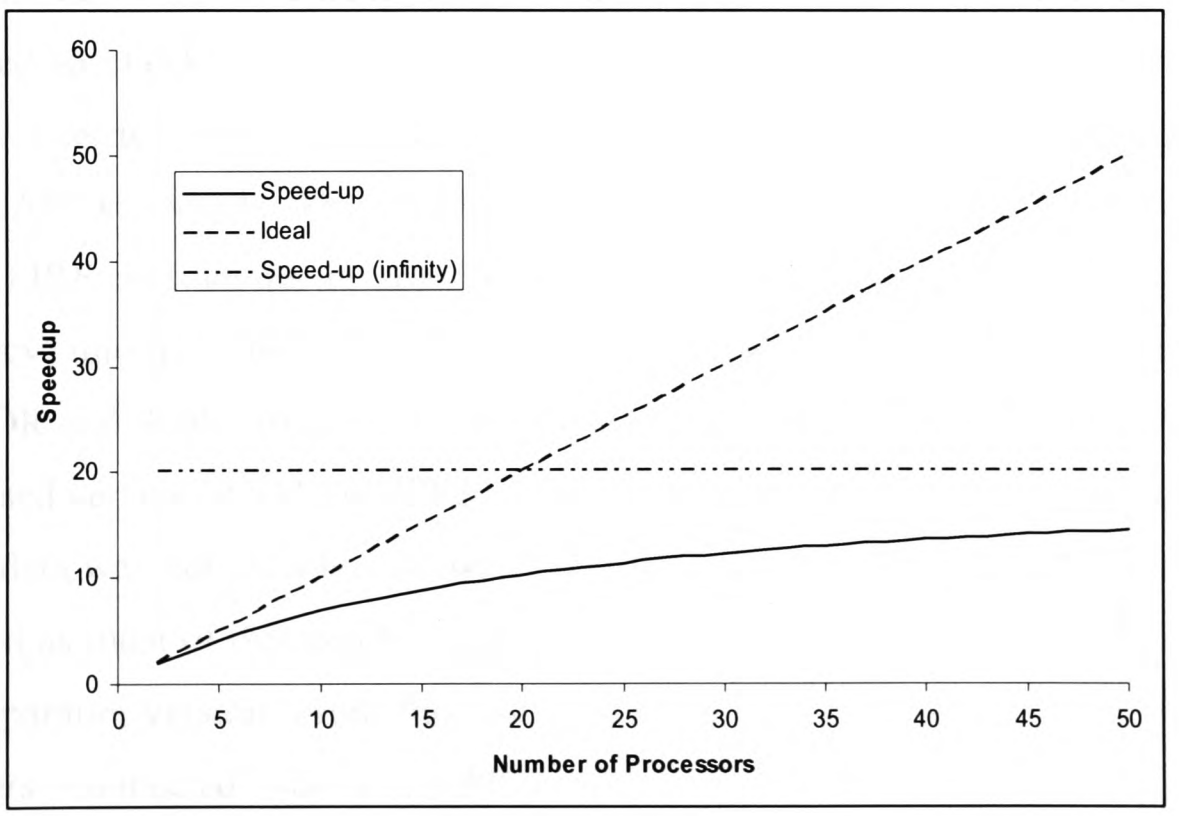


Figure 8-1 – Illustration of Amdahl's law

The ideal load balance for a homogeneous system of processors is simply the number of computational cells divided by the number of processors for each processor. In the more general case of a heterogeneous network the ideal load balance is obtained by weighting the load, the number of computational cells, according to the relative performance of the processor i.

$$W_i = \frac{\kappa_i}{\sum_{j=1}^{n} \kappa_j} \tag{8.1.4}$$

Another measure of parallel performance is the parallel efficiency that gives a measure of the actual performance compared to the ideal performance.

$$eff = \frac{S_{actual}}{S_{ideal}} \times 100 \tag{8.1.5}$$

The results section examines both the use of homogenous PC networks and also heterogeneous PC networks. Both types of networks exist in large numbers in a variety of environments.

8.2 Feasibility testing

Before the task of parallelising SMARTFIRE was undertaken, a feasibility test was performed to check whether a network of standard PCs could return a reasonable speedup. This test was a necessary condition for establishing if parallel processing over a LAN was viable. This was done by taking a standard serial batch version of SMARTFIRE and adding in appropriate dummy parallel calls (i.e. MPI_Isend and MPI_Irecv) into the JOR and SOR solvers. These calls sent 10-20% of the cell values in the JOR and SOR solver to the other processors to mimic the behaviour of a fully parallelised version of SMARTFIRE. This was the only interaction between the codes and the data was not used but allowed the effect of the parallel communication to be simulated as most of the parallel communication would occur within these solvers in the full parallel version. From this test it was found that 4 Intel Pentium III 733Mz computers connected via a 100Mbps Ethernet should provide a speedup of approximately 3.3 for a test case approximately the same size as the LPC-007 test case (~26,000 cells), and a speedup of approximately 3.7 for a case consisting of

~100,000 cells. These results suggested that the parallelisation of SMARTFIRE was viable.

8.3 Test cases

Two fire test cases were derived to test the parallel implementation of SMARTFIRE. These consisted of a small and a large mesh problem. The smaller problem was derived from a fire experiment (LPC-007) and provided a means to gauge the scalability of the parallel implementation. The large problem was designed to utilise the maximum amount of free memory available on a 256MB Windows NT based computer.

In addition, a number of small test cases have been used to ensure the correct operation of the parallel code compared to the serial behaviour. The cases have been run using the JOR solver throughout. This ensures that the result should be the same as the serial code, which was indeed found for all the cases. The use of the SOR solver would still have given a correct result but may have lead to small numerical differences between the final results in some circumstances. When the SOR solvers were used the values obtained were accurate to 5 significant figures or better.

8.3.1 The LPC-007 case

The setup for this test case is described in section 5.5.4. The case is essentially a fire in a small compartment measuring 6m x 4m and 3.3m high with a small vent located in one of the walls. The cell budget for this case was 26040 (31 x 24 x 35) cells.

8.3.2 The Large cell budget case

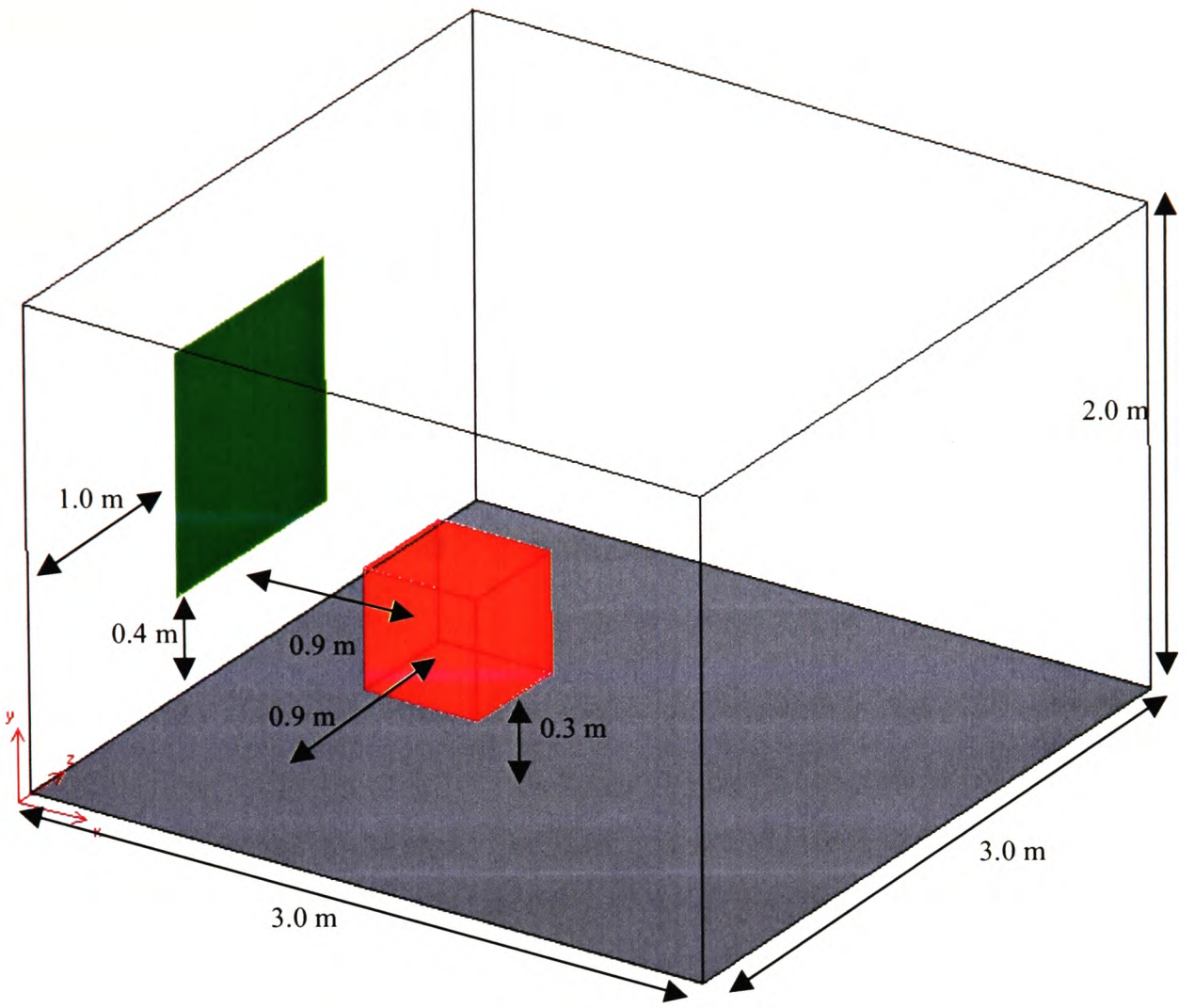


Figure 8-2 – Layout of the large cell budget case.

This test case is composed of (61 x 39 x 43) 102,297 cells and was purely designed for timing purposes of the parallel code on a case with a large cell budget, from this point onwards this case will be referred to as the large case. It is only described as a large cell budget case as this represented a large cell budget on a serial 256Mb machine. This case reflects the lower end of problem size that would be tackled in an engineering office. The large case consists of a simple fire (measuring 0.5m x 0.5m x 0.5m) located within a compartment measuring 3m x 3m plan area with a height of 2m. A single orifice measuring 1m x 1m was located in one of the walls. The location of the fire and vent are illustrated in Figure 8-2 above. The walls and ceiling of the compartment were composed of common brick material, which allows turbulent heat transfer, and the floor was non-conducting. The fire source was a constant simple volumetric enthalpy (heat) source of 50kW. An extended region beyond the vent was added to ensure accurate modelling of the flow through the doorway.

8.4 *Homogeneous results*

8.4.1 Distributed Memory results

A network of Intel Pentium III 800MHz machines was available for testing medium sized homogeneous PC networks (≤ 12 processors). These machines were connected to a 100Mbps Ethernet with 10/100Mbps Network cards. The speedup for varying numbers of PCs on the two test cases is given below in Figure 8-3.

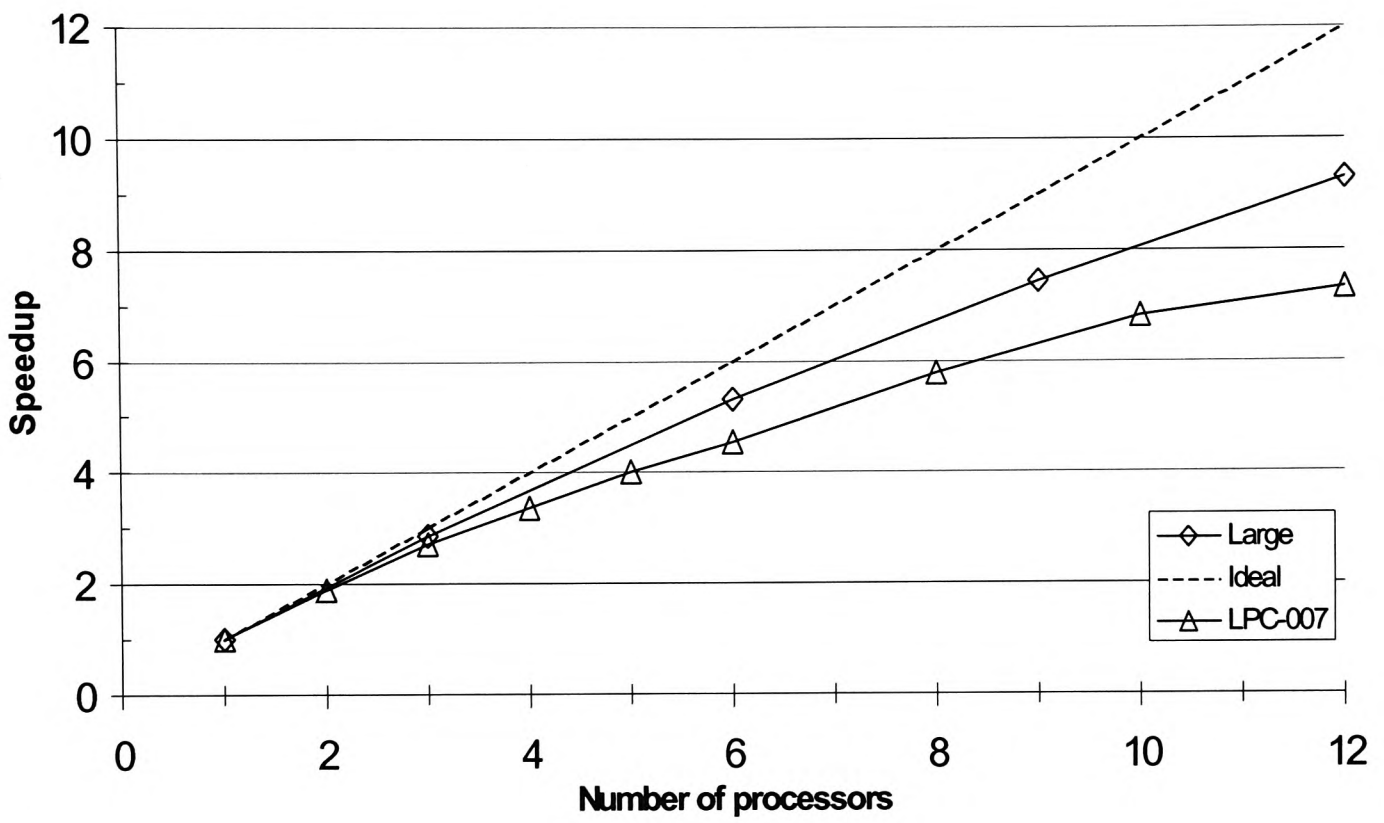


Figure 8-3 – Parallel Speedup Performance of SMARTFIRE on a PC network

As would be expected the parallel performance is better for the large case due to the higher computation to communication ratio that is exhibited by the large case compared to the LPC-007 case. A reasonable limit of profitable parallel power returned can be defined to be the point at which the gradient of the speedup graph becomes 0.5. Beyond this point each additional processor added is returning less than half its potential performance. For the LPC-007 case this limit is reached at 10 processors. This limit is not reached for the large case for the number of PCs used. Amdahl's law (8.1.2) can be used to estimate this point. For 12 processors it is calculated that f_s was 0.0264. Assuming f_s is constant for additional processors the value of N which gives $\frac{dS}{dn} = 0.5$ can be calculated; in this context Amdahl's law incorporates the communication as part of the serial fraction f_s. It was found that the

0.5 gradient would be reached when the number of processors was 16 giving a speedup S_{16} of ~11.5.

Due to time and equipment limitations, it was not feasible to run the problems fully on each network configuration with each problem being run for a few time-steps; 10 time-steps for the LPC-007, and 3 time-steps for the large case. This resulted in runtimes of ~1 hour when a single processor was used (56mins for the LPC-007 case and 58mins for the large case). In practical application a FSE is interested in wall clock times, rather than speed-up, so these have been estimated using the speed-up graphs (Figure 8-3). The cases were run to completion on a single processor to give the actual time taken on a single processor. Table 8-1 shows the estimated time taken if the LPC-007 case was run to completion on each homogeneous network configuration. The time taken for n processors is estimated by rearranging (8.1.1) to give $t_{n,\,full} = t_{1,\,full} / S_n$.

Table 8-1 – Estimated time taken for a given number of processors for LPC-007 case

No. Processors	Speedup	%Parallel Efficiency	Time taken
1	1	100	13hrs 43mins
2	1.9	95	7hrs 13mins
3	2.7	90	5hrs 5mins
4	3.37	84.3	4hrs 4mins
5	4	80	3hrs 26mins
6	4.55	75.8	3hrs1min
8	5.77	72.1	2hrs 23mins
10	6.82	68.2	2hrs 0min
12	7.32	61	1hr 53mins

Table 8-2 below shows the estimated time taken if the large case was run to completion on the homogeneous network configurations.

Table 8-2 – Estimated time taken for a given number of processors for large case

No. Processors	Speedup	%Parallel Efficiency	Time taken
1	1	100	54hrs 10min
3	2.86	95.3	18hrs 57mins
6	5.31	88.5	10hrs 12mins
9	7.44	82.7	7hrs 17mins
12	9.3	77.5	5hrs 50mins
16	11.5*	71.9*	4hrs 43mins*

*Estimated for 16 processors

These results demonstrate the usefulness of parallel SMARTFIRE in the context of using homogeneous PCs on a LAN to considerably reduce the runtime for a fire field simulation. It can be seen for the large case the serial code would have taken over 2 days. With 12 processors this job could be finished well within a working day allowing the FSE to initially set up the problem, run the problem and analyse the problem in a single day. Further runs could be initiated, based on these results, and run overnight thus allowing the FSE to more thoroughly examine the engineering problem in a vastly reduced timeframe.

8.4.2 Effect of Network Usage

Several tests were conducted to see the effect of additional network traffic on the performance of the parallel code. The test were carried out on three 733Mhz Pentium III computers connected to a 100 Mbps Ethernet which were running parallel SMARTFIRE whilst one of the following three tests was performed.

1) Browsing the internet (primarily the BBC news site (news.bbc.co.uk)) on one of the machines (< 5 Mbytes)
2) Downloading a large file from the internet onto one of the machines (~50Mbytes)
3) Uploading a large directory structure via the intranet onto remote disk storage (~500Mbytes)

These tests were repeated 5 times and for two different cases. The cases were the LPC-007 case utilising 1 time step and 50 outer iterations, and the large case utilising 1 time step and 20 outer iterations. These settings were chosen to ensure that the downloading / uploading of cases 2 and 3 was continuous throughout the performed simulations.

Table 8-3 – Effect of network usage on parallel processing network

Tests	LPC-007	Large
No extra network traffic	138 ± 1s	135 ± 1s
Test 1	138 ± 1s	135 ± 1s
Test 2	158 ± 3s	156 ± 2s
Test 3	239 ± 3s	241 ± 4s

It can be seen from Table 8-3 that browsing the Internet had no measurable effect on the performance of parallel SMARTFIRE. It can be seen that downloading a 50

Mbytes file from the Internet adds approximately 20s to the overall runtime and uploading a directory structure via the Ethernet adds approximately 100s to both the simulations considered. These results must be considered in the context of a true simulation that may take several hours. The additional 100s in this context has minimal impact on the overall runtime and a FSE is generally unlikely to be using the network to move around such large amounts of data. A far greater impact on the runtime will be caused by the FSE running extra computationally intensive jobs on one or more of the computers involved in the parallel processing.

8.4.3 Effect of additional computational load

Using the same configuration that was used above (section 8.4.2) one of the nodes was selected to run an additional computational load. The additional computational load is serial SMARTFIRE running the same case but with no interaction with the parallel case.

Table 8-4 - Effect of additional computational on parallel network

Test	LPC-007	Large
No additional load	138 ± 1s	135 ± 1s
Additional load	333 ± 19s	355 ± 14s

It can be seen from Table 8-4 that running the additional computational loads had an adverse effect on the overall runtime of a fire model prediction. Unlike the effect of the network traffic the effect of the additional computational load could possibly last the duration of the parallel simulation undertaken leading to runtimes increasing by a factor of ~3 in this case and completely removing the advantages of parallel processing. With more processors in the parallel network the problem is compounded as all the processors are limited by the performance of the slowest processor (see section 7.4.2), e.g. if a parallel network had a speedup of 12 this could be reduced to a speedup of 4. Obviously it makes no sense to run a parallel job concurrently with another CPU intensive process and should therefore be avoided. Automated computer methods addressing this problem will be described in section 8.5.2.

8.4.4 Shared Memory results (Dual Pentium III equipped PC)

The potential performance of a Shared Memory multiprocessor computer has been assessed to demonstrate the potential usefulness of such a machine to a FSE. It further demonstrates the usability of the DM version of SMARTFIRE on a shared memory environment. The following cases were run on the dual Pentium III 450 MHz machine using both the multithreaded version of SMARTFIRE (see section 7.2) and the DM version of SMARTFIRE.

- Steckler's Room case (13,020 cells; see section 5.5.1 for further details)
- LPC-007 case (26,040 cells; see section 5.5.4 for further details)
- CIB case (42,775 cells; see section 5.5.3 for further details)
- Large Cell budget case (102,297 cells; see section 8.3.2 for further details)

8.4.4.1 Multithreaded SMARTFIRE

Table 8-5 – Speedup of SMARTFIRE (Multithreaded) on dual processor compared to single processor

Case	Speedup
Steckler	1.2
LPC	1.36
CIB	1.38
Large	1.54

In Table 8-5 above the speedups for the various cases previously described are given. It can be seen that the range from poor (Steckler) to reasonable (Large) with the ideal speedup being 2.0. These poor speedups are most likely due to the lack of cache utilisation and to the overhead of creating and destroying the worker threads, particularly inside the solvers.

8.4.4.2 DM-SMARTFIRE

The speedups achieved using DM-SMARTFIRE on the dual Pentium III 450MHz over a single Pentium III 450MHz are tabulated below in Table 8-6.

Table 8-6 – Speedup of DM-SMARTFIRE on dual processor compared to single processor

Case	Speedup
Steckler	1.67
LPC	1.7
CIB	1.73
Large	1.77

The degradation from the ideal performance is mostly due to the shared bus, similar levels of performances have been obtained when two independent serial codes have been run on the dual Pentium system concurrently. The performance of the distributed memory version of SMARTFIRE is good compared to the multithreaded version previously implemented, which gave a best speedup of 1.54, for the large case. This may be due to the multithreaded version of SMARTFIRE not being optimally implemented, or it maybe due to the nature of the threading of the NT O/S with the threads constantly swapping between processors and resulting in less cache hits. The speedup results for DM-SMARTFIRE indicate that there appears to be little disadvantage on a shared memory machine and therefore little to be gained in maintaining a separate shared memory version of SMARTFIRE.

The speedup performance of the shared memory dual Pentium III 450MHz compared to that of two single processor Pentium III 733Mhz PCs connected via a network is disappointing. The two single processor PCs on a network perform far better than the dual processor machine with a speedup of 1.91 for the LPC case and 1.94 for the large case. The fact that the single processors runs at 733MHz is not relevant, indeed it would be expected that a lower MHz rating of the single processor would result in even better speedups due to the increase in computation time compared to communication time. It is apparent from these results that the memory bus contention is far greater than the overhead of passing messages over a network, even for a small number of processors. This has implications for the best value for money when investing in computer hardware between a single dual processor machine and two single processor machines. It must be noted that only one shared memory machine was tested and further testing would be required before any concrete conclusions can be drawn.

8.5 Heterogeneous results

In a FSE's office environment it is also possible, perhaps more likely, that a range of heterogeneous (non-identical) computers would be found. Heterogeneous networks provide additional problems in trying to maximise the parallel performance of such a platform as previously described in section 7.4.2. The computational load on a

processor needs to match the computational performance of that processor to ensure that none of the processors are idle or overworked for too long.

8.5.1 Static Load Balancing

A number of small heterogeneous networks were tested. The processors used for these networks are detailed in Table 8-7 below and the speedup for a number of network configurations are given in Table 8-8 below. In Table 8-7 the relative performance of the processor compared to the Intel PIII 733 is given. The dual processor was given two performance indexes, the first value is when the problem was executed on a single processor, and the second value is when the problems was executed in a dual processor mode on the same machine (see section 8.4.4.2). Memory bus contention prevents the dual processor machine from running two jobs on two processors as effectively as a single job on a single processor. This performance index was simply determined by averaging the serial runtimes of three varying size cases, the LPC-007 case (26,040 cells), the Steckler room case (13,020 cells), and the CIB test case (42,775 cells) on each processor.

Table 8-7 – Processor's performance used in heterogeneous networks based relative to an Intel PIII 733

	CPU	$\kappa_{Steckler}$	κ_{LPC}	κ_{CIB}	$\kappa_{average}$
(a)	Intel PIII 733	1.0	1.0	1.0	1.0
(b)	Intel PII 450	0.64	0.63	0.61	0.63
	2 x Intel PII 450D	1.07	1.08	1.06	1.08
(c)	Intel P200 MMX	0.233	0.23	0.228	0.23

Table 8-8 - Speedup for heterogeneous network configurations

Network Config	CPUs	Predicted Speedup	LPC	LPC %eff	Large	Large %eff
1	(a) + (a)	2.0	1.91	95.5	1.95	97.5
2	(a) + (b)	1.63	1.61	98.8	1.62	99.4
3	(a) + (c)	1.23	1.17	95.1	1.22	99.2
4	(a) + (b) + (b)	2.08	2.03	97.6	2.05	98.6
5	(a) + (b) + (c)	1.86	1.65	88.7	1.73	93.0
6	(a) + (a) + (a)	3.0	2.67	89.0	2.82	94.0
7	(a) + (a) + (b) + (b)	3.08	2.74	89.0	2.91	94.5
8	(a) + (b) + (b) + (c)	2.31	1.88	81.4	2.2	95.2

From Table 8-8 it can be seen that the homogeneous network configurations (1 and 6) perform very well. It can also be seen that the heterogeneous network configurations

(2, 3, 4, 5 and 7) perform very well. They are indeed performing close to their predicted performance. This may be due to an underestimate of the relative performance of the slower processors. With configuration (7) a FSE could expect to run a fire simulation problem, the large case, in 18hrs 37mins as opposed to 54hrs 10min that would be required for a single processor.

The worst performance is that of configuration (8) on the LPC test case which is slower than configuration (4) despite the extra processor. The precise reason for this detrimental performance is unknown although is probably attributable to the non-linear code performance described in section 7.4.2.

8.5.2 Dynamic Load Balancing

It is sometimes difficult to determine the relative performance of the processors due to the effect of differing problem types on various processor architectures i.e. there is no definitive performance index. From Table 8-7 it can be seen that there is some problem dependent variation of the relative performance of a processor, with other problems or processors this could be more extreme. Furthermore a processor may not have been previously benchmarked so that its relative performance κ is unknown. A dynamic load balancing scheme has the potential to address both of these problems. The dynamic load balancing scheme, proposed here, uses the timing data from the problem it is actually running to evaluate the performance index of each processor. This therefore removes the need to accurately determine the performance κ a priori. The dynamic load balancing scheme addresses another issue of extraneous computational loads existing on a particular processor.

The following scheme has been devised and implemented:-

1) The problem is distributed amongst the processors involved using a prescribed processor performance for each processor. The load balance for each processor i is calculated using (7.4.5.2) and (7.4.5.3). If the processor performance is unknown than the performance is assumed to be the same as the maximum processor performance.
2) The time step is incremented and the code is run to solve the problem for this time step.

3) Special operating system interrogation commands are called at the end of the time step from within the code to establish the $\%CPU_{total}$ and $\%CPU_{DM\text{-}SMARTFIRE}$ usage on each processor over the duration of the time step. These would be 100% if the processor was fully utilised running DM-SMARTFIRE.
4) Using these figures the initial processor performance figures κ^{old} the new processor performance indexes are calculated using the following formula :-

$$\kappa^{new} = \frac{\kappa^{old}}{\%CPU_{total}} \times 100$$

This will give a reasonable estimate of the relative CPU performances assuming that the time to process a problem is linearly proportional to the number of cells. This is reasonable for large cell budgets. The processor performance indexes are normalised by dividing by the maximum processor performance index.

5) If $\%CPU_{DM\text{-}smartfire}$ is less than 60% of $\%CPU_{total}$ for a particular processor then it is assumed that the processor is performing another computational task and should therefore be used minimally for the parallel process. The performance κ^{new} is set to 0.05 for the 'busy' processors. This means that only a very small amount of work is placed on the processor for the parallel task. This avoids the entire parallel job being held up by the busy processor and leaves the vast majority of the processor capacity to work on the other job(s) running on it.

6) If all of the new performance indexes are less than 5% different to the old performance indexes then the simulation continues using the old problem distribution. Return to step 2) until the simulation has finished.

If any of the new performance indexes are more than 5% different to the old performance indexes then a restart file is created and the problem is restarted on a new domain decomposition by returning to step1) unless the simulation has finished.

The steps 3 – 5 in the above algorithm effectively take place in step 8 of the SIMPLE solution algorithm described in section 3.11.

The above system is quite simple in nature and utilises a simple 1D decomposition of the problem space onto the PC network. This algorithm is not limited to a 1D decomposition scheme and could easily be replaced with a more advanced decomposition scheme if required.

8.5.2.1 Testing of dynamic load balancing

In Table 8-9 below a number of dynamically load balanced heterogeneous networks are described. In addition to the processors already described, an additional processor, an AMD Athlon 700MHz processor, referred to as (d) in Table 8-9, has been included for configurations (9)-(11). This processor, as well as all the other processors, is initially assumed to have a relative processor performance of 1, this performance was then dynamically adjusted. It was typically found that this processor had a performance ~0.8 compared to the Intel Pentium III 733MHz during these trials. The predicted figures quoted in Table 8-9 are the sum of the relative performances calculated by the dynamic load balancing algorithm.

Table 8-9 - Speedup for heterogeneous network configurations using dynamic load balancing

Network Config	CPUs	Ideal	Predict LPC	Actual LPC	Predict Large	Actual Large
2	(a) + (b)	1.63	1.69	1.639	1.678	1.639
3	(a) + (c)	1.23	1.225	1.19	1.228	1.21
4	(a) + (b) + (b)	2.08	2.09	2.07	2.1	2.07
5	(a) + (b) + (c)	1.86	1.835	1.69	1.863	1.84
7	(a) + (a) + (b) + (b)	3.08	2.97	2.85	3.02	2.91
8	(a) + (b) + (b) + (c)	2.31	2.18	2.12	2.3	2.24
9	(a) + (a) + (a) + (d)	3.8	3.6	3.36	3.61	3.47
10	(a) + (a) + (a) + (b) + (d)	4.43	4.22	3.76	4.18	3.88
11	(a) + (b) + (c) + (a) + (d)	3.66	3.39	2.95	3.19	3.08
12	(a) + (b) + (c) + (a)	2.86	2.66	2.34	2.79	2.67

In Table 8-10 below it can be seen that the dynamic load balancing scheme generally outperforms the static load balance. The static load balance is the best guess available from the known serial processor performance. The one exception to the improved performance was the large case used in network configuration (3) even in this cases

the difference is small, and probably indicates the optimal load balance has been achieved. It is also notable that the processor performance is problem dependent and illustrates the problem of the static load balancing scheme that can not adapt to different problem types or sizes to give the best load balance.

Table 8-10 - Comparison of static and dynamic load balancing

Network Config	CPUs	Static LPC	Dynamic LPC	Static Large	Dynamic Large
2	(a) + (b)	1.61	1.639	1.62	1.639
3	(a) + (c)	1.17	1.19	1.22	1.21
4	(a) + (b) + (b)	2.03	2.07	2.05	2.07
5	(a) + (b) + (c)	1.65	1.69	1.73	1.84
7	(a) + (a) + (b) + (b)	2.74	2.85	2.91	2.91
8	(a) + (b) + (b) + (c)	1.88	2.12	2.2	2.24

The dynamic load balancing system has proved to be successful at providing a good load balance across a heterogeneous PC network and is most useful when the number of processors increases and the network behaviour is more difficult to assess.

8.5.2.2 Testing of load balancing scheme with additional computational load

In a real office environment it is possible that one or more of the machines that a FSE chooses to use as part of their parallel processing system may be used by other FSEs or users in that office at some point during the simulation. The parallel implementation needs to be able to cope with such an event. As previously described in section 8.5.2 the performance of the busy processor is reduced to 5% of the maximum processor performance to ensure that the other user of the computer has a large amount of processing power available to them.

Two network configurations were used to test the ability of the dynamic load balancing scheme to react to an additional serial load being placed on one of the processors. The first network configuration was composed of two Pentium III 733MHz processors. The second configuration was composed of three Pentium III 733MHz processors. Both of these configurations were connected via a 100Mbps Ethernet. The LPC-007 case and the Large case were used to test the performance of the processor configurations. The testing has been conducted on homogeneous

networks to demonstrate the load balancing scheme but does not preclude the possibility of using the dynamic load balancing with an additional load on a heterogeneous network. The parallel cases were run for a couple of time steps and then the additional serial load would be started on one of the processors.

When one of the processors had an additional serial load placed upon it the new speedups for the processors were calculated using the load balancing algorithm (section 8.5.2). The performance of the processor with the additionally serial load was reduced to 0.05. In the networks tested this left 95% of the processing power available for the additional serial job on the additionally loaded processor. The additional serial computational load was an instance of serial SMARTFIRE.

The results from the testing are summarised in Table 8-11 below. The (P) is the parallel performance and (S) is the performance of the additional serial load, SMARTFIRE.

Table 8-11 – Results of testing of dynamic load balancing with an additional serial computational load

Performance index	LPC-007 (2)	Large (2)	LPC-007 (3)	Large (3)
Ideal	2.0	2.0	3.0	3.0
Actual (P)	1.91	1.95	2.67	2.82
Ideal with load (P)	1.05	1.05	2.05	2.05
Actual with load (P)	0.85 ± 0.1	1.02 ± 0.02	1.61 ± 0.13	1.80 ± 0.05
Ideal with load (S)	0.95	0.95	0.95	0.95
Actual with load (S)	0.93 ± 0.01	0.92 ± 0.01	0.94 ± 0.01	0.93 ± 0.01
Combined (P + S)	1.78 ± 0.09	1.95 ± 0.01	2.56 ± 0.12	2.73 ± 0.04

It can be seen from Table 8-11 that the effect of the small parallel load on the processor with the additional serial computational load was quite small with a 6%-8% loss of serial performance compared to a processor with no parallel load. The user of the remote computer used for the parallel processing should hardly notice the small parallel load.

The final row of Table 8-11 gives an indication of the total amount, of computational power extracted from the processors on the parallel network. From Table 8-11 it can be seen that the overall processing power extracted from the network was only slightly less than the processing performance extracted from a plain parallel load.

8.5.2.2.1 Intermediate changeover behaviour

The worst performance of both the serial and the parallel job occurs during the transition phase when the additional serial job is started. The transition phase occurs before a rebalance can be performed to take account of the additional serial load. The performance of both the parallel job and the additional serial job will be compromised by using the wrong load balance and will have a similar behaviour to that obtained in section 8.4.3. This results in the parallel performance being approximately 25-50% of the anticipated parallel performance. The performance of the additional serial load is approximately 50-70% of the normal serial performance. This problem is relatively short lived and should generally only last for a single time step. In some circumstances it may last for two time steps.

When the additional serial job finishes then the parallel performance is slightly compromised as only 5% of the maximum processing power of the processor, which also had the additional serial load, is used. This is worst for 2 processors where only about half the total available power is used. This problem gets progressively better as more processors are used. Theoretically for a set of n homogeneous processors the parallel speedup is (n - 0.95). This problem is only short lived and should generally last for a single time step. In some circumstances it may last for two time steps. However this effect is far less severe than the effect of starting an additional serial load as described in the previous paragraph.

The effect of the changeover behaviour could be alleviated by checking the processor performance indexes more frequently and therefore allowing a problem partition rebalance to be performed earlier. However this process incurs its own time penalty and maybe activated prematurely for short lived additional serial loads. It is difficult to determine the optimum strategy, assuming one actually exists, for choosing when to rebalance the problem partition. The strategy adopted by the author, i.e. checking processor usage and if required rebalancing the computational load at the end of each time step, seems to perform reasonable well.

8.6 Memory usage

With the number of computers available to the FSE using the DM version of SMARTFIRE there is also a vast increase in the potential memory available for running larger simulations than is possible with a single serial version of SMARTFIRE. For a fire modelling problem utilising a volumetric heat source the memory usage for SMARTFIRE, found empirically, is expressed as (8.6.1) in kilobytes.

$$mem = 1.91 * NCELL + 6000 \tag{8.6.1}$$

Ignoring the requirements of any halo cells on the sub-domains due to the partition process means that for a homogeneous network of NP computers the memory requirement of each slave is (8.6.2). The halo cells can not be included in any general fashion as this requirement is problem dependent.

$$mem = \frac{1.91 * NCELL}{NP} + 6000 \tag{8.6.2}$$

The use of temporary arrays for I/O purposes requires that the master process has additional memory requirements over that of the slave processes. In addition to the requirements of the slave, the master process needs to hold a minimum of 2 globally dimensioned arrays, a temporary storage area and the global2local_master_index. In the actual implementation 4 globally sized arrays, i.e. 3 temporary storage areas and the index, were used as this eased the implementation and is slightly quicker as various global arrays do not need to be constantly created and destroyed during the I/O phase. Three temporary storage areas were required as some output file formats required values as vector data, i.e. x, y, and z values. Each stored value is 4 bytes long. For 4 values the memory usage is 0.015625 kilobytes and added to (8.6.2) yields (8.6.3).

$$mem = \frac{1.91 * NCELL}{NP} + 6000 + 0.015625 * NCELL \tag{8.6.3}$$

Using the above relationships (8.6.3) the maximum problem size that can be accommodated on NP homogeneous processors compared to a single processor is plotted in Figure 8-4.

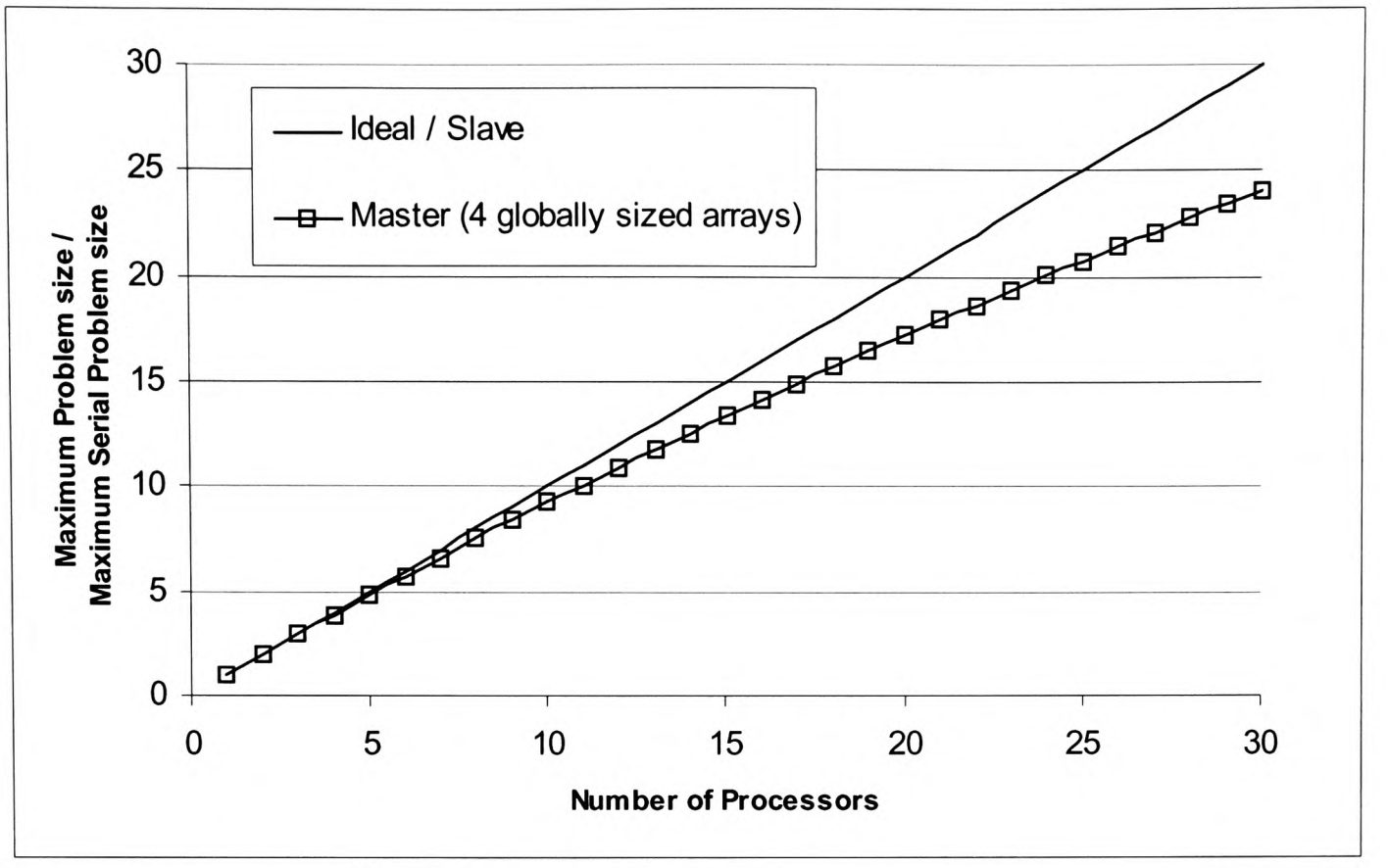


Figure 8-4 – Maximum problem size versus number of processors

From Figure 8-4 it can be seen that the memory usage is not ideal when the master process is considered but must be considered to be extremely good for small scale parallel systems. Even for 30 processors it is possible to run a problem size 24 times larger than a single serial computer.

The halo cells have not been considered in the memory usage due to the problem dependent nature of halo cells. In general this should be a small additional memory requirement; this is illustrated in the following example. A cubic domain of NCELL cells is subdivided into eight equal cubic sub-domains and the number of halo cells for one of the aforementioned sub-domains can be estimated using (8.6.4).

$$\left(\frac{NCELL}{8}\right)^{\frac{2}{3}} * 3 \tag{8.6.4}$$

This leads to the master memory usage equation (8.6.3) with 4 globally sized arrays being modified to (8.6.5) assuming the halo cells require the same amount of storage as the other sub-domain cells. The slave memory usage equation is similarly modified.

$$mem = \frac{1.91 * NCELL}{8} + 6000 + 0.015625 * NCELL + \left(\frac{NCELL}{8}\right)^{\frac{2}{3}} * 3 * 0.015625 \tag{8.6.5}$$

In Table 8-11 below the ratio of maximum parallel problem size to maximum serial problem size is tabulated against maximum serial problem size using (8.6.5) to include the memory usage due to halo cells.

Table 8-12 – Maximum problem sizes including halo cells for a cubic domain split into eight equal sub-domains

Maximum Serial Problem size	Maximum Parallel Problem for 8 processors / Maximum Serial Problem size
10,648	6.6
46,656	6.9
97,336	7.1
1,000,000	7.3
No Halo cells	7.5
Ideal	8.0

From Table 8-12 above it can be seen that the addition of halo cells into the calculation of memory usage only makes a small difference, about 8% for a maximum serial problem size of 46,656 cells, compared to the calculation assuming no halo cells.

If the FSE needs to run very large problems using a very large numbers of processors then the I/O memory requirements of the master process may need to be reduced. This could be achieved by the use of temporary files for storing vector data and thus reducing the additional master memory requirement to two globally sized arrays. This would increase the time required to write the output file, as the data now has to be written and read from the temporary file in addition to the output file. Another possibility is not to use any temporary storage and collect the data one item at a time from the various sub-domains and output this value directly to the output file(s). This approach is attractive as only 1 globally sized array, the global2local_master_index array, is required but this method is excessively slow due to the communication of a huge number of small data items. The following example demonstrates the performance degradation. The computers attached to the University of Greenwich's 100Mbps Ethernet had a latency of ~120μs and a bandwidth of ~10Mbytes/s. If 100,000 data items of 4 bytes (a floating point number or long integer number) were sent to the master process in one communication as a single large array the time taken to transmit the data is approximately 0.04s. If the 100,000 data items were sent to the

master process individually the time taken would be approximately 24s. However it is unlikely these methods would be required for the level of parallelism (< 30 processors) that could be utilised in a FSE's office.

8.7 Future performance of PC technology

Law and Turnock [LT2001] (section 2.4.4) demonstrated that a 10Mbps Ethernet was not fast enough to allow for a good parallel performance, on a coupled domain decomposition problem similar to a CFD type problem, from a set of Pentium II/III ≥ 350Mhz machines. The work in this thesis is all based on using a 100Mbps Ethernet and was found to perform well from both the work performed by the author and from the work of Law and Turnock [LT2001]. However, in the future machines are liable to get faster, indeed there are presently machines rated at 2.2Ghz (Pentium IV) currently on the market compared to the 733 / 800 Mhz Pentium III used in this thesis. These faster machines will be less scalable for parallel computation if the 100Mbps Ethernet technology were to be utilised due to the relatively short computation time compared to the relatively long communication time.

8.7.1 Theoretical study of faster computers

From the results obtained earlier it was possible to obtain a curve fit based on Amdahl's law (8.2) for the 800MHz Pentium III computers on a 100Mbps fast Ethernet. On the large case (section 8.3.2) the serial fraction time f_s was 0.0264. Using this result and assuming that a 2.2GHz Pentium IV is 2.75 times faster than a 800MHz Pentium III and that the data communication is the entire serial fraction overhead and takes the same time on both types of machine, the serial fraction f_s can be estimated for a network of 2.2GHz Pentium IV computers. For a theoretical network of 2.2GHz Pentium IV computers the serial fraction f_s is estimated as 0.0694 for the large case (~100,000 cells). The theoretical speed-up for a network of 2.2GHz Pentium IV machines attached to a 100Mbps Ethernet is illustrated and compared to the theoretical and actual speed-up of a network of 800MHz Pentium III machines in Figure 8-5 below.

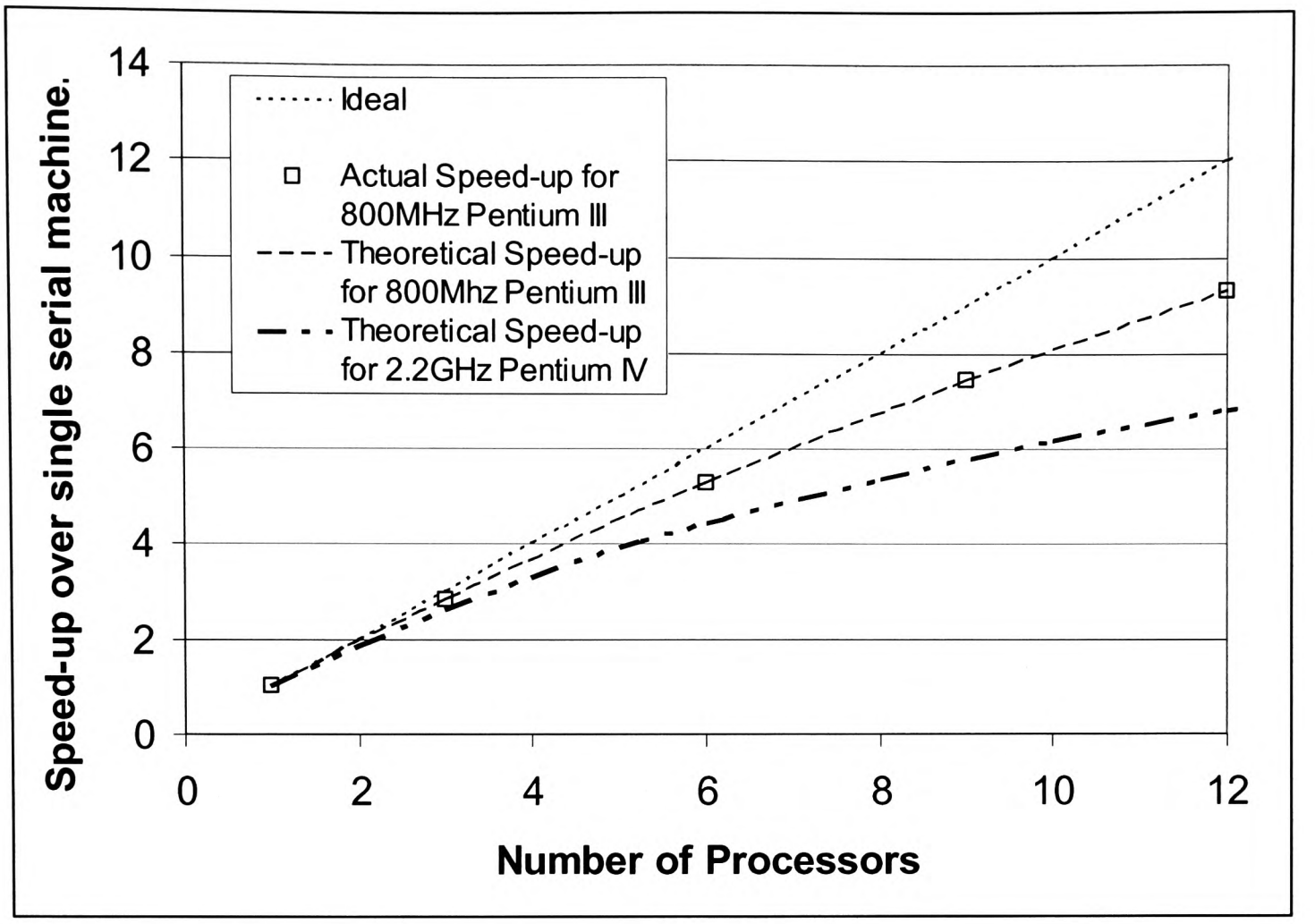


Figure 8-5 – Theoretical speedup for 2.2GHz Pentium IV homogeneous network

Figure 8-5 is a pessimistic view of the achievable speedup performance as the 2.2GHz Pentium IV is probably going to be less than 2.75 times faster so that the actual speed-up should be higher than the above theoretical prediction. Another factor is the communication speed, although the data bandwidth is unlikely to be changed with a faster processor, it is possible that the communication latency could be reduced. Although the relative speed-up is not as good as that achieved using the 800MHz Pentium III machines it is still a worthwhile computational advantage for the FSE. It must be noted that the speed-up for the 2.2GHz Pentium IV is relative to a single 2.2GHz Pentium IV machine; a network of 12 2.2GHz Pentium IV machines would be ~19 times faster than a single 800MHz Pentium III machine although ideally it should be 33 times faster. With fast processors of the future the speedup ratio would become even less favourable if the networking technology remained the same.

8.7.2 Actual testing of DM-SMARTFIRE on faster computers

A small network of faster computers (3 x 1.9GHz Pentium IV) was available for testing DM-SMARTFIRE. A single 1.9GHz Pentium IV was found to be 1.6 times

faster than a 733MHz Pentium III used in section 8.5.1. Results for the LPC-007 case and the 'Large' case are given Table 8-13 and Table 8-14 respectively. The speedup for 733MHz Pentium III processors is also given to illustrate the parallel speedup performance degradation of the faster processor.

Table 8-13 – Performance of 1.9GHz Pentium IV processors on LPC-007 case

Number of CPUs	Speedup (1.9GHz)	%Eff	Speedup (733MHz)
2	1.82	91	1.91
3	2.43	81	2.67

Table 8-14 - Performance of 1.9GHz Pentium IV processors on 'Large' case

Number of CPUs	Speedup (1.9GHz)	%Eff	Speedup (733MHz)
2	1.92	96	1.95
3	2.79	93	2.82

8.7.3 Discussion of future PC performance

It can be seen from both the theoretical study (section 8.7.1) and the testing of faster computers (section 8.7.2) it may seem that as processor speeds increase the benefits of utilising parallel processing diminish but there are two additional factors that affect the possible future parallel performance, namely network speed and memory size. Both of these factors have historically increased as processor performance has increased and should ensure that the parallel methodology applied here should remain valid in the future.

Gigabit Ethernet technology that is "ten times faster" than 100Mbps fast Ethernet technology already exists and is likely to replace fast Ethernet technology in the near future in much the same way as 100Mbps fast Ethernet technology has replaced the 10Mbps Ethernet. This should ensure that speed-ups based on present problem sizes in the future will at least match the speed-ups obtained in the present work despite the increased processor performances.

Other networking technologies exist specifically for high performance computing clusters such as Myrinet networks [BCF+1995]. These are probably too expensive for an average FSE in 2003 but technology advances from this work may enter mainstream networking usage in the future. Presently this type of network has a bandwidth 25 times greater and a latency that is 15 times shorter than 100Mbps fast

Ethernet technology. This level of performance would vastly increase the potential speed-up from future computer networks.

Another factor that must be considered is that speed-up is problem size dependent and speed-up improves with increased problem size as seen in section 8.4.1 with the speed-up comparison with a case of ~26,000 cells compared to a case of ~100,000 cells. If present trends in average memory size continue then the typical problem size a FSE can consider will also increase in the future helping to improve speedup performance.

8.8 Case study of Home Office's Large Warehouse case

The UK office of the Deputy Prime Minister's Fire Research Division (FRD) performed a study of a large warehouse to assess the safe evacuation time of such a building for the occupants and the safety of fire fighters entering the building to fight the fire [HE2002]. Increasing numbers of large warehouses with floor areas in the range of 2,000m^2 to 20,000m^2 or more and heights of 12 metres or more are being constructed in the UK. These buildings present different characteristics to "normal" warehouses with the potential for rapid fire spread across a very large area and the consequential hazards to firefighters and the environment. This task was more concerned with demonstrating the use of CFD modelling, in particular SMARTFIRE, in the context of fire safety analysis rather than providing a definitive judgement on the warehouse. The building consisted of six bays arranged in a 3 x 2 fashion. A side view of the warehouse is given in Figure 8-6. The warehouse had a floor area of 94m by 137m.

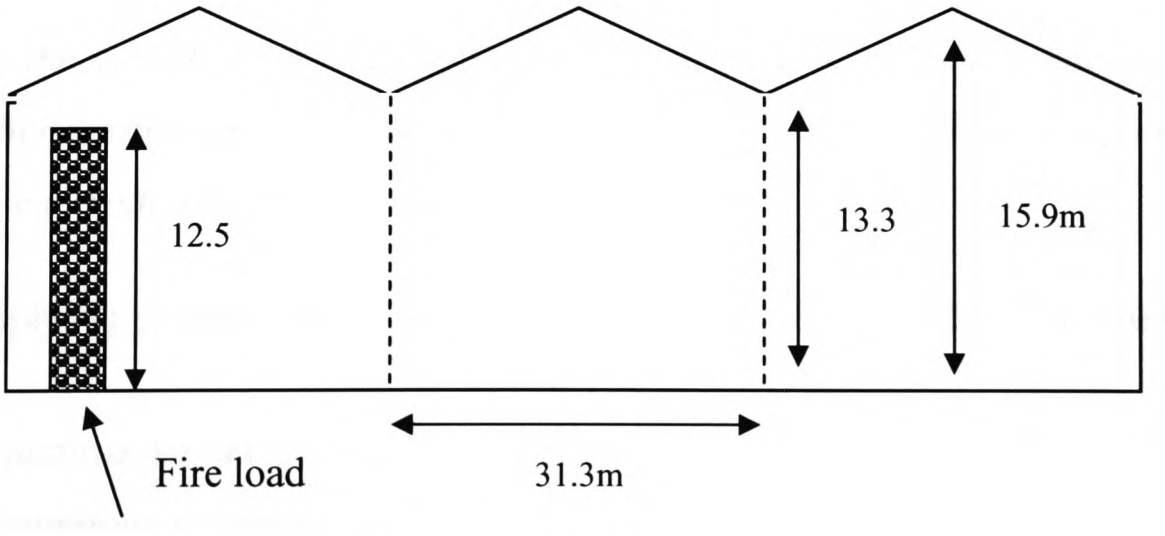


Figure 8-6 - Side view of FRD warehouse scenario

For this study seven runs of the serial SMARTFIRE fire modelling software were performed. The seven runs are briefly described in Table 8-15 below.

Table 8-15 – Description of the scenarios modelled for FRD Warehouse case

Case 1	An empty warehouse (1 of 6 bays modelled)
Case 2	As Case 1 but with obstacles throughout the warehouse to represent stored goods
Case 3	As Case 2 but with a coarser computational grid
Case 4	As Case 3 but with vents in roof (previously in top of walls)
Case 5	As Case 4 but with modelling of fire by gaseous combustion (previously volumetric heat source)
Case 6	As Case 4 but with insulating material for walls and ceiling (previously brick)
Case 7	As Case 4 but with full 6-bay warehouse (previously only single bay was modelled)

The additional cases (2-7) were essentially a sensitivity analysis examining the effect of varying a single aspect of the model with each successive simulation. It was found that small variations existed between the results but these were consistent with the changes made. Furthermore all the simulations led to the same overall conclusion. A main conclusion from the report [HE2002] was:-

These results indicate that safe evacuation of occupants is likely to be achieved providing that the building is maintained and run according to the design, and that the occupants are adequately trained in fire safety. However, these results depend on a number of assumptions and limitations inherent in the fire safety analysis and CFD fire modelling.

Regarding the conditions for firefighting, the results show that the conditions at 10 minutes after ignition are highly hazardous throughout all parts of the bay containing the fire, due to radiation from the [hot] upper layer.

8.8.1 Applying parallel processing to the warehouse simulations of the FRD

The total runtime for all the simulations was 388 hours and 32 minutes or over 16 days of continuous computation. The work was performed on a computer equipped with a 1.0GHz Pentium III CPU and 1 GB RAM. It is apparent that the runtime, of the

CFD code serial SMARTFIRE, is very large. Parallel processing could significantly reduce the time needed to perform the study or the time saved could be used to run additional simulations to further check grid / parameter sensitivity or to run further scenarios.

The number of cells used and the time taken to run each case is given in Table 8-16 below.

Table 8-16 – Number of cells used for each scenario of FRD warehouse

Case	Number of Cells	Time on 1GHz
1	169,000	57hrs
2	209,456	71hrs 38mins
3	108,528	37hrs
4	89,760	31hrs
5	164,388	68hrs
6	89,760	31hrs
7	273,000	93hrs 56mins

From private communications with Dr Hume of the FRD it was determined that there were approximately 6 additional machines that could have been accessed for the purposes of parallel processing. The exact specification of these machines was unknown although it was believed that they were Pentium III class. For the purposes of this study it will be assumed that they are 800MHz rated machines. It will be further assumed that the 1.0GHz Pentium III is 25% faster than the 800MHz Pentium III and that when used in parallel the 1.0GHz machine is operating as part of a homogeneous network of 800MHz Pentium III machine, i.e. the 1.0GHz Pentium III idles for 20% time and has the performance of an 800MHz Pentium III computer. It is also assumed that the computers are attached to a 100Mbps fast Ethernet.

With these machines available one could reasonably suggest that the machines could be used in a serial fashion with all the jobs running simultaneously on the machines reducing the overall runtime to 93hrs and 56 minutes. However this has a serious disadvantage compared to running the problems in parallel in order. This has the disadvantage of not utilising all the CPU cycles available as each case has a different runtime. In parallel all the computers are used throughout all the simulations and should lead to an overall shorter wall clock time to perform the simulations although

this is dependent on the number of simulations being performed and the time that each simulation takes to perform. It also presupposes that the FSE knows exactly what simulations they wish to run. In the parallel mode of operation the FSE can choose their simulations based on the results of previous simulations. This is an important advantage ensuing that inappropriate and unnecessary simulations are not performed.

It is reasonable to assume that a network of seven 800MHz Pentium III computer, in fact six 800Mhz and one 1GHz Pentium III would exhibit similar speedup characteristics to that found for the 'large' case (section 8.4.1), ~100,000 cells, for the majority of the cases performed by Hume. For case (7) the speedup behaviour should be better but it will be assumed that this case also has the same speedup behaviour as the 'large' case. For seven 800MHz Pentium III computer it is assumed that the speedup compared to serial machine is 6.0. In the table below the time taken on an 800MHz serial computer is estimated from the time taken on the 1.0 GHz Pentium III. The time taken a parallel network of seven 800MHz computers is estimated using the estimated time for the serial 800MHz computer. The time taken on a small parallel network of three computers is also estimated. From section 8.4.1 the speedup for three computers is estimated to be 2.7. The estimated times taken to run the cases on the parallel network are given in Table 8-17.

Table 8-17 – Estimated runtimes on parallel network for FRD Warehouse Scenarios

Case	Time on 1GHz	Est. Time on 800MHz	Est. Time on 7 800MHz	Est. Time on 3 800Mhz
1	57hrs	71hrs 15mins	11 hrs 52mins	26hrs 23mins
2	71hrs 38mins	89hrs 22mins	18hrs 54 mins	33hrs 5mins
3	37hrs	46hrs 15mins	7hrs 42mins	17hrs 7 mins
4	31hrs	38hrs 45mins	6hrs 27mins	14hrs 35mins
5	68hrs	85hrs	14hrs 10mins	31hrs 28mins
6	31hrs	38hrs 45mins	6hrs 27mins	14hrs 35mins
7	93hrs 56mins	117hrs 25mins	19hrs 34mins	43hrs 49mins

When seven computers are used in parallel the total runtime is now 85 hours; when three computers are used in parallel the total runtime is ~180 hours. Both of these times compare favourable to the 388 hours that was previously obtained using a single serial 1.0GHz Pentium III computer. The runtime is overestimated due to the

assumption that the 1.0GHz machine only has the performance of an 800 MHz Pentium III when used as part of the parallel network. The speedup is also problem size dependent and is therefore probably underestimated for the majority of the cases.

Another factor to consider is the problem sizes used, in particular for case 7. The 1GHz Pentium III was unusually equipped with 1GB of RAM. Typically computers equipped with this type of processor are endowed with 256MB of RAM and case 7 would not have been accommodated. It is estimated that the memory usage for case 7 was about 516MB. DM-SMARTFIRE could have easily fitted the problem onto seven or three computers assuming all the computers were equipped with 256MB ram. If the computers were equipped with 128MB ram the problem would fit onto the 7 computer network.

8.8.2 Running Case 1 on an actual network

Case 1 of the FRD warehouse study was run on a network of computers at the University of Greenwich to simulate a typical FSE's office engineering environment. It was run during the daytime when the computers that formed the network were also being used by other users to perform such tasks as word processing, data analysis, surfing the internet and running serial CFD based fire simulations. The computers that formed the parallel network are given in Table 8-18. The network consisted of three 1.9GHz Pentium IV and two 733MHz Pentium III class computers.

Table 8-18 – Performance of CPUs used in parallel network for FRD Warehouse simulation

CPU	Performance index (relative to 1.9GHz)
1.9GHz Pentium IV (x 3)	1
733MHz Pentium III (x 2)	0.62

The time taken to run the simulation on a single 1.9GHz Pentium IV computer was 46 hours and 49 minutes. Ideally the runtime on the parallel network would be about 11 hours. When the problem was run on the network the overall runtime was 15 hours and 22 minutes. However it must be noted that dynamic load balancing was used and the computers were performing various tasks other than the parallel simulation of the FRD warehouse. This case has demonstrated the practical applicability of parallel

processing methods, when used with dynamic load balancing within a FSE's office environment.

8.9 Concluding remarks

Parallel processing was implemented within the SMARTFIRE fire modelling software. Both the Shared Memory (SM) and Distributed Memory (DM) approaches were implemented and tested. It was found that the DM approach was the most suitable approach and even outperformed the SM approach on a shared memory machine and the SM software approach was rejected in favour of the DM approach. Another problem with the SM approach is the current rarity of SM based PC hardware in an office environment. The DM approach utilised computers attached via a LAN.

As the parallel processing takes place within an office environment the software was implemented to take into account the possible heterogeneous nature of the available computers. This objective was met by the implementation of a dynamic load balancing mechanism that would distribute the workload based on the performance of each computer used within the parallel processing network to minimise the wall clock time of the fire simulation.

The possible impact of network usage and processor usage by an additional user was examined. It was found that only the most extreme usage of network facilities would affect the parallel performance. However the addition of extra computational load on one of the processing nodes had an adverse effect on the parallel performance and needed to be addressed. The dynamic load balancing mechanism ensured that other computer users did not adversely affect the parallel performance. Furthermore this dynamic load balancing scheme ensured that the additional computer user was not adversely affected by the FSE running a parallel computation.

The parallel processing performance was found to be good for the types of machines used. The results were not particularly close to the ideal performance or that of dedicated parallel platforms when a large number of processors were utilised but this was never expected from general office base PCs. However, it was demonstrated that

significant performance benefits could be extracted from both homogeneous and heterogeneous PC networks of a moderate size.

The theoretical memory usage study found that the vast majority of the memory could be utilised for a small number of processors. Even for a 30 processor network, problem sizes 24 times larger than the maximum serial problem size can be performed. This means that a FSE can not only perform calculations faster than is possible on a serial machine but that larger more complex simulations are also feasible.

Future developments of computing hardware could reduce the effectiveness of the parallel processing techniques developed in this thesis unusable due to the increase in serial processing power reducing the speed-up scalability to lower levels. However developments in networks and available memory should alleviate this problem so that the methods developed in this thesis will still be applicable in the future.

Finally a real fire safety analysis is used as a case study for the practical benefits of parallel processing. It has been demonstrated that parallel techniques can be used to provide a powerful computing resource for the FSE using readily available pre-existing equipment. The use of parallel processing technology allows the FSE to take advantage of additional computers which would normally only use a small amount of their processing potential. By using this computing resource the FSE can reduce the timeframes necessary for modelling the advanced fire modelling scenarios that will be necessary for gaining acceptance in performance based safety codes.

9 Conclusions

In this chapter, a review of how this thesis has addressed the original research questions posed in chapter 1 is given. More detailed conclusions have already been provided at the end of each section so the conclusions that follow will be of a more general nature.

From the work of FIREDASS (chapter 4) a numerical model was developed to aid in the development and optimisation of water mist based fire suppression and extinguishment systems. The aim was to provide industry with a tool that can be used to help speed up, and reduce the costs involved in, the design of such systems. The full model included submodels to simulate the fire, radiation field, water mist, fire suppression, temperature sensors and the misting nozzle activation system. Given the assumptions of the model, to be considered a practical tool, the FIREDASS model requires accurate release rate and boundary data and a sufficiently powerful computer. If these conditions are met, the model can be used to better target the testing necessary for development and approval of the detection and water mist system.

The FSEs raised concerns over the possible reliability of the results and the time taken to obtain these results from CFD codes. These concerns are applicable to both the FIREDASS model and CFD fire models in general. Both of these issues need to be addressed to allow the more widespread use of CFD based fire methods in performance based codes. These two issues provided the direction for the research within this thesis.

Can a suitable benchmark/standard for CFD based fire-modelling codes, which is free from manufacturer and user bias, be developed?

- Which types of problem should be selected for the benchmark process?

The benchmark process was split into 2 phases. Each of these phases was split into two categories, CFD and fire test cases. Unlike some previous tests of fire models the CFD capability of the software product (SP) was also tested, this was necessary to ensure that the basic physics of the SP operated correctly this has generally been

ignored. For the fire cases both simple heat source scenarios and combustion model scenarios were simulated. The first phase was rigidly controlled and ensured that all the test cases for each code were specified as near identically as possible. In the second phase the software producers were free to specify the problem as they wished within the limitations of the information supplied for a scenario allowing their code to be shown in its best light. The cases selected here were designed to demonstrate the concept and is not a definitive test suite. Further cases will need to be added to fully demonstrate the CFD based fire modelling codes.

- What methods will reduce / eliminate user bias? Typically bias play a part in most validation processes as developers are allowed to use their own judgement in simulating a fire modelling scenario.

The first phase was rigidly controlled and ensured that all the test cases for each code were specified as near identically as possible. This included the mesh specification, time step size and the parameters used in combustion and turbulence models. This allowed a fair comparison between the codes, removing many of the ambiguities that may exist with different users being able to set up problems using a range of parameters and physical sub-models. This problem of user bias has occurred in many model comparisons before (see section 2.3.1.1.2).

Each phase 1 simulation was performed at least once. The participants were also requested to run at least two of the 10 phase 1 simulations using their SP. Participants were free to choose which two simulations to run. Participants were of course free to (and were indeed encouraged to) run all 10 of the phase 1 simulations. It was however imperative that the participants did not inform the assessor/author which of the phase 1 simulation they intended to run. The purpose of repeating the simulations was to ensure that the author/assessor had not fabricated results or incorrectly set up the scenario. This therefore eliminated bias that may have been caused by lack of experience with a particular SP or perhaps personal bias that may be held by the assessor.

- How can software developers demonstrate features that differentiate their field model from other field models that exist?

The problem with phase 1 of the benchmarking was that the models would not be performing particularly well due to the simplifications made. The purpose of the phase 1 protocol was to demonstrate that a particular code could achieve a minimum standard that was attainable by all the SPs involved. Furthermore there is no way to differentiate between the codes which may possess better sub-models or boundary condition handling. This issue was addressed by allowing the software producers to proceed to phase 2 once their SP had demonstrated that phase 1 could be achieved. In this phase of the assessment the models full capabilities could be demonstrated. Phase 2 was free format in nature but the simulations could only be performed with features that exist within the general version of the code i.e. additional code or external routines are generally not permitted. This was demonstrated for SMARTFIRE and the cases selected for phase 2 showed considerable improvements over the phase 1 testing protocols. All participants would have to complete a similar pro-forma as those supplied for the phase 1 simulations so that the results could be verified.

- Can CFD methods in general be demonstrated to be useful for fire modelling?

Three software vendors agreed to participate in phase 1 and all the codes demonstrated that they could meet the phase 1 standard. By having a range of SPs, and not just a single product, that can achieve the standard it is hoped that the CFD methodology can be demonstrated to be useful for the purposes of fire safety engineering. Previous to the work in this thesis little had been demonstrated on formalising independent verification, validation and cross comparison of CFD software utilised for fire modelling purposes.

Can parallel processing techniques be usefully applied to standard office based PCs to increase the computational power available to a FSE for the purposes of CFD based fire modelling?

- How much faster will the code run on a parallel processing system? Ideally this would be to the sum of the processing power of the computers involved in the parallel computation. In reality this is not even achieved on dedicated parallel processing hardware but can enough power be extracted from a network of PCs connected via a conventional LAN to make a useful resource for a FSE.

From chapters 7 and 8 it is apparent that standard office based PCs can be used profitably for parallel fire based CFD calculations when a moderate number of PCs were used (< 16). Currently shared memory environments within an office context are rare although this may change in the future with manufacturers pursuing shared memory multiprocessor technology more vigorously. This current effort is directed towards server applications but may have use for CFD modelling. However single processor PC boxes networked together are very common and provide a suitable platform for parallelism. For a network of 800MHz Pentium III computers attached to a 100Mbps LAN a problem composed of ~100,000 cells, the 'Large' case ran 5.31 times faster with 6 processors and 9.3 times faster with 12 processors.

The type of parallel processing described in this thesis only required the use of commonly available computer hardware. This consisted of a number of standard Microsoft Windows based PCs equipped with a 100Mbps Ethernet card connected to a 100Mbps Ethernet LAN. This is a common configuration for PCs in the office environment and it is quite likely that a FSE would require no further investment in hardware to take advantage of the parallel processing techniques described in this thesis.

In the past most of the work involving parallel processing on PCs has required the use of the Linux operating system. All the work in this thesis has been based on using Windows NT based technologies that are commonly used by most FSEs.

- What problem sizes can be tackled on a parallel processing system? Ideally this would be linearly related to the sum of the memory available on all the PCs, i.e. double the memory allows the problem size to be doubled.

From section 8.6 it can be seen that with eight homogeneous computers a problem that is seven times larger than the largest problem size that can be accommodated on a single computer. Even with a large number of computers it is anticipated that most of the memory will be well utilised. For thirty computers it is anticipated that a problem size twenty four times larger than can be accommodated on a single computer.

- What are the limitations of PC parallel processing? At what point would the use of parallel processing become unprofitable. What is the potential future of using parallel processing on PCs with future developments of PC based technologies.

From the work performed in this thesis it can be seen that very large scale parallel processing is impractical (see section 8.4.1). For the large case using 800MHz Pentium III computers the practical limit was about 16 processors. However if very large problems, i.e. problems that only fit onto a large number of processors, are run then the practical limit is much higher. As processors get faster the relative parallel speedup will deteriorate if networking technologies remain the same (see section 8.7). This problem is likely to be alleviated by advances in networking technology and also by the increase in memory size.

- Can methods be devised to efficiently take advantage of a network of heterogeneous (non-identical) PCs? A FSE's engineering environment may consist of non-identical computers that have no known performance benchmark. Ideally the parallel processing software itself should determine the performance of each of the computers used for parallel processing.

A dynamic load balancing scheme (see section 8.5.2) was designed, implemented and tested that could determine and monitor the relative processing power of a set of heterogeneous PC; the computational load is then distributed according to the relative processing powers of the PCs involved.

- Can parallel processing be implemented to maximise the potential processing power without adversely affecting other computer users? Within a FSE's office environment other users will be using computers that could potentially be used as

part of a parallel processing job. Methods need to be devised that ensure that neither the FSE using parallel processing of other computer users are adversely affected by each other.

It was found from testing that parallel processing would adversely affect other computer users if they attempted to concurrently run other computational jobs on computers that were part of the parallel fire modelling simulation (see section 8.4.3). In addition to this the parallel computation would be adversely affected by the other concurrent jobs as well. An adaptive mechanism based on the dynamic load balancing scheme was extended to monitor the work load exerted on the computers within the parallel network and helped to alleviate this problem for both the parallel computation and other serial computation (see section 8.5.2).

- Can performance improvements be made to the serial code without affecting the object-orientated structure of the code? There is little point in parallelising software if the performance can be easily speeded through modification to the serial code; this could include algorithm and implementation details.

In section 7.4.10 it was found that the use of 1D-storage arrays speeded up the serial code by 50% on a 733MHz Pentium III computer. However, this was a minor modification in terms of changes to the source code with no change to the object orientated nature of the code.

10 Further work

Although much has been achieved there is always more that can be done. In this chapter, ideas for further work are expressed. Some of these ideas will be carried through in the near future and others may or may not happen depending on circumstances.

10.1 Implementation of FIREDASS in parallel

Using SMARTFIRE as the core CFD engine the FIREDASS modules could be ported to a parallel version to improve the throughput of the code. Some difficulties would exist with parallelising the particle tracking Eulerian-Lagrangian model as the particles would move over domains and would therefore vary the computational load on each node as they travel across the sub-domain partitions.

10.2 Benchmark – More cases

Generation of more and perhaps better comparison cases for benchmarking is required for the future. This is required so the class of problem the CFD code is validated for can be extended. This may simply be with reference to literature or commissioning of cases specifically for CFD validation purposes. Many fire experiments are unsuitable for validation purposes as they are conducted without any modelling agenda. One possible candidate case by Isaksson et al [IPT1997] gives experimental data and simulation data from JASMINE and SOFIE for a fire in a room with a perforated suspended ceiling. Another possible source of good experimental data concerns a room fire trial conducted by Neilson [Nie2000].

10.3 Improved Error Reporting

Improved methods of reporting numerical results including error bars need to be devised. This is not just suitable for this benchmarking process but for reporting numerical results in general. The main obstacle to this is the time taken to run a CFD code. An accurate error estimate can be obtained by using 3 meshes successively refined by a factor of 2 from the previous mesh. However, this is currently impractical due to the timeframe required to run these exceptionally refined meshes. Until a better

method can be found, utilising two meshes of differing refinements may be the most practical way of analysing grid convergence error.

10.4 Interactive Parallel SMARTFIRE CFD

As mentioned previously one of the major benefits of the serial version of the SMARTFIRE CFD engine was its interactive nature. This allows user to interactively change solvers, solver relaxations, time step size etc. This was removed from the parallel version to remove this complexity from this initial feasibility study of parallel processing on a network of PCs. Another difficulty was the incompatibility between the graphics libraries and the MPI libraries. However from the point of view of a FSE this interactivity is important and should be incorporated back into parallel version.

The incompatibility between the MPI libraries and graphic libraries can be overcome by creation of a separate GUI program that communicates to the master process. This communication could be conducted via fast memory files to minimise the effect of the GUI on the overall runtime of DM-SMARTFIRE. The serial code although fully interactive only really acts on the user interaction when a complete outer iteration of the solver has been completed. This is an obvious hook from which the batch mode parallel programs can interact with a separate GUI to implement the interactive nature required. The following code fragment could be placed in the DM-SMARTFIRE source code after the end of the outer iteration:-

```
Do {
   If (my_mpi_id == 0) get_interation(); // master gets interaction
   MPI_Bcast(&interaction,..,..,0,..); // interaction delivered to
                                       // slaves
   Switch (interaction) {
      Case 1 : // set relaxation

      Case 2 : // set solver iterations
      ..
      ..
      ..

   }
} while (interaction != END_INTERACTION);
```

10.5 Parallel Scheduling

One problem with parallel processing on a network of PCs is that it is unlikely that a FSE would find themselves in the position of having lots of PCs freely available. They may have 1 or 2 available during the daytime with perhaps a large number of PCs becoming available when other users go home for the evening. Given that a dynamic load balancing code already exists it should be possible to put in some extra constraints so that during the day only a minimal amount of work will be exerted on other user machines with their workload changing according to time of day, usage or login status. The scheduling could be extended to automatically launch new parallel tasks on completion of a task to ensure that the computers are not idling for any length of time even when the FSE is not around to manually launch a job in a fashion similar to the batch and scheduling tools used on old mainframe machines.

10.6 Improved Dynamic Load Balancing

In the dynamic load balancing system implemented in this thesis the rebalancing is performed by saving the results of the simulation and then restarting the simulation with the new load balance. This has the disadvantage of requiring that all the results from the problem are first written and then read from disk storage which incurs a time penalty. In addition the memory on each processor needs to be de-allocated and re-allocated for each rebalance which also incurs a time penalty. Another possibility would be to migrate the cells from one processor to another directly using message passing. This would require some additional bookkeeping and memory re-allocation within the code but should be faster than the current method. This method was not initially implemented as it was unknown at the beginning of the study whether this extra implementation effort would be worthwhile.

Another potential improvement to both static and dynamic load balancing schemes could be the introduction of more advanced domain decomposition techniques. These should reduce the size of the communications between processors compared to the simple 1D portion scheme used and could therefore improve the speed-up scalability. One such domain decomposition tool is Jostle [Wal1995]. At the time of writing

Jostle had been tested with DM-SMARTFIRE but Jostle had a bug causing the software to crash. Jostle will be tested again in the future when a bug-fixed version becomes available.

10.7 Data analysis and Visualisation

With the parallel CFD based fire model that has been developed it is quite possible to run a larger problem than can be accommodated on a serial machine. This may have consequences for data analysis as the output data may also not fit onto a serial machine for analysis. This is particularly true if serial SMARTFIRE in interactive mode is used as the visualisation program. This problem can be circumvented to some extent by serial amendments to only read a limited set of variables although it is quite possible that just one field variable could exhaust all the memory of a serial machine. Parallel methods could be applied to this problem to allow the visualisation to be created.

References

[AIAA1998] AIAA, "Guide for the Verification and Validation of Computational Fluid Dynamics Simulations", AIAA G-077-1998.

[Alp1984] Alpert, R. L., "Calculated interactions of Sprays with large-scale buoyant flows", ASME Trans. Heat Transfer, 106, pp310-317, 1984.

[Alp1985] Alpert, R. L., "Numerical modeling of the interaction between automatic sprinkler sprays and fire plumes", Fire Safety Journal, 9, pp123-136, 1985.

[Amd1967] Amdahl, G. M.,"Validity of the single-processor approach to achieving large scale computing capabilities", In Proc AFIPS, pages 483-485, 1967

[Bak1999] Baker, M. A., "MPI on NT: The Current Status and Performance of the Available Environments", NHSE Review, Volume 4, No. 1, September 1999.

[BCF+1995] Boden, N. J., Cohen, D., Felderman, R. E., Kulawik, A. E., Seitz, C. L., Seizovic, J. N., and Weng-King Su, "Myrinet: A Gigabit-per-Second Local Area Network", IEEE Micro, Vol. 15(1), pp29-36, February 1995.

[Bea1992] Beard, A., "The Limitations of Computer Models", Fire Safety Journal, 18, pp375-391, 1992.

[Bea1997] Beard, A. N., "Fire Models and Design", Fire Safety Journal, 28, pp117-138, 1997.

[Bea2000] Beard, A. N., "On *a priori*, *blind* and *open* comparisons between theory and experiment", Fire Safety Journal, 35, pp63-66, 2000.

[Bil1993] Bill, R. G., "Numerical Simulation of actual delivered density measurements", Fire Safety Journal, 20, pp227-290, 1993.

[BK1996] Björkman, J., and Keski-Rahkonen, O., "Simulation of the Steckler Room fire experiments by using SOFIE CFD model. Technical Report 265, VTT Technical Research Centre of Finland, 1996

[Bou1877] Boussinesq, J., "Theorie de L'Ecoulement Toubillant", Mem. Acad. Sci., 23(46), 1877.

[BS1994] Baltas, N. D., Spalding, D. B., "MIMD PHOENICS: Porting a Computational Fluid Dynamics Application to a Distributed Memory MIMD Computer", Massively Parallel Processing Applications and Development, Eds L.Dekker, et.al., Elsevier, Amsterdam, 1994.

[Cam2001] Camp, P., "Computer Modelling vs Reality – Can we predict the behaviour of Fires?", Proc. Interflam 2001, pp811-822, Interscience 2001.

[CF1991] Chow, W. K., and Fong, N. K., "Numerical simulation on cooling of the fire-induced air flow by sprinkler water sprays", Fire Safety Journal, 17, pp263-290, 1991.

[CF2001] Chitty, R. and Foster, J., "Application of computer modelling to real fire incidents", Proc. Interflam 2001, pp939-950, Interscience 2001.

[CFX1997] CFX User Guide. AEA technology 1997.

[CFX2003] CFX-5.6 User Manual, ANSYS CFX, Waterloo, 2003.

[Cha] http://www.cham.co.uk/

[CKZ1986] Cheesewright, R., King, K. J., and Ziai, S., "Experimental data for the validation of Computer Codes for the prediction of Two-Dimensional Buoyant Cavity flows", Significant Questions in Buoyance Affected Enclosure or Cavity Flows, vol. HTD-60, pp 75-81, ASME, New York, 1986

[CK1986] Cox, G., and Kumar, S., "Field Modelling of Fire in Forced Ventilated Enclosures", Combustion Science and Technology, vol 52, pp 7-23, 1986.

[Cox1990] Cox, G., "Fire Simulation in the design evaluation process", Interflam '90, Fire Research Station, Building Research Establishment, Borehamwood, Herts, 1990.

[Cox1995] "Combustion Fundamentals of Fire", Editor: Cox G., Academic Press, 1995.

[Cro1998] Croft, T. N., "Unstructured Mesh - Finite Volume Algorithms for Swirling, Turbulent, Reacting Flows", University of Greenwich, London, April 1998.

[Cru2000] Crutchington, R., "MPI on NT: An evaluation and comparison of current environments.", Final Year Project, Distributed Systems Group, Division of Computer Science, University of Portsmouth, 2000.

[CSS1977] Crowe, C.T., Sharma, M.P. and Stock, D.E.; "The Particle-Source-In Cell (PSI-Cell) Model for Gas-Droplet Flows."; *Trans. of ASME - J. of Fluids Engr.*; Jun. 1977, pp325-332.

[Dey2002] Dey, M. K., "Evaluation of Fire Models for Nuclear Power Plant Applications: Cable Tray Fires", NISTIR 6872, 2002

[EGP+1999] Ewer J., Galea E.R., Patel M.K., Taylor S., Knight B. and Petridis M., "SMARTFIRE: An Intelligent CFD Based Fire Model", Fire protection Engineering, vol 10, no 1, pp 13-27, 1999.

[EJG+2002] Ewer, J., Jia, F., Grandison, A., Galea, E., Patel, M., "SMARTFIRE V3.0 User guide and technical manual", University of Greenwich, 2002.

[Eur] http://www.gmd.de/SCAI/europort-d/safety-analysis1.html

[Ewe2000] Ewer, J., "An investigation into the feasibility problems and benefits of re-engineering a legacy procedural CFD code into an event driven, object orientated system that allows dynamic user interaction", PhD Thesis, University of Greenwich, July 2000.

[Far1988] Farhat, C., A simple and efficient automatic FEM domain decomposer. Computers and Structures, 28:579-602, 1988.

[Fiv1988] Fiveland, W. A., "Three-Dimensional Radiative Heat-Transfer Solutions by the Discrete-Ordinates Method", J. Thermophysics, Vol 2 (4), pp309-316, October 1988.

[FKE+1993] Fardell, P. J., Kumar, S., Ellwood, J. A., Rowley, J. A., and Vollam, S., "A Study of Life Threat in Bus Fires", Proc. Interflam'93, pp401, 1993.

[FKAG1994] Fletcher, D. F., Kent, J. H., Apte, V. B., and Green, A. R., "Numerical Simulation of Smoke Movement from a Pool Fire in a Ventilated Tunnel", Fire Safety Journal, 23, pp305-325, 1994.

[For1994] Message Passing Interface Forum. MPI: A Message-Passing Interface Standard. University of Tennessee, May 1994.

[Fox1988] Fox, G. C., Numerical Algorithms for Modern Parallel Computers, Springer-Verlag 1988.

[Fre1995] Freitas, C.J., "Perspective: Selected Benchmarks From Commercial CFD Codes", Journal of Fluids Engineering, 117(2), pp210-218. 1995

[GAC1997] Glocking, J.L.D, Annable, K., Campbell, S.C. "Fire Spread in multi-storey buidings – *'Fire break out from heavyweight unglazed curtain wall system – Run 007'* ", LPC Laboratories rep. TE 88932-43, 25 Feb 1997.

[Gal1989] Galea E.R., "On the field modelling approach to the simulation of enclosure fires", Journal of Fire Protection Engineering, vol 1 (1), 1989, pp 11-22.

[GBD+1994] Geist, A. Beguelin, J. Dongarra, J. Weicheng, R. Manchek, and V. Sunderam. "PVM: Parallel Virtual Machine – A Users' Guide and Tutorial for Networked Parallel Computing". MIT Press, 1994.

[GBH1996] Galea, E. R., Berhane, D. and Hoffman, N. A., "CFD Analysis of Fire Plumes Emerging from Windows with External Protrusions in High-Rise Buildings", Proc. Interflam'96, Compiled by C. Franks and S Grayson, ISBN 0 9516320 94, pp835-839, 1996.

[GDJ1990] Gidaspow, D., Ding, J., Jayaswal, U., K., Multiphase Navier-Stokes Equation solver", Num. Meth. For Multiphase Flow, FED-Vol. 91 pp47-56, ASME, 1990.

[GGP2003a] Grandison, A. J., Galea, E. R., Patel, M. K., "Development of Standards for Fire Field Models – Report on Phase 1 Simulations", FRD Publication 2/2003, Fire Research Division, ODPM (Office of the Deputy Prime Minister), 2002.

[GGP2003b] Grandison, A. J., Galea, E. R., Patel, M. K., "Development of Standards for Fire Field Models – Report on SMARTFIRE Phase 2 Simulations", FRD Publication 1/2003, Fire Research Division, ODPM (Office of the Deputy Prime Minister), 2002.

[GGPM1998a] Grandison, A., Galea, E., Patel, M. K., Mawhinney, R. N., "The Fire Model Validation Report."; *FIREDASS report*; Proj. ref. no. 2.2-4, Mar. 1998.

[GGPM1998b] Grandison, A., Galea, E., Patel, M. K., Mawhinney, R. N., "The Interaction Model Development and Validation Report."; *FIREDASS report*; Proj. ref. no. 3.2-1/3.2-2, Jun. 1998.

[GGPM1999] Grandison, A., Galea, E., Patel, M. K., Mawhinney, R. N., FIREDASS report – "The Cargo Interaction model development and validation report"; pro. ref. no. 3.4-1; 3.4-2, 1999

[GI1992] Galea, E. R. and Ierotheou, C., "Fire-Field Modelling on Parallel Computers", Fire Safety Journal, 19, pp251-266, 1992.

[GI1993] Galea, E. R. and Ierotheou, C., "A Parallel Implementation of a General Purpose Fluid Flow Code and its Application to Fire Field Modelling", Proceed. Parallel Computing 1991. Ed:Joubert, Evans and Liddel.Elsevier Press 1993.

[GKP1996] Geist, G. A., Kohl, J. A., and Papadopoulos, P. M., "PVM and MPI: A Comparison of Features", Calculateurs Paralleles Vol. 8 No. 2 pp 137-150, June 1996

[GKP+1999] Galea E., Knight B., Patel M., Ewer J., Petridis M., and Taylor S., "SMARTFIRE V2.01 build 369D, User Guide and Technical Manual", SMARTFIRE CD and bound manual, 1999

[GL1997] Gropp, W. D, and Lusk, E., "Why are PVM and MPI so Different?", Proceedings of 4th European PVM/MPI Users' Group Meeting, Cracow, Poland, November 1997, Vol 1332, pp 3-10.

[Goo2000] Goodchild, C., "Firedass", Air & Space Europe, Volume 2, Issue 1, January-February 2000, pp96-100

[Ham1947a] Hamaker, H. C., "Radiation and Heat Conduction in Light-Scattering Material: Part I Reflection and Transmission", Philips Research Report, 2, 55-67 (1947).

[Ham1947b] Hamaker, H. C., "Radiation and Heat Conduction in Light-Scattering Material: Part II", Philips Res. Rep., 2, 103 (1947).

[Has1996] Hassan, M.A., Theoretical simulation of fire extinction by water spray in a computer cabinet, Applied Mathematical Modelling, Volume 20, Issue 11, November 1996, Pages 804-813

[HE2002] Hume, B., and Eady, M., "The Use of CFD Computer Models for Fire Safety Design in Buildings: Large Warehouse Case Study", FRD Report 4/2002, Fire Research Division, ODPM (Office of the Deputy Prime Minister), 2002.

[Hes1981] Heskestad, G.; "Peak Gas Velocities and Flame Heights of Buoyancy-Controlled Turbulent Diffusion Flames."; *Proc. 18th Int. Symp. on Combustion*; 1981, pp951-960.

[HGM1989] Hoffman, N., Galea, E. R., Markatos, N. C., Mathematical Modelling of Fire Sprinkler Systems, Appl. Math. Modelling, Vol 13 (May), pp298-306, 1989.

[HK1998] Hostikka S and Keski-Rahkonen O., Results of CIB W14 Round Robin for Code Assessment Scenario B. Draft 31/08/98, VTT Technical Research Centre of Finland, 1998.

[HKKX2002] Jinsong Hua, J., Kumar, K., Khoo, B. C., and Xue, H., "A numerical study of the interaction of water spray with a fire plume", Fire Safety Journal, 37, Issue 7, pp631-657, October 2002

[HM1988] Hoffman, N. and Markatos, N. C., "Thermal radiation effects on fires in enclosures", Appl. Math. Modelling, vol. 12, pp129-140, 1988.

[HN1967] Harlow, F. H. and Nakayama, P.I., "Turbulent Transport Equations", Phys. Fluids, Vol. 10, pp2323-2332, 1967.

[How1968] Howell, J. R., "Application of Monte Carlo to heat transfer problems", Advances in Heat Transfer, vol. 5, Academic Press, New York, pp1-54, 1968.

[How1988] Howell, J. R., "Thermal Radiation in Participating Media: The Past, the Present and Some Possible Futures", Transactions of the ASME, Journal of Heat Transfer, 110, pp1220-1229, 1988.

[HS1952] Hestenes, M. R. and Stiefel, E., "Methods of Conjugate Gradients for Solving Linear Systems", Journal of Research of the National Bureau of Standards, 49(6), 409-436, 1952.

[HS1967] Hottel, H. C. and Sarofin, A. F., Radiative transfer, McGraw-Hill, N.Y., 1967.

[HT1978] Hubbard, G.L, and Tien, C.L., "Infrared Mean Absorption Coefficients of Luminous Flames", Journal of Heat Transfer, vol. 100, p 235, May 1978.

[Huh1989] Huhtanen, R., "Numerical fire modelling of a turbine hall"; Fire Safety Science, Proceedings of the 2nd International Symposium, pp771-779, IAFSS, 1989.

[HW1999] Hempel, R., and Walker, D. W., "The Emergence of the MPI Message Passing Standard for Parallel Computing", Computer Standards and Interfaces, Vol. 21, pages 51-62, 1999.

[IG1992] Ierotheou, C. and Galea, E. R., "A Fire-Field Model Implemented in a Parallel Computing Environment", Int J.Numerical Methods in Fluids vol 14, pp175-187 1992.

[ISO1999] ISO TC92/SC4, "Fire safety engineering -- Part 3: Assessment and verification of mathematical fire models", ISO/TR 13387-3, 1999.

[IPT1997] Isaksson S., Persson B., and Tuovinen H., "CFD-Simulatons of Fire Detection in a Room with a Perforated Suspended Ceiling, BRANDFORSK Project 628-951", SP Report 1997:43, SP Swedish National Testing and Research Institute, 1997

[Jay1969] Jayatileke, C. L. V., "The Influence of Prandtl Number and Surface Roughness on the Resistance of the Laminar Sublayer to Momentum and Heat Transfer", Prog. Heat Mass Transfer, Vol 1, pp193, 1969.

[JEI+1997] Johnson, A. D., Ebbinghaus, A., Imanari, T., Lennon, S. P. and Marie N., "Large-scale free and impinging turbulent jet flames - modelling and experiments", Trans. IChemE, Vol.75, 1997.

[JEG+2001] Janes, D., Ewer, J. A. C., Galea, E. R., Patel, M. K., and Knight, B., "Automatic Dynamic Control of CFD Based Fire Modelling Simulations", Proc. Interflam 2001, pp811-822, Interscience 2001.

[JFPR2000] Jones, W. W., Forney, G. P., Peacock, R. D., Reneke, P. A., "A Technical Reference for CFAST: An Engineering Tool For Estimating

Fire And Smoke Transport, NIST Technical Note 1431, Building and Fire Research Laboratory, January 2000.

[JGP1997] Jia, F., Galea, E.R. and Patel, M.K.; "Simulating 'FLASHOVER' and 'BACKDRAFT' Type Events Using Fire Field Models - A First Approximation."; *Journal of Fire Prot. Engineering*; Vol. 9, pp1-17, 1997.

[JGP1999] Jia, F., Galea, E.R. and Patel, M.K.; "The Numerical Simulation of the Noncharring Pyrolysis Process and Fire Development within a Compartment."; *Applied Math. Modelling*; Vol. 23, 1999, pp587-607.

[Jia1999] Jia, F., "The Simulation of Fire Growth and Spread within Enclosures using an integrated CFD Fire Field Model", PhD thesis, University of Greenwich, UK, 1999.

[Jia2001] Jia, F., Personal communication

[Joh1992] Johnson, S. P., "Mapping Numerical Software onto Distributed Memory Parallel Systems", PhD thesis, University of Greenwich, 1992.

[Kar1992] Karlsson, B., "Modelling fire growth on combustible lining materials in enclosures", 992, Report TVBB-1009, Lund University, Department of Fire Safety Engineering, Lund, Sweden, 1992.

[Kar2002] Karniadakis, G. E., "Quantifying Uncertainty in CFD", Journal of Fluids Engineering, 124(1), pp2–3, 2002.

[KB1997] Keramida, E., Boudouvis, A., "Radiation Model Implementation."; *FIREDASS report*; Proj. ref. no. 3.1.3-5, Feb. 1997.

[Ker1997a] Keramida, E., "The Discrete Transfer and the Six Flux Radiation Models on Steckler's Room."; *FIREDASS report*; Proj. ref. no. 3.1.3-4, Jul. 1997.

[Ker1997b] Keramida, E., "Attenuation of Fire Radiation Through Water Droplets."; *FIREDASS report*; Proj. ref. no. 3.1.3-2, Apr. 1997.

[KGC1991] Kumar S., Gupta A.K. and Cox G., "Effects of Thermal Radiation on the Fluid Dynamics of Compartment Fires", Fire Safety Science - Proc. of the Third Intl. Symp., pp 345-354, 1991.

[KGHP1994] Kerrison, L., Galea, E. R., Hoffmann, N. and Patel, M. K.; "A Comparison of a FLOW3D Based Fire Field Model with Experimental Room Fire Data."; *Fire Safety J.*; Vol. 23, 1994, pp387-411.

[KHL1997] Kumar S, Heywood GM, Liew SK. Superdrop modelling of a sprinkler spray in a two-phase CFD-particle tracking model. Fire Safety Science. Proceedings of the Fifth International Symposium, 1997. p. 889–900.

[KK1999] Karypis, G. and Kumar, V.,"A fast and highly quality multilevel scheme for partitioning irregular graphs". *SIAM Journal on Scientific Computing*, 20(1), 1999.

[KKJ1980] Kim, J., Kline, S. J., Johnston, J. P., "Investigation of a Reattachment Turbulent Shear Layer: Flow over a Backward-Facing Step", Transactions of the ASME, Journal of Fluids Engineering, 102, 302-308, 1980

[KMG+1994] Kerrison, L., Mawhinney, N., Galea, E.R., Hoffmann, N. and Patel, M.K.; "A Comparison of Two Fire Field Models with Experimental Room Fire Data."; *Proc. 4th Int. Symp. on Fire Safety Sci.*; 1994, pp161-172.

[KRSK2001] Kaurinkoski, P., Rautaheimo, P., Siikonen, T., and Koski, K., "Performance of a Parallel CFD-Code on a Linux Cluster"; Parallel Computational Fluid Dyanmics – Trends and Applications, Ed by Jensson, C. B., Elsevier Science pp107-114, 2001.

[KSBM2000] Keramida, E.P., Souris, N.N., Boudouvis, A.G. and Markatos, N.C.; "Numerical Modeling of Radiative Heat Transfer in Integrated CFD Fire Modeling."; *J. of Applied Fire Science*; Vol. 9(1), 2000, pp3-19.

[Kum1983] Kumar, S., "Mathematical Modelling of Natural Convection in Fire – A state of the Art Review of the Field Modelling of Variable Density Turbulent Flow", Fire and Materials; Vol. 7 (1), pp1-24, 1983.

[Lar1983] Larsen, M. E., "Exchange Factor Method and Alternative Zonal Formulation for Analysis of Radiating Enclosures Containing Participating Media", Ph. D. Thesis, University of Texas, Austin, 1983.

[LC1965] Lathrop, K. and Carlson, B., "Discrete-ordinates angular quadrature of the neutron transport equation in (x, y, z) geometry", Los Alamos Scientific Laboratory, Los Alamos, NM, Reports LASL-3186, 1965.

[Lin] http://www.linux.org

[LMR1997] Lewis M.J., Moss M.B. and Rubini P.A., "CFD Modelling of Combustion and Heat Transfer in Compartment Fires", Fire Safety Science, Proc. of the 5th Int. Symp., pp 463-474, 1997.

[LS1981] Lockwood, F. C. and Shah, N. G., "A new radiation solution method for incorporation in general combustion prediction procedures", 18th Symp. (Int.) on Combustion, pp1405-1414, 1981.

[LSH+1999] Levy, S., Sandzimier, J., Harvey, N., Rosenbluth, E., Karki, K., Patankar, S., "CFD Model for Transverse Ventilation Systems", Proceedings of the First International Conference on Tunnel Fires and One Day Seminar on Escape from Tunnels, pp223-233, Lyon, France, 5-7 May 1999.

[LT2001] Law, R. A., and Turnock, S. R., "Utilising Existing Computational Resources to Create a Commodity PC Network suitable for fast CFD Computation"; Parallel Computational Fluid Dyanmics – Trends and Applications, Ed by Jensson, C. B., Elsevier Science pp115-122, 2001.

[Mau1990] Mauzerall, D.L.; "Protecting the Ozone Layer. Phasing out Halon by 2000."; *Fire Journal (Boston)*; Vol. 84, Sep.-Oct. 1990, pp22-31.

[MBR+2001] McGratten, K. B., Baum, H. R., Rehm, R. G., Hamins, A., Forney, G. P., Floyd, J. E., Hostikka, S., "Fire Dynamics Simulator (Version 2) – Technical Reference Guide", NISTIR 6783, November 2001.

[MC1984] Markatos, N.C. and Cox, G.; "Hydrodynamics and Heat Transfer in Enclosures."; *Physico-Chem. Hydrody.*; Vol. 5, 1984, pp53-66.

[MC1996] Miles, S. D., and Cox, G., "Predictions of fire hazards associated with chemical warehouses", Fire Safety Journal, 27, pp265-287, 1996.

[McM1996] McManus, K., "A Strategy for Mapping Unstructured Mesh Computational Mechanics Programs onto Distributed Memory Parallel Architectures.", PhD Thesis, University *of* Greenwich, 1996.

[MGG+2000] Mawhinney, R. N., Grandison, A. J., Galea, E. R., Patel, M. K., Ewer, J., "The Development of a CFD Based Simulator for Water Mist Suppression Systems: The Development of the Fire Submodel", J. Applied Fire Science, Vol 9 (4) 311-345, 1999-2000.

[MGPG1998] Mawhinney, R. N., Galea, E., Patel, M. K., Grandison, A., "Mist Model - Development, Implementation and Validation."; *FIREDASS report*; Proj. ref. no. 3.1.1-2, Jun. 1998.

[MH1977] Magnussen, B. F., Hjertager, B. H., "On mathematical modelling of turbulent, combustion with special emphasis on soot formation and combustion", 16th Symp. (Int.) on Combustion, pp1383-1393, 1977.

[Mic] Microsoft Corporation – Windows NT operating system.

[MV2000] Madrzykowski, D., and Vettori, R. L., "Simulation of the Dynamics of the Fire at 3146 Cherry Road NE Washington D. C., May 30, 1999", NISTIR 6510, NIST publication. April 2000.

[NACB1997] Norris, J. O. W., Astill, A. G., Cursley, C., Burton, G., "Characterisation of the Spray from the GMAv Nozzle."; *AEA report*; File ref. no. AEAT-1195, Jan. 1997.

[Nam1996] Nam, S., "Development of a Computational Model Simulating the Interaction Between a Fire Plume and a Sprinkler Spray", Fire Safety Journal, 26, pp1-33, 1996.

[Nam1999] Nam, S., "Numerical simulation of the penetration capability of sprinkler sprays", Fire Safety Journal, 32, Issue 4, pp307-329, June 1999.

[Nie2000] Nielson, C., "An analysis of pre-flashover fire experiments with field modelling comparisons", Fire Engineering Research Report 2000/10, Department of Civil Engineering, University of Canterbury, New Zealand, March 2000, ISSN 1173-5996

[NMFK1996] Novozhilov, V., Moghtaderi, B., Fletcher, D. F., and Kent, J. H., "A Computational Fluid Dynamic Modelling of Wood Combustion", Fire Safety Journal, 27(1), pp69-84, 1996.

[Odi1999] Odic, R., FIREDASS report "Optimised System Specification and Demonstrator Design Specification"; proj. ref. No.5.2.3 and 5.2.4, 1999

[OM1998] Odic, R., Mangon, P., "Detection/Activation Model Development and Verification."; *FIREDASS report*; Proj. ref. no. 4.2.2-2/4.3.1-1, Jul. 1998.

[Oli1998] Oliefka, L., DLR document - Abstract and Result, ID No. FD-DS-450-01/0, 1998

[Pat1980] Patankar, S., "Numerical Heat Transfer and Fluid Flow", Intertext Books, McGraw Hill, New York, 1980

[PC1989] Patel, M. K., Cross, M. C., "The Modelling of Fluidized Beds for Ore Reduction", Int. Conf. Num. Meths. in Laminar and Turbulent flow, ed Taylor, Gresho, Sani and Hauser, Pineridge Press, pp 2051-2068, 1989.

[PLK1999] Prasad, K., Li, C., and Kailasanath, K., "Simulation of Water Mist suppression of a small scale methanol liquid pool fires", Fire Safety Journal, 33, pp185- 212, 1999.

[PI1995] Pelletier, D. and Ignat, L., "On the accuracy of the Grid Convergence Index and Zhu-Zienkiewicz Error Estimator" pp31-36, in Quantification of Uncertainty in Computational Fluid Dynamics – 1995, Joint JSME-ASME Fluid Mechanics Meeting, ASME FED Volume 213, Hilton Head, South Carolina August 14-18, Edited by Johnson, R. W., and Hughes, E. D., 1995.

[PRFK1998] Peacock, R. D., Reneke, P. A., Forney, L., Kostreva, M. M., "Issues in evaluation of complex fire models", Fire Safety Journal, 30, pp103-136, 1998.

[PS1972] Patankar, S. V. and Spalding, D. B., `A Calculation Procedure for Heat, Mass and Momentum Transfer in Three-Dimensional Parabolic Flows', International Journal of Heat and Mass Transfer, 15, 1787-1806, 1972.

[PWC1989] Pericleous, K. A., Worthington, D. R. E., and Cox, G., "The field modelling of fire in an air-supported structure.", Fire Safety Science - Proceedings of the 2nd International Symposium, International Association for Fire Safety Science, pp871-880, Hemisphere Publishing Corporation, 1989.

[QRJ1995] Quintere, J. G., Rinkin, W. J. and Jones, W. W., "Applications of a model to predict flame spread over interior finish materials in a compartment", J. Fire Protection Engineering, Vol. 7, pp1-13, 1995.

[Raw1994] Raw, M. J., "A Coupled Algebraic Multigrid Method for the £D Navier Stokes Equations", 10th GAMM Seminar, Kiel, 1994.

[Raw1996] Raw, M. J., "Robustness of Coupled Algebraic Multigrid for the Navier Stokes Equations", presented at 34th Aerospace Sciences Meeting & Exhibit, Reno, Nevada, January 15-18, 1996.

[RC1983] Rhie, C. M. and Chow, W. L., "Numerical Study of the Turbulent Flow Past an Airfoil with Trailing Edge Separation", AIAA Journal, 21(11), 1525-1532, 1983.

[RC1990] Raithby G. D., Chui E. H., "A finite volume method for predicting a radiant heat transfer in enclosures with participating media", Journal of Heat Transfer, vol 112, pp 415-423, May 1990.

[Roa1998] Roache, P. J., Verification and Validation in Computational Science and Engineering., Hermosa Publishers, New Mexico, 1998.

[Roa2001] Roache, P. J., private communication

[RST1983] Rosten, H. I., Spalding, D. B., and Tatchell, D. G., "PHOENICS: A General-Purpose Program for Fluid-Flow, Heat-Transfer and Chemical-Reaction Processes", in Proceedings of the 3rd International Conference on Engineering Software, 639-655, 1983.

[Rub1997] Rubini, P. A.,"SOFIE - Simulation of Fires in Enclosures", Proceedings of 5th International Symposium on Fire Safety Science, Melbourne, Australia, March 1997, International Association for Fire Safety Science, ISBN 4-9900625-5-5.

[RVD1993] Roose D., Van Driessche R., Distributed memory parallel computers and computational fluid dynamics. TW Report 186, Department of Computer Science, Katholieke Universiteit Leuven, Belgium, March 1993.

[Sar1986] Sarofim, A. F., "Radiative heat transfer in combustion: friend or foe", 21st Sym. (Int.) on Combustion, The Combustion Institute, pp1-23, 1986.

[Sat1985] Satoh, K., "Three-dimensional field model analysis of fire induced flow in an enclosure with a doorway opening – comparison with NBS fire tests. Report of Fire Research Institute of Japan, Vol 60, pp53-62, 1985.

[SBS1991] Smithies, J.N., Burry, P.E. and Spearpoint, M.J.; "Background Signals from Fire Detectors: Measurement, Analysis, Application."; *Fire Safety Journal*; Vol. 17(6), pp445-459, 1991.

[SBS+1995] Sterling, T., Becker, D., Savarese, D., et al. "BEOWULF: A Parallel workstation for Scientific Computation", Proceedings of the 1995 International Conference on Parallel Processing (ICPP), August 1995, Vol. 1, pp. 11-14.

[Sch1905] Schuster, A., `Radiation Through a Foggy Atmosphere', The Astrophysical Journal, 21(1), 1-22, 1905.

[Sin2000] Sinai Y L, "Exploratory CFD modelling of pool fire instabilities without cross-wind", Fire Safety Journal, Vol. 35 No. 1, pp 51-61, July 2000.

[Sin2003] Sinai, Y.L., "Field Modelling of a Steckler Experiment: An Example of the Relationship Between the Level of Modelling and Accuracy", presented Fourth International Seminar on Fire & Explosion Hazards, Londonderry, UK, 8-12 September 2003, University of Belfast, 2003.

[Sma1963] Smagorinsky, J., Mon. Weath. Rev. vol 91 part 3, pp99

[SFPE1995] "The SFPE Handbook of Fire Protection Engineering." 2nd edition., *NFPA*; 1995

[SO1995] Sinai, Y. L., Owens, M. P., "Validation of CFD modelling of unconfined pool fires with cross-wind: Flame geometry", Fire Safety Journal, 24, pp1-34, 1995.

[Spa] D. B. Spalding, `Proposal for a Diffusional Radiation Model', Unpublished Technical Memorandum, CHAM, London.

[Spa1983] Spalding, D. B., "Developments in the IPSA Procedure for Numerical Computation of Multiphase Flow Phenomena with Interphase Slip, Unequal Temperature etc". Numerical Properties and Methodologies in

Heat Transfer, Ed by T. M. Shih; Hemisphere Pub. Corp. pp421-436, 1983.

[SQR1982] Steckler, K.D., Quintiere, J.G. and Rinkinen, W.J.; "Flow induced by Fire in a Compartment."; *U.S. Dept. of Commerce*; NBSIR 82-2520, 1982.

[SRM1999] Sanderson, V., Rubini, P. A., and Moss J. B., "The effect of vent size of a compartment fire: Numerical simulation and validation", Proceedings 8th International Conference INTERFLAM'99. Interscience 1999.

[SSEW2001] Sinai, Y. L., Stopford, P. J., Edwards, M. and Watkins, S., "CFD Modelling of Fire Suppression by Water Spray : A Feasibility Study Examining a Pool Fire in a Simple Enclosure", Proc. Interflam 2001, pp1445-1451, Interscience 2001.

[Stu1997] Stuben, K., "Europort-D: parallel computing for European industry", IEEE Concurrency, 5(4): 7-10, October/December 1997.

[SWB+1997] Sullivan III, W. T., Werthimer, D., Bowyer, S., Cobb, J., Gedye, D., Anderson, D., "A new major SETI project based on Project Serendip data and 100,000 personal computers". Published in: "Astronomical and Biochemical Origins and the Search for Life in the Universe", Proc. of the Fifth Intl. Conf. on Bioastronomy. 1997

[SWJ1992] Simcox, S., Wilkes, N.S. and Jones, I.P.; "Computer Simulation of the Flows of Hot Gases from the Fire at Kings Cross Underground Station."; *Fire Safety Journal*; Vol. 18, pp 49-73, 1992.

[Tho1999] Thole, C. A., "DOWNPORT - Industrial Simulation on Workstation Clusters", Public Final Report of Espirit HPCN PST Activity, Rev 2.1, GMD-SCAI, May26th, 1999

[Tho2000] Thole, C. A., "DOWNPORT - Industrial Simulation on Workstation Clusters". In *SNA200 - The Fourth International Conference on Supercomputing in Nuclear Applications*. Japan Atomic Energy Research Institute, Tokyo, ISBN 4-9900652-0-4, 2000.

[TS1999] Tatum, K. E., Slater, J. W., "The Validation Archive of the NPARC Alliance", AIAA 37th Aerospace Sciences Meeting and Exhibit, AIAA-99-0747, 1999.
See also http://www.grc.nasa.gov/www/wind/valid/homepage.html.

[Tub1994] Tubbs, J. S., "Modelling the NIST high bay fire experiment with JASMINE", Master's Thesis, Worcester Polytechnic Institute, 1994.

[VR1984] Van Doormal, J. P. and Raithby, G. D., "Enhancements of the SIMPLE Method for Predicting Incompressible Fluid Flows", Numer. Heat Transfer, Vol 7, pp.147-163, 1984.

[WAD1997] Wighus, R., Aune, P., Drangsholt, G., "FIREDASS - Fire Detection and Suppression Simulation. Task 1.3.2. SINTEF Trials."; *FIREDASS report*; Proj. ref. no. 1.3.2, Jul. 1997.

[Wal1995] Walshaw C., "A parallelisable algorithm for optimising unstructured mesh partitions.", Technical Report P95/IM/03, School of Computing and Mathematical Science, January 1995.

[Was2001] Wasson, S http://www.tech-report.com/reviews/2001q3/pentium4-2ghz/index.x?pg=5

[Wat1986] Waters, R., "Air and Smoke Movement within a Large Enclosure", Numerical Simulation of Fluid Flow and Heat/Mass Transfer Processes, Ed. By Markatos, N. C., Tatchell, D. G., Cross, M. C., pp135-147, Springer Verlag, Berlin, 1986.

[Wig1998] Wighus, R., "An Empirical Model for Extinguishment of Enclosed Fires with Water Mist."; *Presented: HOTWC '98, Albuquerque, New Mexico, USA*; May 1998.

[WJG+2000] Wang, Z., Jia, F., Galea, E.R., Patel, M.K. and Ewer, J.; "Simulating One of the CIB W14 Round Robin Test Cases Using the SMARTFIRE Fire Field Models."; *CMS Press*; Paper no. 00/IM/53, ISBN 1 899991 54 9, 2000.

[YCL1995] Yeoh, G.H., Chandrasekaran, V. and Leonardi, E.; "Numerical Prediction of Fire and Smoke."; *AIRAH Journal*; Vol. 49, pp13-18, 1995.

[YH1996] Yan, Z. and Holmstedt, G.; "CFD and Experimental Studies of Room Fire Growth on Wall Lining Materials."; *Fire Safety J.*; Vol. 27, pp201-238, 1996.

[YH2001] Yan, Z. and Holmstadt, G., "Investigation of the dance hall fire in Gothenburg, October 1998 – A Comparison between human observations and CFD simulation", Proc. Interflam 2001, pp951-963, Interscience 2001.

Appendix A - MPI Commands

MPI is an API for parallel processing. This appendix is a subset of all MPI commands and is the full set of MPI commands utilised in parallel SMARTFIRE.

MPI_Bcast

Broadcasts a message from the process with rank "root" to all other processes of the group.

int MPI_Bcast (void *buffer, int count, MPI_Datatype datatype, int root,
MPI_Comm comm)
Input/output Parameters

buffer	starting address of buffer (choice)
count	number of entries in buffer (integer)
datatype	data type of buffer (handle)
root	rank of broadcast root (integer)
comm	communicator (handle)

MPI_Barrier

Blocks until all process have reached this routine.

int MPI_Barrier (
MPI_Comm comm)
Input Parameters
comm
communicator (handle)

MPI_Finalize

Terminates MPI execution environment

int MPI_Finalize()

MPI_Init

Initialize the MPI execution environment

int MPI_Init(int *argc, char ***argv)
Input Parameters

argc	Pointer to the number of arguments

MPI_Allgather

Gathers data from all tasks and distribute it to all

int MPI_Allgather (void *sendbuf, int sendcount, MPI_Datatype sendtype,
void *recvbuf, int recvcount, MPI_Datatype recvtype,
MPI_Comm comm)
Input Parameters
sendbuf starting address of send buffer (choice)
sendcount number of elements in send buffer (integer)
sendtype data type of send buffer elements (handle)
recvcount number of elements received from any process (integer)
recvtype data type of receive buffer elements (handle)
comm communicator (handle)
Output Parameter
recvbuf
address of receive buffer (choice)

MPI_Irecv

Begins a nonblocking receive

int MPI_Irecv(void *buf, int count, MPI_Datatype datatype, int source,
int tag, MPI_Comm comm, MPI_Request *request)
Input Parameters
buf initial address of receive buffer (choice)
count number of elements in receive buffer (integer)
datatype datatype of each receive buffer element (handle)
source rank of source (integer)
tag message tag (integer)
comm communicator (handle)
Output Parameter
request
communication request (handle)

MPI_Isend

Begins a nonblocking send

int MPI_Isend(void *buf, int count, MPI_Datatype datatype, int dest, int tag,
MPI_Comm comm, MPI_Request *request)
Input Parameters
buf initial address of send buffer (choice)
count number of elements in send buffer (integer)
datatype datatype of each send buffer element (handle)
dest rank of destination (integer)
tag message tag (integer)
comm communicator (handle)
Output Parameter
request

communication request (handle)

MPI_Recv

Basic receive

int MPI_Recv(void *buf, int count, MPI_Datatype datatype, int source,
int tag, MPI_Comm comm, MPI_Status *status)
Output Parameters
buf initial address of receive buffer (choice)
status status object (Status)
Input Parameters
count maximum number of elements in receive buffer (integer)
datatype datatype of each receive buffer element (handle)
source rank of source (integer)
tag message tag (integer)
comm communicator (handle)

MPI_Send

Performs a basic send

int MPI_Send(void *buf, int count, MPI_Datatype datatype, int dest,
int tag, MPI_Comm comm)
Input Parameters
buf initial address of send buffer (choice)
count number of elements in send buffer (nonnegative integer)
datatype datatype of each send buffer element (handle)
dest rank of destination (integer)
tag message tag (integer)
comm communicator (handle)

MPI_Startall

Starts a collection of requests

int MPI_Startall(int count, MPI_Request array_of_requests[])
Input Parameters
count list length (integer)
array_of_requests array of requests (array of handle)

MPI_Waitall

Waits for all given communications to complete

```
int MPI_Waitall(
      int count,
      MPI_Request array_of_requests[],
      MPI_Status array_of_statuses[] )
```

Input Parameters

count	lists length (integer)
array_of_requests	array of requests (array of handles)

Output Parameter

array_of_statuses

array of status objects (array of Status). May be `MPI_STATUSES_NULL`

MPI_Recv_init

Builds a handle for a receive

```
int MPI_Recv_init( void *buf, int count, MPI_Datatype datatype, int source,
            int tag, MPI_Comm comm, MPI_Request *request )
```

Input Parameters

buf	initial address of receive buffer (choice)
count	number of elements received (integer)
datatype	type of each element (handle)
source	rank of source or `MPI_ANY_SOURCE` (integer)
tag	message tag or `MPI_ANY_TAG` (integer)
comm	communicator (handle)

Output Parameter

request

communication request (handle)

MPI_Send_init

Builds a handle for a standard send

```
int MPI_Send_init( void *buf, int count, MPI_Datatype datatype, int dest,
            int tag, MPI_Comm comm, MPI_Request *request )
```

Input Parameters

buf	initial address of send buffer (choice)
count	number of elements sent (integer)
datatype	type of each element (handle)
dest	rank of destination (integer)
tag	message tag (integer)
comm	communicator (handle) Output Parameter:

request

communication request (handle)

Appendix B – CFD Problem specification sheets

B.1 - Test case : Two-dimensional turbulent flow over a backward facing step - 2000/1/1

Fire Safety Engineering Group
Maritime Greenwich Campus,
Cooper Building,
University of Greenwich,
King William Walk,
London SE10 9JH, UK.

Date : 4/2/2000

PART 1 – CONTROLLED TEST SETUP

Test case : **Two-dimensional turbulent flow over a backward facing step - 2000/1/1**

Document Version 1.0

PART 1 – CONTROLLED TEST SETUP

Case: **Two-dimensional turbulent flow over a backward facing step - 2000/1/1**

User details	
Run by: Date: Phone no: email:	Address:

Fire modelling Software

SMARTFIRE	CFX	PHOENICS			

Version/build number_____
Date of release_____

Operating System

Windows 95/98/2000	Windows NT	Unix	Dos	

Version/build number_____

Machine

PC	Unix Workstation	

CPU:
Memory:

Case description

This test examines the CFD fire modelling software's turbulence model. The flow is incompressible, fully turbulent and isothermal. The fluid has a density of 1.0 kg/m^3 and a laminar viscosity of 1.101E-5 kg/ms. The geometry of the case is illustrated in Figure 0-1.

The upper and lower surfaces are walls and there is a solid obstruction below the inlet. The fluid enters the chamber at 13.0 m/s.

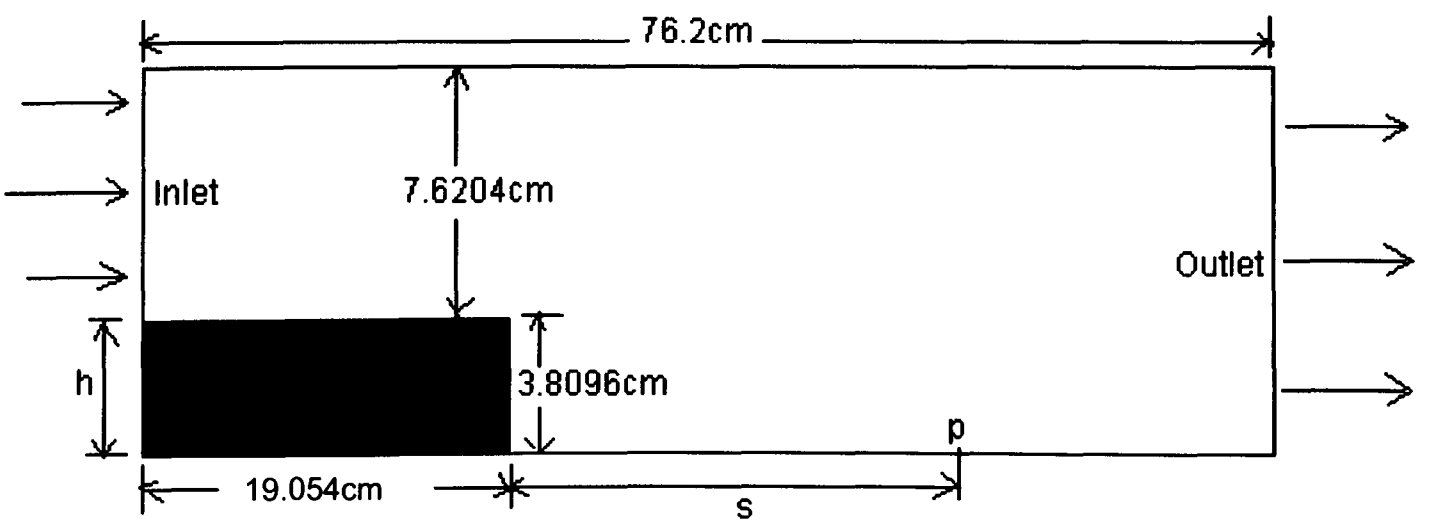


Figure 0-1 - Backward facing step configuration

Required Results

The results should be supplied as graphs and as Excel97 worksheets

This case will be compared to experimental results[1,2] and across the codes

The reattachment point is the downstream location in the x direction where there is no longer any flow re-circulation due to the backward facing step.

In Figure 0-1 the reattachment point is denoted by P and the distance from the step to point P is s. The ratio of s to the height of the step needs to be provided.

Graphs

A u-velocity component profile should be provided at the outlet (the last cell centre value) and also 0.285m downstream from the inlet.

A velocity vector field plot must be provided.

1) J. Kim, S. J. Kline and J. P. Johnston, "Investigation of a Reattachment Turbulent Shear Layer: Flow over a Backward-Facing Step", Transactions of the ASME, Journal of Fluids Engineering, 102, 302-308, 1980.

2) J. K. Eaton and J. P. Johnston, `A Review of Research on Subsonic Turbulent Flow Reattachment', AIAA, Paper AIAA-80-1438, 1980.

CFD set up

1D	2D	3D

Transient	Steady State

Differencing Schemes

Temporal:

Fully Implicit	Crank-Nicolson	Explicit	Exponential	

Spatial:

Hybrid	Central Difference	Upwind		

Notes:

Physical Models

Radiation Model *(if not listed please specify in the space provided)*

None	Six flux	Discrete Transfer	Monte Carlo	Radiosity	***10.7.1.1***

Notes:

Parameters

Turbulence model *(if not listed please specify in the space provided)*

Laminar	k- ε	Buoyancy modified k-ε	RNG	

Notes:

Turbulence Parameters*:

C_μ	σ_k	σ_ε	$C_{1\varepsilon}$	$C_{2\varepsilon}$	C_3
0.09	1.0	1.3	1.44	1.92	1.0

*If different parameters are being used please specify in the table above.

Combustion Model *(if not listed please specify in the space provided)*

none	Volumetric heat source	Mixed is burnt	Eddy break up
Magnussen soot model			

Combustion Parameters:

Compressibility

Incompressible	Boussinesq	Weakly compressible	Fully compressible

Compressibility Parameters:

External Pressure 1.01325e+05

Buoyancy

Yes	No

Gravity	0.0 m/s

Material Properties

Material Name	Air
Density	1 kg/m^3
Viscosity	1.101E-5 kg/ms
Conductivity	0.0
Specific heat capacity	0.0

Initial Values

U-VELOCITY	0.0
V-VELOCITY	0.0
W-VELOCITY	0.0
PRESSURE	0.0
TEMPERATURE	293.75
KINETIC ENERGY	0.01
DISSIPATION RATE	0.01

Boundary conditions

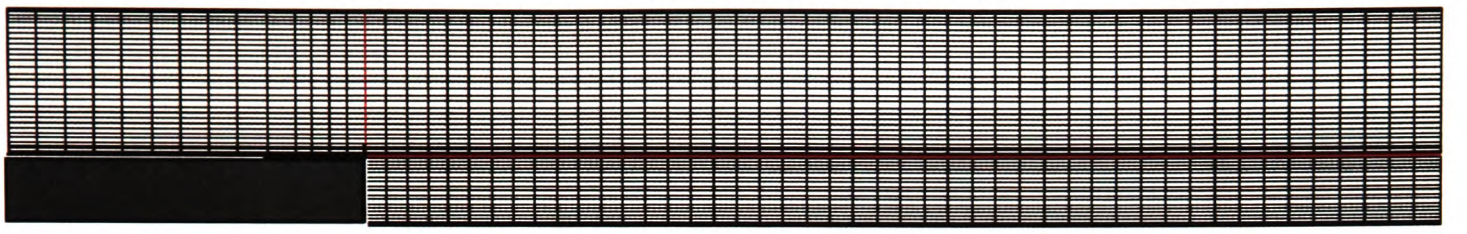
Inlet
Velocity: 13.0 m/s,
Kinetic energy: 0.7605 m^2/s^2,
Dissipation rate: 31.78 m^2/s^3.

Outlet
Pressure 0.0 Pa

Stationary Walls

Turbulent wall functions

Mesh



60 50 1

X

```
0.0 0.01524 0.03048 0.04572 0.06096 0.0762 0.09144 0.10668
0.12192 0.13716 0.1524 0.160645 0.170073 0.180007 0.19054
0.200822 0.211576 0.222511 0.233604 0.244837 0.256195
0.267668 0.279246 0.290921 0.302686 0.314536 0.326464
0.338468 0.350543 0.362684 0.37489 0.387157 0.399483
0.411865 0.424301 0.436788 0.449326 0.461912 0.474545
0.487223 0.499945 0.51271 0.525515 0.538361 0.551246
0.564168 0.577128 0.590123 0.603154 0.61622 0.629318
0.64245 0.655613 0.668808 0.682034 0.69529 0.708575
0.721889 0.735232 0.748602 0.762
```

Y

```
0.0 0.000955 0.002351 0.003982 0.005789 0.007737 0.009806
0.011982 0.014253 0.016612 0.01905 0.021488 0.023847
0.026118 0.028294 0.030363 0.032311 0.034118 0.035749
0.037145 0.0381 0.03896 0.040369 0.042103 0.044088 0.046284
0.048664 0.051208 0.053902 0.056735 0.059697 0.06278
0.065977 0.069283 0.072692 0.0762 0.079708 0.083117
0.086423 0.08962 0.092703 0.095665 0.098498 0.101192
0.103736 0.106116 0.108312 0.110297 0.112031 0.11344 0.1143
```

Z

```
0.0 1.0
```

Model Definition files

Convergence

Please specify your convergence criteria including type of error estimator and tolerance value for each variable

Runtime

Results files/Archiving:

Document cross-reference:

User Guides, etc

Comments

B.2 - Test case : Turbulent long duct flow – 2000/1/2

Group

Fire Safety Engineering

Maritime Greenwich Campus,
Cooper Building,
University of Greenwich,
King William Walk,
London SE10 9JH, UK.

Date : 4/2/2000

PART 1 – CONTROLLED TEST SETUP

Test case : **Turbulent long duct flow – 2000/1/2**
Document Version 1.0

PART 1 – CONTROLLED TEST SETUP

Case: **Turbulent long duct flow – 2000/1/2**

User details	
Run by: Date: Phone no: email:	Address:

Fire modelling Software

SMARTFIRE	CFX	PHOENICS			

Version/build number_____
Date of release_____

Operating System

Windows 95/98/2000	Windows NT	Unix	Dos	

Version/build number_____

Machine

PC	Unix Workstation	

CPU:
Memory:

Case description

This test case examines the CFD fire modelling software's κ-ε turbulence model in conjunction with turbulent heat transfer. This case has been well investigated numerically. The geometry of the case is depicted Figure 0-2. The flow is non buoyant, fully turbulent, incompressible with heat transfer but no radiation. Flow enters the inlet at 50m/s with an enthalpy of 50 J/Kg. The wall has a fixed enthalpy value of 1 J/Kg. The fluid density is 1.0 kg/m^3, the conductivity is 0.07179 W/mK, the density is 1.0 kg/m^3, laminar viscosity is 5e-5 kg/ms, specific heat is 1005 J/kgK

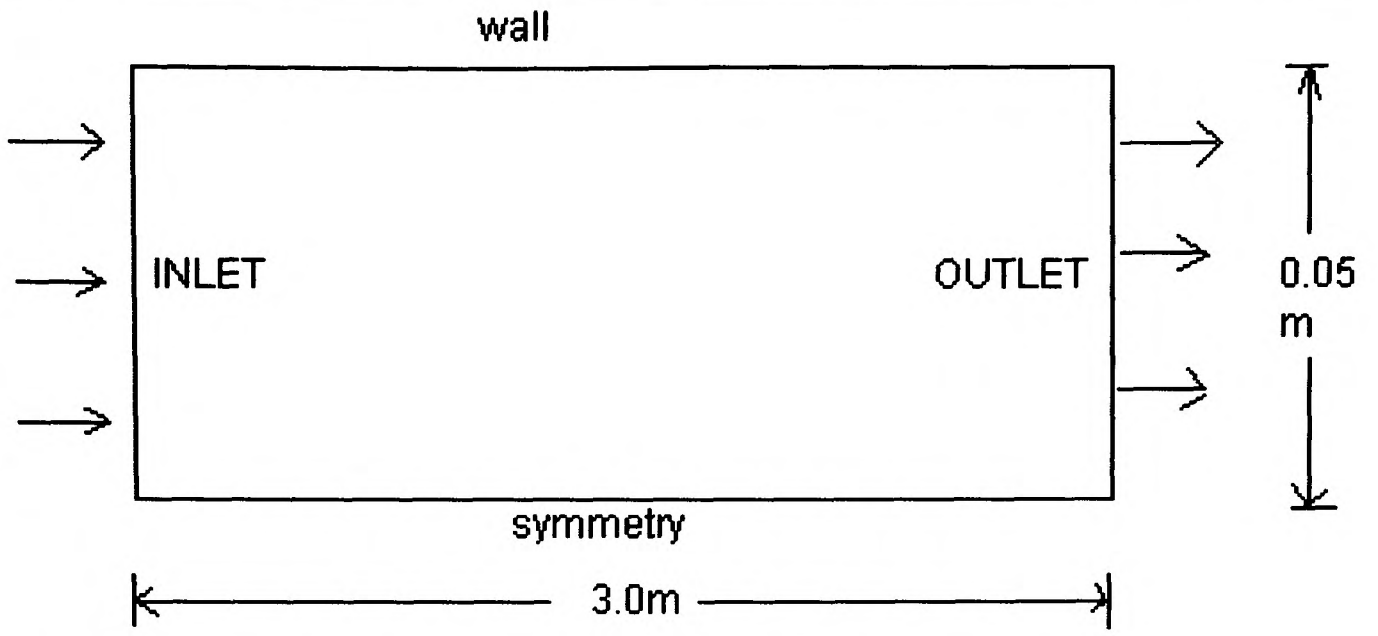


Figure 0-2- Turbulent long duct flow configuration

Required Results
The results should be supplied as graphs and as Excel97 worksheets
This case is used to compare between codes.
An enthalpy and u-velocity profile at the outlet need to be provided.

CFD set up

1D	**2D**	3D

Transient	**Steady State**

Differencing Schemes

Temporal:

Fully Implicit	Crank-Nicolson	Explicit	Exponential	

Spatial:

Hybrid	Central Difference	Upwind		

Notes:

Physical Models

Radiation Model (if not listed please specify in the space provided)

None	Six flux	Discrete Transfer	Monte Carlo	Radiosity	

Notes:

Parameters

Turbulence model *(if not listed please specify in the space provided)*

Laminar	k- ε	buoyancy modified k-ε	RNG	

Notes:

Turbulence Parameters*:

C_μ	σ_k	σ_ε	$C_{1\varepsilon}$	$C_{2\varepsilon}$	C_3
0.09	1.0	1.3	1.44	1.92	1.0

*If different parameters are being used please specify in the table above.

Combustion Model *(if not listed please specify in the space provided)*

none	Volumetric heat source	Mixed is burnt	Eddy break up
Magnussen soot model			

Combustion Parameters:

Compressibility

Incompressible	Boussinesq	Weakly compressible	Fully compressible

Compressibility Parameters:

Buoyancy

Yes	No

Gravity	0.0m/s

Material Properties

Material Name	Air
Density	1.0
Laminar Viscosity	5e-05
Conductivity	0.07179
Specific heat capacity	1005.0

Initial Values

U-VELOCITY	50.0
V-VELOCITY	0.0
W-VELOCITY	0.0
PRESSURE	0.0
TEMPERATURE	293.75
KINETIC ENERGY	11.25
DISSIPATION RATE	1387.0

Boundary conditions

Inlet

Velocity : 50 m/s
Turbulent kinetic energy: 11.25 (m^2/s^2)
Dissipation rate: 1378.0 (m^2/s^3)
Enthalpy: 10 (J/kg)

Wall

Fixed enthalpy value :(1 J/kg).
Standard turbulent wall functions on the wall

Outlet

Pressure : 0.0 Pa

Mesh

Mesh data

The mesh is non-uniformly distributed and the cell budget is 600(20×30x1).

X
0.0 0.15 0.3 0.45 0.6 0.75 0.9 1.05 1.2 1.35 1.5 1.65 1.8 1.95 2.1 2.25 2.4 2.55 2.7 2.85 3.0
Y
0.0 0.003901 0.00656 0.008891 0.011033 0.013042 0.014953 0.016786 0.018554 0.020268 0.021935 0.02356 0.025149 0.026705 0.028231 0.02973 0.031205 0.032656 0.034087 0.035497 0.036889 0.038264 0.039623 0.040966 0.042295 0.04361 0.044912 0.046201 0.047479 0.048745 0.05
Z
0.0 1.0

Model Definition files

Convergence

Please specify your convergence criteria including type of error estimator and tolerance value for each variable

Runtime

Results files/Archiving:

Document cross-reference:

User Guides, etc

Comments

B.3 - Test case : Symmetry boundary condition test – 2000/1/3

Fire Safety Engineering Group
Maritime Greenwich Campus,
Cooper Building,
University of Greenwich,
King William Walk,
London SE10 9JH, UK.

Date : 4/2/2000

PART 1 – CONTROLLED TEST SETUP

Test case : **Symmetry boundary condition test – 2000/1/3**
Document Version 1.0

PART 1 – CONTROLLED TEST SETUP

Case: **Symmetry boundary condition test – 2000/1/3**

User details	
Run by: Date: Phone no: email:	Address:

Fire modelling Software

SMARTFIRE	CFX	PHOENICS			

Version/build number_____
Date of release_____

Operating System

Windows 95/98/2000	Windows NT	Unix	Dos	

Version/build number_____

Machine

PC	Unix Workstation	

CPU:
Memory:

Case description

This case is intended to test if the symmetry function works correctly for turbulent isothermal flow. The case involves flow expansion from a small duct into a larger duct. The configuration is shown in Figure 0-3 below. The case was simulated using the whole flow domain and then repeated using a symmetry boundary condition along the central axis. Two tests must be conducted using the full domain and using a half domain with a symmetry plane. The results from these two tests should agree with one another. The flow enters the domain at 1.0m/s.

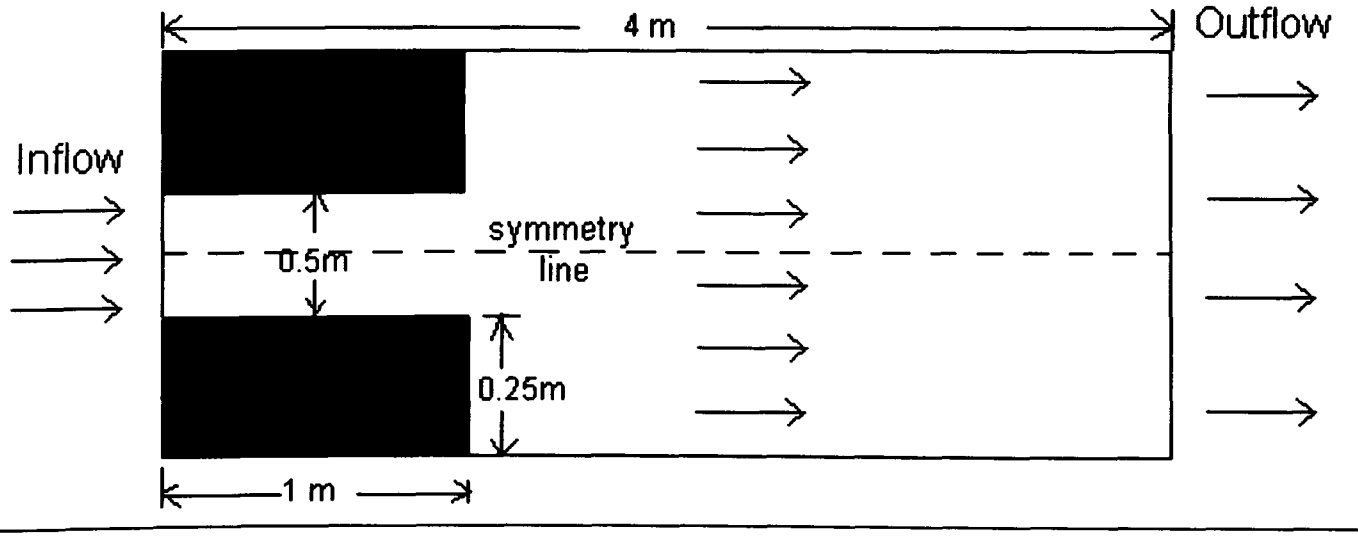


Figure 0-3 - Expanding duct with symmetry line indicated

Required Results

The results should be supplied as graphs and as Excel97 worksheets

This case is specifically designed for testing the symmetry treatment of each case. A whole field velocity vector plot should be supplied for both cases along with a u-velocity profile at the outlet for both cases.

CFD set up

1D	2D	3D

Transient	Steady State

Differencing Schemes

Temporal:

Fully Implicit	Crank-Nicolson	Explicit	Exponential	

Spatial:

Hybrid	Central Difference	Upwind		

Notes:

Physical Models

Radiation Model *(if not listed please specify in the space provided)*

None	Six flux	Discrete Transfer	Monte Carlo	Radiosity	

Notes:

Parameters

Turbulence model *(if not listed please specify in the space provided)*

Laminar	k- ε	buoyancy modified k-ε	RNG	

Notes:

Turbulence Parameters*:

C_μ	σ_k	σ_ε	$C_{1\varepsilon}$	$C_{2\varepsilon}$	C_3
0.09	1.0	1.3	1.44	1.92	1.0

*If different parameters are being used please specify in the table above.

Combustion Model *(if not listed please specify in the space provided)*

none	Volumetric heat source	Mixed is burnt	Eddy break up
Magnussen soot model			

Combustion Parameters:

Compressibility

Incompressible	Boussinesq	Weakly compressible	Fully compressible

Compressibility Parameters:

External Pressure 1.01325e+05 Pa

Buoyancy

Yes	No

Gravity	0.0m/s

Material Properties

Material Name	Air
Density	Determined by compressibility (Ideal Gas Law) Molecular Weight of air is 29.35
Viscosity	Laminar 1.798e-005kg/m.s + Value determined from turbulence model
Conductivity	0.02622 W/m.K
Specific heat capacity	1007.0 J/kg.K

Initial Values

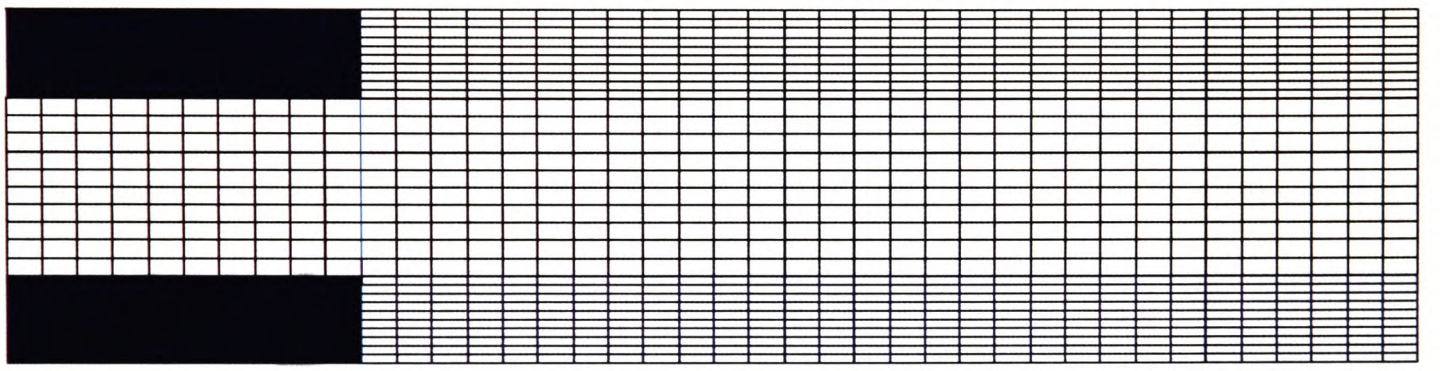
U-VELOCITY	0.0
V-VELOCITY	0.0
W-VELOCITY	0.0
PRESSURE	0.0
TEMPERATURE	n/a
KINETIC ENERGY	0.01
DISSIPATION RATE	0.01

Boundary conditions

Inlet
U-velocity:1.0 m/s
Dissipation rate:0.01
Kinetic energy:0.01

Outlet: (Pressure boundary) 0.0 Pa

Mesh



Half

40 15 1
X
0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1 1.1 1.2 1.3 1.4
1.5 1.6 1.7 1.8 1.9 2 2.1 2.2 2.3 2.4 2.5 2.6 2.7 2.8 2.9 3
3.1 3.2 3.3 3.4 3.5 3.6 3.7 3.8 3.9 4.0
Y
0.0 0.05 0.1 0.15 0.2 0.25 0.275 0.3 0.325 0.35 0.375 0.4
0.425 0.45 0.475 0.5
Z
0.0 1.0

Whole

40 30 1

X
0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1 1.1 1.2 1.3 1.4
1.5 1.6 1.7 1.8 1.9 2 2.1 2.2 2.3 2.4 2.5 2.6 2.7 2.8 2.9 3
3.1 3.2 3.3 3.4 3.5 3.6 3.7 3.8 3.9 4.0
Y
0.0 0.025 0.05 0.075 0.1 0.125 0.15 0.175 0.2 0.225 0.25
0.3 0.35 0.4 0.45 0.5 0.55 0.6 0.65 0.7 0.75 0.775 0.8
0.825 0.85 0.875 0.9 0.925 0.95 0.975 1
Z
0.0 1.0

Model Definition files

Convergence

Please specify your convergence criteria including type of error estimator and tolerance value for each variable

Runtime

Results files/Archiving:

Document cross-reference:

User Guides, etc

Comments

B.4 - Test case : Turbulent buoyancy flow in a cavity – 2000/1/4

Fire Safety Engineering Group
Maritime Greenwich Campus,
Cooper Building,
University of Greenwich,
King William Walk,
London SE10 9JH, UK.

Date : 15/6/2000

PART 1 – CONTROLLED TEST SETUP

Test case : **Turbulent buoyancy flow in a cavity – 2000/1/4**
Document Version 1.1

PART 1 – CONTROLLED TEST SETUP

Case: **Turbulent buoyancy flow in a cavity – 2000/1/4**

User details	
Run by: Date: Phone no: email:	Address:

Fire modelling Software

SMARTFIRE	CFX	PHOENICS			

Version/build number_____
Date of release_____

Operating System

Windows 95/98/2000	Windows NT	Unix	Dos	

Version/build number_____

Machine

PC	Unix Workstation	

CPU:
Memory:

Case description

This test case examines the turbulence model, turbulent heat transfer and buoyancy model of a CFD fire modelling code. The test case is a standard test case that has been used by a number of other investigators*.

The geometry used for this case is depicted in Figure **0-4** below.

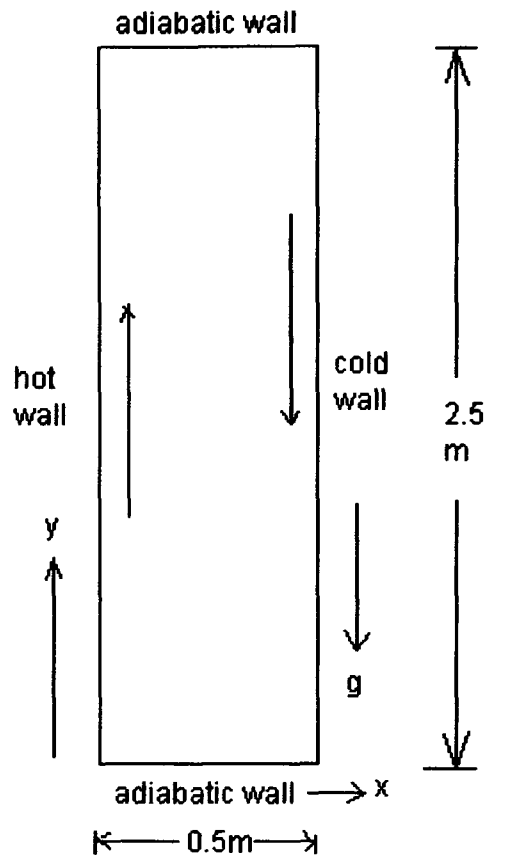


Figure 0-4 – Configuration for buoyancy flow in a duct

The flow is fully turbulent, buoyant and fully compressible but with no radiation heat transfer. The hot wall is at a temperature of 353K and the cold wall is at 307.2K. The other walls are adiabatic. The acceleration due to gravity (g) is –9.81m/s

*L. Davidson, "Calculation of the turbulent buoyancy-driven flow in a rectangular cavity using an efficient solver and two different low Reynolds number κ-ε turbulence models", Numerical Heat Transfer, vol. 18, pp. 129-147, 1990.

Fluid properties

conductivity is 2.852158e-02 (W/mK)
density is 1.071 (kg/m^3) determined by ideal gas law as fully compressible.
specific heat is 1.008e+03 (J/kgK)
laminar viscosity is 2.0383e-05 (kg/ms)
thermal expansion is 3.029385e-03 (1/K).

Required Results
The results should be supplied as graphs and as Excel97 worksheets

These results will be compared against experiment and across codes.

The v-velocity profile at y/H = 0.5
The normalised temperature profile at y/H = 0.5 and x/L = 0.5
where $T_{normalised} = (T_{actual} - T_{cold})/(T_{hot} - T_{cold})$
The turbulent fluctuations, $\sqrt{k}$, at y/H = 0.5
The turbulent viscosity scaled with the laminar viscosity at y/H = 0.5.

where L is full length across the x direction of the duct (0.5m) and H is the full height of

the duct in the y direction (2.5m).

CFD set up

1D	**2D**	3D

Transient	**Steady State**

Differencing Schemes

Temporal:

Fully Implicit	Crank-Nicolson	Explicit	Exponential	

Spatial:

Hybrid	Central Difference	Upwind		

Notes:

Physical Models

Radiation Model *(if not listed please specify in the space provided)*

None	Six flux	Discrete Transfer	Monte Carlo	Radiosity	

Notes:

Parameters

Turbulence model *(if not listed please specify in the space provided)*

Laminar	k- ε	**buoyancy modified k-ε**	RNG	

Notes:

Turbulence Parameters*:

C_μ	σ_k	σ_ε	$C_{1\varepsilon}$	$C_{2\varepsilon}$	C_3
0.09	1.0	1.3	1.44	1.92	1.0

*If different parameters are being used please specify in the table above.

Combustion Model *(if not listed please specify in the space provided)*

none	Volumetric heat source	Mixed is burnt	Eddy break up
Magnussen soot model			

Combustion Parameters:

Compressibility

Incompressible	Boussinesq	Weakly compressible	Fully compressible

Compressibility Parameters:

External Pressure 1.01325e+05 Pa

Buoyancy

Yes	No

Material Properties

Material Name	Air
Density	Determined by compressibility (Ideal Gas Law) Molecular Weight of air is 29.35
Laminar Viscosity	2.0383e-05 kg/m.s
Conductivity	2.852158e-02 W/m.K
Specific heat capacity	1008.0 J/kg.K

Initial Values

U-VELOCITY	1.0e-6
V-VELOCITY	1.0e-6
W-VELOCITY	0.0
PRESSURE	0.0
TEMPERATURE	307.2
KINETIC ENERGY	0.0034
DISSIPATION RATE	0.001369

Boundary conditions
hot wall (t_h): constant temperature (353.0 K)
cold wall(t_c): constant 307.2 (K).
The other walls are adiabatic.

Mesh

The cell budget is 14641(121×121x1) with non-uniformly distributed mesh. There is no mesh diagram due to the fineness of the mesh.

```
X 0.0 0.001819 0.004179 0.006797 0.0096 0.012548 0.015616
0.018789 0.022055 0.025403 0.028827 0.03232 0.035877
0.039494 0.043167 0.046892 0.050668 0.054492 0.058361
0.062273 0.066226 0.070219 0.074251 0.078319 0.082423
0.086561 0.090732 0.094936 0.099171 0.103436 0.107731
0.112054 0.116406 0.120784 0.12519 0.129621 0.134078
0.138559 0.143065 0.147595 0.152148 0.156724 0.161322
0.165942 0.170583 0.175246 0.17993 0.184634 0.189358
0.194102 0.198865 0.203647 0.208448 0.213268 0.218105
0.222961 0.227834 0.232725 0.237633 0.242558 0.2475 0.2525
0.257442 0.262367 0.267275 0.272166 0.277039 0.281895
0.286732 0.291552 0.296353 0.301135 0.305898 0.310642
0.315366 0.32007 0.324754 0.329417 0.334058 0.338678
0.343276 0.347852 0.352405 0.356935 0.361441 0.365922
0.370379 0.37481 0.379216 0.383594 0.387946 0.392269
0.396564 0.400829 0.405064 0.409268 0.413439 0.417577
0.421681 0.425749 0.429781 0.433774 0.437727 0.441639
0.445508 0.449332 0.453108 0.456833 0.460506 0.464123
0.46768 0.471173 0.474597 0.477945 0.481211 0.484384
0.487453 0.4904 0.493203 0.495821 0.498181 0.5
Y 0.0 0.002657 0.007516 0.013808 0.021258 0.029709 0.039054
0.049214 0.060128 0.071747 0.084031 0.096946 0.110462
0.124553 0.139198 0.154375 0.170067 0.186257 0.202931
0.220075 0.237676 0.255723 0.274204 0.293111 0.312433
0.332162 0.35229 0.372808 0.393711 0.414989 0.436638
0.458651 0.481022 0.503745 0.526815 0.550227 0.573976
0.598057 0.622466 0.647198 0.672249 0.697615 0.723293
0.749278 0.775567 0.802156 0.829043 0.856223 0.883694
0.911453 0.939496 0.967822 0.996426 1.025307 1.054461
1.083887 1.113582 1.143543 1.173767 1.204254 1.235 1.265
1.295746 1.326233 1.356457 1.386418 1.416113 1.445539
1.474693 1.503574 1.532178 1.560504 1.588547 1.616306
1.643777 1.670957 1.697844 1.724433 1.750722 1.776707
1.802385 1.827751 1.852802 1.877534 1.901943 1.926024
1.949773 1.973185 1.996255 2.018978 2.041349 2.063362
2.085011 2.106289 2.127192 2.14771 2.167838 2.187567
2.206889 2.225796 2.244277 2.262324 2.279925 2.297069
2.313743 2.329933 2.345625 2.360802 2.375447 2.389538
2.403054 2.415969 2.428253 2.439872 2.450786 2.460946
2.470291 2.478742 2.486192 2.492484 2.497343 2.5
Z 0.0 1.0
```

Model Definition files

Convergence

Please specify your convergence criteria including type of error estimator and tolerance value for each variable

Runtime

Results files/Archiving:

Document cross-reference:

User Guides, etc

Comments

B.5 - Test case : Radiation in 3 dimensional cavity 2000/1/5

Fire Safety Engineering Group
Maritime Greenwich Campus,
Cooper Building,
University of Greenwich,
King William Walk,
London SE10 9JH, UK.

Date : 14/2/2000

PART 1 – CONTROLLED TEST SETUP

Test case : **Radiation in 3 dimensional cavity 2000/1/5**
Document Version 1.1

PART 1 – CONTROLLED TEST SETUP

Case: **Radiation in 3 dimensional cavity 2000/1/5**

User details	
Run by: Date: Phone no: email:	Address:

Fire modelling Software *(if not listed please specify in the space provided)*

SMARTFIRE	CFX	PHOENICS			

Version/build number_____
Date of release_____

Operating System *(if not listed please specify in the space provided)*

Windows 95/98/2000	Windows NT	Unix	Dos	

Version/build number_____

Machine *(if not listed please specify in the space provided)*

PC	Unix Workstation	

CPU:
Memory:

Case description

This test case tests the fire modelling software's radiation model. The cavity is a unit cube (1m x 1m x 1m) with three walls with planes x=1, y=0 and z =0 set to an unit emissive power and the three other walls set to zero emissive power. All the walls are considered radiatively black have unit emissivity and the fluid has a unit absorption coefficient. Scattering is neglected. No fluid flow is considered

Required Results

The results should be supplied as graphs and as Excel97 worksheets

The results will be compared with analytically derived results[1,2] and between the codes.

The emissive power variation with x must be provided for the following locations y= 0.1, 0.3, 0.5, 0.7, 0.9 on the planes z = 0.5, 0.9.

1) Larsen, M. E., "Exchange Factor Method and Alternative Zonal Formulation for Analysis of radiating enclosures containing paticipating media", PhD thesis, University of Texas, Austin, 1983.

2) Fiveland, W. A., "Three dimensional Radiative Heat-Transfer Solutions by Discrete-Ordinate Method", Journal of Thermophysics, Vol. 2, No. 4, October 1988, pp 309-316.

CFD set up

1D	2D	**3D**

Transient	**Steady State**

Differencing Schemes

Temporal:

Fully Implicit	Crank-Nicolson	Explicit	Exponential	

Spatial:

Hybrid	Central Difference	Upwind		

Notes:

Physical Models

Radiation Model *(if not listed please specify in the space provided)*

None	**Six flux**	Discrete Transfer	Monte Carlo	Radiosity	

Notes:

(1) If the fire modelling software does not possess the six-flux model, a discrete transfer model may be used in place of the six-flux model. If the discrete transfer model must be used instead of a six flux model then the discrete model must be made to emulate the behaviour of the six-flux model. This can be achieved by using 6 rays in the co-ordinate directions. If a radiation mesh needs to be specified, this should be identical to the flow mesh. If this is not possible, then at least the same number of cells in each direction must be specified. The details of the mesh must also be provided with your results.

Parameters

$a = 1.0$

It is assumed there is no scattering so s = 0.0.

Turbulence model *(if not listed please specify in the space provided)*

Laminar	k- ε	buoyancy modified k-ε	RNG	

Notes:

Turbulence Parameters*:

C_μ	σ_k	σ_ε	$C_{1\varepsilon}$	$C_{2\varepsilon}$	C_3
0.09	1.0	1.3	1.44	1.92	1.0

*If different parameters are being used please specify in the table above.

Combustion Model *(if not listed please specify in the space provided)*

none	Volumetric heat source	Mixed is burnt	Eddy break up
Magnussen soot model			

Combustion Parameters:

Compressibility

Incompressible	Boussinesq	Weakly compressible	Fully compressible

Compressibility Parameters:

External Pressure 1.01325e+05

Buoyancy

Yes	No

Gravity	

Material Properties

Material Name	Optical Fluid
Density	N/a
Viscosity	N/a
Conductivity	1E-100
Specific heat capacity	0.1

Initial Values

TEMPERATURE	0K

Boundary conditions

All the walls have an emissivity of 1.0

The walls at x=1, y = 0 and z = 0 have a temperature of 64.8052186K, all the other walls have a temperature of 0K.

Mesh

11 11 11
A uniformly meshed cube.
X, Y and Z
0.000000 0.090909 0.181818 0.272727 0.363636 0.454545 0.545455 0.636364
0.727273 0.818182 0.909091 1.000000

Model Definition files

Convergence

Please specify your convergence criteria including type of error estimator and tolerance value for each variable

Runtime

Results files/Archiving:

Document cross-reference:

User Guides, etc

Comments

Appendix C - Fire cases specification sheets

C.1 - Steckler Room Fire – volumetric heat – 2000/2/1

Fire Safety Engineering Group
Maritime Greenwich Campus,
Cooper Building,
University of Greenwich,
King William Walk,
London SE10 9JH, UK.

Date : 4/2/2000

PART 1 – CONTROLLED TEST SETUP

Test case : **Steckler Room Fire – volumetric heat – 2000/2/1**
Document Version 1.0

PART 1 – CONTROLLED TEST SETUP

Case: **Steckler Room Fire – volumetric heat – 2000/2/1**

User details	
Run by: Date: Phone no: email:	Address:

Fire modelling Software

SMARTFIRE	CFX	PHOENICS			

Version/build number_____
Date of release_____

Operating System

Windows 95/98/2000	Windows NT	Unix	Dos	

Version/build number_____

Machine

PC	Unix Workstation	

CPU:
Memory:

Case description

The experimental data obtained from Steckler's fire tests* which have been used as part of the validation process for both zone and field fire models. The data represents non-spreading fires in small compartments. The non-spreading fire was created using a centrally located (position A in Figure 0-1) 62.9kW methane burner with a diameter of 0.3m and a height of 0.3m. The experiments were conducted by Steckler et al. in a compartment measuring 2.8m × 2.8m in plane and 2.18m in height (see Figure 0-1) with a doorway centrally located in one of the walls measuring 0.74m wide by 1.83m high. The walls and ceiling were 0.1m thick and they were covered with a ceramic fibre insulation board to establish near steady state conditions within 30 minutes.

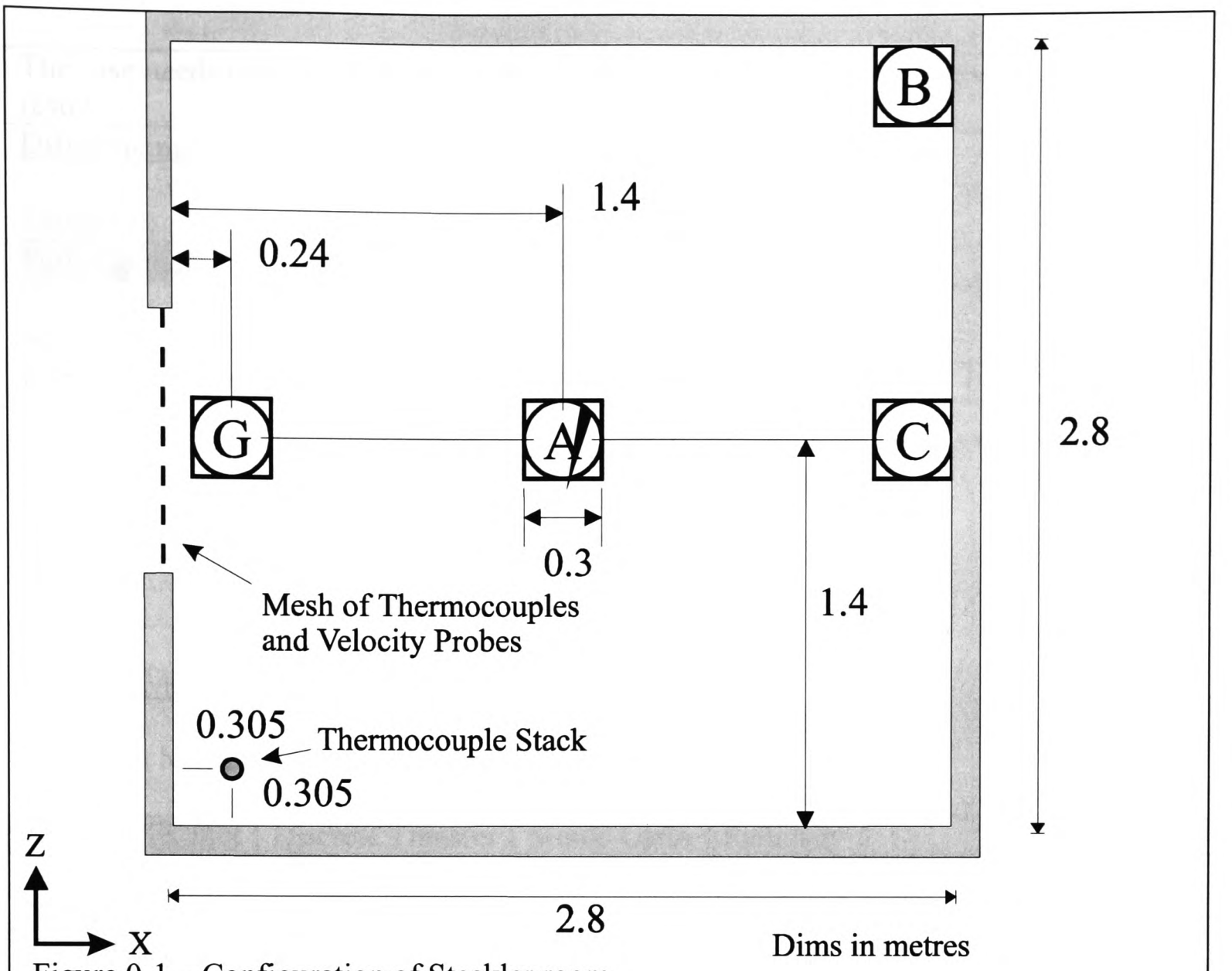


Figure 0-1 – Configuration of Steckler room

The door measures 0.74m wide and 1.83m high and is centrally located in one of the walls.

*Steckler, K.D, Quintiere, J.G and Rinkinen, W.J.[1982], "Flow induced by fire in a compartment", NBSIR 82-2520, National Bureau of Standards.

Required Results

The results should be supplied as graphs and as Excel97 worksheets

Vertical Corner Stack temperatures at 0.305 from the front and side walls.
Vertical Doorway temperature profile in the middle of the doorway.
Horizontal velocity profile for a vertical stack in the middle of the doorway.

These should all be plotted with height of the variable on the y-axis and the variable value (temperature or horizontal velocity) on the x-axis.

CFD set up

1D	2D	3D

Transient	Steady State

The case needs to be run for 200s using 1s time steps. This effectively gives a steady state result.

Differencing Schemes

Temporal:

Fully Implicit	Crank-Nicolson	Explicit	Exponential	

Spatial:

Hybrid	Central Difference	Upwind		

Notes:

Physical Models

Radiation Model *(if not listed please specify in the space provided)*

None	**Six flux**	Discrete Transfer	Monte Carlo	Radiosity	

Notes:

(2) If the fire modelling software does not possess the six-flux model, a discrete transfer model may be used in place of the six-flux model. If the discrete transfer model must be used instead of a six flux model then the discrete model must be made to emulate the behaviour of the six-flux model. This can be achieved by using 6 rays in the co-ordinate directions. If a radiation mesh needs to be specified, this should be identical to the flow mesh. If this is not possible, then at least the same number of cells in each direction must be specified. The details of the mesh must also be provided with your results.

Parameters

The absorption coefficient (a) assumed the following form:

$a = 0.315$

It is assumed there is no scattering so s = 0.0.

Turbulence model *(if not listed please specify in the space provided)*

Laminar	k- ε	**buoyancy modified k-ε**	RNG	

Notes:

Turbulence Parameters*:

C_μ	σ_k	σ_ε	$C_{1\varepsilon}$	$C_{2\varepsilon}$	C_3

0.09	1.0	1.3	1.44	1.92	1.0

*If different parameters are being used please specify in the table above.

Combustion Model *(if not listed please specify in the space provided)*

none	Volumetric heat source	Mixed is burnt	Eddy break up
Magnussen soot model			

Combustion Parameters:

The volumetric heat source is assumed to be centrally located within the room with dimension of 0.3m x 0.3m x 0.3m with a total heat source of 62.9kW.

Compressibility

Incompressible	Boussinesq	Weakly compressible	Fully compressible

Compressibility Parameters:

External Pressure 1.01325e+05 Pa

Buoyancy

Yes	No

Gravity	-9.81m/s in the v-velocity direction.

Material Properties

Material Name	Air
Density	Determined by compressibility (Ideal Gas Law) Molecular Weight of air is 29.35
Viscosity (dynamic)	Laminar 1.798e-005 kg/m.s
Conductivity	0.02622 W/m.K
Specific heat capacity	1007.0 J/kg.K

Initial Values

U-VELOCITY	0.0
V-VELOCITY	0.0
W-VELOCITY	0.0
PRESSURE	0.0
TEMPERATURE	303.75
KINETIC ENERGY	0.01
DISSIPATION RATE	0.01

Boundary conditions

All walls are assumed to be adiabatic for the first phase of the validation process. In the first phase of validation the walls are perfect reflectors of radiation, i.e. the emissivity of

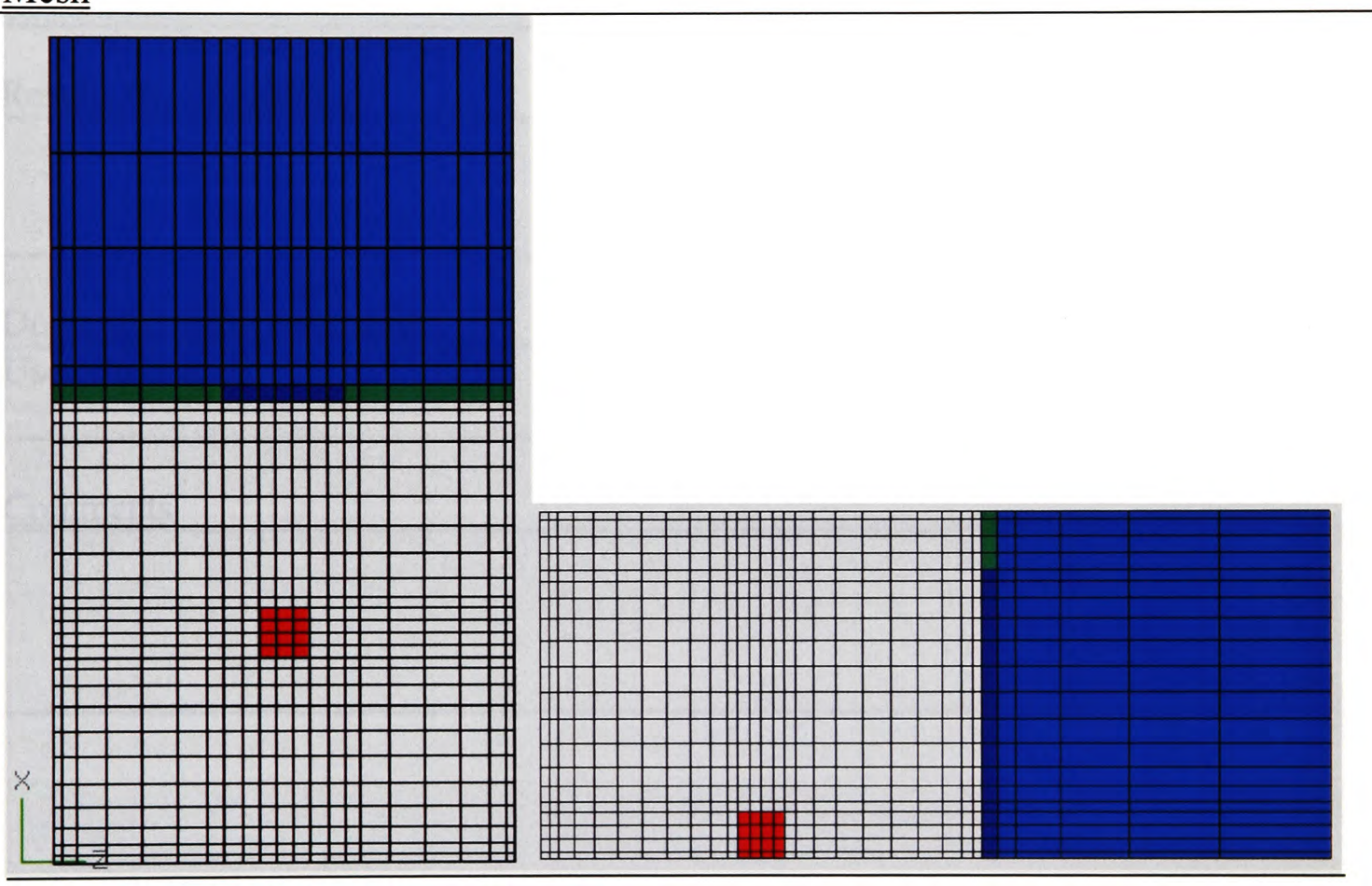
the walls is 0.

The doorway measures 0.74m wide and 1.83m high and is centrally located in one of the walls. This doorway is modelled using three solid non conducting obstructions to create the walls around the doorway. An extended region for this doorway is required to ensure that the airflow in the door is correctly modelled.

On the extended region all the boundary patches are fixed pressure (outlet) boundaries set to 0.0 Pa apart from the floor which is adiabatic.

The fire is modelled as a volumetric heat source which is assumed to be centrally located within the room on the floor with dimensions of 0.3m x 0.3m x 0.3m with a total heat source of 62.9kW.

Mesh




```
31 20 21
X
0.0 0.056 0.1157 0.2137 0.3407 0.4872 0.6438 0.8011 0.9498
1.0805 1.1837 1.25 1.325 1.4 1.475 1.55 1.6163 1.7303
1.8787 2.0481 2.2252 2.3966 2.5491 2.6694 2.744 2.8 2.9
3.0115 3.2949 3.7272 4.2979 5.0
Y
0.0 0.044 0.1293 0.2147 0.3 0.3638 0.4612 0.5858 0.7308
0.8896 1.0555 1.222 1.3824 1.53 1.6582 1.7605 1.83 1.9387
2.0473 2.156 2.2
Z
0.0 0.056 0.142 0.3151 0.5328 0.7527 0.9326 1.03 1.14 1.25
1.35 1.45 1.55 1.66 1.77 1.856 2.0291 2.2468 2.4667 2.6466
```

2.744 2.8

Model Definition files

Convergence

Please specify your convergence criteria including type of error estimator and tolerance value for each variable

Runtime

Results files/Archiving:

Document cross-reference:

User Guides, etc

Comments

C.2 – Steckler Room Fire – combustion model – 2000/2/2

Fire Safety Engineering Group
Maritime Greenwich Campus,
Cooper Building,
University of Greenwich,
King William Walk,
London SE10 9JH, UK.

Date : 4/2/2000

PART 1 – CONTROLLED TEST SETUP

Test case : **Steckler Room Fire – combustion model – 2000/2/2**
Document Version 1.0

PART 1 – CONTROLLED TEST SETUP

Case: **Steckler Room Fire – combustion model – 2000/2/2**

User details	
Run by: Date: Phone no: email:	Address:

Fire modelling Software

SMARTFIRE	CFX	PHOENICS			

Version/build number_____
Date of release_____

Operating System

Windows 95/98/2000	Windows NT	Unix	Dos	

Version/build number_____

Machine

PC	Unix Workstation	

CPU:
Memory:

Case description

The experimental data obtained from Steckler's fire tests* which have been used as part of the validation process for both zone and field fire models. The data represents non-spreading fires in small compartments. The non-spreading fire was created using a centrally located (position A in Figure 0-2) 62.9kW methane burner with a diameter of 0.3m and a height of 0.3m. The experiments were conducted by Steckler et al. in a compartment measuring 2.8m × 2.8m in plane and 2.18m in height (see Figure 0-2) with a doorway centrally located in one of the walls measuring 0.74m wide by 1.83m high. The walls and ceiling were 0.1m thick and they were covered with a ceramic fibre insulation board to establish near steady state conditions within 30 minutes.

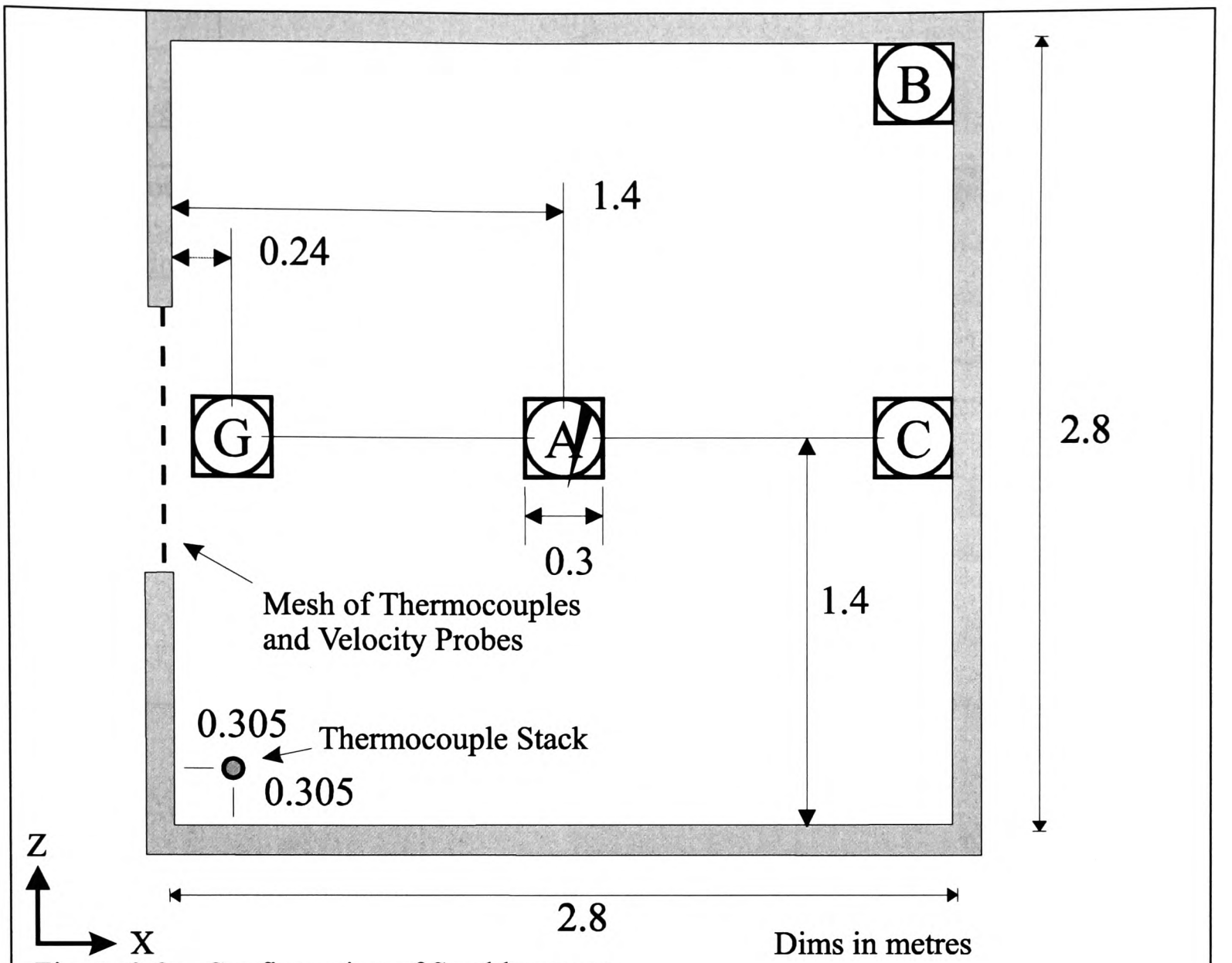


Figure 0-2 – Configuration of Steckler room

The door measures 0.74m wide and 1.83m high and is centrally located in one of the walls.

*Steckler, K.D, Quintiere, J.G and Rinkinen, W.J.[1982], "Flow induced by fire in a compartment", NBSIR 82-2520, National Bureau of Standards.

Required Results

The results should be supplied as graphs and as Excel97 worksheets

These results will be compared against experiment and across the codes.

Vertical Corner Stack temperatures at 0.305 from the front and side walls.
Vertical Doorway temperature profile in the middle of the doorway.
Horizontal velocity profile for a vertical stack in the middle of the doorway.

These should all be plotted with height of the variable on the y-axis and the variable value (temperature or horizontal velocity) on the x-axis.

CFD set up

1	2D	3D

D		

Transient	Steady State

The case needs to be run for 200s using 1s time steps. This effectively gives a steady state result.

Differencing Schemes

Temporal:

Fully Implicit	Crank-Nicolson	Explicit	Exponential	

Spatial:

Hybrid	Central Difference	Upwind		

Notes:

Physical Models

Radiation Model *(if not listed please specify in the space provided)*

None	**Six flux**	Discrete Transfer	Monte Carlo	Radiosity	

Notes:

(3) If the fire modelling software does not possess the six-flux model, a discrete transfer model may be used in place of the six-flux model. If the discrete transfer model must be used instead of a six flux model then the discrete model must be made to emulate the behaviour of the six-flux model. This can be achieved by using 6 rays in the co-ordinate directions. If a radiation mesh needs to be specified, this should be identical to the flow mesh. If this is not possible, then at least the same number of cells in each direction must be specified. The details of the mesh must also be provided with your results.

Parameters

The absorption coefficient (a) assumed the following form:

$a = 0.315$

It is assumed there is no scattering so s = 0.0.

Turbulence model *(if not listed please specify in the space provided)*

Laminar	k- ε	**buoyancy modified k-ε**	RNG	

Notes:

Turbulence Parameters*:

C_μ	σ_k	σ_ε	$C_{1\varepsilon}$	$C_{2\varepsilon}$	C_3
0.09	1.0	1.3	1.44	1.92	1.0

*If different parameters are being used please specify in the table above.

Combustion Model *(if not listed please specify in the space provided)*

none	Volumetric heat source	Mixed is burnt	Eddy break up
Magnussen soot model			

Combustion Parameters:

For phase-1 testing the Eddy Break up model must be used with the collision mixing model and infinite rate chemistry.

$$S_{m_f} = -\rho \frac{\varepsilon}{k} C_R \min\left(m_f, \frac{m_o}{i} \right),$$

where S_{mf} is the source term for the fuel mass fraction equation,
$C_R = 4.0$ (rate constant for collision mixing model),
m_f is the mass fraction of fuel
m_o is the mass fraction of oxident.

i is the amount of oxygen used for combustion every unit fuel, i.e

1kg Fuel + i kg oxidant => (1+i) kg products

Compressibility

Incompressible	Boussinesq	Weakly compressible	Fully compressible

Compressibility Parameters:

External Pressure 1.01325e+05

Buoyancy

Yes	No

Gravity	-9.81m/s in the v-velocity direction.

Material Properties

Material Name	Air
Density	Determined by compressibility (Ideal Gas Law) Molecular Weight of air is 29.35
Viscosity	Laminar 1.798e-005 + Value determined from turbulence model
Conductivity	0.02622
Specific heat capacity	1007.0

Initial Values

U-VELOCITY	0.0
V-VELOCITY	0.0
W-VELOCITY	0.0
PRESSURE	0.0
TEMPERATURE	303.75
KINETIC ENERGY	0.01
DISSIPATION RATE	0.01

Boundary conditions

All walls are assumed to be adiabatic for the first phase of the validation process. In the first phase of validation the walls are perfect reflectors of radiation, i.e. the emissivity of the walls is 0. The default log-law turbulent wall functions should be used.

The doorway measures 0.74m wide and 1.83m high and is centrally located in one of the walls. This doorway is modelled using three solid non conducting obstructions to create the walls around the doorway. An extended region for this doorway is required to ensure that the airflow in the door is correctly modelled.

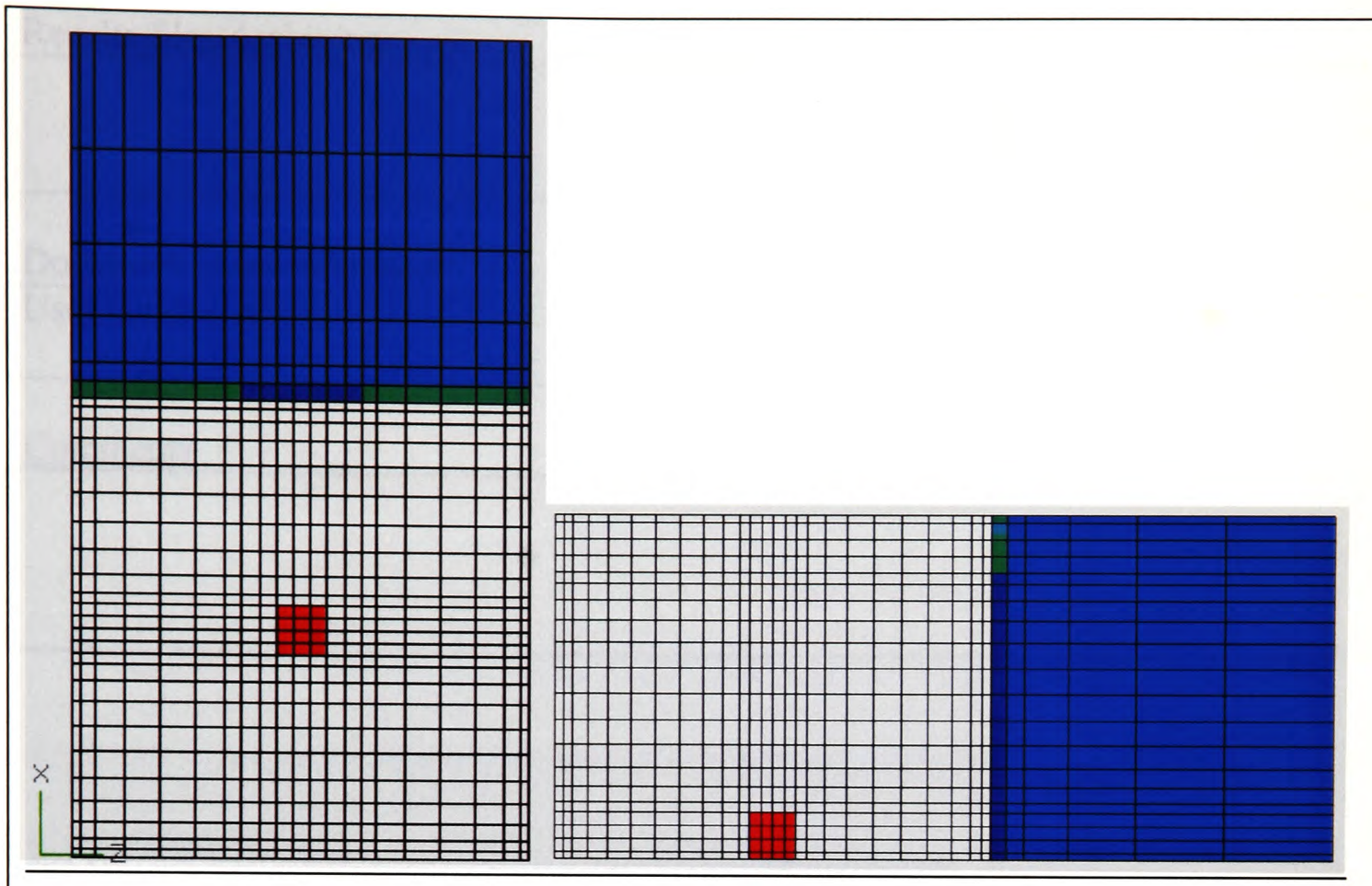
On the extended region all the boundary patches are fixed pressure (outlet) boundaries set to 0.0 Pa apart from the floor which is adiabatic.

For the gaseous combustion model the heat of combustion of pure methane is 5.001×10^7 J/Kg. The commercial grade fuel used in the gas burner is 91% methane by mass the rest of the fuel mass is assumed to be non-combustible. The fuel supply rate is 1.38e-3 Kg/s. The chemical equation for the combustion of pure methane is shown below.

$CH_4 + 2O_2 \rightarrow 2H_2O + CO_2$

For this case the fuel is from an inlet 0.3 x 0.3 centrally located within the room and placed on the floor. The inlet has a v-velocity component of 0.0239m/s and a mass fraction of 1.0 and fuel fraction 1.0. The ambient temperature is 303.75

Mesh




```
31 20 21
X
0.0 0.056 0.1157 0.2137 0.3407 0.4872 0.6438 0.8011 0.9498
1.0805 1.1837 1.25 1.325 1.4 1.475 1.55 1.6163 1.7303
1.8787 2.0481 2.2252 2.3966 2.5491 2.6694 2.744 2.8 2.9
3.0115 3.2949 3.7272 4.2979 5.0
Y
0.0 0.044 0.1293 0.2147 0.3 0.3638 0.4612 0.5858 0.7308
0.8896 1.0555 1.222 1.3824 1.53 1.6582 1.7605 1.83 1.9387
2.0473 2.156 2.2
Z
0.0 0.056 0.142 0.3151 0.5328 0.7527 0.9326 1.03 1.14 1.25
1.35 1.45 1.55 1.66 1.77 1.856 2.0291 2.2468 2.4667 2.6466
2.744 2.8
```

Model Definition files

Convergence

Please specify your convergence criteria including type of error estimator and tolerance value for each variable

All variable residuals should be converged to 0.1%. The mass source tolerance is set to 0.0001.

Runtime

steckler_comb.inf; steckler_comb.geo

Results files/Archiving:

Document cross-reference:

User Guides, etc

Comments

C.3 – Fire in a completely open compartment with lid case – 2000/2/3

Fire Safety Engineering Group
Maritime Greenwich Campus,
Cooper Building,
University of Greenwich,
King William Walk,
London SE10 9JH, UK.

Date : 4/2/2000

PART 1 – CONTROLLED TEST SETUP

Test case : **Fire in a completely open compartment with lid case – 2000/2/3**

Document Version 1.1

PART 1 – CONTROLLED TEST SETUP

Case: **Simple volumetric fire under a lid case – 2000/2/3**

User details	
Run by: Date: Phone no: email:	Address:

Fire modelling Software

SMARTFIRE	CFX	PHOENICS			

Version/build number_____
Date of release_____

Operating System

Windows 95/98/2000	Windows NT	Unix	Dos	

Version/build number_____

Machine

PC	Unix Workstation	

CPU:
Memory:

Case description

This Fire case utilises a volumetric heat source. The compartment is completely open apart from a solid ceiling. The fire is located on the floor at the centre of the building. The prescribed fire volume is 1m x 1m x 1m. The fire power is defined as $H = 0.188t^2(kW)$ (i.e. t squared fire and t is measured in seconds). The compartment is 5m(wide) × 5m(long) × 3m(high). It should be noted that this is a hypothetical case for which there is no experimental data. The walls are adiabatic. The ambient temperature is 303.75K.

Required Results

The results should be supplied as graphs and as Excel97 worksheets
This case is used for comparison between the codes.

All the results are instantaneous results for the 110^{th} second.
Temperature profile across the cabin 0.1m below the ceiling.
Temperature profile across the cabin 0.3m below the ceiling.

CFD set up

1D	2D	**3D**

Transient	Steady State

110*1s time steps (110s total)

Differencing Schemes

Temporal:

Fully Implicit	Crank-Nicolson	Explicit	Exponential	

Spatial:

Hybrid	Central Difference	Upwind		

Notes:

Physical Models

Radiation Model *(if not listed please specify in the space provided)*

None	**Six flux**	Discrete Transfer	Monte Carlo	Radiosity	

Notes:

(4) If the fire modelling software does not possess the six-flux model, a discrete transfer model may be used in place of the six-flux model. If the discrete transfer model must be used instead of a six flux model then the discrete model must be made to emulate the behaviour of the six-flux model. This can be achieved by using 6 rays in the co-ordinate directions. If a radiation mesh needs to be specified, this should be identical to the flow mesh. If this is not possible, then at least the same number of cells in each direction must be specified. The details of the mesh must also be provided with your results.

Parameters

The absorption coefficient (a) is equal to 0.7

It is assumed there is no scattering so s = 0.0.

Turbulence model *(if not listed please specify in the space provided)*

Laminar	k- ε	**buoyancy modified k-ε**	RNG	

Notes:

Turbulence Parameters*:

C_μ	σ_k	σ_ε	$C_{1\varepsilon}$	$C_{2\varepsilon}$	C_3
0.09	1.0	1.3	1.44	1.92	1.0

*If different parameters are being used please specify in the table above.

Combustion Model *(if not listed please specify in the space provided)*

none	Volumetric heat source	Mixed is burnt	Eddy break up
Magnussen soot model			

Combustion Parameters:

Compressibility

Incompressible	Boussinesq	Weakly compressible	Fully compressible

Compressibility Parameters:

External Pressure 1.01325e+05

Buoyancy

Yes	No

Gravity	-9.81m/s in the v-velocity direction.

Material Properties

Material Name	Air
Density	Determined by compressibility (Ideal Gas Law) Molecular Weight of air is 29.35
Viscosity	1.6e-005 + Value determined from turbulence model
Conductivity	0.02622
Specific heat capacity	1045.78

Initial Values

U-VELOCITY	0.0
V-VELOCITY	0.0
W-VELOCITY	0.0
PRESSURE	0.0
TEMPERATURE	303.75
KINETIC ENERGY	0.01
DISSIPATION RATE	0.01

Boundary conditions

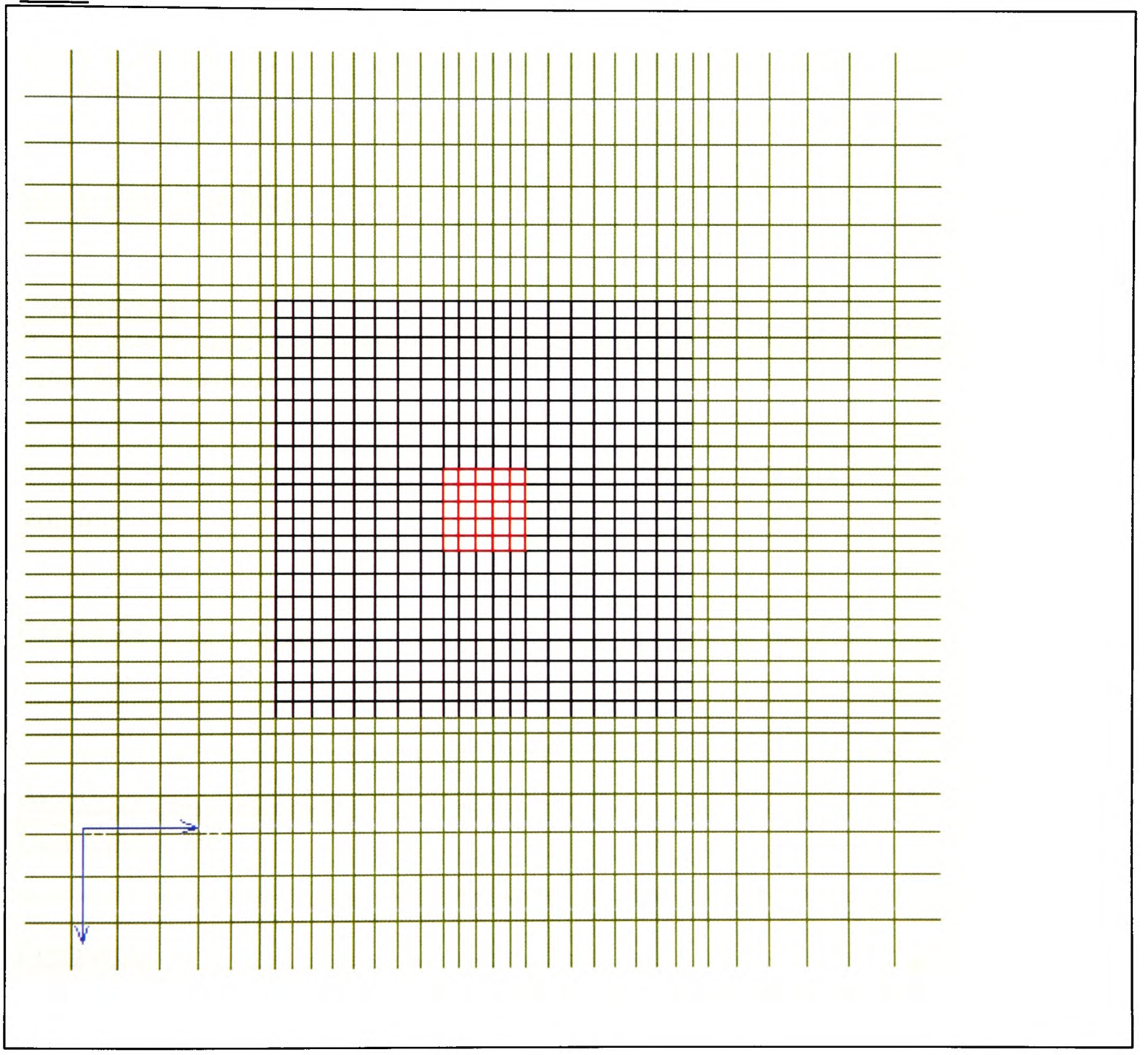
All walls are assumed to be adiabatic for the first phase of the validation process. In the first phase of validation the walls are perfect reflectors of radiation, i.e. the emissivity of the walls is 0. The default log-law turbulent wall functions should be used.

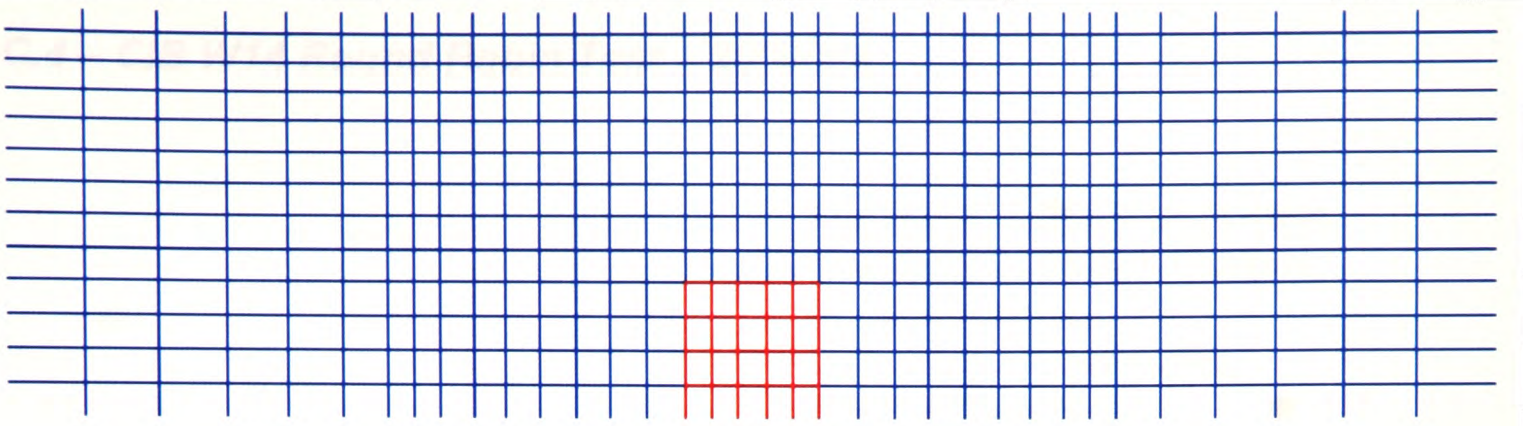
An adiabatic floor covering the whole of the bottom of the domain
An adiabatic ceiling centrally located 3m above the floor measuring 5m x 5m.

The centrally located fire volume is 1m x 1m x 1m. The fire power is defined by the standard method, i.e., $H = 0.188t^2(kW)$ (i.e. t squared fire)

Extended regions are required all around the compartment and outlet boundary conditions are applied to these patches with pressure set to equal 0.0Pa.

Mesh





35 13 35
X & Z 0.0 0.582337 1.126982 1.629535 2.083877 2.480691 2.803219 3.0 3.203063 3.435275 3.67993 3.933033 4.192609 4.457462 4.726788 5.0 5.2 5.4 5.6 5.8 6.0 6.273212 6.542538 6.807391 7.066967 7.32007 7.564725 7.796937 8.0 8.196781 8.519309 8.916123 9.370465 9.873018 10.417663 11.0
Y 0.0 0.25 0.5 0.75 1.0 1.243039 1.483051 1.719647 1.952316 2.180348 2.402694 2.617619 2.821613 3.0

Model Definition files

Convergence

Please specify your convergence criteria including type of error estimator and tolerance value for each variable

Runtime

Results files/Archiving:

Document cross-reference:

User Guides, etc

Comments

C.4 – CIB W14 Round Robin Test – 2000/2/4

Fire Safety Engineering Group
Maritime Greenwich Campus,
Cooper Building,
University of Greenwich,
King William Walk,
London SE10 9JH, UK.

Date : 3/3/2000

PART 1 – CONTROLLED TEST SETUP

Test case : **CIB W14 Round Robin Test – 2000/2/4**
Document Version 1.2

PART 1 – CONTROLLED TEST SETUP

Case: **CIB W14 Round Robin Test – 2000/2/4**

User details

Run by: Date: Phone no: email:	Address:

Fire modelling Software

SMARTFIRE	CFX	PHOENICS			

Version/build number_____
Date of release_____

Operating System

Windows 95/98/2000	Windows NT	Unix	Dos	

Version/build number_____

Machine

PC	Unix Workstation	

CPU:
Memory:

Case description

This case arises from the CIB round robin tests* of which subscenario B1 is the case of interest. The fire compartment measured 14.4 m × 7.2 m in plan and 3.53 m in height and contained a doorway of dimensions 2.97 m × 2.13 m. The walls of the compartment were made of aerated concrete blocks (with siporex mortar) with thickness 0.3 m and the following material properties: specific heat 1.05 kJ/kg.K, thermal conductivity 0.12 W/m.K and density 500 kg/m^3. The initial air temperature was measured as 20.0 °C.

The fire was located on the floor in the centre of the room. The fire fuel consisted of softwood (Pinea ecelsa) timber cribs nailed into 40mm x 40mm battens. The crib measured 2.4m in length, 2.4 m in width and 1.4 m in height.

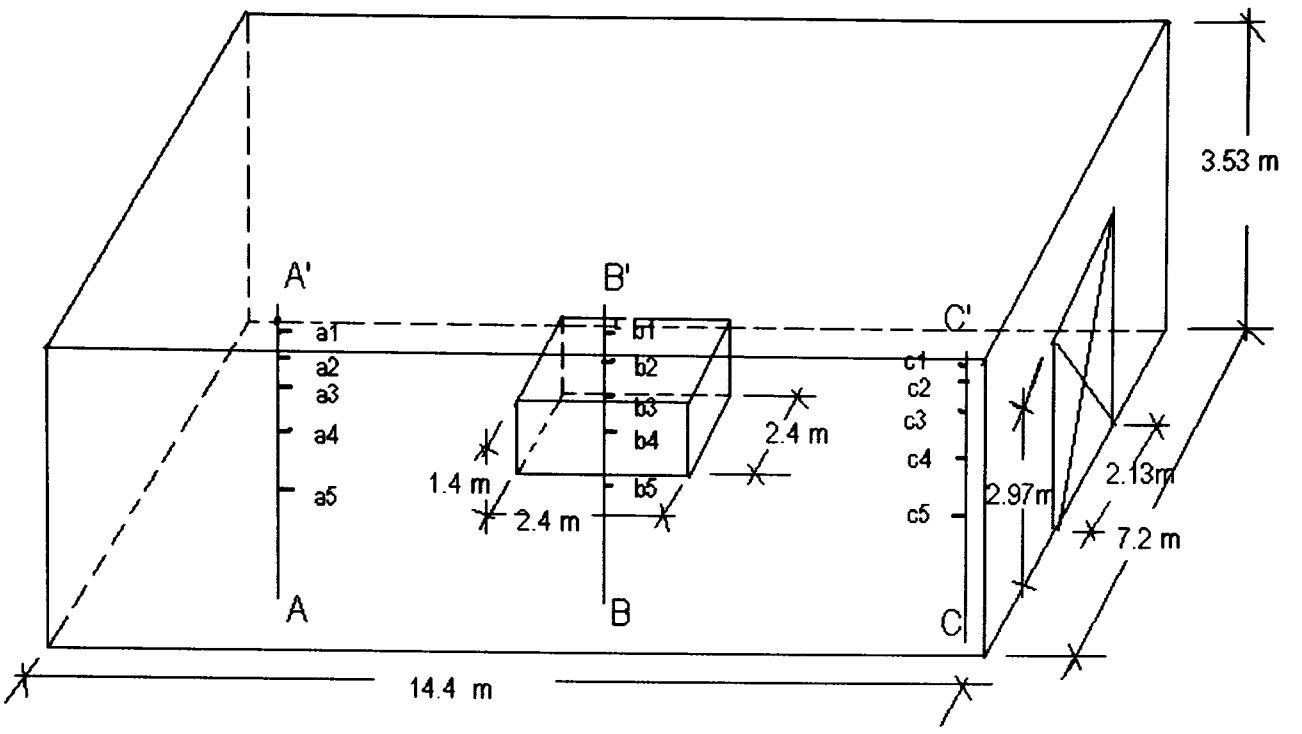


Figure 0-3 – Depiction of fire compartment geometry showing location of fire source.

The heat release rate ($\dot{Q}$) is given by the following calculation :-

$$\dot{Q} = \chi \cdot \Delta H_c \cdot \dot{m}$$

The efficiency factor (χ) and heat of combustion (ΔH_c) were given as χ =0.7 and ΔH_c is 17.8 MJ/kg for burning wood with a 10% moisture content and the mass loss rate ($\dot{m}$) (kg/s) for the wood crib is presented in the table below. A maximum heat release rate of approximately 11 MW was produced. It is assumed that the fuel molecule is $CH_{1.7}O_{0.83}$.

*Hostikka S and Keski-Rahkonen O., Results of CIB W14 Round Robin for Code Assessment Scenario B. Draft 31/08/98, VTT Technical Research Centre of Finland.

Time (s)	0	60	120	180	240	300	360	420	480	540	600
Mass loss rate(kg/s)	0	0.005	0.004	0.009	0.013	0.014	0.019	0.033	0.052	0.08	0.207

Required Results:

Temperature histories locations (Ta, Tb and Tc) illustrated in Figure 0-4 which is a plan view of the compartment with the doorway at the right hand side of the compartment. In the vertical direction spot values are needed at 20cm, 50cm, 100cm, 180cm and 250cm below the ceiling. These spot values should be produced at the end of every time step.

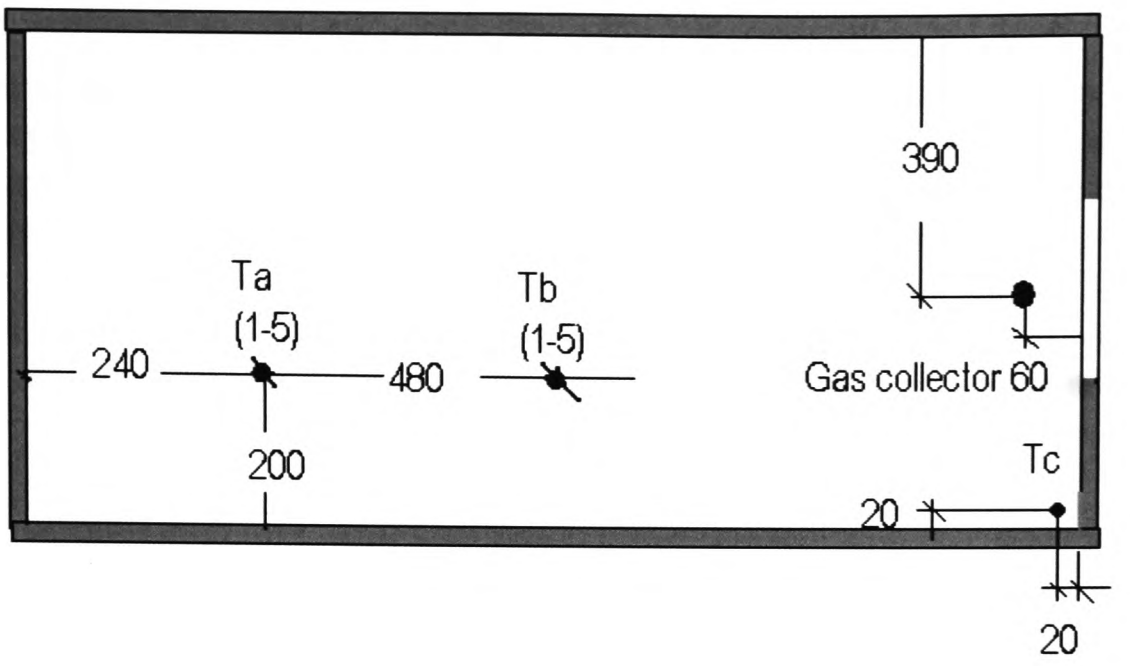


Figure 0-4 - location of the three thermocouple trees in the fire compartment

CFD set up

1D	2D	3D

Transient	Steady State

The case needs to be run for 10 minutes using 5 s time steps.

Differencing Schemes

Temporal:

Fully Implicit	Crank-Nicolson	Explicit	Exponential	

Spatial:

Hybrid	Central Difference	Upwind		

Notes:

Physical Models

Radiation Model *(if not listed please specify in the space provided)*

None	Six flux	Discrete Transfer	Monte Carlo	Radiosity	

Notes:

(5) If the fire modelling software does not possess the six-flux model, a discrete transfer model may be used in place of the six-flux model. If the discrete transfer model must be used instead of a six flux model then the discrete model must be made to emulate the behaviour of the six-flux model. This can be achieved by using 6 rays in the co-ordinate directions. If a radiation mesh needs to be specified, this should be identical to the flow mesh. If this is not possible, then at least the same number of cells in each direction must be specified. The details of the mesh must also be provided with your results.

Parameters

The absorption coefficient (a) assumed the following form:

$a = 0.315$

It is assumed there is no scattering so s = 0.0.

Turbulence model *(if not listed please specify in the space provided)*

Laminar	k- ε	buoyancy modified k-ε	RNG	

Notes:

For the first phase validation process the standard k- ε turbulence model with the standard buoyancy modification, $C_3 = 1.0$ (see ε-equation below) must be used with the parameters below.

$$\frac{\partial \rho \varepsilon}{\partial t} + \nabla.(\rho U \varepsilon) - \nabla\left(\left(\mu_L + \frac{C_\mu \rho \frac{k^2}{\varepsilon}}{\sigma_\varepsilon}\right)\nabla\varepsilon\right) = C_1 \frac{\varepsilon}{k}(P + C_3 \max(G,0)) - C_2 \rho \frac{\varepsilon^2}{k}$$

Turbulence Parameters*:

C_μ	σ_k	σ_ε	$C_{1\varepsilon}$	$C_{2\varepsilon}$	C_3
0.09	1.0	1.3	1.44	1.92	1.0

*If different parameters are being used please specify in the table above.

Combustion Model *(if not listed please specify in the space provided)*

none	Volumetric heat source	Mixed is burnt	Eddy break
Magnussen soot model			

Combustion Parameters:

For phase-1 testing the Eddy Break up model must be used with the collision mixing model and infinite rate chemistry.

$$S_{m_f} = -\rho \frac{\varepsilon}{k} C_R \min\left(m_f, \frac{m_o}{i}\right),$$

where S_{mf} is the source term for the fuel mass fraction equation,

C_R = 4.0 (rate constant for collision mixing model),
m_f is the mass fraction of fuel
m_o is the mass fraction of oxident.
i is the amount of oxygen used for combustion every unit fuel, i.e

1kg Fuel + ikg -> (1+i) kg products

Compressibility

Incompressible	Boussinesq	Weakly compressible	**Fully compressible**

Compressibility

Incompressible	Boussinesq	Weakly compressible	Fully compressible

Compressibility Parameters:

External Pressure 1.013e+05 Pa

Buoyancy

Yes	No

Material Properties

Material Name	Air
Density	Determined by compressibility (Ideal Gas Law) Molecular Weight of air is 29.35
Viscosity (dynamic)	Laminar 1.798e-005kg/m.s + Value determined from turbulence model
Conductivity	0.02622 W/m.K
Specific heat capacity	1007.0 J/kg.K

The fuel and combustion products are assumed to have the same physical properties of air for the first phase of validation.

Initial Values

U-VELOCITY	0.0
V-VELOCITY	0.0
W-VELOCITY	0.0
PRESSURE	0.0
TEMPERATURE	293.75
KINETIC ENERGY	0.01
DISSIPATION RATE	0.01

Boundary conditions

All walls are assumed to be adiabatic for the first phase of the validation process. In the first phase of validation the walls are perfect reflectors of radiation, i.e. the emissivity of the walls is 0. The default log-law turbulent wall functions should be used.

The doorway measures 2.97 m high × 2.13 m wide and is centrally located in one of the small walls. This wall is constructed as a solid non-conducting obstruction with a thickness 0.3m An extended region for this door is required to ensure that the airflow in the door is correctly modelled.

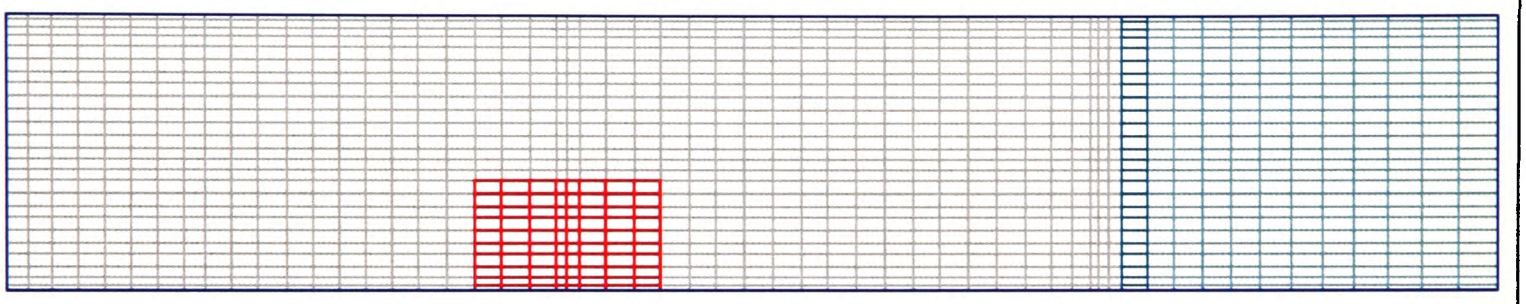
On the extended region all the boundary patches are fixed pressure (outlet) boundaries set to 0.0 Pa apart from the floor which is an adiabatic floor.

The fire is modelled as a volumetric source of fuel with the same location, position and using the mass fuel rate as the wood crib in the case description.

The fire needs to be modelled as a volumetric source of fuel with the same dimensions as the crib illustrated above using the fuel mass source specified above.

Mesh

The mesh consists of 42775 (i.e. 59 x 25 x 29) computational cells.



59 25 29

```
X 0.0 0.119432 0.315183 0.556021 0.831774 1.136787 1.467348 1.820784
2.195064 2.588575 3.0 3.511529 3.994572 4.44635 4.863213 5.240085
5.569238 5.836772 6.0 6.3 6.6 6.9 7.2 7.5 7.8 8.1 8.4 8.563228 8.830762
9.159915 9.536787 9.95365 10.405428 10.888471 11.4 11.774744 12.134776
12.479132 12.806659 13.115946 13.405214 13.672127 13.913434 14.124187
14.295486 14.4 14.5 14.599999 14.7 14.797048 14.994193 15.262831
15.591825 15.974489 16.406185 16.883438 17.403511 17.964167 18.56353
19.200001
Y 0.0 0.2 0.4 0.6 0.8 1.0 1.2 1.4 1.482473 1.617647 1.783956 1.974375
2.185 2.337378 2.479893 2.611395 2.730281 2.834114 2.918509 2.97
3.030143 3.12872 3.25 3.37128 3.469857 3.53
Z 0.0 0.126073 0.332710 0.586939 0.878026 1.2 1.597829 1.945285
2.227695 2.4 2.535 2.724465 2.970275 3.243066 3.535 3.665 3.956934
4.229725 4.475535 4.665 4.8 4.972305 5.254715 5.602171 6.0 6.321974
6.613061 6.86729 7.073927 7.2
```

Input files

Convergence

All variable residuals should be converged to 0.1%. The mass source tolerance is set to 0.0001.

Runtime

Results files/Archiving:

Document cross-reference:

Comments

C.5 – LPC-007 – 2000/2/5

Fire Safety Engineering Group
Maritime Greenwich Campus,
Cooper Building,
University of Greenwich,
King William Walk,
London SE10 9JH, UK.

Date : 14/2/2000

Test case : **LPC-007 – 2000/2/5**
Document Version 1.1

PART 1 – CONTROLLED TEST SETUP

Case: **LPC-007 – 2000/2/5**

User details	
Run by: Date: Phone no: email:	Address:

Fire modelling Software

SMARTFIRE	CFX	PHOENICS			

Version/build number_____
Date of release_____

Operating System

Windows 95/98/2000	Windows NT	Unix	Dos	

Version/build number_____

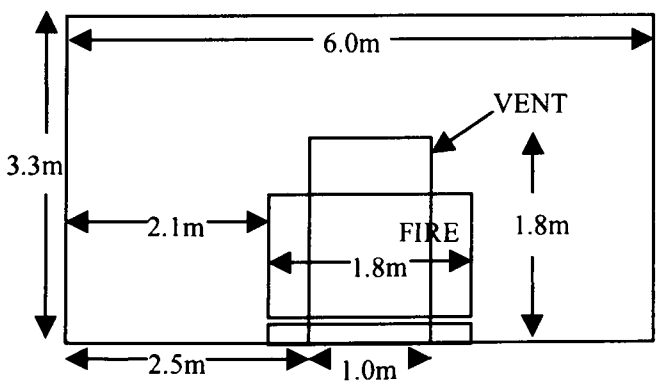
Machine

PC	Unix Workstation	

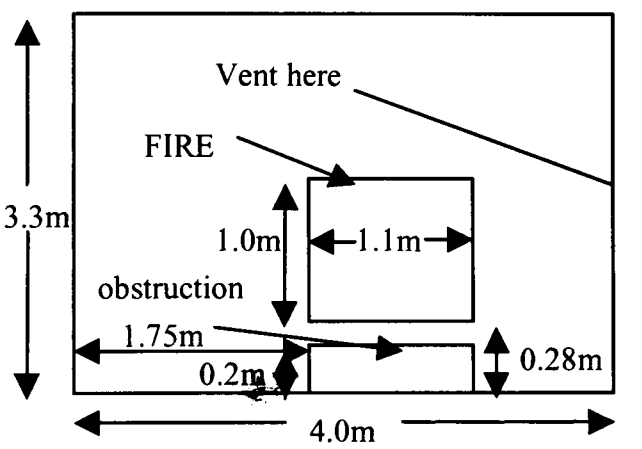
CPU:
Memory:

Case description

This test case arises from a fire test conducted by the Loss Prevention Council (LPC)*. The test is a burning wood crib within an enclosure with a single opening. The test compartment is illustrated below and had a floor area of 6m x 4m and a 3.3m high ceiling. The compartment contained a doorway (vent) measuring 1.0m x 1.8m located on the rear 6m x 3.3m wall. The walls and ceiling of the compartment were made of fire resistant board (Asbestos) which were 0.1m thick. The floor was made of concrete.



y-z view



x-y view

The heat release rate ($\dot{Q}$) is given by the following calculation (see equation 1).

$$\dot{Q} = \chi \cdot \Delta H_c \cdot \dot{m} \qquad (1)$$

The efficiency factor (χ) and heat of combustion (ΔH_c) were given as χ =0.7 and ΔH_c is 17.8 MJ/kg for burning wood with a 10% moisture content and the mass loss rate ($\dot{m}$) (kg/s) for the wood crib is presented in the table below. It is assumed that the fuel molecule is $CH_{1.7}O_{0.83}$.

Time(s) |	0	150	450	460	1650
$\dot{m}$ (kg/s)|	0	0.01835	0.18636	0.1978	0.1978

* Glocking, J.L.D, Annable, K., Campbell, S.C. "Fire Spread in multi-storey buidings – *'Fire break out from heavyweight unglazed curtain wall system – Run 007'* ", LPC Laboratories rep. TE 88932-43, 25 Feb 1997.

Required Results

The results should be supplied as graphs and as Excel97 worksheets

The results from this case will be compared against experimental results and between the codes.

The required results are the temperature history curves for the first 900 seconds for the following locations:-

For the corner thermocouple stack located at 0.57m away form the side wall and 0.5m away from the front wall containing the vent.

The thermocouples within this stack are located at 1.5m, 2.0m, 2.5m and 3.0m above the floor.

The plume temperature measurements were taken at 3.0m away from the side wall and 2.392m away form the back wall of the compartment with the low measurement 1.5m above the floor and the high measurement at 3.0m above the floor.

CFD set up

1D	2D	**3D**

Transient	Steady State

180 * 5s time steps (900s total)

Differencing Schemes

Temporal:

Fully Implicit	Crank-Nicolson	Explicit	Exponential	

Spatial:

Hybrid	Central Difference	Upwind		

Notes:

Physical Models

Radiation Model *(if not listed please specify in the space provided)*

None	**Six flux**	Discrete Transfer	Monte Carlo	Radiosity	

Notes:

(6) If the fire modelling software does not possess the six-flux model, a discrete transfer model may be used in place of the six-flux model. If the discrete transfer model must be used instead of a six flux model then the discrete model must be made to emulate the behaviour of the six-flux model. This can be achieved by using 6 rays in the co-ordinate directions. If a radiation mesh needs to be specified, this should be identical to the flow mesh. If this is not possible, then at least the same number of cells in each direction must be specified. The details of the mesh must also be provided with your results.

Parameters

The absorption coefficient (a) assumed the following form: $a = 0.315$ It is assumed there is no scattering so s = 0.0.

Turbulence model *(if not listed please specify in the space provided)*

Laminar	k- ε	**Buoyancy modified k-ε**	RNG	

Notes:

Turbulence Parameters*:

C_μ	σ_k	σ_ε	$C_{1\varepsilon}$	$C_{2\varepsilon}$	C_3
0.09	1.0	1.3	1.44	1.92	1.0

*If different parameters are being used please specify in the table above.

Combustion Model *(if not listed please specify in the space provided)*

none	Volumetric heat source	Mixed is burnt	Eddy break up
Magnussen soot model			

Combustion Parameters:

For phase-1 testing the Eddy Break up model must be used with the collision mixing model and infinite rate chemistry.

$$S_{m_f} = -\rho \frac{\varepsilon}{k} C_R \min\left(m_f, \frac{m_o}{i} \right),$$

where S_{mf} is the source term for the fuel mass fraction equation,
$C_R = 4.0$ (rate constant for collision mixing model),
m_f is the mass fraction of fuel
m_o is the mass fraction of oxident.
i is the amount of oxygen used for combustion every unit fuel, i.e

1kg Fuel + ikg -> (1+i) kg products

Compressibility

Incompressible	Boussinesq	Weakly compressible	Fully compressible

Compressibility Parameters:

External Pressure 1.01325e+05

Buoyancy

Yes	No

Gravity	-9.81m/s in the v-velocity direction.

Material Properties

Material Name	Air
Density	Determined by compressibility (Ideal Gas Law) Molecular Weight of air is 29.35
Viscosity	Laminar 1.798e-005 + Value determined from turbulence model
Conductivity	0.02622
Specific heat capacity	1007.0

Initial Values

U-VELOCITY	0.0
V-VELOCITY	0.0
W-VELOCITY	0.0
PRESSURE	0.0
TEMPERATURE	293.75
KINETIC ENERGY	0.01
DISSIPATION RATE	0.01

Boundary conditions

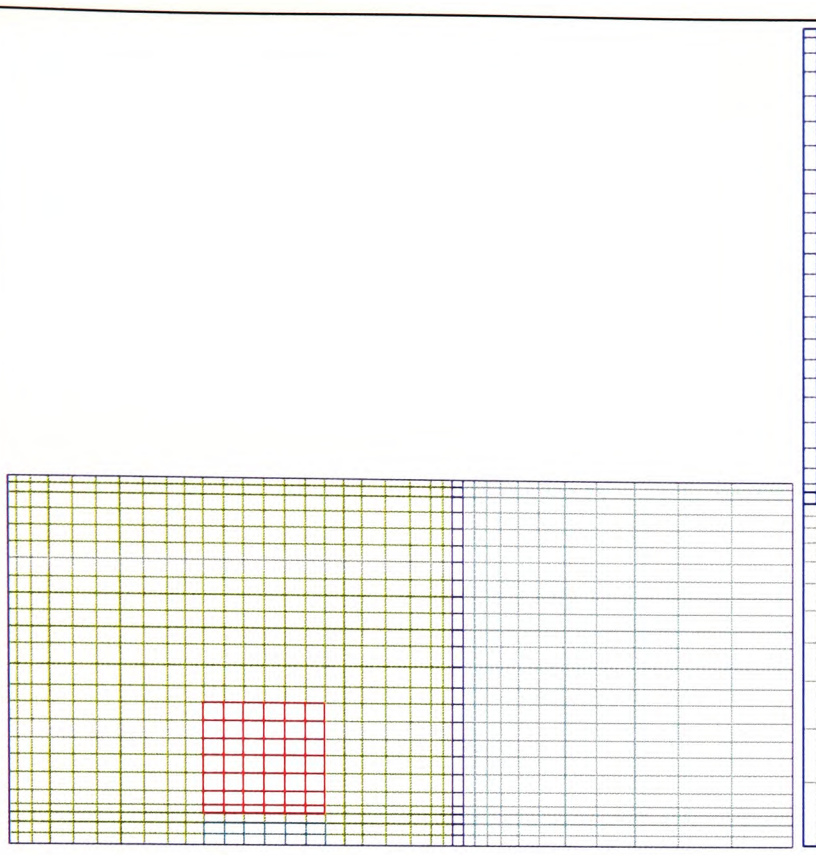
All walls are assumed to be adiabatic for the first phase of the validation process. In the first phase of validation the walls are perfect reflectors of radiation, i.e. the emissivity of the walls is 0. The default log-law turbulent wall functions should be used.

The door measures 1.0m wide and 1.8m high and is centrally located in the front wall. An extended region for this door is required to ensure that the airflow in the doorway is correctly modelled.

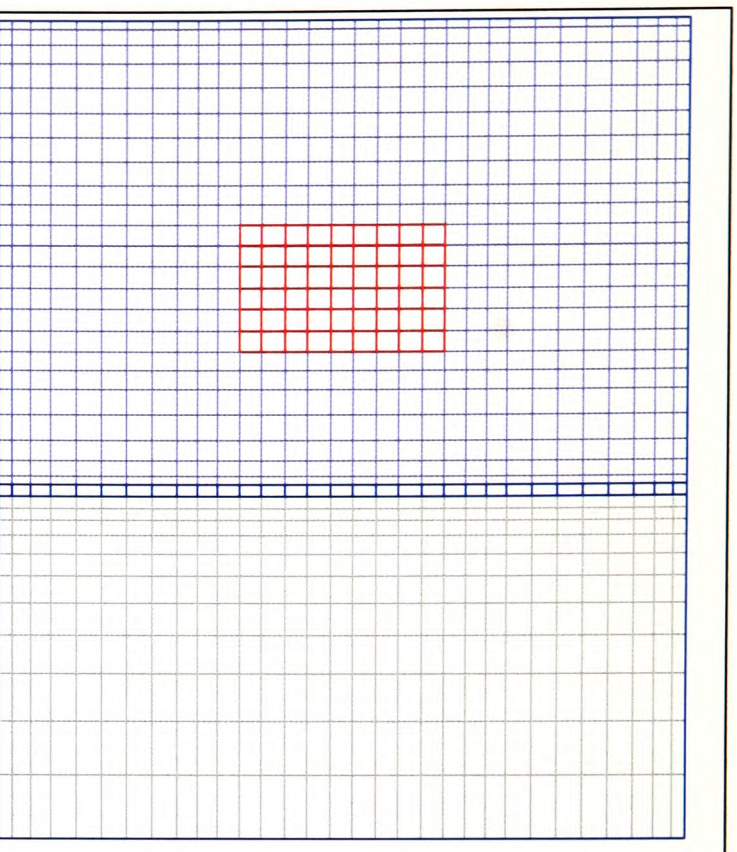
On the extended region all the boundary patches are fixed pressure (outlet) boundaries set to 0.0 Pa.

The fire needs to be modelled as a volumetric source of fuel with the same dimensions as the crib illustrated above using the fuel mass source specified above.

Mesh



Side view



Top view

```
31 24 35

X 0.0 0.12 0.28 0.46 0.68 0.9 1.12 1.34 1.56 1.74 1.92 2.1 2.3 2.5 2.7
2.9 3.1 3.3 3.5 3.7 3.9 4.08 4.26 4.44 4.66 4.88 5.1 5.32 5.54 5.72
5.88 6.0
Y 0.0 0.1 0.2 0.28 0.36 0.49 0.648 0.806 0.964 1.122 1.28 1.43 1.62 1.8
1.952 2.104 2.256 2.408 2.56 2.712 2.864 3.016 3.145 3.234 3.3
Z 0.0 0.08 0.21 0.37 0.58 0.79 1.0 1.21 1.42 1.585 1.75 1.9333 2.1167
2.3 2.4833 2.6667 2.85 3.015 3.18 3.4 3.62 3.8 3.92 4.0 4.1 4.2 4.312
4.4435 4.6021 4.7951 5.0303 5.315 5.6568 6.0633 6.5418 7.1
```

Model Definition files

Convergence

Please specify your convergence criteria including type of error estimator and tolerance value for each variable

Runtime

Results files/Archiving:

Document cross-reference:

User Guides, etc

Comments

Appendix D – Phase 2 specification sheets

D.1 - Multray radiation model with 6 rays

Fire Safety Engineering Group
Maritime Greenwich Campus,
Cooper Building,
University of Greenwich,
King William Walk,
London SE10 9JH, UK.

Date : 14/1/2001

PART 2 – FREE TEST SETUP

Test case : **Radiation in 3 dimensional cavity 2000/1/5 – multiray radiation model with 24 rays.**
Document Version 1.0

Case: Radiation in 3 dimensional cavity 2000/1/5

User details	
Run by: Angus Grandison Date: 14/1/2001 Phone no: 020 8331 7912 email: ga02@gre.ac.uk	Address: School of CMS University of Greenwich Maritime Greenwich Campus Old Royal Naval College Greenwich SE10 9LS. UNITED KINGDOM

Fire modelling Software (if not listed please specify in the space provided)

SMARTFIRE	CFX	PHOENICS			

Version/build number : v2.01 b369d
Date of release : 19/4/2000

Operating System *(if not listed please specify in the space provided)*

Windows 95/98/2000	Windows NT	Unix	Dos	

Version/build number v4.00 sp6

Machine *(if not listed please specify in the space provided)*

PC	Unix Workstation	

CPU: Intel Pentium III 733Mhz
Memory: 256Mb

Case description
This test case tests the fire modelling software's radiation model. The cavity is a unit cube (1m x 1m x 1m) with three walls with planes x=1, y=0 and z =0 set to an unit emissive power and the three other walls set to zero emissive power. All the walls are considered radiatively black have unit emissivity and the fluid has a unit absorption coefficient. Scattering is neglected. No fluid flow is considered

Required Results
The results should be supplied as graphs and as Excel97 worksheets
The results will be compared with analytically derived results[1,2] and between the codes.

The emissive power variation with x must be provided for the following locations
y= 0.1, 0.3, 0.5, 0.7, 0.9 on the planes z = 0.5, 0.9.

1) Larsen, M. E., "Exchange Factor Method and Alternative Zonal Formulation for Analysis of radiating enclosures containing paticipating media", PhD thesis, University of Texas, Austin, 1983.

2) Fiveland, W. A., "Three dimensional Radiative Heat-Transfer Solutions by Discrete-Ordinate Method", Journal of Thermophysics, Vol. 2, No. 4, October 1988, pp 309-316.

CFD set up

1D	2D	3D

Transient	Steady State

Differencing Schemes

Temporal:

Fully Implicit	Crank-Nicolson	Explicit	Exponential	

Spatial:

Hybrid	Central Difference	Upwind		

Notes:

Physical Models

Radiation Model *(if not listed please specify in the space provided)*

None	Six flux	Discrete Transfer	Monte Carlo	Radiosity	Multi-ray 6 rays

Notes:

The Multiray radiation must be set up from the .inf file and can not be set up from the GUI.

Parameters

$a = 1.0$

For the multiray radiation model with 6 rays the following parameters were used to specify the 6 rays this should simulate the use of six rays.

The first 3 values form the unit vectors describing the direction of that ray and the last value indicates the weight of that ray direction (total sum of ray directions is 4π). In this case each ray has a weight of $4\pi/6$.

RAY-1 -1.0 0.0 0.0 2.09439507
RAY-2 1.0 0.0 0.0 2.09439507
RAY-3 0.0 1.0 0.0 2.09439507
RAY-4 0.0 -1.0 0.0 2.09439507

RAY-5 0.0 0.0 1.0 2.09439507
RAY-6 0.0 0.0 -1.0 2.09439507

It is assumed there is no scattering so s = 0.0.

Turbulence model *(if not listed please specify in the space provided)*

Laminar	k- ε	buoyancy modified k-ε	RNG	

Notes:

Turbulence Parameters*:

C_μ	σ_k	σ_ε	$C_{1\varepsilon}$	$C_{2\varepsilon}$	C_3
0.09	1.0	1.3	1.44	1.92	1.0

*If different parameters are being used please specify in the table above.

Combustion Model *(if not listed please specify in the space provided)*

none	Volumetric heat source	Mixed is burnt	Eddy break up
Magnussen soot model			

Combustion Parameters:

Compressibility

Incompressible	Boussinesq	Weakly compressible	Fully compressible

Compressibility Parameters:

Buoyancy

Yes	**No**

Gravity	

Material Properties

Material Name	Static Optical Fluid
Density	N/a
Viscosity	N/a
Conductivity	1E-100
Specific heat capacity	0.1

Initial Values

TEMPERATURE	0K

Boundary conditions

All the walls have an emissivity of 1.0

The walls at x=1, y = 0 and z = 0 have a temperature of 64.8052186K, all the other walls have a temperature of 0K.

Mesh

11 11 11
A uniformly meshed cube.
X, Y and Z
0.000000 0.090909 0.181818 0.272727 0.363636 0.454545 0.545455 0.636364 0.727273 0.818182 0.909091 1.000000

Model Definition files

Convergence

Please specify your convergence criteria including type of error estimator and tolerance value for each variable

Runtime

Results files/Archiving:

Document cross-reference:

User Guides, etc

Comments

6 ray Multiray radiation model gives similar performance to 6-flux model.

D.2 – Multray radiation model with 24 rays

Fire Safety Engineering Group
Maritime Greenwich Campus,
Cooper Building,
University of Greenwich,
King William Walk,
London SE10 9JH, UK.

Date : 14/1/2001

PART 2 – FREE TEST SETUP

Test case : **Radiation in 3 dimensional cavity 2000/1/5 – multiray radiation model with 24 rays.**
Document Version 1.0

PART 2 – FREE TEST SETUP

Case: **Radiation in 3 dimensional cavity 2000/1/5**

User details	
Run by: Angus Grandison Date: 14/1/2001 Phone no: 020 8331 7912 email: ga02@gre.ac.uk	Address: School of CMS University of Greenwich Maritime Greenwich Campus Old Royal Naval College Greenwich SE10 9LS. UNITED KINGDOM

Fire modelling Software *(if not listed please specify in the space provided)*

SMARTFIRE	CFX	PHOENICS			

Version/build number : v2.01 b369d
Date of release : 19/4/2000

Operating System *(if not listed please specify in the space provided)*

Windows 95/98/2000	Windows NT	Unix	Dos	

Version/build number v4.00 sp6

Machine *(if not listed please specify in the space provided)*

PC	Unix Workstation	

CPU: Intel Pentium III 733Mhz
Memory: 256Mb

Case description

This test case tests the fire modelling software's radiation model. The cavity is a unit cube (1m x 1m x 1m) with three walls with planes x=1, y=0 and z =0 set to an unit emissive power and the three other walls set to zero emissive power. All the walls are considered radiatively black have unit emissivity and the fluid has a unit absorption coefficient. Scattering is neglected. No fluid flow is considered

Required Results

The results should be supplied as graphs and as Excel97 worksheets

The results will be compared with analytically derived results[1,2] and between the codes.

The emissive power variation with x must be provided for the following locations
y= 0.1, 0.3, 0.5, 0.7, 0.9 on the planes z = 0.5, 0.9.

1) Larsen, M. E., "Exchange Factor Method and Alternative Zonal Formulation for Analysis of radiating enclosures containing paticipating media", PhD thesis, University of Texas, Austin, 1983.

2) Fiveland, W. A., "Three dimensional Radiative Heat-Transfer Solutions by Discrete-Ordinate Method", Journal of Thermophysics, Vol. 2, No. 4, October 1988, pp 309-316.

CFD set up

1D	2D	3D

Transient	Steady State

Differencing Schemes

Temporal:

Fully Implicit	Crank-Nicolson	Explicit	Exponential	

Spatial:

Hybrid	Central Difference	Upwind		

Notes:

Physical Models

Radiation Model *(if not listed please specify in the space provided)*

None	Six flux	Discrete Transfer	Monte Carlo	Radiosity	Multi-ray 6 rays

Notes:

If the fire modelling software does not possess the six-flux model, a discrete transfer model may be used in place of the six-flux model. If the discrete transfer model must be used instead of a six flux model then the discrete model must be made to emulate the behaviour of the six-flux model. This can be achieved by using 6 rays in the co-ordinate directions. If a radiation mesh needs to be specified, this should be identical to the flow mesh. If this is not possible, then at least the same number of cells in each direction must be specified. The details of the mesh must also be provided with your results.

Parameters

$a = 1.0$

For the multiray radiation model with 24 rays the following parameters were used to specify the rays this should give improved resolution over six rays/flux.

The first 3 values form the unit vectors describing the direction of that ray and the last value indicates the weight of that ray direction (total sum of ray directions is 4π).

RAY 1 0.2958759 0.2958759 0.9082483 0.5235987
RAY-2 0.9082483 0.2958759 0.2958759 0.5235987
RAY-3 0.2958759 0.9082483 0.2958759 0.5235987
RAY-4 -0.2958759 0.2958759 0.9082483 0.5235987
RAY-5 -0.9082483 0.2958759 0.2958759 0.5235987
RAY-6 -0.2958759 0.9082483 0.2958759 0.5235987
RAY-7 0.2958759 -0.2958759 0.9082483 0.5235987
RAY-8 0.9082483 -0.2958759 0.2958759 0.5235987
RAY-9 0.2958759 -0.9082483 0.2958759 0.5235987
RAY-10 0.2958759 0.2958759 -0.9082483 0.5235987
RAY-11 0.9082483 0.2958759 -0.2958759 0.5235987
RAY-12 0.2958759 0.9082483 -0.2958759 0.5235987
RAY-13 -0.2958759 -0.2958759 0.9082483 0.5235987
RAY-14 -0.9082483 -0.2958759 0.2958759 0.5235987
RAY-15 -0.2958759 -0.9082483 0.2958759 0.5235987
RAY-16 -0.2958759 0.2958759 -0.9082483 0.5235987
RAY-17 -0.9082483 0.2958759 -0.2958759 0.5235987
RAY-18 -0.2958759 0.9082483 -0.2958759 0.5235987
RAY-19 0.2958759 -0.2958759 -0.9082483 0.5235987
RAY-20 0.9082483 -0.2958759 -0.2958759 0.5235987
RAY-21 0.2958759 -0.9082483 -0.2958759 0.5235987
RAY-22 -0.2958759 -0.2958759 -0.9082483 0.5235987
RAY-23 -0.9082483 -0.2958759 -0.2958759 0.5235987
RAY-24 -0.2958759 -0.9082483 -0.2958759 0.5235987

It is assumed there is no scattering so s = 0.0.

Turbulence model *(if not listed please specify in the space provided)*

Laminar	k- ε	buoyancy modified k-ε	RNG	

Notes:

Turbulence Parameters*:

C_μ	σ_k	σ_ε	$C_{1\varepsilon}$	$C_{2\varepsilon}$	C_3
0.09	1.0	1.3	1.44	1.92	1.0

*If different parameters are being used please specify in the table above.

Combustion Model *(if not listed please specify in the space provided)*

none	Volumetric heat source	Mixed is burnt	Eddy break up
Magnussen soot model			

Combustion Parameters:

Compressibility

Incompressible	Boussinesq	Weakly compressible	Fully compressible

Compressibility Parameters:

Buoyancy

Yes	No

Gravity	

Material Properties

Material Name	Static Optical Fluid
Density	N/a
Viscosity	N/a
Conductivity	1E-100
Specific heat capacity	0.1

Initial Values

TEMPERATURE	0K

Boundary conditions

All the walls have an emissivity of 1.0

The walls at x=1, y = 0 and z = 0 have a temperature of 64.8052186K, all the other walls have a temperature of 0K.

Mesh

11 11 11
A uniformly meshed cube.
X, Y and Z
0.000000 0.090909 0.181818 0.272727 0.363636 0.454545 0.545455 0.636364 0.727273 0.818182 0.909091 1.000000

Model Definition files

Convergence

Please specify your convergence criteria including type of error estimator and tolerance value for each variable

Runtime

Results files/Archiving:

Document cross-reference:

User Guides, etc

Comments

D.3 - Multiray radiation model with 48 rays

Fire Safety Engineering Group
Maritime Greenwich Campus,
Cooper Building,
University of Greenwich,
King William Walk,
London SE10 9JH, UK.

Date : 14/1/2001

PART 2 – FREE TEST SETUP

Test case : **Radiation in 3 dimensional cavity 2000/1/5 – multiray radiation model with 48 rays.**
Document Version 1.0

PART 2 – FREE TEST SETUP

Case: **Radiation in 3 dimensional cavity 2000/1/5**

User details	
Run by: Angus Grandison Date: 14/1/2001 Phone no: 020 8331 7912 email: ga02@gre.ac.uk	Address: School of CMS University of Greenwich Maritime Greenwich Campus Old Royal Naval College Greenwich SE10 9LS. UNITED KINGDOM

Fire modelling Software (if not listed please specify in the space provided)

SMARTFIRE	CFX	PHOENICS			

Version/build number : v2.01 b369d
Date of release : 19/4/2000

Operating System *(if not listed please specify in the space provided)*

Windows 95/98/2000	Windows NT	Unix	Dos	

Version/build number v4.00 sp6

Machine *(if not listed please specify in the space provided)*

PC	Unix Workstation	

CPU: Intel Pentium III 733Mhz
Memory: 256Mb

Case description

This test case tests the fire modelling software's radiation model. The cavity is a unit cube (1m x 1m x 1m) with three walls with planes x=1, y=0 and z =0 set to an unit emissive power and the three other walls set to zero emissive power. All the walls are considered radiatively black have unit emissivity and the fluid has a unit absorption coefficient. Scattering is neglected. No fluid flow is considered

Required Results

The results should be supplied as graphs and as Excel97 worksheets

The results will be compared with analytically derived results[1,2] and between the codes.

The emissive power variation with x must be provided for the following locations
y= 0.1, 0.3, 0.5, 0.7, 0.9 on the planes z = 0.5, 0.9.

1) Larsen, M. E., "Exchange Factor Method and Alternative Zonal Formulation for Analysis of radiating enclosures containing paticipating media", PhD thesis, University of Texas, Austin, 1983.

2) Fiveland, W. A., "Three dimensional Radiative Heat-Transfer Solutions by Discrete-Ordinate Method", Journal of Thermophysics, Vol. 2, No. 4, October 1988, pp 309-316.

CFD set up

1 D	2D	3D

Transient	Steady State

Differencing Schemes

Temporal:

Fully Implicit	Crank-Nicolson	Explicit	Exponential	

Spatial:

Hybrid	Central Difference	Upwind		

Notes:

Physical Models

Radiation Model *(if not listed please specify in the space provided)*

None	Six flux	Discrete Transfer	Monte Carlo	Radiosity	Multi-ray 6 rays

Notes:

Parameters

$a = 1.0$

For the multiray radiation model with 48 rays the following parameters were used to specify the rays this should give improved resolution over six rays/flux and 24 rays model.

The first 3 values form the unit vectors describing the direction of that ray and the last value indicates the weight of that ray direction (total sum of ray directions is 4π).

RAY-1 0.1838670 0.1838670 0.9656013 0.1609517
RAY-2 0.6950514 0.1838670 0.6950514 0.3626469
RAY-3 0.9656013 0.1838670 0.1838670 0.1609517
RAY-4 0.1838670 0.6950514 0.6950514 0.3626469

RAY-5 0.6950514 0.6950514 0.1838670 0.3626469
RAY-6 0.1838670 0.9656013 0.1838670 0.1609517
RAY-7 -0.1838670 0.1838670 0.9656013 0.1609517
RAY-8 -0.6950514 0.1838670 0.6950514 0.3626469
RAY-9 -0.9656013 0.1838670 0.1838670 0.1609517
RAY-10 -0.1838670 0.6950514 0.6950514 0.3626469
RAY-11 -0.6950514 0.6950514 0.1838670 0.3626469
RAY-12 -0.1838670 0.9656013 0.1838670 0.1609517
RAY-13 0.1838670 -0.1838670 0.9656013 0.1609517
RAY-14 0.6950514 -0.1838670 0.6950514 0.3626469
RAY-15 0.9656013 -0.1838670 0.1838670 0.1609517
RAY-16 0.1838670 -0.6950514 0.6950514 0.3626469
RAY-17 0.6950514 -0.6950514 0.1838670 0.3626469
RAY-18 0.1838670 -0.9656013 0.1838670 0.1609517
RAY-19 0.1838670 0.1838670 -0.9656013 0.1609517
RAY-20 0.6950514 0.1838670 -0.6950514 0.3626469
RAY-21 0.9656013 0.1838670 -0.1838670 0.1609517
RAY-22 0.1838670 0.6950514 -0.6950514 0.3626469
RAY-23 0.6950514 0.6950514 -0.1838670 0.3626469
RAY-24 0.1838670 0.9656013 -0.1838670 0.1609517
RAY-25 -0.1838670 -0.1838670 0.9656013 0.1609517
RAY-26 -0.6950514 -0.1838670 0.6950514 0.3626469
RAY-27 -0.9656013 -0.1838670 0.1838670 0.1609517
RAY-28 -0.1838670 -0.6950514 0.6950514 0.3626469
RAY-29 -0.6950514 -0.6950514 0.1838670 0.3626469
RAY-30 -0.1838670 -0.9656013 0.1838670 0.1609517
RAY-31 -0.1838670 0.1838670 -0.9656013 0.1609517
RAY-32 -0.6950514 0.1838670 -0.6950514 0.3626469
RAY-33 -0.9656013 0.1838670 -0.1838670 0.1609517
RAY-34 -0.1838670 0.6950514 -0.6950514 0.3626469
RAY-35 -0.6950514 0.6950514 -0.1838670 0.3626469
RAY-36 -0.1838670 0.9656013 -0.1838670 0.1609517
RAY-37 0.1838670 -0.1838670 -0.9656013 0.1609517
RAY-38 0.6950514 -0.1838670 -0.6950514 0.3626469
RAY-39 0.9656013 -0.1838670 -0.1838670 0.1609517
RAY-40 0.1838670 -0.6950514 -0.6950514 0.3626469
RAY-41 0.6950514 -0.6950514 -0.1838670 0.3626469
RAY-42 0.1838670 -0.9656013 -0.1838670 0.1609517
RAY-43 -0.1838670 -0.1838670 -0.9656013 0.1609517
RAY-44 -0.6950514 -0.1838670 -0.6950514 0.3626469
RAY-45 -0.9656013 -0.1838670 -0.1838670 0.1609517
RAY-46 -0.1838670 -0.6950514 -0.6950514 0.3626469
RAY-47 -0.6950514 -0.6950514 -0.1838670 0.3626469
RAY-48 -0.1838670 -0.9656013 -0.1838670 0.1609517

It is assumed there is no scattering so s = 0.0.

Turbulence model *(if not listed please specify in the space provided)*

Laminar	k- ε	buoyancy modified k-ε	RNG	

Notes:

Turbulence Parameters*:

C_μ	σ_k	σ_ε	$C_{1\varepsilon}$	$C_{2\varepsilon}$	C_3
0.09	1.0	1.3	1.44	1.92	1.0

*If different parameters are being used please specify in the table above.

Combustion Model *(if not listed please specify in the space provided)*

none	Volumetric heat source	Mixed is burnt	Eddy break up
Magnussen soot model			

Combustion Parameters:

Compressibility

Incompressible	Boussinesq	Weakly compressible	Fully compressible

Compressibility Parameters:

Buoyancy

Yes	No

Gravity	

Material Properties

Material Name	Static Optical Fluid
Density	N/a
Viscosity	N/a
Conductivity	1E-100
Specific heat capacity	0.1

Initial Values

TEMPERATURE	0K

Boundary conditions

All the walls have an emissivity of 1.0

The walls at x=1, y = 0 and z = 0 have a temperature of 64.8052186K, all the other walls have a temperature of 0K.

Mesh

11 11 11
A uniformly meshed cube.
X, Y and Z
0.000000 0.090909 0.181818 0.272727 0.363636 0.454545 0.545455 0.636364 0.727273 0.818182 0.909091 1.000000

Model Definition files

Convergence

Please specify your convergence criteria including type of error estimator and tolerance value for each variable

Runtime

Results files/Archiving:

Document cross-reference:

User Guides, etc

Comments

D.4 – Using refined mesh 41 x 41 x 41

Fire Safety Engineering Group
Maritime Greenwich Campus,
Cooper Building,
University of Greenwich,
King William Walk,
London SE10 9JH, UK.

Date : 14/1/2001

PART 2 – FREE TEST SETUP

Test case : **Radiation in 3 dimensional cavity 2000/1/5 using large mesh 41 x 41 x 41**
Document Version 1.0

PART 2 – FREE TEST SETUP

Case: **Radiation in 3 dimensional cavity 2000/1/5**

User details	
Run by: Angus Grandison Date: 14/1/2001 Phone no: 020 8331 7912 email: ga02@gre.ac.uk	Address: School of CMS University of Greenwich Maritime Greenwich Campus Old Royal Naval College Greenwich SE10 9LS. UNITED KINGDOM

Fire modelling Software (if not listed please specify in the space provided)

SMARTFIRE	CFX	PHOENICS			

Version/build number:v2.01 b369d
Date of release:

Operating System *(if not listed please specify in the space provided)*

Windows 95/98/2000	Windows NT	Unix	Dos	

Version/build number___v4.00 sp6_____

Machine *(if not listed please specify in the space provided)*

PC	Unix Workstation	

CPU: Intel Pentium III 733Mhz
Memory: 256Mb

Case description

This test case tests the fire modelling software's radiation model. The cavity is a unit cube (1m x 1m x 1m) with three walls with planes x=1, y=0 and z =0 set to an unit emissive power and the three other walls set to zero emissive power. All the walls are considered radiatively black have unit emissivity and the fluid has a unit absorption coefficient. Scattering is neglected. No fluid flow is considered

Required Results

The results should be supplied as graphs and as Excel97 worksheets

The results will be compared with analytically derived results[1,2] and between the codes.

The emissive power variation with x must be provided for the following locations
y= 0.1, 0.3, 0.5, 0.7, 0.9 on the planes z = 0.5, 0.9.

1) Larsen, M. E., "Exchange Factor Method and Alternative Zonal Formulation for Analysis of radiating enclosures containing paticipating media", PhD thesis, University of Texas, Austin, 1983.

2) Fiveland, W. A., "Three dimensional Radiative Heat-Transfer Solutions by Discrete-Ordinate Method", Journal of Thermophysics, Vol. 2, No. 4, October 1988, pp 309-316.

CFD set up

1 D	2D	3D

Transient	Steady State

Differencing Schemes

Temporal:

Fully Implicit	Crank-Nicolson	Explicit	Exponential	

Spatial:

Hybrid	Central Difference	Upwind		

Notes:

Physical Models

Radiation Model *(if not listed please specify in the space provided)*

None	Six flux	Discrete Transfer	Monte Carlo	Radiosity	Multi-ray 6 rays

Notes:

Parameters

$a = 1.0$

It is assumed there is no scattering so s = 0.0.

Turbulence model *(if not listed please specify in the space provided)*

Laminar	k- ε	buoyancy modified k-ε	RNG	

Notes:

Turbulence Parameters*:

C_μ	σ_k	σ_ε	$C_{1\varepsilon}$	$C_{2\varepsilon}$	C_3
0.09	1.0	1.3	1.44	1.92	1.0

*If different parameters are being used please specify in the table above.

Combustion Model *(if not listed please specify in the space provided)*

none	Volumetric heat source	Mixed is burnt	Eddy break up
Magnussen soot model			

Combustion Parameters:

Compressibility

Incompressible	Boussinesq	Weakly compressible	Fully compressible

Compressibility Parameters:

External Pressure 1.01325e+05

Buoyancy

Yes	No

Gravity	

Material Properties

Material Name	Optical Fluid
Density	N/a
Viscosity	N/a
Conductivity	1E-100
Specific heat capacity	0.1

Initial Values

TEMPERATURE	0K

Boundary conditions

All the walls have an emissivity of 1.0

The walls at x=1, y = 0 and z = 0 have a temperature of 64.8052186K, all the other walls have a temperature of 0K.

Mesh

41 41 41
A uniformly meshed cube.
X, Y and Z
0.0 0.024390 0.048780 0.073171 0.097561 0.121951 0.146341 0.170732 0.195122
0.219512 0.243902 0.268293 0.292683 0.317073 0.341463 0.365854 0.390244 0.414634
0.439024 0.463415 0.487805 0.512195 0.536585 0.560976 0.585366 0.609756
0.634146 0.658537 0.682927 0.707317 0.731707 0.756098 0.780488 0.804878 0.829268
0.853659 0.878049 0.902439 0.926829 0.951219 0.975610 1.000000

Model Definition files

Convergence

Please specify your convergence criteria including type of error estimator and tolerance value for each variable

Runtime

Results files/Archiving:

Document cross-reference:

User Guides, etc

Comments

Refined mesh to see if 6 flux model result is improved.

D.5 – Steckler room with improved physical properties and boundary conditions.

Fire Safety Engineering Group
Maritime Greenwich Campus,
Cooper Building,
University of Greenwich,
King William Walk,
London SE10 9JH, UK.

Date : 4/2/2000

PART 2 – FREE TEST SETUP

Test case : **Steckler Room Fire – volumetric heat – 2000/2/1**
Document Version 1.0

PART 2 – FREE TEST SETUP

Case: **Steckler Room Fire – volumetric heat – 2000/2/1**

User details	
Run by: Angus Grandison Date: Phone no: 0208 331 7912 email: ga02@gre.ac.uk	Address: School of CMS University of Greenwich Maritime Greenwich Campus Old Royal Naval College Greenwich SE10 9LS. UNITED KINGDOM

Fire modelling Software

SMARTFIRE	CFX	PHOENICS			

Version/build number: v2.01 b369d
Date of release: 14/4/2000

Operating System

Windows 95/98/2000	Windows NT	Unix	Dos	

Version/build number: v4.00 sp6

Machine

PC	Unix Workstation	

CPU: Intel Pentium III 733 Mhz
Memory: 256 Mb

Case description

The experimental data obtained from Steckler's fire tests* which have been used as part of the validation process for both zone and field fire models. The data represents non-spreading fires in small compartments. The non-spreading fire was created using a centrally located (position A in Figure 5-5) 62.9kW methane burner with a diameter of 0.3m and a height of 0.3m. The experiments were conducted by Steckler et al. in a compartment measuring 2.8m × 2.8m in plane and 2.18m in height (see Figure 5-5) with a doorway centrally located in one of the walls measuring 0.74m wide by 1.83m high. The walls and ceiling were 0.1m thick and they were covered with a ceramic fibre insulation board to establish near steady state conditions within 30 minutes.

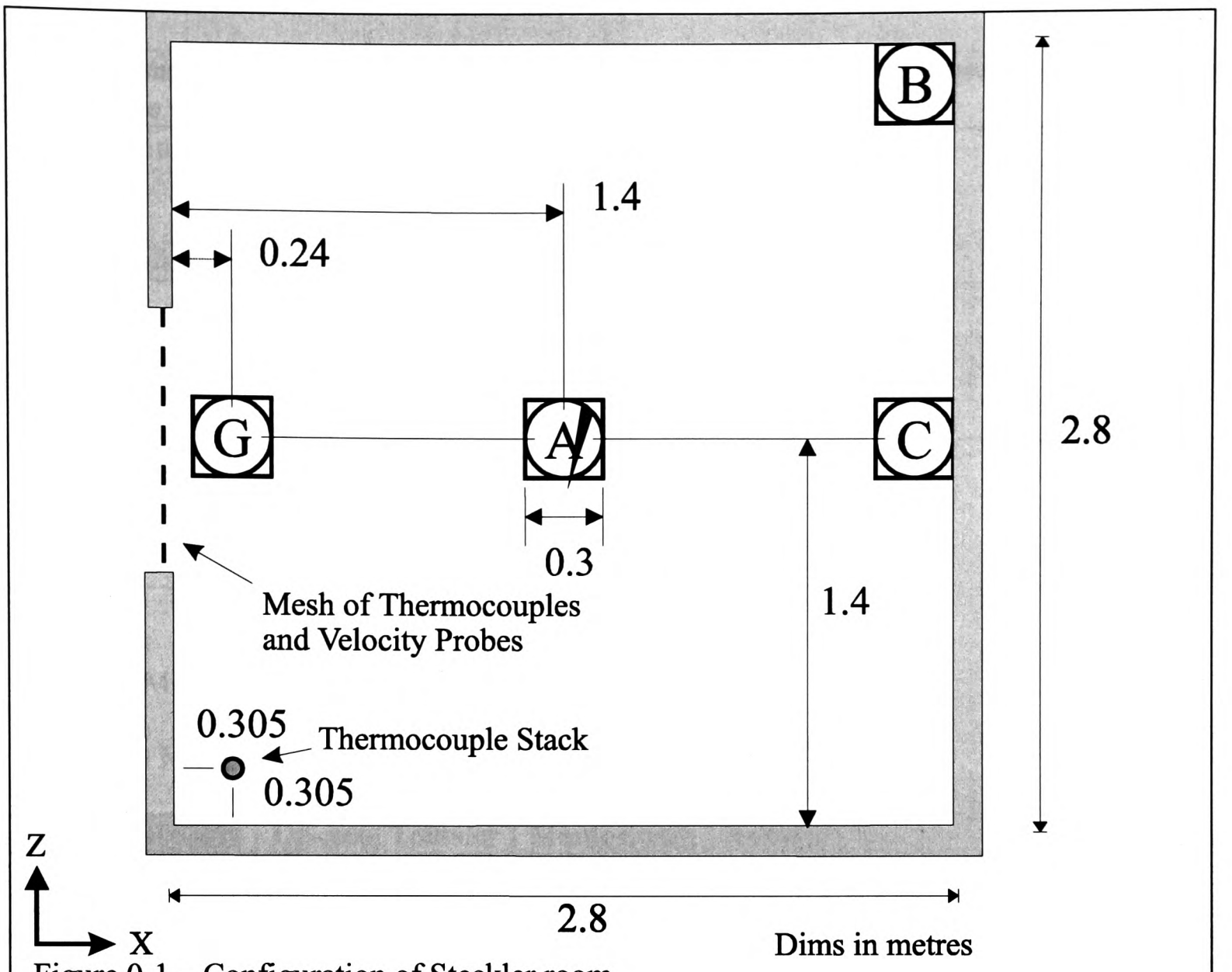


Figure 0-1 – Configuration of Steckler room

The door measures 0.74m wide and 1.83m high and is centrally located in one of the walls.

*Steckler, K.D, Quintiere, J.G and Rinkinen, W.J.[1982], "Flow induced by fire in a compartment", NBSIR 82-2520, National Bureau of Standards.

Required Results
The results should be supplied as graphs and as Excel97 worksheets
Vertical Corner Stack temperatures at 0.305 from the front and side walls.
Vertical Doorway temperature profile in the middle of the doorway.
Horizontal velocity profile for a vertical stack in the middle of the doorway.

These should all be plotted with height of the variable on the y-axis and the variable value (temperature or horizontal velocity) on the x-axis.

CFD set up

1D	2D	3D

Transient	Steady State

The case needs to be run for 200s using 1s time steps. This effectively gives a steady state result. The walls are assumed to be made of common brick.

Differencing Schemes

Temporal:

Fully Implicit	Crank-Nicolson	Explicit	Exponential	

Spatial:

Hybrid	Central Difference	Upwind		

Notes:

Physical Models

Radiation Model *(if not listed please specify in the space provided)*

None	**Six flux**	Discrete Transfer	Monte Carlo	Radiosity	

Notes:

Parameters

The absorption coefficient (a) assumed the following form:

$a = 0.01$, if $T < 323$K;

$a = 0.01 + (3.49/377)(T–323)$, if $323 <= T < 700$;

$a = 3.5 + (3.5/700)(T–700)$, if $T > 700$.

The walls are assumed to have an emissivity of 0.8.

It is assumed there is no scattering so s = 0.0.

Turbulence model *(if not listed please specify in the space provided)*

Laminar	k- ε	**buoyancy modified k-ε**	RNG	

AAA

Notes:

Turbulence Parameters*:

C_μ	σ_k	σ_ε	$C_{1\varepsilon}$	$C_{2\varepsilon}$	C_3
0.09	1.0	1.3	1.44	1.92	1.0

*If different parameters are being used please specify in the table above.

Combustion Model *(if not listed please specify in the space provided)*

none	**Volumetric heat source**	Mixed is burnt	Eddy break up
Magnussen soot model			

Combustion Parameters:

The volumetric heat source is assumed to be centrally located within the room with dimension of 0.3m x 0.3m x 0.3m with a total heat source of 62.9kW.

Compressibility

Incompressible	Boussinesq	Weakly compressible	**Fully compressible**

Compressibility Parameters:

External Pressure 1.01325e+05 Pa

Buoyancy

Yes	No

Material Properties

Material Name	Air
Density	Determined by compressibility (Ideal Gas Law) Molecular Weight of air is 29.35
Viscosity (dynamic)	Laminar 1.798e-005 kg/m.s
Conductivity	0.02622 W/m.K
Specific heat capacity	1007.0 J/kg.K

Material Name	Common brick
Density	1600 Kg/m^3
Viscosity (dynamic)	Laminar 1.0e+10 kg/m.s
Conductivity	0.69 W/m.K
Specific heat capacity	840J/kg.K

Initial Values

U-VELOCITY	0.0
V-VELOCITY	0.0
W-VELOCITY	0.0
PRESSURE	0.0

TEMPERATURE	293.75
KINETIC ENERGY	0.01
DISSIPATION RATE	0.01

Boundary conditions

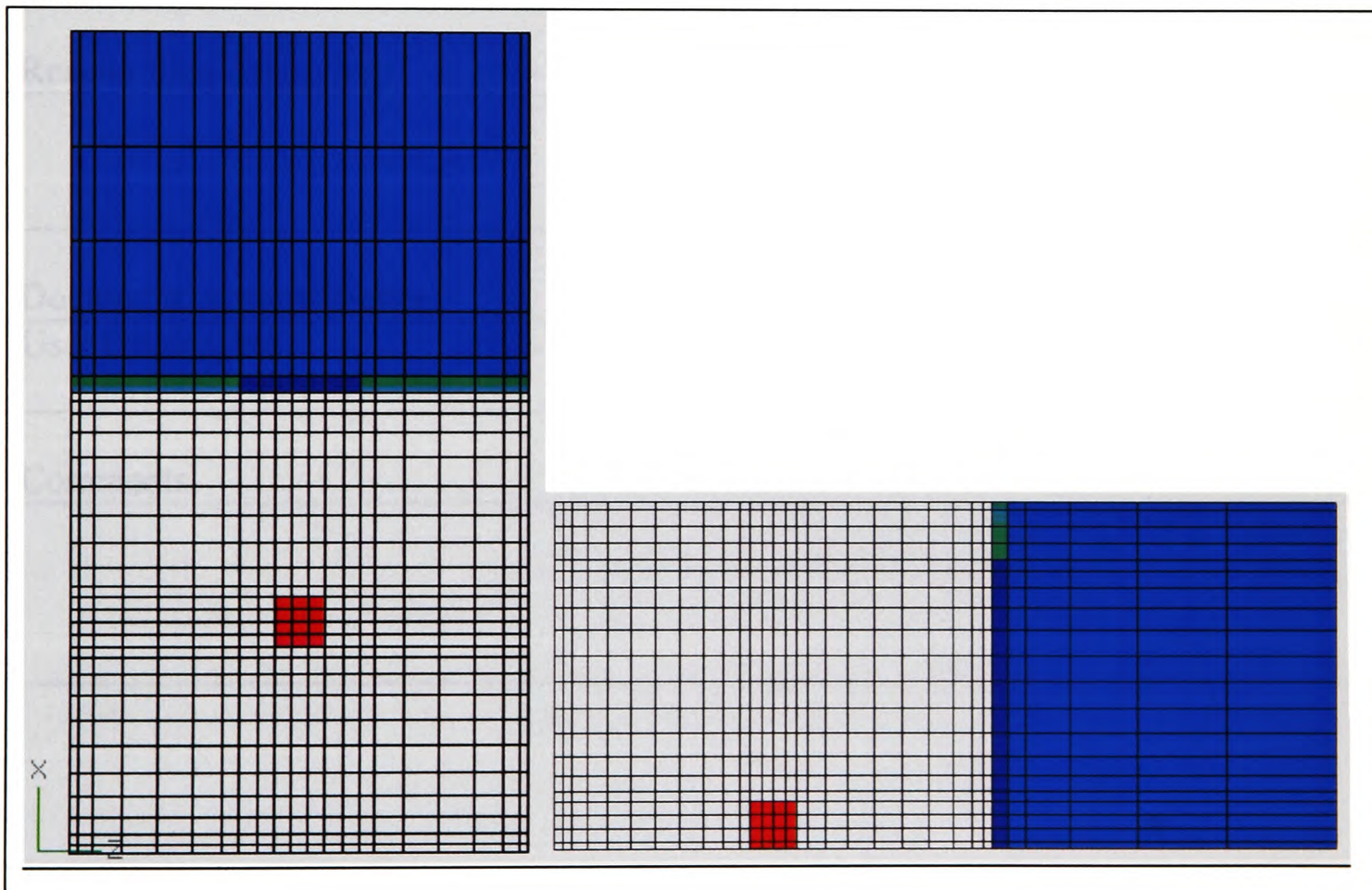
All walls are assumed to be made of common brick material. The emissivity of the walls is 0.8.

The doorway measures 0.74m wide and 1.83m high and is centrally located in one of the walls. This doorway is modelled using three solid conducting obstructions (again made from the brick material) to create the walls around the doorway. An extended region for this doorway is required to ensure that the airflow in the door is correctly modelled.

On the extended region all the boundary patches are fixed pressure (outlet) boundaries set to 0.0 Pa apart from the floor which is adiabatic.

The fire is modelled as a volumetric heat source which is assumed to be centrally located within the room on the floor with dimensions of 0.3m x 0.3m x 0.3m with a total heat source of 62.9kW.

Mesh




```
31 20 21
X
0.0 0.056 0.1157 0.2137 0.3407 0.4872 0.6438 0.8011 0.9498
1.0805 1.1837 1.25 1.325 1.4 1.475 1.55 1.6163 1.7303
1.8787 2.0481 2.2252 2.3966 2.5491 2.6694 2.744 2.8 2.9
3.0115 3.2949 3.7272 4.2979 5.0
```

```
Y
0.0  0.044  0.1293  0.2147  0.3  0.3638  0.4612  0.5858  0.7308
0.8896  1.0555  1.222  1.3824  1.53  1.6582  1.7605  1.83  1.9387
2.0473  2.156  2.2
Z
0.0  0.056  0.142  0.3151  0.5328  0.7527  0.9326  1.03  1.14  1.25
1.35  1.45  1.55  1.66  1.77  1.856  2.0291  2.2468  2.4667  2.6466
2.744  2.8
```

Model Definition files

Convergence

Please specify your convergence criteria including type of error estimator and tolerance value for each variable

Runtime

Results files/Archiving:

Document cross-reference:

User Guides, etc

Comments

D6 – Improved Steckler case with multiray radiation model (24 rays).

Fire Safety Engineering Group
Maritime Greenwich Campus,
Cooper Building,
University of Greenwich,
King William Walk,
London SE10 9JH, UK.

Date : 4/2/2000

PART 2 – FREE TEST SETUP

Test case : **Steckler Room Fire – volumetric heat – 2000/2/1 using multiray radiaion model**
Document Version 1.0

PART 2 – FREE TEST SETUP

Case: **Steckler Room Fire – volumetric heat – 2000/2/1 using multiray radiation model**

User details	
Run by: Angus Grandison Date: Phone no: 0208 331 7912 email: ga02@gre.ac.uk	Address: School of CMS University of Greenwich Maritime Greenwich Campus Old Royal Naval College Greenwich SE10 9LS. UNITED KINGDOM

Fire modelling Software

SMARTFIRE	CFX	PHOENICS			

Version/build number: v2.01 b369d
Date of release: 14/4/2000

Operating System

Windows 95/98/2000	Windows NT	Unix	Dos	

Version/build number: v4.00 sp6

Machine

PC	Unix Workstation	

CPU: Intel Pentium III 733 Mhz
Memory: 256 Mb

Case description

The experimental data obtained from Steckler's fire tests* which have been used as part of the validation process for both zone and field fire models. The data represents non-spreading fires in small compartments. The non-spreading fire was created using a centrally located (position A in Figure 5-5) 62.9kW methane burner with a diameter of 0.3m and a height of 0.3m. The experiments were conducted by Steckler et al. in a compartment measuring 2.8m × 2.8m in plane and 2.18m in height (see Figure 5-5) with a doorway centrally located in one of the walls measuring 0.74m wide by 1.83m high. The walls and ceiling were 0.1m thick and they were covered with a ceramic fibre insulation board to establish near steady state conditions within 30 minutes.

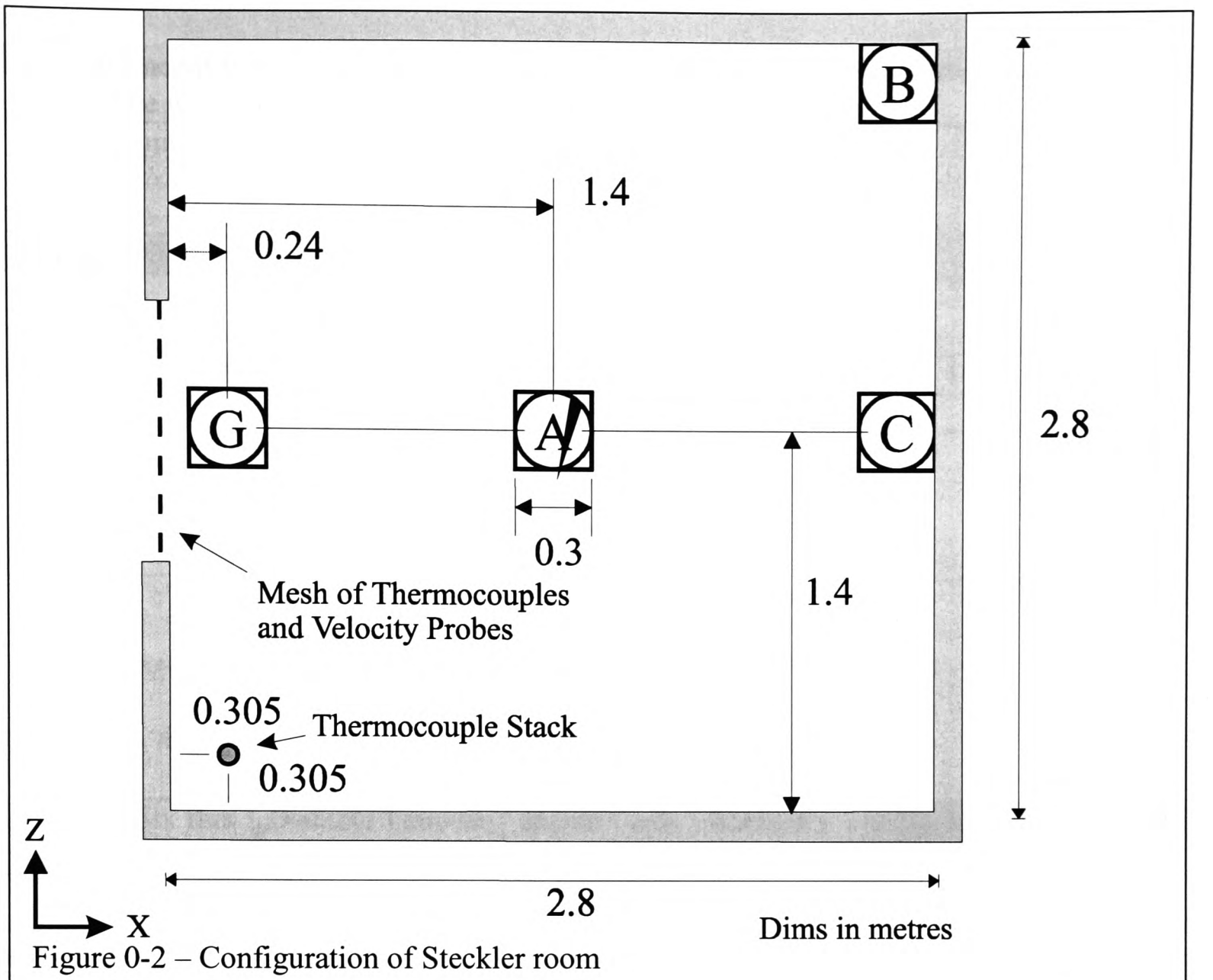


Figure 0-2 – Configuration of Steckler room

The door measures 0.74m wide and 1.83m high and is centrally located in one of the walls.

*Steckler, K.D, Quintiere, J.G and Rinkinen, W.J.[1982], "Flow induced by fire in a compartment", NBSIR 82-2520, National Bureau of Standards.

Required Results

The results should be supplied as graphs and as Excel97 worksheets

Vertical Corner Stack temperatures at 0.305 from the front and side walls.
Vertical Doorway temperature profile in the middle of the doorway.
Horizontal velocity profile for a vertical stack in the middle of the doorway.

These should all be plotted with height of the variable on the y-axis and the variable value (temperature or horizontal velocity) on the x-axis.

CFD set up

1D	2D	3D

Transient	Steady State

The case needs to be run for 200s using 1s time steps. This effectively gives a steady state result. The walls are assumed to be made of common brick.

Differencing Schemes

Temporal:

Fully Implicit	Crank-Nicolson	Explicit	Exponential	

Spatial:

Hybrid	Central Difference	Upwind		

Notes:

Physical Models

Radiation Model *(if not listed please specify in the space provided)*

None	Six flux	Discrete Transfer	Monte Carlo	Radiosity	Multi-ray radiation model

Notes:

Parameters

The absorption coefficient (a) assumed the following form:

$a = 0.01$, if $T < 323$K;

$a = 0.01 + (3.49/377)(T–323)$, if $323 <= T < 700$;

$a = 3.5 + (3.5/700)(T–700)$, if $T > 700$.

The walls are assumed to have an emissivity of 0.8.

For the multiray radiation model with 24 rays the following parameters were used to specify the rays this should give improved resolution over six rays/flux.

The first 3 values form the unit vectors describing the direction of that ray and the last value indicates the weight of that ray direction (total sum of ray directions is 4π).

RAY 1 0.2958759 0.2958759 0.9082483 0.5235987
RAY-2 0.9082483 0.2958759 0.2958759 0.5235987
RAY-3 0.2958759 0.9082483 0.2958759 0.5235987
RAY-4 -0.2958759 0.2958759 0.9082483 0.5235987

RAY-5 -0.9082483 0.2958759 0.2958759 0.5235987
RAY-6 -0.2958759 0.9082483 0.2958759 0.5235987
RAY-7 0.2958759 -0.2958759 0.9082483 0.5235987
RAY-8 0.9082483 -0.2958759 0.2958759 0.5235987
RAY-9 0.2958759 -0.9082483 0.2958759 0.5235987
RAY-10 0.2958759 0.2958759 -0.9082483 0.5235987
RAY-11 0.9082483 0.2958759 -0.2958759 0.5235987
RAY-12 0.2958759 0.9082483 -0.2958759 0.5235987
RAY-13 -0.2958759 -0.2958759 0.9082483 0.5235987
RAY-14 -0.9082483 -0.2958759 0.2958759 0.5235987
RAY-15 -0.2958759 -0.9082483 0.2958759 0.5235987
RAY-16 -0.2958759 0.2958759 -0.9082483 0.5235987
RAY-17 -0.9082483 0.2958759 -0.2958759 0.5235987
RAY-18 -0.2958759 0.9082483 -0.2958759 0.5235987
RAY-19 0.2958759 -0.2958759 -0.9082483 0.5235987
RAY-20 0.9082483 -0.2958759 -0.2958759 0.5235987
RAY-21 0.2958759 -0.9082483 -0.2958759 0.5235987
RAY-22 -0.2958759 -0.2958759 -0.9082483 0.5235987
RAY-23 -0.9082483 -0.2958759 -0.2958759 0.5235987
RAY-24 -0.2958759 -0.9082483 -0.2958759 0.5235987

It is assumed there is no scattering so s = 0.0.

Turbulence model *(if not listed please specify in the space provided)*

Laminar	k- ε	**buoyancy modified k-ε**	RNG	

AAA

Notes:

Turbulence Parameters*:

C_μ	σ_k	σ_ε	$C_{1\varepsilon}$	$C_{2\varepsilon}$	C_3
0.09	1.0	1.3	1.44	1.92	1.0

*If different parameters are being used please specify in the table above.

Combustion Model *(if not listed please specify in the space provided)*

none	**Volumetric heat source**	Mixed is burnt	Eddy break up
Magnussen soot model			

Combustion Parameters:

The volumetric heat source is assumed to be centrally located within the room with dimension of 0.3m x 0.3m x 0.3m with a total heat source of 62.9kW.

Compressibility

Incompressible	Boussinesq	Weakly compressible	**Fully compressible**

Compressibility Parameters:

External Pressure 1.01325e+05 Pa

Buoyancy

Yes	No

Gravity	-9.81m/s in the v-velocity direction.

Material Properties

Material Name	Air
Density	Determined by compressibility (Ideal Gas Law) Molecular Weight of air is 29.35
Viscosity (dynamic)	Laminar 1.798e-005 kg/m.s
Conductivity	0.02622 W/m.K
Specific heat capacity	1007.0 J/kg.K

Material Name	Common brick
Density	1600 Kg/m^3
Viscosity (dynamic)	Laminar 1.0e+10 kg/m.s
Conductivity	0.69 W/m.K
Specific heat capacity	840J/kg.K

Initial Values

U-VELOCITY	0.0
V-VELOCITY	0.0
W-VELOCITY	0.0
PRESSURE	0.0
TEMPERATURE	293.75
KINETIC ENERGY	0.01
DISSIPATION RATE	0.01

Boundary conditions

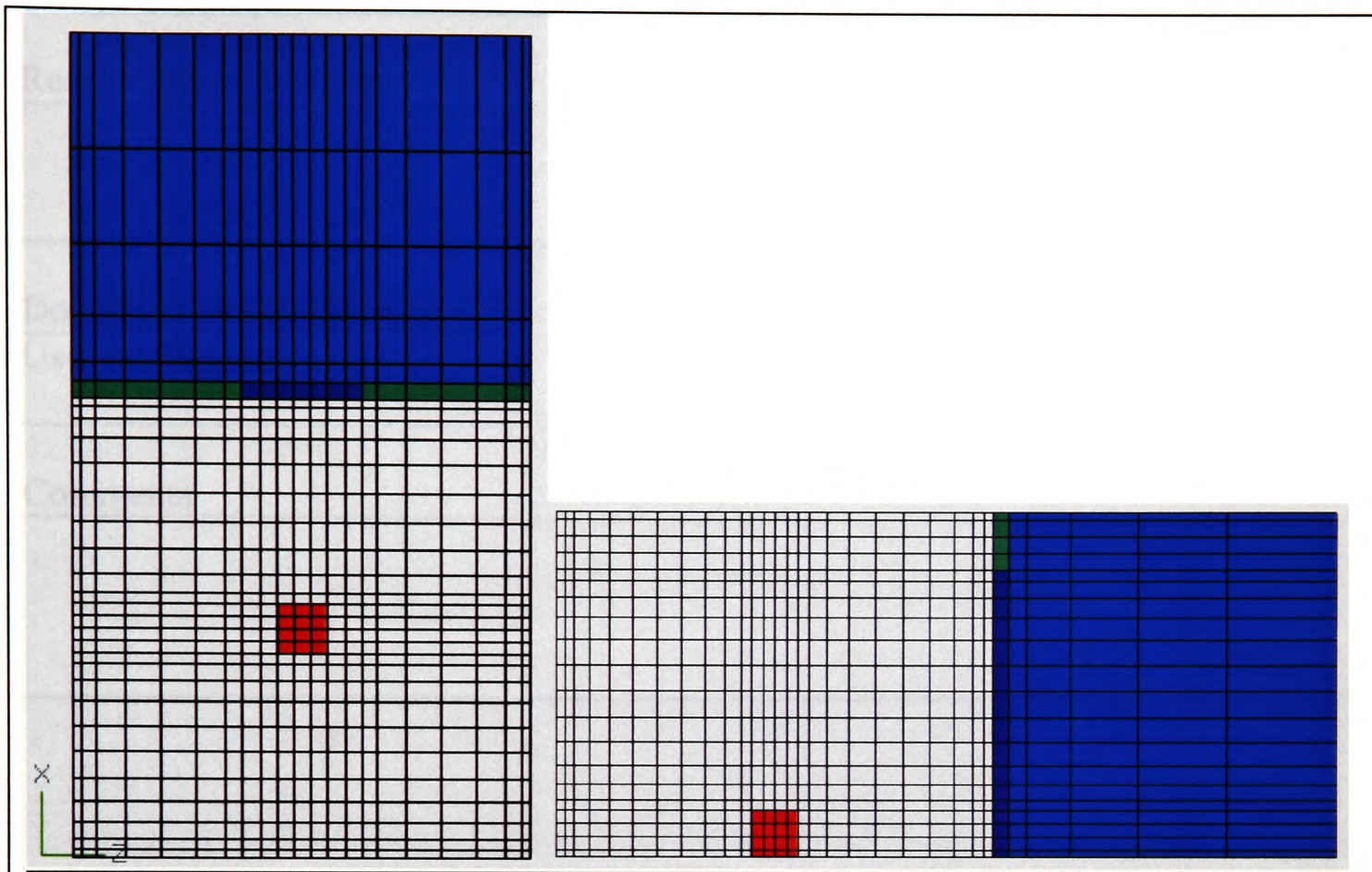
All walls are assumed to be made of common brick material. The emissivity of the walls is 0.8.

The doorway measures 0.74m wide and 1.83m high and is centrally located in one of the walls. This doorway is modelled using three solid conducting obstructions (again made from the brick material) to create the walls around the doorway. An extended region for this doorway is required to ensure that the airflow in the door is correctly modelled.

On the extended region all the boundary patches are fixed pressure (outlet) boundaries set to 0.0 Pa apart from the floor which is adiabatic.

The fire is modelled as a volumetric heat source which is assumed to be centrally located within the room on the floor with dimensions of 0.3m x 0.3m x 0.3m with a total heat source of 62.9kW.

Mesh




```
31 20 21
X
0.0 0.056 0.1157 0.2137 0.3407 0.4872 0.6438 0.8011 0.9498
1.0805 1.1837 1.25 1.325 1.4 1.475 1.55 1.6163 1.7303
1.8787 2.0481 2.2252 2.3966 2.5491 2.6694 2.744 2.8 2.9
3.0115 3.2949 3.7272 4.2979 5.0
Y
0.0 0.044 0.1293 0.2147 0.3 0.3638 0.4612 0.5858 0.7308
0.8896 1.0555 1.222 1.3824 1.53 1.6582 1.7605 1.83 1.9387
2.0473 2.156 2.2
Z
0.0 0.056 0.142 0.3151 0.5328 0.7527 0.9326 1.03 1.14 1.25
1.35 1.45 1.55 1.66 1.77 1.856 2.0291 2.2468 2.4667 2.6466
2.744 2.8
```

Model Definition files

Convergence

Please specify your convergence criteria including type of error estimator and tolerance

value for each variable

Runtime

Results files/Archiving:

Document cross-reference:

User Guides, etc

Comments

D7 – Improved Steckler case with refined mesh.

Fire Safety Engineering Group
Maritime Greenwich Campus,
Cooper Building,
University of Greenwich,
King William Walk,
London SE10 9JH, UK.

Date : 4/2/2000

PART 2 – FREE TEST SETUP

Test case : **Steckler Room Fire – volumetric heat – 2000/2/1 using a refined mesh**
Document Version 1.0

PART 2 – FREE TEST SETUP

Case: **Steckler Room Fire – volumetric heat – 2000/2/1 using a refined mesh**

User details	
Run by: Angus Grandison Date: Phone no: 0208 331 7912 email: ga02@gre.ac.uk	Address: School of CMS University of Greenwich Maritime Greenwich Campus Old Royal Naval College Greenwich SE10 9LS. UNITED KINGDOM

Fire modelling Software

SMARTFIRE	CFX	PHOENICS			

Version/build number: v2.01 b369d
Date of release: 14/4/2000

Operating System

Windows 95/98/2000	Windows NT	Unix	Dos	

Version/build number: v4.00 sp6

Machine

PC	Unix Workstation	

CPU: Intel Pentium III 733 Mhz
Memory: 256 Mb

Case description

The experimental data obtained from Steckler's fire tests* which have been used as part of the validation process for both zone and field fire models. The data represents non-spreading fires in small compartments. The non-spreading fire was created using a centrally located (position A in Figure 5-5) 62.9kW methane burner with a diameter of 0.3m and a height of 0.3m. The experiments were conducted by Steckler et al. in a compartment measuring 2.8m × 2.8m in plane and 2.18m in height (see Figure 5-5) with a doorway centrally located in one of the walls measuring 0.74m wide by 1.83m high. The walls and ceiling were 0.1m thick and they were covered with a ceramic fibre insulation board to establish near steady state conditions within 30 minutes.

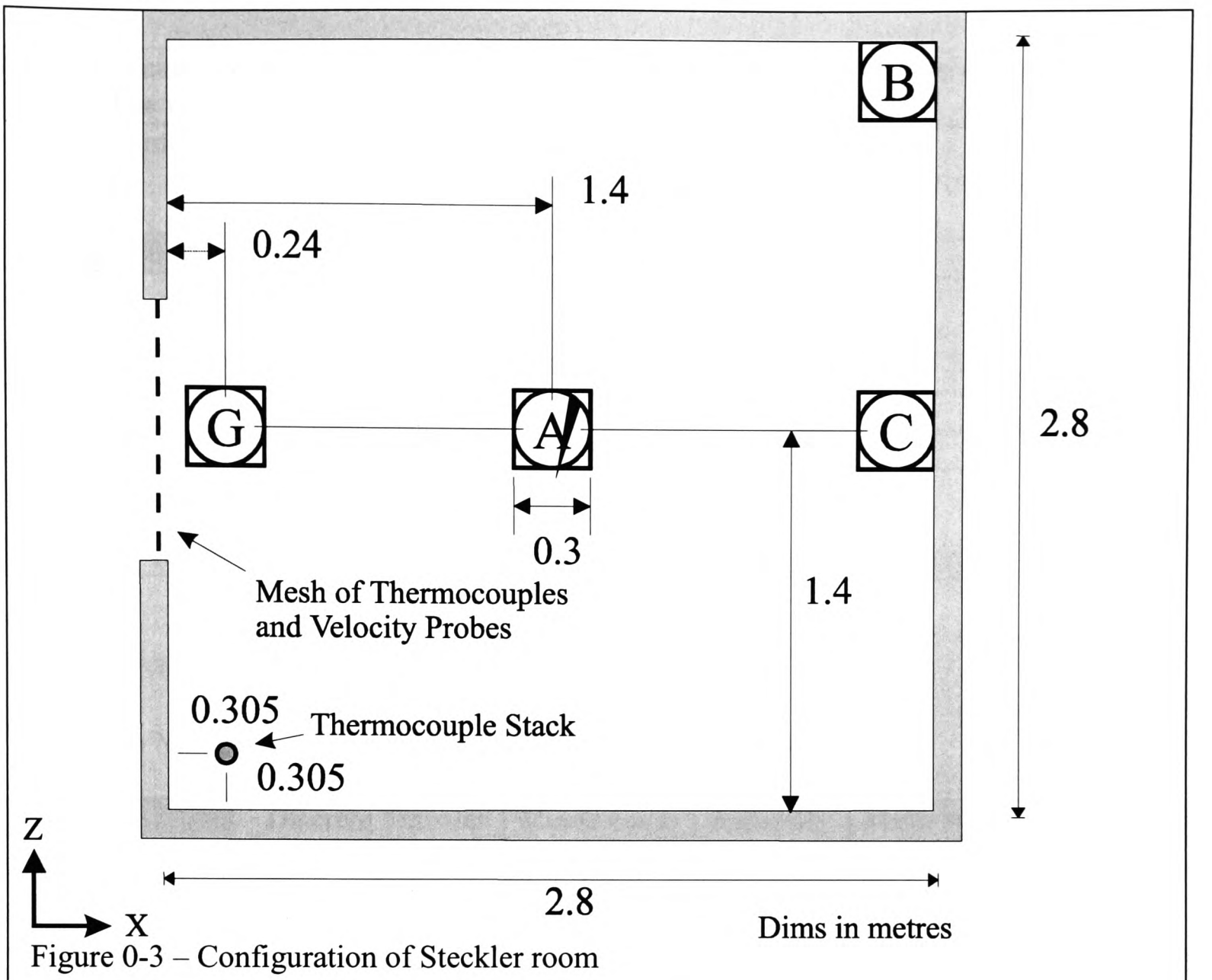


Figure 0-3 – Configuration of Steckler room

The door measures 0.74m wide and 1.83m high and is centrally located in one of the walls.

*Steckler, K.D, Quintiere, J.G and Rinkinen, W.J.[1982], "Flow induced by fire in a compartment", NBSIR 82-2520, National Bureau of Standards.

Required Results
The results should be supplied as graphs and as Excel97 worksheets
Vertical Corner Stack temperatures at 0.305 from the front and side walls.
Vertical Doorway temperature profile in the middle of the doorway.
Horizontal velocity profile for a vertical stack in the middle of the doorway.

These should all be plotted with height of the variable on the y-axis and the variable value (temperature or horizontal velocity) on the x-axis.

CFD set up

1D	2D	3D

Transient	Steady State

The case needs to be run for 200s using 1s time steps. This effectively gives a steady state result. The walls are assumed to be made of common brick.

Differencing Schemes

Temporal:

Fully Implicit	Crank-Nicolson	Explicit	Exponential	

Spatial:

Hybrid	Central Difference	Upwind		

Notes:

Physical Models

Radiation Model *(if not listed please specify in the space provided)*

None	Six flux	Discrete Transfer	Monte Carlo	Radiosity	Multi-ray radiation model

Notes:

If the fire modelling software does not possess the six-flux model, a discrete transfer model may be used in place of the six-flux model. If the discrete transfer model must be used instead of a six flux model then the discrete model must be made to emulate the behaviour of the six-flux model. This can be achieved by using 6 rays in the co-ordinate directions. If a radiation mesh needs to be specified, this should be identical to the flow mesh. If this is not possible, then at least the same number of cells in each direction must be specified. The details of the mesh must also be provided with your results.

Parameters

The absorption coefficient (a) assumed the following form:

$a = 0.01$, if $T < 323$K;

$a = 0.01 + (3.49/377)(T–323)$, if $323 <= T < 700$;

$a = 3.5 + (3.5/700)(T–700)$, if $T > 700$.

The walls are assumed to have an emissivity of 0.8.

It is assumed there is no scattering so s = 0.0.

Turbulence model *(if not listed please specify in the space provided)*

Laminar	k- ε	**buoyancy modified k-ε**	RNG	

AAA

Notes:

Turbulence Parameters*:

C_μ	σ_k	σ_ε	$C_{1\varepsilon}$	$C_{2\varepsilon}$	C_3
0.09	1.0	1.3	1.44	1.92	1.0

*If different parameters are being used please specify in the table above.

Combustion Model *(if not listed please specify in the space provided)*

none	**Volumetric heat source**	Mixed is burnt	Eddy break up
Magnussen soot model			

Combustion Parameters:

The volumetric heat source is assumed to be centrally located within the room with dimension of 0.3m x 0.3m x 0.3m with a total heat source of 62.9kW.

Compressibility

Incompressible	Boussinesq	Weakly compressible	**Fully compressible**

Compressibility Parameters:

External Pressure 1.01325e+05 Pa

Buoyancy

Yes	No

Gravity	-9.81m/s in the v-velocity direction.

Material Properties

Material Name	Air
Density	Determined by compressibility (Ideal Gas Law) Molecular Weight of air is 29.35
Viscosity (dynamic)	Laminar 1.798e-005 kg/m.s
Conductivity	0.02622 W/m.K
Specific heat capacity	1007.0 J/kg.K

Material Name	Common brick
Density	1600 Kg/m^3
Viscosity (dynamic)	Laminar 1.0e+10 kg/m.s
Conductivity	0.69 W/m.K
Specific heat capacity	840J/kg.K

Initial Values

U-VELOCITY	0.0
V-VELOCITY	0.0
W-VELOCITY	0.0
PRESSURE	0.0
TEMPERATURE	293.75
KINETIC ENERGY	0.01
DISSIPATION RATE	0.01

Boundary conditions

All walls are assumed to be made of common brick material. The emissivity of the walls is 0.8.

The doorway measures 0.74m wide and 1.83m high and is centrally located in one of the walls. This doorway is modelled using three solid conducting obstructions (again made from the brick material) to create the walls around the doorway. An extended region for this doorway is required to ensure that the airflow in the door is correctly modelled.

On the extended region all the boundary patches are fixed pressure (outlet) boundaries set to 0.0 Pa apart from the floor which is adiabatic.

The fire is modelled as a volumetric heat source which is assumed to be centrally located within the room on the floor with dimensions of 0.3m x 0.3m x 0.3m with a total heat source of 62.9kW.

Only half the geometry is modelled as a symmetry plane is utilised to half the computational domain.

Mesh

```
49 34 30
X
0.0  0.089286  0.178571  0.267857  0.357143  0.446429  0.535714
0.625  0.714286  0.803571  0.892857  0.982143  1.07143  1.16071
1.25  1.325  1.4  1.475  1.55  1.63929  1.72857  1.81786  1.90714
1.99643  2.08571  2.175  2.26429  2.35357  2.44286  2.53214
2.62143  2.71071  2.8  2.83333  2.86667  2.9  2.98783  3.10889
3.24676 3.39682 3.55666 3.72474 3.9 3.99839 4.15966 4.35806
4.58523 4.83651 5.10883 5.4
Y
0.0  0.06  0.12  0.18  0.24  0.3  0.369545  0.439091  0.508636
0.578182  0.647727  0.717273  0.786818  0.856364  0.925909
0.995455  1.065  1.13454  1.20409  1.27364  1.34318  1.41273
```

```
1.48227 1.55182 1.62136 1.69091 1.76046 1.83 1.88 1.93 1.98
2.03 2.08 2.13 2.18
z
0.0  0.0375  0.075  0.1125  0.15  0.186667  0.223333  0.26
0.296667
0.333333 0.37 0.4215 0.473 0.5245 0.576 0.6275 0.679 0.7305
0.782 0.8335 0.885 0.9365 0.988 1.0395 1.091 1.1425 1.194
1.2455 1.297 1.3485 1.4
```

Model Definition files

Convergence

Please specify your convergence criteria including type of error estimator and tolerance value for each variable

Runtime

Results files/Archiving:

Document cross-reference:

User Guides, etc

Comments

D8 – LPC case with improved physical properties and boundary conditions.

Fire Safety Engineering Group
Maritime Greenwich Campus,
Cooper Building,
University of Greenwich,
King William Walk,
London SE10 9JH, UK.

Date : 14/2/2001

Test case : **LPC-007 – 2000/2/5 – phase 2 improved physical properties and boundary conditions**
Document Version 1.0

PART 1 – CONTROLLED TEST SETUP

Case: **LPC-007 – 2000/2/5**

User details	
Run by: Angus Grandison Date: Phone no: 0208 331 7912 email: ga02@gre.ac.uk	Address: School of CMS University of Greenwich Maritime Greenwich Campus Old Royal Naval College Greenwich SE10 9LS. UNITED KINGDOM

Fire modelling Software

SMARTFIRE	CFX	PHOENICS			

Version/build number: v2.01 b369d
Date of release: 14/4/2000

Operating System

Windows 95/98/2000	Windows NT	Unix	Dos	

Version/build number: v4.00 sp6

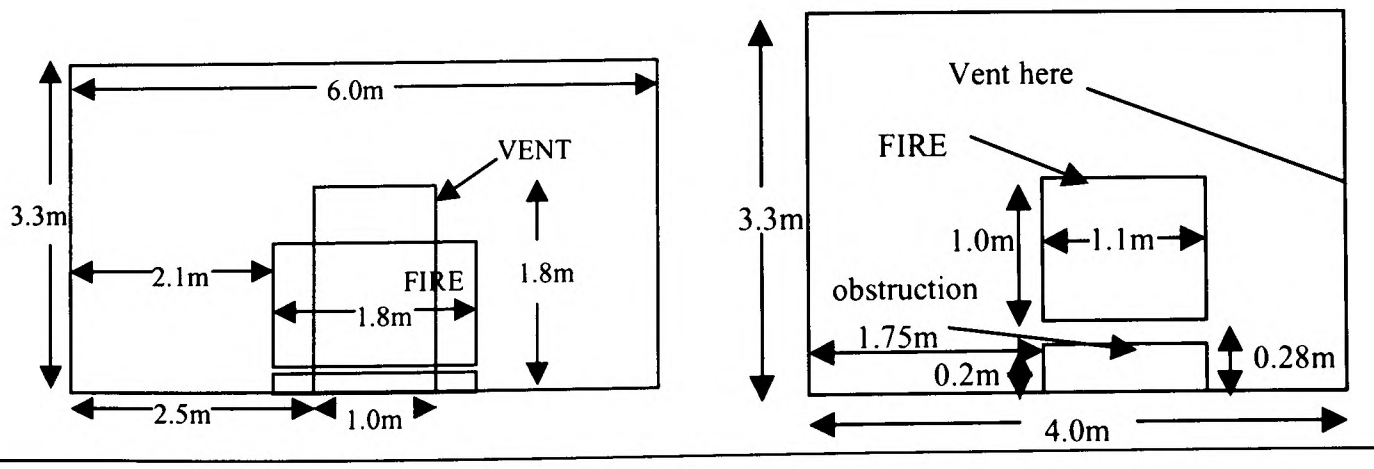
Machine

PC	Unix Workstation	

CPU: Intel Pentium III 733 Mhz
Memory: 256 Mb

Case description

This test case arises from a fire test conducted by the Loss Prevention Council (LPC)*. The test is a burning wood crib within an enclosure with a single opening. The test compartment is illustrated below and had a floor area of 6m x 4m and a 3.3m high ceiling. The compartment contained a doorway (vent) measuring 1.0m x 1.8m located on the rear 6m x 3.3m wall. The walls and ceiling of the compartment were made of fire resistant board (Asbestos) which were 0.1m thick. The floor was made of concrete.



y-z view x-y view

The heat release rate ($\dot{Q}$) is given by the following calculation (see equation 1).

$$\dot{Q} = \chi \cdot \Delta H_c \cdot \dot{m} \quad (1)$$

The efficiency factor (χ) and heat of combustion (ΔH_c) were given as χ=0.7 and ΔH_c is 17.8 MJ/kg for burning wood with a 10% moisture content and the mass loss rate ($\dot{m}$) (kg/s) for the wood crib is presented in the table below. It is assumed that the fuel molecule is $CH_{1.7}O_{0.83}$.

Time(s) |	0	150	450	460	1650
$\dot{m}$ (kg/s)|	0	0.01835	0.18636	0.1978	0.1978

* Glocking, J.L.D, Annable, K., Campbell, S.C. "Fire Spread in multi-storey buidings – *'Fire break out from heavyweight unglazed curtain wall system – Run 007'* ", LPC Laboratories rep. TE 88932-43, 25 Feb 1997.

Required Results

The results should be supplied as graphs and as Excel97 worksheets

The results from this case will be compared against experimental results and between the codes.

The required results are the temperature history curves for the first 900 seconds for the following locations:-

For the corner thermocouple stack located at 0.57m away form the side wall and 0.5m away from the front wall containing the vent.

The thermocouples within this stack are located at 1.5m, 2.0m, 2.5m and 3.0m above the floor.

The plume temperature measurements were taken at 3.0m away from the side wall and 2.392m away form the back wall of the compartment with the low measurement 1.5m above the floor and the high measurement at 3.0m above the floor.

CFD set up

1D	2D	3D

Transient	Steady State

180 * 5s time steps (900s total)

Differencing Schemes

Temporal:

Fully Implicit	Crank-Nicolson	Explicit	Exponential	

Spatial:

Hybrid	Central Difference	Upwind		

Notes:

Physical Models

Radiation Model *(if not listed please specify in the space provided)*

None	**Six flux**	Discrete Transfer	Monte Carlo	Radiosity	

Notes:

If the fire modelling software does not possess the six-flux model, a discrete transfer model may be used in place of the six-flux model. If the discrete transfer model must be used instead of a six flux model then the discrete model must be made to emulate the behaviour of the six-flux model. This can be achieved by using 6 rays in the co-ordinate directions. If a radiation mesh needs to be specified, this should be identical to the flow mesh. If this is not possible, then at least the same number of cells in each direction must be specified. The details of the mesh must also be provided with your results.

Parameters

The absorption coefficient (a) assumed the following form:

$a = 0.01$, if $T < 323$K;

$a = 0.01 + (0.314/377)(T–323)$, if $323 <= T < 700$;

$a = 0.315 + (0.315/700)(T–700)$, if $T > 700$.

The walls are assumed to have an emissivity of 0.8.

It is assumed there is no scattering so s = 0.0.

Turbulence model *(if not listed please specify in the space provided)*

Laminar	k- ε	**Buoyancy modified k-ε**	RNG	

Notes:

Turbulence Parameters*:

C_μ	σ_k	σ_ε	$C_{1\varepsilon}$	$C_{2\varepsilon}$	C_3
0.09	1.0	1.3	1.44	1.92	1.0

*If different parameters are being used please specify in the table above.

Combustion Model *(if not listed please specify in the space provided)*

none	Volumetric heat source	Mixed is burnt	Eddy break up
Magnussen soot model			

Combustion Parameters:

For phase-2 testing the Eddy Break up model must be used with the collision mixing model and infinite rate chemistry.

$$S_{m_f} = -\rho \frac{\varepsilon}{k} C_R \min\left(m_f, \frac{m_o}{i} \right),$$

where S_{mf} is the source term for the fuel mass fraction equation,
$C_R = 4.0$ (rate constant for collision mixing model),
m_f is the mass fraction of fuel
m_o is the mass fraction of oxident.
i is the amount of oxygen used for combustion every unit fuel, i.e

1kg Fuel + ikg -> (1+i) kg products

Compressibility

Incompressible	Boussinesq	Weakly compressible	Fully compressible

Compressibility Parameters:

External Pressure 1.01325e+05

Buoyancy

Yes	No

Gravity	-9.81m/s in the v-velocity direction.

Material Properties

Material Name	Air
Density	Determined by compressibility (Ideal Gas Law) Molecular Weight of air is 29.35
Viscosity	Laminar 1.798e-005 + Value determined from turbulence model
Conductivity	0.02622
Specific heat capacity	1007.0

Material Name	Asbestos
Density	577 Kg/m^3

Viscosity	1.0e+010
Conductivity	0.15
Specific heat capacity	1050.0

Material Name	Concrete
Density	2300 Kg/m^3
Viscosity	1.0e+010
Conductivity	1.4
Specific heat capacity	880.0

Material Name	Steel
Density	7850
Viscosity	1.0e+010
Conductivity	45.8
Specific heat capacity	460.0

Initial Values

U-VELOCITY	0.0
V-VELOCITY	0.0
W-VELOCITY	0.0
PRESSURE	0.0
TEMPERATURE	293.75
KINETIC ENERGY	0.01
DISSIPATION RATE	0.01

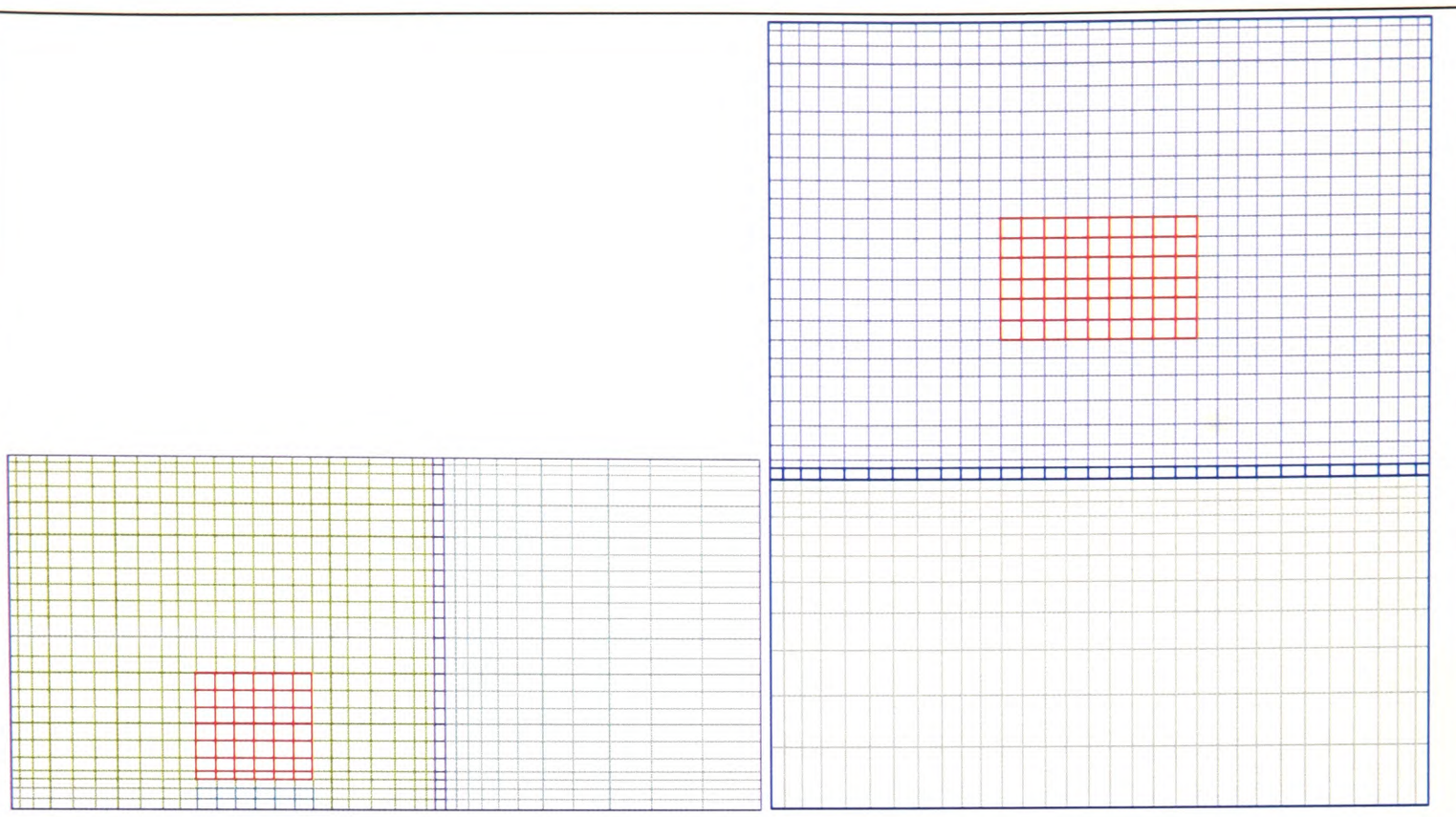
Boundary conditions
All walls are assumed to be made of asbestos for the second phase of the validation process. The wall emissivity is assummed to be 0.8. The floor is made of concrete and the obstruction below the fire is made of steel.

The door measures 1.0m wide and 1.8m high and is centrally located in the front wall. An extended region for this door is required to ensure that the airflow in the doorway is correctly modelled.

On the extended region all the boundary patches are fixed pressure (outlet) boundaries set to 0.0 Pa.

The fire needs to be modelled as a volumetric source of fuel with the same dimensions as the crib illustrated above using the fuel mass source specified above.

Mesh



Side view

Top view

31 24 35

X 0.0 0.12 0.28 0.46 0.68 0.9 1.12 1.34 1.56 1.74 1.92 2.1 2.3 2.5 2.7 2.9 3.1 3.3 3.5 3.7 3.9 4.08 4.26 4.44 4.66 4.88 5.1 5.32 5.54 5.72 5.88 6.0
Y 0.0 0.1 0.2 0.28 0.36 0.49 0.648 0.806 0.964 1.122 1.28 1.43 1.62 1.8 1.952 2.104 2.256 2.408 2.56 2.712 2.864 3.016 3.145 3.234 3.3
Z 0.0 0.08 0.21 0.37 0.58 0.79 1.0 1.21 1.42 1.585 1.75 1.9333 2.1167 2.3 2.4833 2.6667 2.85 3.015 3.18 3.4 3.62 3.8 3.92 4.0 4.1 4.2 4.312 4.4435 4.6021 4.7951 5.0303 5.315 5.6568 6.0633 6.5418 7.1

Model Definition files

Convergence

Please specify your convergence criteria including type of error estimator and tolerance value for each variable

Runtime

Results files/Archiving:

Document cross-reference:

User Guides, etc

Comments

D9 – LPC case with improved physical properties and boundary conditions and multiray radiation model with 24 rays.

Fire Safety Engineering Group
Maritime Greenwich Campus,
Cooper Building,
University of Greenwich,
King William Walk,
London SE10 9JH, UK.

Date : 14/2/2001

Test case : **LPC-007 – 2000/2/5 – phase 2 improved physical properties and boundary conditions and using the multiray radiation model with 24 rays**
Document Version 1.0

PART 1 – CONTROLLED TEST SETUP

Case: **LPC-007 – 2000/2/5**

User details	
Run by: Angus Grandison Date: Phone no: 0208 331 7912 email: ga02@gre.ac.uk	Address: School of CMS University of Greenwich Maritime Greenwich Campus Old Royal Naval College Greenwich SE10 9LS. UNITED KINGDOM

Fire modelling Software

SMARTFIRE	CFX	PHOENICS			

Version/build number: v2.01 b369d
Date of release: 14/4/2000

Operating System

Windows 95/98/2000	Windows NT	Unix	Dos	

Version/build number: v4.00 sp6

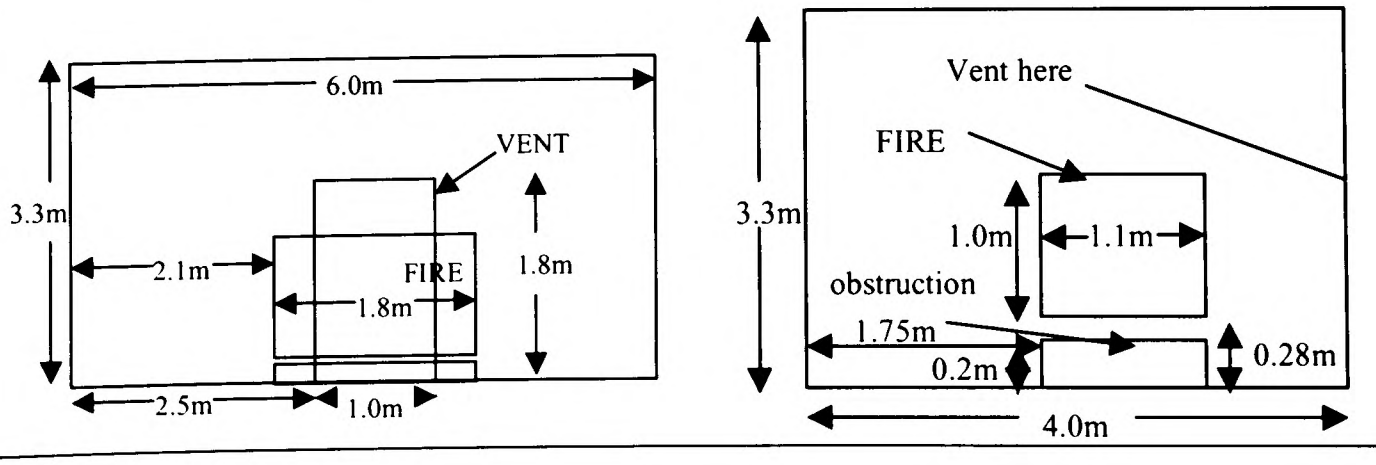
Machine

PC	Unix Workstation	

CPU: Intel Pentium III 733 Mhz
Memory: 256 Mb

Case description

This test case arises from a fire test conducted by the Loss Prevention Council (LPC)*. The test is a burning wood crib within an enclosure with a single opening. The test compartment is illustrated below and had a floor area of 6m x 4m and a 3.3m high ceiling. The compartment contained a doorway (vent) measuring 1.0m x 1.8m located on the rear 6m x 3.3m wall. The walls and ceiling of the compartment were made of fire resistant board (Asbestos) which were 0.1m thick. The floor was made of concrete.



y-z view x-y view

The heat release rate ($\dot{Q}$) is given by the following calculation (see equation 1).

$$\dot{Q} = \chi \cdot \Delta H_c \cdot \dot{m} \qquad (1)$$

The efficiency factor (χ) and heat of combustion (ΔH_c) were given as χ=0.7 and ΔH_c is 17.8 MJ/kg for burning wood with a 10% moisture content and the mass loss rate ($\dot{m}$) (kg/s) for the wood crib is presented in the table below. It is assumed that the fuel molecule is $CH_{1.7}O_{0.83}$.

Time(s) |	0	150	450	460	1650
$\dot{m}$ (kg/s)|	0	0.01835	0.18636	0.1978	0.1978

* Glocking, J.L.D, Annable, K., Campbell, S.C. "Fire Spread in multi-storey buidings – *'Fire break out from heavyweight unglazed curtain wall system – Run 007'* ", LPC Laboratories rep. TE 88932-43, 25 Feb 1997.

Required Results

The results should be supplied as graphs and as Excel97 worksheets

The results from this case will be compared against experimental results and between the codes.

The required results are the temperature history curves for the first 900 seconds for the following locations:-

For the corner thermocouple stack located at 0.57m away form the side wall and 0.5m away from the front wall containing the vent.

The thermocouples within this stack are located at 1.5m, 2.0m, 2.5m and 3.0m above the floor.

The plume temperature measurements were taken at 3.0m away from the side wall and 2.392m away form the back wall of the compartment with the low measurement 1.5m above the floor and the high measurement at 3.0m above the floor.

CFD set up

1D	2D	3D

Transient	Steady State

180 * 5s time steps (900s total)

Differencing Schemes

Temporal:

Fully Implicit	Crank-Nicolson	Explicit	Exponential	

Spatial:

Hybrid	Central Difference	Upwind		

Notes:

Physical Models

Radiation Model *(if not listed please specify in the space provided)*

None	Six flux	Discrete Transfer	Monte Carlo	Radiosity	Multiray radiation

Notes:

Parameters

The absorption coefficient (a) assumed the following form:

$a = 0.01$, if $T < 323$K;

$a = 0.01 + (0.314/377)(T–323)$, if $323 <= T < 700$;

$a = 0.315 + (0.315/700)(T–700)$, if $T > 700$.

The walls are assumed to have an emissivity of 0.8.

It is assumed there is no scattering so s = 0.0.

For the multiray radiation model with 24 rays the following parameters were used to specify the rays this should give improved resolution over six rays/flux.

The first 3 values form the unit vectors describing the direction of that ray and the last value indicates the weight of that ray direction (total sum of ray directions is 4π).

RAY 1 0.2958759 0.2958759 0.9082483 0.5235987
RAY-2 0.9082483 0.2958759 0.2958759 0.5235987
RAY-3 0.2958759 0.9082483 0.2958759 0.5235987
RAY-4 -0.2958759 0.2958759 0.9082483 0.5235987
RAY-5 -0.9082483 0.2958759 0.2958759 0.5235987
RAY-6 -0.2958759 0.9082483 0.2958759 0.5235987
RAY-7 0.2958759 -0.2958759 0.9082483 0.5235987

RAY-8 0.9082483 -0.2958759 0.2958759 0.5235987
RAY-9 0.2958759 -0.9082483 0.2958759 0.5235987
RAY-10 0.2958759 0.2958759 -0.9082483 0.5235987
RAY-11 0.9082483 0.2958759 -0.2958759 0.5235987
RAY-12 0.2958759 0.9082483 -0.2958759 0.5235987
RAY-13 -0.2958759 -0.2958759 0.9082483 0.5235987
RAY-14 -0.9082483 -0.2958759 0.2958759 0.5235987
RAY-15 -0.2958759 -0.9082483 0.2958759 0.5235987
RAY-16 -0.2958759 0.2958759 -0.9082483 0.5235987
RAY-17 -0.9082483 0.2958759 -0.2958759 0.5235987
RAY-18 -0.2958759 0.9082483 -0.2958759 0.5235987
RAY-19 0.2958759 -0.2958759 -0.9082483 0.5235987
RAY-20 0.9082483 -0.2958759 -0.2958759 0.5235987
RAY-21 0.2958759 -0.9082483 -0.2958759 0.5235987
RAY-22 -0.2958759 -0.2958759 -0.9082483 0.5235987
RAY-23 -0.9082483 -0.2958759 -0.2958759 0.5235987
RAY-24 -0.2958759 -0.9082483 -0.2958759 0.5235987

Turbulence model *(if not listed please specify in the space provided)*

Laminar	k- ε	Buoyancy modified k-ε	RNG	

Notes:

Turbulence Parameters*:

C_μ	σ_k	σ_ε	$C_{1\varepsilon}$	$C_{2\varepsilon}$	C_3
0.09	1.0	1.3	1.44	1.92	1.0

*If different parameters are being used please specify in the table above.

Combustion Model *(if not listed please specify in the space provided)*

none	Volumetric heat source	Mixed is burnt	Eddy break up
Magnussen soot model			

Combustion Parameters:

For phase-2 testing the Eddy Break up model was used with the collision mixing model and infinite rate chemistry.

$$S_{m_f} = -\rho \frac{\varepsilon}{k} C_R \min\left(m_f, \frac{m_o}{i}\right),$$

where S_{mf} is the source term for the fuel mass fraction equation,
$C_R = 4.0$ (rate constant for collision mixing model),
m_f is the mass fraction of fuel
m_o is the mass fraction of oxident.
i is the amount of oxygen used for combustion every unit fuel, i.e

1kg Fuel + ikg -> (1+i) kg products

Compressibility

Incompressible	Boussinesq	Weakly compressible	Fully compressible

Compressibility Parameters:

External Pressure 1.01325e+05

Buoyancy

Yes	No

Gravity	-9.81m/s in the v-velocity direction.

Material Properties

Material Name	Air
Density	Determined by compressibility (Ideal Gas Law) Molecular Weight of air is 29.35
Viscosity	Laminar 1.798e-005 + Value determined from turbulence model
Conductivity	0.02622
Specific heat capacity	1007.0

Material Name	Asbestos
Density	577 Kg/m^3
Viscosity	1.0e+010
Conductivity	0.15
Specific heat capacity	1050.0

Material Name	Concrete
Density	2300 Kg/m^3
Viscosity	1.0e+010
Conductivity	1.4
Specific heat capacity	880.0

Material Name	Steel
Density	7850
Viscosity	1.0e+010
Conductivity	45.8
Specific heat capacity	460.0

Initial Values

U-VELOCITY	0.0
V-VELOCITY	0.0
W-VELOCITY	0.0
PRESSURE	0.0
TEMPERATURE	293.75
KINETIC ENERGY	0.01

DISSIPATION RATE	0.01

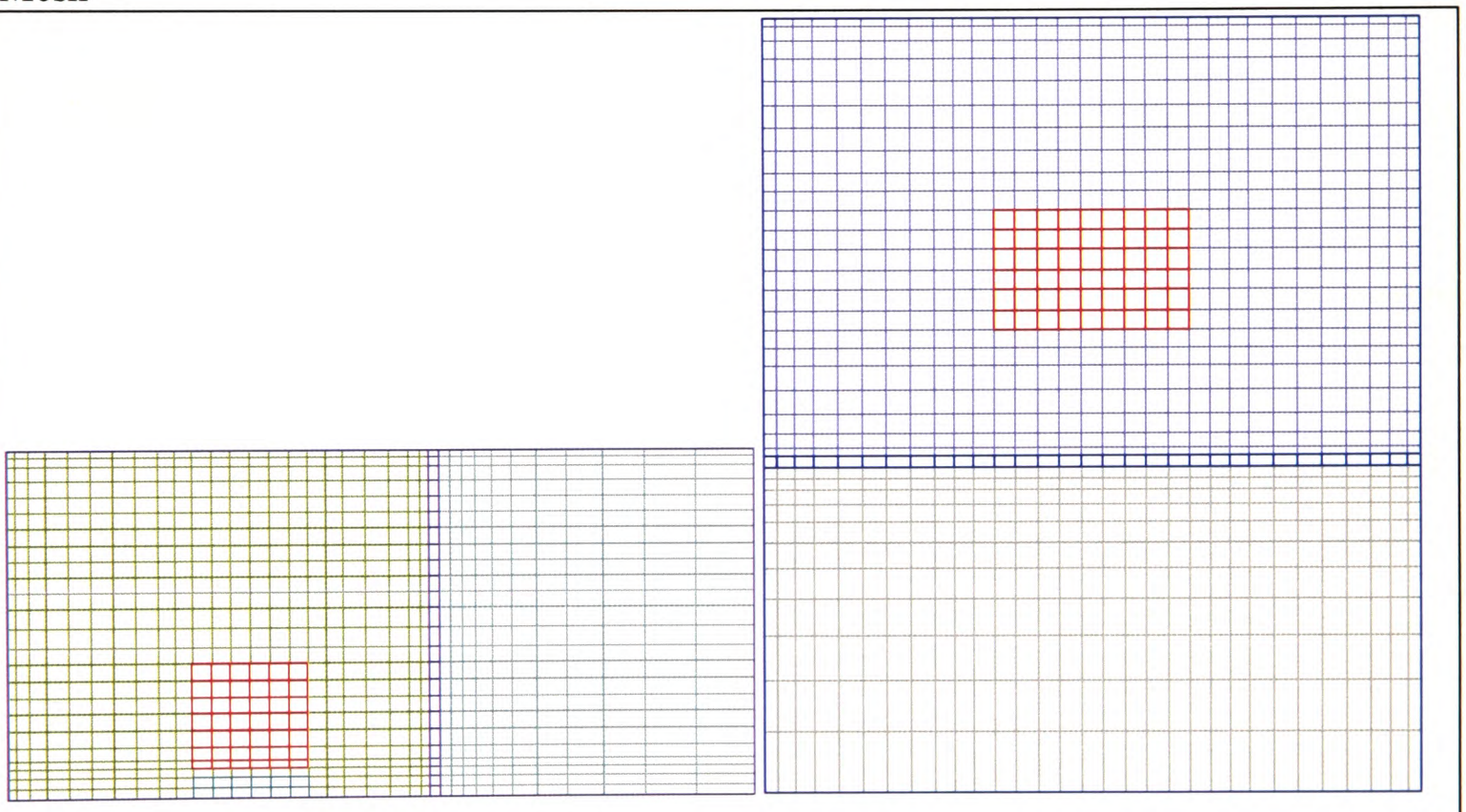
Boundary conditions
All walls are assumed to be made of asbestos for the second phase of the validation process. The wall emissivity is assummed to be 0.8. The floor is made of concrete and the obstruction below the fire is made of steel.

The door measures 1.0m wide and 1.8m high and is centrally located in the front wall. An extended region for this door is required to ensure that the airflow in the doorway is correctly modelled.

On the extended region all the boundary patches are fixed pressure (outlet) boundaries set to 0.0 Pa.

The fire needs to be modelled as a volumetric source of fuel with the same dimensions as the crib illustrated above using the fuel mass source specified above.

Mesh



Side view

Top view

31 24 35

X 0.0 0.12 0.28 0.46 0.68 0.9 1.12 1.34 1.56 1.74 1.92 2.1 2.3 2.5 2.7 2.9 3.1 3.3 3.5 3.7 3.9 4.08 4.26 4.44 4.66 4.88 5.1 5.32 5.54 5.72 5.88 6.0
Y 0.0 0.1 0.2 0.28 0.36 0.49 0.648 0.806 0.964 1.122 1.28 1.43 1.62 1.8 1.952 2.104 2.256 2.408 2.56 2.712 2.864 3.016 3.145 3.234 3.3
Z 0.0 0.08 0.21 0.37 0.58 0.79 1.0 1.21 1.42 1.585 1.75 1.9333 2.1167 2.3 2.4833 2.6667 2.85 3.015 3.18 3.4 3.62 3.8 3.92 4.0 4.1 4.2 4.312 4.4435 4.6021 4.7951 5.0303 5.315 5.6568 6.0633 6.5418 7.1

Model Definition files

Convergence

Please specify your convergence criteria including type of error estimator and tolerance value for each variable

Runtime

Results files/Archiving:

Document cross-reference:

User Guides, etc

Comments