AMERICAN UNIVERSITY OF BEIRUT

COMPARATIVE STUDY USING OPENFOAM© AND FLUENT SOLVERS OF SMOKE PROPAGATION INSIDE OCCUPIED SPACES

by
JAD EMILE HAWI

A thesis
submitted in partial fulfillment of the requirements
for the degree of Master of Engineering
to the Department of Mechanical Engineering
of the Faculty of Engineering and Architecture
at the American University of Beirut

Beirut, Lebanon
February 2016

AMERICAN UNIVERSITY OF BEIRUT

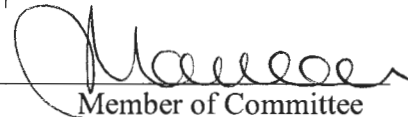# COMPARATIVE STUDY USING OPENFOAM© AND FLUENT SOLVERS OF SMOKE PROPAGATION INSIDE OCCUPIED SPACES
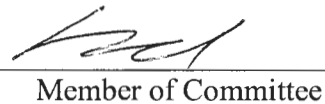

by
JAD EMILE HAWI

Approved by:

_____
Dr. Fadl Mukalled, Associate Dean & Professor   Advisor
Department of Mechanical Engineering

_____
Dr. Marwan Darwish, Professor   Member of Committee
Department of Mechanical Engineering

_____
Dr. Kamel Ghali, Chairperson & Professor   Member of Committee
Department of Mechanical Engineering

Date of thesis defense: February 01, 2016

AMERICAN UNIVERSITY OF BEIRUT

THESIS, DISSERTATION, PROJECT RELEASE FORM

Student Name:
_____
Last   First   Middle

◯ Master's Thesis   ◯ Master's Project   ◯ Doctoral
Dissertation

☐ I authorize the American University of Beirut to: (a) reproduce hard or electronic copies of my thesis, dissertation, or project; (b) include such copies in the archives and digital repositories of the University; and (c) make freely available such copies to third parties for research or educational purposes.

☐ I authorize the American University of Beirut, **three years after the date of submitting my thesis, dissertation, or project,** to: (a) reproduce hard or electronic copies of it; (b) include such copies in the archives and digital repositories of the University; and (c) make freely available such copies to third parties for research or educational purposes.

_____

Signature   Date

ACKNOWLEDGMENTS

AN ABSTRACT OF THE THESIS OF

Jad Emile Hawi   for   Master of Engineering
                      Major: Applied Energy

Title: Comparative Study Using OpenFOAM© and Fluent Solvers of Smoke Propagation inside Occupied Spaces

Smoke composed of toxic gases is considered the most dangerous aspect in case of fire in occupied buildings; thus, it is essential to design an effective ventilation system capable of creating a suitable environment for people to evacuate. The decision criteria of a proper ventilation system for a certain engineering application requires prior understanding of the transport physical phenomena associated with smoke propagation inside closed buildings and its thermal and hydrodynamic interactions with surrounding environment. The aim of this study is to build and implement a 3D based computational code in an open source platform (OpenFOAM$^©$), providing fire protection engineers with an effective, free, and customizable open source tool to simulate and analyze smoke propagation in occupied spaces. Results are validated by comparison to similar ones obtained using the FLUENT commercial CFD solver. The finite volume method is used to numerically mimic the existence of fire via a modified energy and smoke concentration equations. A source term resembling the time-dependent power of fire is implemented in a pre-existing modified energy equation. Further, to account for the transport of smoke, a concentration equation governing the temporal, convection, diffusion, and mass generation variations of smoke is numerically implemented and coupled to the OpenFOAM$^©$ solver. Similarly, the FLUENT solver is coupled to a developed C code to account for the existence of fire. To validate the implemented model in OpenFOAM$^©$, simulations are conducted on a test case (1mx1mx1m) topologically decomposed into 244,776 cells. The heat release and smoke generation rate due to fire are triggered via an integrated source term in each of the energy and species equations using the t-squared method in the growth region. Temporal and spatial variations of velocity magnitude, temperature, smoke concentration, visibility, and smoke and heat exposure are presented and analyzed in terms of contours and spatial profiles. Results predicted via the developed open source code and the well-known FLUENT solver are in good agreement.

CONTENTS

Chapter

ILLUSTRATIONS

TABLES

CHAPTER I

INTRODUCTION

Fire is a chemical reaction that occurs when a carbon based material mixes with air (i.e. oxygen) and ignites as it comes in contact with a hot enough surface. This is why it has a high frequency of occurrence. Worldwide, nearly 15000 deaths from fires and burns and 70 billion USD of damages are sustained yearly with most victims of fires die from smoke or toxic gases and not from burns [13]. Hence, smoke is the most dangerous aspect of fires.

Smoke in fires is made up of toxic gases that irritate the eyes and lung, reduce visibility, and decrease mental acuity making it difficult for occupants to escape the building. For this reason, providing safe conditions by assuring two meters above floor level of smoke free zone that gives good visibility of exit signs and small amount of smoke inhalation is recently becoming more and more important in all building codes according to NFPA [27] e.g. the time of incapacitation should be less than 20 minutes, the range of vision should be greater than 5 m., the temperature should be less than 100°C or 373 K with an exposure time less than 10 min, etc. This gives the occupants a tenable environment to escape the building in ten minutes after detection of fire. To apply this codes, companies are relying on commercial Computational Fluid Dynamics (CFD) packages such as ANSYS FLUENT [1] and Pyrosim. However, many designers avoid using such programs due to their high cost.

The current work deals with the analysis of fire propagation in large buildings using an open source CFD code known as "OpenFOAM©", which is a free CFD toolbox

with many built in features. OpenFOAM$^©$ adopts a pressure-based finite volume

formulation with its structure allowing for a complete freedom to extend and modify its

initial functionality. Throughout this report, the theory behind it is discussed, highlighting

its importance and showing its validity by comparing it to reliable commercial CFD

software packages.

CHAPTER II

LITERATURE REVIEW

A. Fire design using CFD: Beginnings and Early Uses

Modeling techniques for fire and smoke simulation allow predicting both the propagation of smoke in buildings and the assessment of performance of smoke exhaust systems. The accuracy of predictions depends on the adopted model and its implementation in a CFD code. Due to the fact that the main concern in CFD is the computational cost, and since turbulence is a major factor in smoke modeling, the treatment of turbulence rely on different techniques such as the Reynolds averaged Navier-Stokes (RANS) approach, Large eddy simulation (LES), Detached eddy simulation (DES), and Direct numerical simulation (DNS). The RANS equations are represented by a statistical averaging description of the fluid flow motion; such that the velocity and the pressure of the flow are divided into time averaged and fluctuations which is called Reynolds decomposition [29]. On the other hand, large eddy simulation (LES) is based on the filtering concept where large scale turbulent structures are directly simulated whereas small turbulent scales are modeled using sub-grid scale (SGS) models. Detached eddy simulation (DES) is a mixture of RANS and LES, where this model uses the RANS mode to treat near wall regions and switches to LES mode for bulk flow [29]. Finally, Direct numerical simulation (DNS) solves the Navier–Stokes equations without any turbulence model, in other words, the whole range of spatial and temporal scales of the turbulence must be resolved.

3

Fire development can be illustrated as shown in Figure 1. The hot combustion products caused by a fire rise due to variations in pressure and entrain with them the nearby cold air to form a plume. This intensifying plume reaches the ceiling and forms two streams of smoke that flow horizontally in opposite directions.
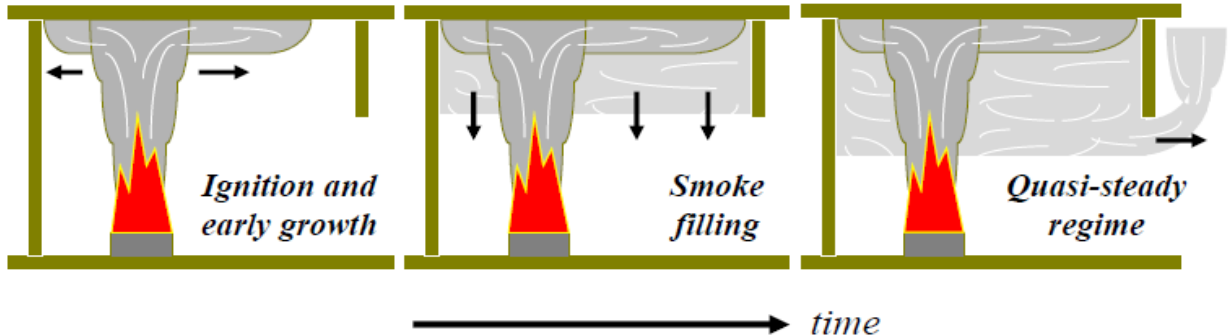
Figure 1: Smoke filling in a room in function of time

This process was first studied by Yang and Chang [40] using the computer code UNDSAFE-I and then in the late 1980s by the application of JASMINE developed by the Fire Research Station, UK [8], [35].FLOW-3D [8], [35]which was developed in early 1990s by the Atomic Energy Authority, Harwell, of the UK was used to reconstruct the case of Kings Cross Fire in the London Underground station which resulted in many fatalities [8], [35]. Kandola and Morris carried out a numerical study using the AEA CFD-FLOW-3D package [5] in which they focused on hazards of smoke accompanying fire as it affects people and discussed fire safety procedures and equipment to be used. Sinai et al. reported on the AEA attempt to validate fire predictions using different CFD codes [39]. The domestic fire was simulated by Mawhinney et al. using Phoenics CFD software [19]. Novozhilov relied on the same experimental data to discuss the significance of experimental studies to validate CFD simulations [28]. Most papers tackling this subject

have simulated, numerically using CFD packages and experimentally by considering a full scale model, fires in tunnels and car parking where the source of fire is a car explosion. Den Boer et al. performed full scale fire tests in a tunnel in the Netherland and compared it with predictions generated using the PHOENICS CFD software [9]. Results showed an acceptable accuracy of the software but the workers insisted that the real site conditions could not be implemented accurately. In addition, Tabarra et al. used scale model tests on tunnels to validate the CFD results [25] while Deng et al. used CFD to model fire on trains in underground stations [26].

Due to the limited computer capacity and to reduce cost, early fire simulations did not reproduce accurately real fires. In fact, many of the models were idealized and many factors such as radiation and soot emission were not accounted for to simplify the mesh and reduce the solution time. With recent advances in computer technology and developments in numerical methods, CFD simulations have crossed a long way in fire engineering to become a necessity in the design of fire systems.

B. Fire Types

According to European Standard Classification of Fires, there are six types of fire (Table 1) depending on the kind of fuel that is burning. This categorization makes it easier to decide on the most appropriate type of fire extinguishers to contain a fire.

1. Class A: Ordinary Combustible Fires

Class A type is the most common type of fire, where the materials involved are: paper, textiles, rubber, wood, plastics and organic carbon based compounds. Class A fire

occurs when the material is heated to its ignition temperature, in the presence of oxygen in the air and fuel to burn. Water or foam fire extinguishers are the most suitable way to overcome this type of ordinary combustible fire [11].

2. Class B: Flammable Liquids

The ignition temperature of flammable liquids is less than 100°C which make them easy to burn [11]. In addition, class B liquids can burn at any temperature if near a source of flame. Some examples of class B fire are: gasoline, kerosene, paint, paint thinners and others. To extinguish Class B fire, one should use a foam fire extinguisher since by using water, the fuel will scatter; thus, the fire will spread.

3. Class C: Flammable Gases

Class C as butane, propane and petroleum gases are the most dangerous types of fire since they have a high potential to create an explosion and are very hard to fight. Therefore flammable gases are stored in sealed containers. The most efficient way is to use dry power extinguishers to defeat this type of fire [11].

4. Class D: Metal Fires

Although it requires a lot of heat to ignite most metals, they can cause fire. Powdered metals are easier to ignite than solid metals, thus causing a higher risk. Examples of Class D metals: potassium, magnesium, aluminum, sodium, etc. To defeat fire caused by Class D, a special type D powder for fire extinguishers is used. Neither water nor foam can be used as they increase the intensity of the fire flame and cause more damages [11].

5. Class E: Electrical Fires

Appliances, Switches, Panel boxes, Power tools and others can cause electrical fires. When dealing with a Class E fire, electricity supply should be isolated as quickly as posible. Carbon dioxide and dry powder fire extinguishers should only be used to tackle electrical fire, and water or foam should not be used even when the source of electricity has been cut off [11].

6. Class F: Cooking Oil Fires

Home and professional kitchens are common place for the presence of cooking oil and fats which may be the cause of class F fire although they require high temperatures. Special wet chemicals extinguishers should be used for class F in order to cool the fire and prevent its re-ignition [11].

Table 1: Example of the different types of fire and their extinguishers [11].

Type	Example	Fire extinguishers
Class A: Ordinary Fires	Wood, Paper, Cloth, Rubber, some Plastics ...	Water or foam
Class B: Flammable Liquids	Gasoline, Kerosene, Paint, Paint thinners...	Foam
Class C: Flammable Gases	Butane, Propane, Petroleum gases...	Dry power
Class D: Metal Fires	Magnesium, Titanium, Potassium, Sodium...	Special type D powder
Class E: Electrical Fires	Appliances, Switches, Panel boxes, Power tools...	Carbon Dioxide and dry powder
Class F: Cooking Oil Fires	Cooking oil, Fats...	Wet chemical

C. Smoke Hazards

The main reason for deaths resulting from fire is smoke inhalation rather than burns since people become incapable of reaching the exit on time. The presence of synthetic materials in homes nowadays causes the production of dangerous substances. Most of the available oxygen in a building on fire gets consumed in the process of incomplete combustion slowing down the burning process and releasing smoke containing toxic gases and particles which can be fatal. These particles are small substances which are not yet burned, partially burned or completely burned and can enter the protective filters of respiratory systems and go into the lungs. Some of them might be actively toxic or irritating to the eyes and the digestive system, while others may be foggy composed of vapors. These can be poisonous if they are inhaled or absorbed by the skin. Most commonly, smokes are composed of toxic gases like carbon monoxide (CO) that can be toxic even in small amounts as it replaces oxygen in the blood. Another toxic gas is hydrogen cyanide caused by plastic burning, like PVC pipe, as it intervenes with cellular respiration. Phosgene is also toxic as it causes itchy eyes and a sore throat and it is caused by the burning of household products like vinyl materials and at high levels, it can result in pulmonary edema and death. Other than producing smoke, fire can be dangerous as it reduces the levels of oxygen by consuming it (Table 2) or by displacing it with other gases. Another respiratory risk is heat as the respiratory tract can be burned by super-heated gases and death can be the consequence of one breath if the air contains such heat [27].

Table 2: What an individual will experience at different oxygen levels [27].

Oxygen levels	Individual experiences
21%	Normal outside air
17 %	Impaired judgment and coordination
12 %	Headache, dizziness, nausea, fatigue
9 %	Unconsciousness
6 %	Respiratory arrest, cardiac arrest, death

CHAPTER III

METHODOLOGY

A. Methodology

The objective of this project is to develop a solver in OpenFOAM© able to simulate the propagation of smoke resulting from fire. The solver should be capable, regardless of geometry and boundary and initial conditions, of producing accurate smoke propagation results. Computations involve predicting very complex fields of velocity, temperature, and concentration, visibility, time to lose consciousness due to heat and smoke. Such numerical tool will be helpful for checking and modifying designs of ventilation and smoke extraction systems to guarantee maintaining a lower "smoke-free" layer for a specified period of time for occupants to safely exit and fire fighters to contain and extinguish the fire. The correctness and accuracy of results generated using the OpenFOAM© based solver will be checked by direct comparison with similar predictions obtained using the ANSYS software, which is the most trusted CFD tool for predicting smoke propagation by engineers.

B. Software Description

1. ANSYS Fluent

ANSYS Fluent is one of many advanced computational fluid dynamics software (e.g. Smart Fire, FDS, etc.), that is able to simulate different fire scenarios in complex geometries. ANSYS, which stands for "Analysis System", was first developed by Swanson

Analysis Systems, Inc. (SASI) company to simulate stationary, moving, and heat transfer problems. ANSYS Fluent uses the C computer language and benefits from its flexibility that increases the analysis of the flow and optimize its capabilities; such that any study, whether single or multi-phase, isothermal or reacting, compressible or incompressible, will get reliable and valuable solutions [1]. In addition, ANSYS Fluent gives the user a wide range of mesh, e.g. triangular & quadrilateral for the 2-dimensional geometries and tetrahedral, hexahedral, pyramid, wedge, polyhedral, etc. for three dimensional geometries [1]. The user friendly interface permits the implementation of the other operations such as the boundary conditions and the fluid properties, then it executes the solution, post processes it, and interprets the results.

2. OpenFOAM$^{©}$

a. Description

OpenFOAM$^{©}$, which stands for "Open Field Operation and Manipulation", is an open source object oriented C++ framework that can be used to build a variety of computational solvers in continuum mechanics with a focus on finite volume discretization. It also includes several ready solvers, utilities, and applications that can be directly used. It particularly allows developing solvers to deal with complex flows involving chemical reactions, turbulence, heat transfer, solid dynamics, and others [29]. It was originally developed at Imperial College, London in the late 1980s in an attempt to compete with existing FORTRAN programs by providing flexibility due to the use of the C++ computer language. It was first sold by the Nabla Ltd. company then it was released as open source in 2004 [29]. In addition to the meshing CAD geometries tools, it contains pre- and post-

11

processing codes. Since it is not a black box, OpenFOAM© offers complete freedom to modify and extend its functionality. Different applications have been developed to be executed by simply linking them to the library. Over 80 solvers and 170 utility applications to simulate problems in engineering mechanics and perform pre- and post-processing tasks have been reported [29]. As shown in Figure 2, this includes meshing, data visualization, etc.
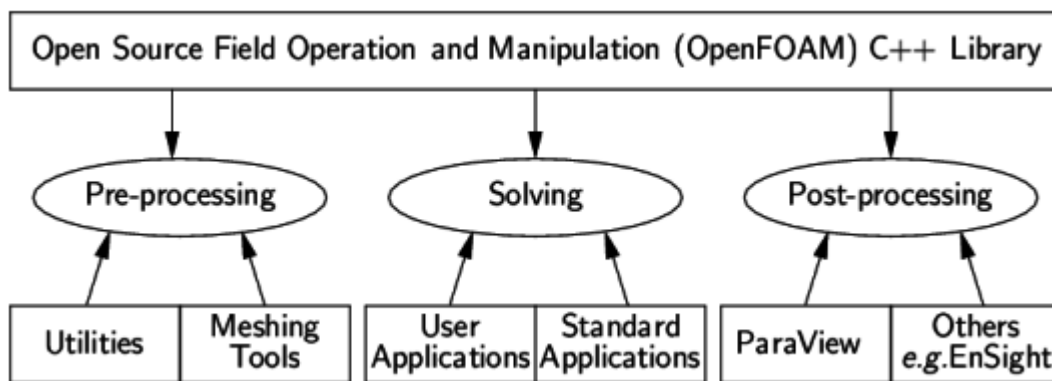
Figure 2: Overview of OpenFOAM© structure [29]

b. Discretization and Numerical Methods

i. Discretization

Time and spatial discretization are the main keys in solving CFD problems. Time or temporal discretization is the integration of the different terms in the equations over a time step Δt; thus, in transient flows, a careful selection of Δt should be done to take into consideration every detail of the flow.

Space or spatial discretization is the subdivision of the computational domain into a finite number of non-overlapping cells as shown in Figure 3. The conservation equations are first discretized and then solved using numerical methods.
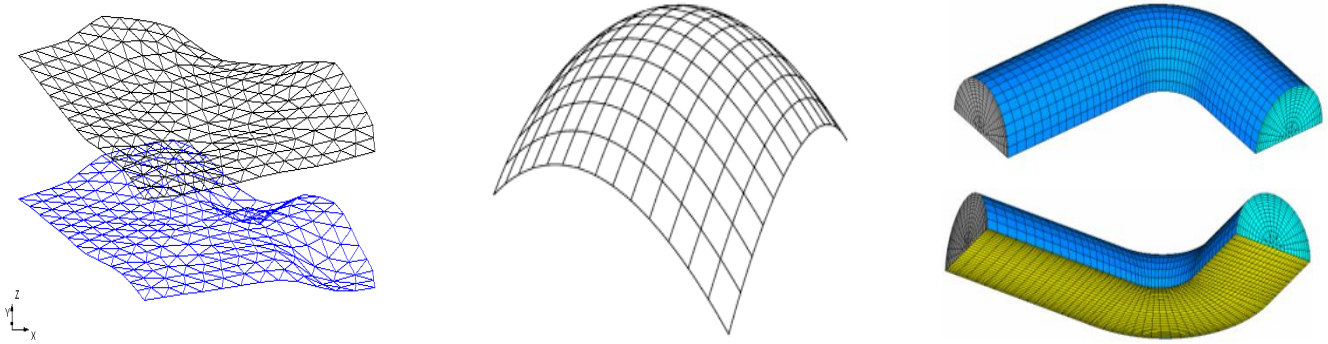
Figure 3: Examples of spatial discretization

ii. Numerical Methods

For each control volume, the governing equations are converted to algebraic ones, and they are assembled in a global matrix of the form A[T]=B; where T is the solution vector that represents the properties of the flow field which are stored at the center of the control volume [24].

In general, for each term of the equation, several schemes are permitted. The diffusion term is usually central-differenced and second order accurate. Whereas the convection term can be first or higher order accurate. Examples of convection schemes include the first order Upwind [30] and the Second Order Upwind (SOU) [30]. On the other hand, the transient scheme is Crank Nicolson [30] schemes.

In this study, for the simulation of smoke using OpenFOAM$^{©}$, a customized solver is developed called SmokeFOAM based on the SimpleFOAM code where momentum and continuity equations are solved. Temperature and smoke concentration equations are added in order to analyze the effect of smoke in buildings by means of heat and concentration. In SmokeFOAM, the transport of smoke by convection and diffusion are accounted for by solving the Navier-Stokes equations. Processes of buoyancy, convection, turbulence, and boundary heat transfer relevant to the movement of smoke is included in

the code. Specifically the scheme selected for solving the momentum, energy, species and turbulence equations is implicit and second order accurate. Its implicit formulation allows for large time steps since it provides stable solutions. It is a finite-volume code using an unstructured grid and based on the SIMPLE pressure-correction algorithm.

The overall solution algorithm for predicting the flow field can be summarized as follows [24]:

Solve the discretized momentum equation

- Solve the discretized continuity (or pressure correction) equation

- Correct and update pressure and velocities.

- Solve the turbulence model equations

- Solve the energy equation

- Solve the species equation

- Repeat the above steps until convergence

Then, in the post-processing, equations for visibility and time for losing consciousness as a result heat and smoke concentrations are inserted so that the consequence of fire are easily read.

C. Theoretical Formulation

The equations governing the transport phenomena of interest in this work represent the conservation of mass, momentum, energy, and mass species. In addition, turbulence effects are modeled using the two-equation $k - \varepsilon$ turbulence model. The forms of these conservation equations are presented next [24].

1. *Mathematical Model*

a. Continuity Equation

Since the flow is considered to be incompressible, density is independent of time and the continuity equation is reduced to

$$\frac{\partial \bar{u}_j}{\partial x_j} = 0 \tag{3.1}$$

b. Momentum Equation

The momentum equation for incompressible flow with turbulence accounted for within the Boussinesq hypothesis is expressed as

$$\frac{\partial \bar{u}_i}{\partial t} + \frac{\partial}{\partial x_j}\left(\bar{u}_j \bar{u}_i\right) = -\frac{\partial}{\partial x_i}\left(\frac{\bar{p}}{\rho_0}\right) + \frac{1}{\rho_0}\frac{\partial}{\partial x_i}\left(\tau_{ij} + \tau_{tij}\right) + \frac{\bar{\rho}}{\rho_0}g_i \tag{3.2}$$

where:

- g_i is the gravity acceleration.

- τ_{tij} is the turbulent stress tensor.

- τ_{ij} is the mean stress tensor due to molecular viscosity

$$\tau_{ij} = \mu\left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i}\right) - \frac{2}{3}\frac{\partial \bar{u}_k}{\partial x_k}\delta_{ij} \tag{3.3}$$

Substituting stresses by their equivalent expressions and using the Boussinesq approximation to model variation in density, the momentum equation becomes:

$$\frac{\partial \bar{u}_i}{\partial t} + \frac{\partial}{\partial x_j}\left(\bar{u}_j \bar{u}_i\right) = -\frac{\partial P}{\partial x_i} + \frac{\partial}{\partial x_j}\left[\vartheta_{eff}\left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i}\right) - \frac{2}{3}\frac{\partial \bar{u}_k}{\partial x_k}\delta_{ij}\right] + \rho_k \tag{3.4}$$

where:

- ϑ_{eff} is the effective kinematic viscosity. Within the Boussinesq approximation, the density is related to temperature according to

15

$$\rho_k = g_i[1 - \beta(\bar{T} - T_0)] \tag{3.5}$$

where:

- β is the coefficient of expansion with temperature of the fluid in

Kelvin^{-1}

- $\bar{T}$ is the mean temperature in Kelvin

- T_0 is the reference temperature in Kelvin

c. Energy Equation

The energy equation in terms of enthalpy is expressed as

$$\frac{\partial}{\partial t}(\rho h) + \frac{\partial}{\partial x_j}(\rho h u_j) = -p\frac{\partial u_k}{\partial x_k} + \tau_{ij}\frac{\partial u_i}{\partial x_j} - \frac{\partial q_k}{\partial x_k} + S_t \tag{3.6}$$

- $\frac{\partial u_k}{\partial x_k}$ represents the divergence of the velocity vector, which is equal to zero

as revealed by the continuity equation.

- $\tau_{ij}\frac{\partial u_i}{\partial x_j}$ represents the viscous shearing force and for incompressible flows,

this term is equal to zero according to Ferziger[16], Peric [16]and White [17].

Then, energy equation becomes:

$$\frac{\partial}{\partial t}(\rho h) + \frac{\partial}{\partial x_j}(\rho h u_j) = -\frac{\partial q_k}{\partial x_k} + S_t \tag{3.7}$$

The diffusion fluxes are computed using Fourier's law [30] of heat conduction

relates the heat flux to the local temperature gradient as

$$q_i = -k\frac{\partial T}{\partial x_i} \,\&\, q_{t_i} = -k_t\frac{\partial T}{\partial x_i} \tag{3.8}$$

Replacing h by T using the enthalpy temperature relation $h = c_p T$, the energy

equation written in terms of temperature is shown in equation (3.9)

$$\frac{\partial}{\partial t}(\overline{T}) + \frac{\partial}{\partial x_j}(\overline{T}\overline{u}_j) = \frac{\partial}{\partial x_i}\left[\left(\frac{\mu_0 k}{\mu_0 \rho_0 c_p}\right)\frac{\partial \overline{T}}{\partial x_i}\right] + \frac{\partial}{\partial x_i}\left[\left(\frac{\mu_t k_t}{\mu_t \rho_0 c_p}\right)\frac{\partial \overline{T}}{\partial x_i}\right] + S_t \tag{3.9}$$

Thus, the final form of the energy equation used in this study can be written as

$$\frac{\partial}{\partial t}(\overline{T}) + \frac{\partial}{\partial x_j}(\overline{T}\overline{u}_j) = \frac{\partial}{\partial x_i}\left[\left(\frac{\vartheta}{Pr} + \frac{\vartheta_t}{Pr_t}\right)\frac{\partial \overline{T}}{\partial x_i}\right] + S_t \tag{3.10}$$

where:

- ϑ is the kinematic viscosity equal to $\vartheta = \mu/\rho$ in m^2/s.

- Pr is Prandtl number equal to $Pr = \frac{c_p \mu}{k}$

d. Smoke Concentration Equation (mass fraction of smoke in the air)

Smoke is tracked by solving its conservation equation given by

$$\frac{\partial}{\partial t}(\overline{C}) + \frac{\partial}{\partial x_j}(\overline{C}\overline{u}_j) = \frac{\partial}{\partial x_i}\left[\left(D_i + \frac{\mu_t}{Sc_t}\right)\frac{\partial \overline{C}}{\partial x_i}\right] + S_t \tag{3.11}$$

where:

- Sc is the global laminar Schmidt number that is usually prescribed as Sc=0.7

- μ is the viscosity of the smoke that can be set equal to the viscosity of air

- $D_i = \frac{\mu}{Sc_i} = \frac{\mu}{Sc}$ is the diffusion coefficient

As discussed by Moghtaderi et al. [41], in fire regions, flow will be strongly buoyancy driven; and not taking turbulence into account, simulations will show relatively uniform thermal layers rather than strong stratification. Thus, turbulence will be accounted for using the two-equation k- ε model since it is widely used and best validated. The k- ε model constants are presented in Table 3 according to Launder and Spalding [23].

e. Turbulence Equations

The most widely used turbulence model is the standard k-epsilon model. It is based on the boussinesq approximation to derive two equations (i.e. Eq. (3.13) and (3.14)) from which the turbulent viscosity is computed.

i. Transport equation for the turbulence kinetic energy

$$\frac{\partial}{\partial t}(\rho k) + \frac{\partial}{\partial x_j}(\rho U_j k) = P_k + \frac{\partial}{\partial x_i}\left[\left(\mu + \frac{\mu_T}{\sigma_k}\right)\frac{\partial k}{\partial x_i}\right] - \rho\varepsilon \tag{3.12}$$

ii. Transport equation for the turbulence dissipation rate

$$\frac{\partial}{\partial t}(\rho\varepsilon) + \frac{\partial}{\partial x_j}(\rho U_j \varepsilon) = C_{\varepsilon 1}\frac{\varepsilon}{k}P_k - C_{\varepsilon 2}\rho\frac{\varepsilon^2}{k} + \frac{\partial}{\partial x_i}\left[\left(\mu + \frac{\mu_t}{\sigma_\varepsilon}\right)\frac{\partial \varepsilon}{\partial x_i}\right] \tag{3.13}$$

where:

- The volumetric production rate of k by shear forces for incompressible flow:

$$P_k = \tau^R : \nabla U \tag{3.14}$$

- The turbulent viscosity and thermal diffusivity are respectively computed as

$$\mu_t = C_\mu \rho \frac{k^2}{\varepsilon} \text{ and } k_t = \frac{C_p \mu_t}{Pr_t} \tag{3.15}$$

Table 3: Set of k-ε model constants [23]

σ_k	σ_ε	C_μ	$C_{\varepsilon 1}$	$C_{\varepsilon 2}$	Pr_t	Sh_t
1	1.3	0.09	1.44	1.92	0.9	0.81

2. Source Terms

Generation of heat and smoke are boundary phenomena that are accounted for through two source terms, a heat source term and a smoke source term, added to the energy and smoke concentration equations, respectively.

a. Heat Source

In fire simulations, growth and fully developed are the main stages in determining the initial heat release rate for the flaming fire as described in Figure 4.
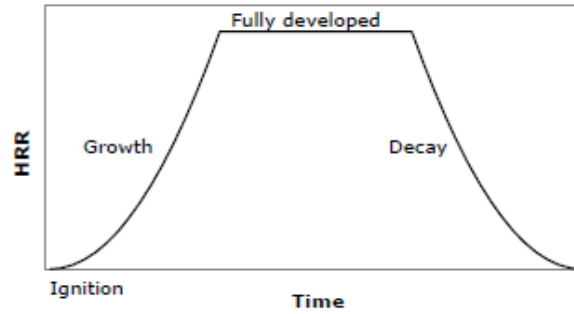
Figure 4: Different stages of fire showing the variation of HRR in function of time [34].

i. Growth Stage – Heat Release Rate

The heat release rate (HRR) in the growth stage is described by the t²-method [36] as

$$\dot{Q} = \alpha t^2 \tag{3.16}$$

where:

- $\dot{Q}$ = heat release rate in kW

- α = fire growth coefficient in kW/s² from Table 2

- t = time in s

The values of the fire growth coefficient presented in Table 4 and Figure 5 are calculated according to

$$\alpha = \frac{\dot{Q}_0}{t_0^2} \tag{3.17}$$

where:

- $\dot{Q}_0$ = reference heat release rate in kW, usually taken to be 1055 kW (= 1000 Btu/s)

- t_0 = fire growth time in s

Table 4: Values for α and time to reach 1055 kW for different growth rates [4]

Growth Rate	α in kW/s²	Time to reach 1055 kW in s
Slow	0.003	600
Medium	0.012	300
Fast	0.047	150
Ultra-Fast	0.188	75
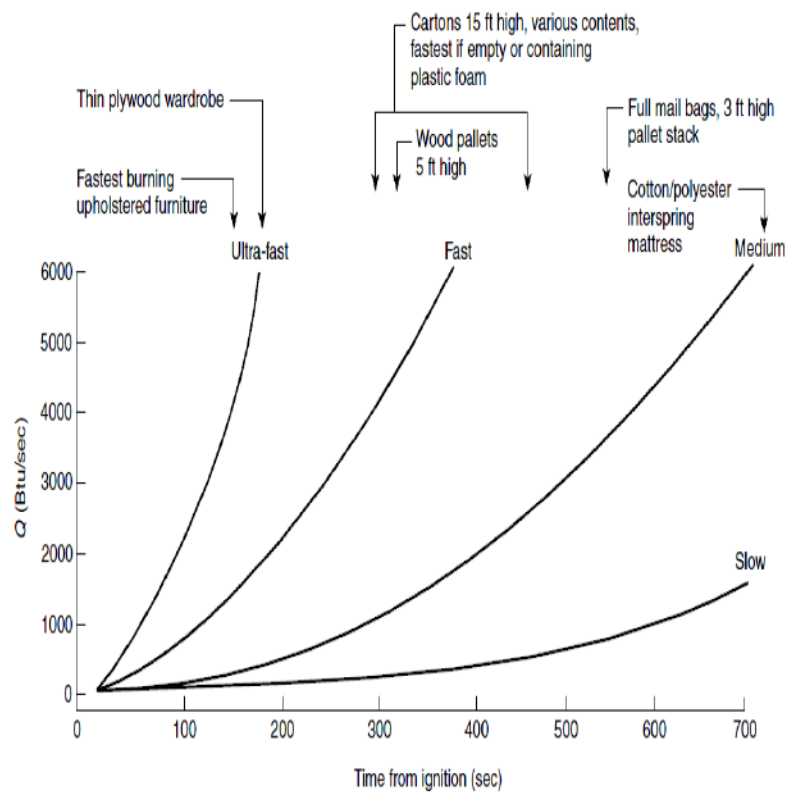
Figure 5: Heat release rate for different growth rate according to t-squared method [4]

Table 5: Fire growth rates for different examples

Occupancy	Growth rate
Dwellings	Fast[43] Medium[7,42]
Schools, offices	Fast[43] Medium[7]
Hotels, nursing homes, etc.	Fast[43]
Shopping centers, entertainment centers	Ultra-Fast[42] Fast[7]

20

According to Boverket [6] and British Standard Institution [7], the fire growth rates in schools and offices occupancy is assumed to be medium but fast for Karlsson and Quintiere [4] as shown in Table 5. This depends on the material burned in the simulations.

ii. Fully Developed Fire – Peak Heat Release Rate

After determining the type of fire growth, the next step is to specify the fire characteristics in the fully developed regime. Thus, we should identify the maximum value of heat release rate, which is limited either by the presence of ventilation or the presence of combustible material. Therefore, peak HRR for both ventilation and fuel-controlled fires should be calculated and the lower value between them should be selected as described by Karlsson and Quintiere [4].

- Peak Heat Release for the Fuel or Combustible Surface-Controlled Fire

The peak heat release rate for the fuel-controlled fire is estimated according to

$$\dot{Q} = \dot{Q}^{''} \times A_f \tag{3.18}$$

where:

- $\dot{Q}$= total heat release rate in kW

- $\dot{Q}^{''}$ = heat release rate per unit area in kW/m²

- Af = horizontal burning area of the fuel in m²

For offices and schools, the European standard [12] indicates that the heat release rate per unit area is equal 250 kW/m^2.

- Peak Heat Release for the Ventilated-Controlled fire

Assuming each kilogram of oxygen produces in combustion approximately 13.1 MJ and since 23% of the mass of air is oxygen, the peak HRR for complete combustion is given according to Karlsson and Quintiere [4] as

$$\dot{Q}_V = 1500 A_0 \sqrt{H_0} \qquad (3.19)$$

where:

- $\dot{Q}_V$ = maximum heat release rate for the ventilation-controlled fire in kW

- $H_0 = \frac{A_1 H_1 + A_2 H_2 + \cdots + A_6 H_6}{A_0}$ as shown in Figure 4

- $A_0 = A_1 + A_2 + \ldots + A_6 = b_1 H_1 + b_2 H_2 + \ldots + b_6 H_6$
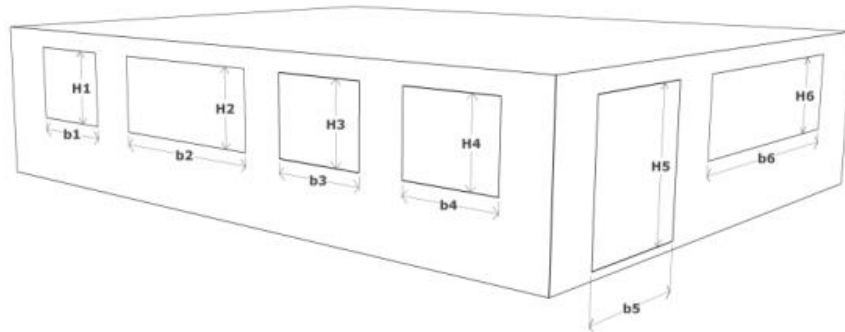
Figure 6: Room showing the way of calculation of H_0 and A_0 for peak HRR [4].

b. Smoke Source

Smoke mass is calculated according to equations 3.22 and 3.23 established by the Fire Protection Handbook [40].

i. Growth Stage

$$Smokemass = \frac{\dot{Q}}{\Delta H_{eff}} \qquad (3.20)$$

ii. Fully Developed Stage

$$Smokemass = \frac{\dot{Q}_{peak}}{\Delta H_{eff}} \tag{3.21}$$

where:

- $\dot{Q}$ is the heat release rate in kW

- $\dot{Q}_{peak}$ is the peak release rate in kW

- ΔH_{eff} is the effective heat of combustion in kJ/kg in Table 6

Table 6: Heat of combustion (HOF) and effective HOF for different type of wood [34]

Wood type	Heat of Combustion ΔH_c in MJ/Kg	Effective heat of combustion $\Delta H_{eff\ in}$ MJ/Kg
Beech	20	18.7
Birch	20	18.7
Douglas Fir	21	19.6
Maple	19.1	17.8
Red Oak	20.2	18.7
Spruce	21.8	20.4
While Pine	19.2	17.8

In addition, values for the different boundaries of the model describing walls, inlets and exhausts are implemented in the case.

3. Visibility

Klote and Milke [22] describe the visibility according to:

$$S = \frac{K}{\alpha_m \times m_p} \tag{3.22}$$

where:

- S = visibility through smoke in m

- K = proportionality constant from Table 7

- α_m = specific extinction coefficient (m²/kg) from Table 8

- m_p = mass concentration of particulate (kg/m³)

The predicted smoke mass fraction values are used to calculate m_p. Since m_p is in kg/m³ while concentration is in kg of smoke per kg of air, the results from the concentration equation will be multiplied by the density of air to get the required mass concentration of particulate to compute the target "visibility".

Table 7: Proportionality constant for different situations [22]

Situation	Proportionality Constant K
Illuminated Signs	8
Reflecting Signs	3
Building Components in Reflected Light	3

Table 8: Specific Extinction Coefficient for Smoldering and Flaming Combustion [22]

Mode of Combustion	Specific Extinction Coefficient α_m (m²/kg)
Smoldering Combustion	4301
Flaming Combustion	7578

4. Exposure to Smoke

The relationship between the concentration of smoke in the air and the time to lose consciousness is presented by Purser, D. in SFPE handbook of Fire Protection Engineering [34] for an average person with high level of activity as follows:

$$t = \frac{30}{8.2925 \times 10^{-4} \times (10^4 \times C)^{1.036}}$$

(3.23)

where:

- t = time to incapacitation in min

- C = concentration calculated by the simulation in kg_{smoke}/kg_{air}

5. *Exposure to Heat*

For the exposure of human to heat [34], Purser proposes the following equation that relates temperature to the time of consciousness is:

$$t = e^{5.1849 - 0.0273 \times (T - 273)}$$ (3.24)

where:

- t is the time to incapacitation in min

- T is the temperature in kelvin

D. Finite Volume Discretization

Discretization transform the Navier-Stokes equations from analytical state into a numeric form readable by the computer. It is first discretized in time followed by a spatial semi-discretization at each time step. Several spatial discretization forms can be used such as finite difference, finite element and finite volume methods; but in this report, we will present the finite volume method.

The general transport equation [24] is presented in equation (3.25):

$$\frac{\partial \rho \emptyset}{\partial t} + div(\rho \vec{u} \emptyset) = div(\Gamma grad \emptyset) + S$$ (3.25)

where

- $\frac{\partial \rho \emptyset}{\partial t}$ is the unsteady term

- $div(\rho \vec{u} \emptyset)$ is the convection term

- $div(\Gamma grad \emptyset)$ is the diffusion term

- S is the source term
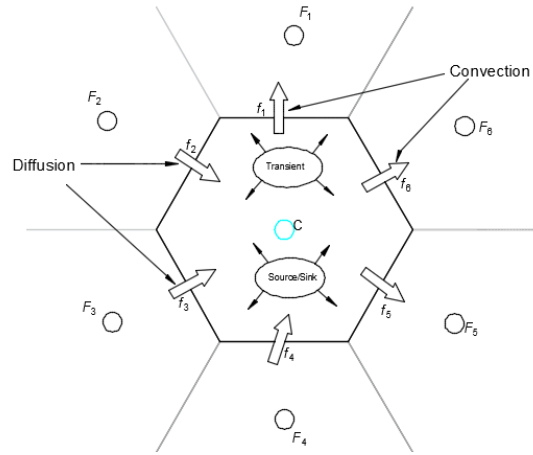
Figure 7: Explanatory control volume showing the contribution of each term of the general transport equation [24]

The integration of general transport equation over the control volume V is presented in equation (3.26) [24]:

$$\int_V \int_t^{t+\Delta t} \frac{\partial \rho \emptyset}{\partial t} \, dt dV + \int_t^{t+\Delta t} \int_A (\vec{n}. \rho \vec{u} \emptyset) \, dA dt =$$

$$\int_t^{t+\Delta t} \int_A (\vec{n}. \Gamma \vec{\nabla} \emptyset) \, dA dt + \int_V \int_t^{t+\Delta t} S \, dt dV \qquad (3.26)$$

1. Temporal Discretization

For the time discretization, numerous time schemes can be considered such as Explicit Euler Method, Implicit Euler Method, Crank–Nicholson Method, etc. [30].

The Explicit Euler Method uses a forward differencing for the time derivative that gives

$$\frac{u^{k+1} - u^k}{\tau} = -(u^k. \nabla)u^k + v\Delta u^k - \nabla p^k + f^k \qquad (3.27)$$

for the velocity u^{k+1}, given u^k and $f^k = f(x, k\tau)$

26

The implicit Euler method is based on a backward time differencing and gives

$$\frac{u^{k+1}-u^k}{\tau} + \left(u^{k+1}.\nabla\right)u^{k+1} - v\Delta u^{k+1} + \nabla p^{k+1} = f^{k+1} \ \&\mathbf{div}\ \mathbf{u}^{k+1} = \mathbf{0} \qquad (3.28)$$

with $f^{k+1}(x) = f(x,(k+1)\tau)$.

In this method, the modified Navier Stokes equations are solved in each time step.

In the Crank Nicholson method, the average between the implicit and explicit Euler method is considered:

$$\frac{u^{k+1}-u^k}{\tau} + \left(u^{k+1/2}.\nabla\right)u^{k+1/2} - v\Delta u^{k+1/2} + \nabla p^{k+1/2} = f^{k+1/2} \ \&\mathbf{div}\ \mathbf{u}^{k+1} = \mathbf{0} \qquad (3.29)$$

with $u^{k+1/2} = \frac{1}{2}(u^{k+1}+u^k), p^{k+1/2} = \frac{1}{2}(p^{k+1}+p^k), f^{k+1/2} = \frac{1}{2}(f^{k+1}+f^k)$

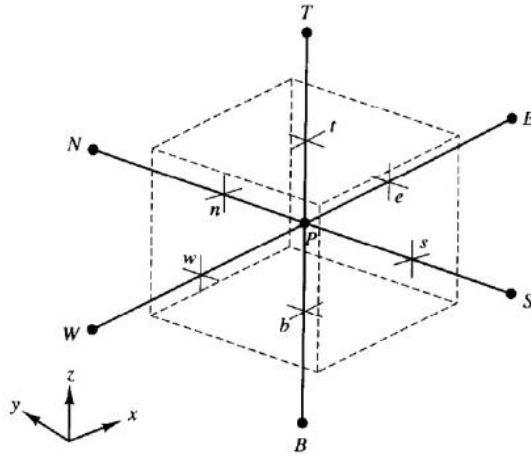## *2. Spatial Discretization*

Figure 8: 3-D control volume and its neighbors [30]

As shown in Figure 8, a three-dimensional control volume is presented; where P represents the central control volume and the neighbor control volumes are symbolized by W, E, S, N, T & B for west, east, south, north, top and bottom cells respectively. Nodes and faces are represented by capital and small letters respectively.

a. Diffusion Term Discretization

i. Orthogonal Grid
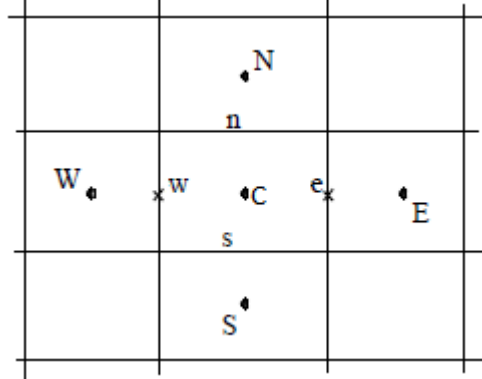
Figure 9: 2-D orthogonal grid showing the central control volume and its neighbors

For simplicity purposes, an orthogonal 2D grid (Figure 9) and a steady state diffusion equation are considered to proceed with the discretization [24].

The steady state diffusion equation:

$$-\nabla.\left(\Gamma^{\emptyset}\nabla\emptyset\right) = Q^{\emptyset} \tag{3.30}$$

Where $\emptyset$ is scalar variable (e.g. temperature, mass fraction of species, etc.)

Now, consider the diffusion flux: $J^{\emptyset,D} = -\Gamma^{\phi}\nabla\emptyset$

Then equation becomes

$$\nabla.J^{\emptyset,D} = Q^{\emptyset} \tag{3.31}$$

Applying the first stage discretization, equation (3.31) can be written as

$$\sum_{f\sim nb(C)}\left(J^{\emptyset,D}\right).S_f = Q_C^{\emptyset}V_C \tag{3.32}$$

Following several integration and substitution, the diffusion equation can be formulated as

$$a_C\emptyset_C + \sum_{F\sim NB(C)}a_F\emptyset_F = b_C \tag{3.33}$$

with

- $a_F = Flux\ F_f = -\Gamma_f^{\phi} gDiff_f$

- $a_C = \sum_{f \sim nb(C)} FluxC_f$

- $b_C = Q_C^{\phi} V_C - \sum_{f \sim nb(C)} FluxV_f$

where

- $gDiff_f = \dfrac{(\Delta y)_f}{\delta x_f} = \dfrac{\|\vec{S}_f\|}{\|\vec{d}_{CF}\|} = \dfrac{S_f}{d_f}$

- $FluxT_f = -\Gamma_f^{\phi} (\Delta y)_f \dfrac{(\phi_F - \phi_C)}{\delta x_f} = \Gamma_f^{\phi} \dfrac{(\Delta y)_f}{\delta x_f} (\phi_C - \phi_F) = FluxC_f \phi_C +$

$FluxF_f \phi_F + FluxV_f$

F represents the neighbors of element C (E, W, N, S) and f represents the neighboring faces

of element C (e, w, n, s)

ii. Non-Orthogonal Unstructured Grid
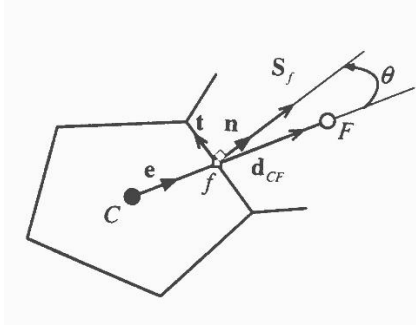
Figure 10: Non-orthogonal unstructured grid showing the central control volume [24]

In the non-orthogonal unstructured grid, $\vec{S}_f$ isn't collinear with the $\overrightarrow{CF}$ as shown in

Figure 10. Thus,

$$\vec{S}_f = \vec{E}_f + \vec{T}_f \tag{3.34}$$

$$(\nabla\phi)_f \cdot \vec{S}_f = (\nabla\phi)_f \cdot \vec{E}_f + (\nabla\phi)_f \cdot \vec{T} = E_f \left(\frac{\partial\phi}{\partial e}\right)_f + (\nabla\phi)_f \cdot \vec{T} = E_f \frac{\phi_F - \phi_C}{d_{CF}} + (\nabla\phi)_f \cdot \vec{T} \tag{3.35}$$

Table 9: Values of $\overrightarrow{E_f}$ & $(\nabla\emptyset)_f.\overrightarrow{T}$ for the different approaches [24]

Approach	$\overrightarrow{E_f}$	$(\nabla\emptyset)_f.\overrightarrow{T}$
Minimum Correction	$(S_f\cos\theta)\vec{e}$	$(\nabla\emptyset)_f.(\vec{n}-\cos\theta\,\vec{e})S_f$
Orthogonal Correction	$S_f\vec{e}$	$(\nabla\emptyset)_f.(\vec{n}-\vec{e})S_f$
Over-Relaxed	$\dfrac{\vec{S_f}.\vec{S_f}}{\vec{e}.\vec{S_f}}\vec{e}$	$(\nabla\emptyset)_f.(\vec{n}-\dfrac{1}{\cos\theta}\vec{e})S_f$

The final form of the discretized diffusion term in a non-orthogonal grid is

$$\mathbf{a_C\emptyset_C} + \sum_{\mathbf{F\sim NB(C)}} \mathbf{a_F\emptyset_F} = \mathbf{b_C} \tag{3.36}$$

with

- $a_F = Flux\ F_f = -\Gamma_f^\phi gDiff_f$

- $a_C = \sum_{f\sim nb(C)} FluxC_f = \Gamma_f^\phi gDiff_f$

- $b_C = Q_C^\emptyset V_C - \sum_{f\sim nb(C)} FluxV_f = Q_C^\emptyset V_C - \sum_{f\sim nb(C)}\left(\Gamma^\phi\nabla\emptyset\right)_f.\overrightarrow{T_f}$

where

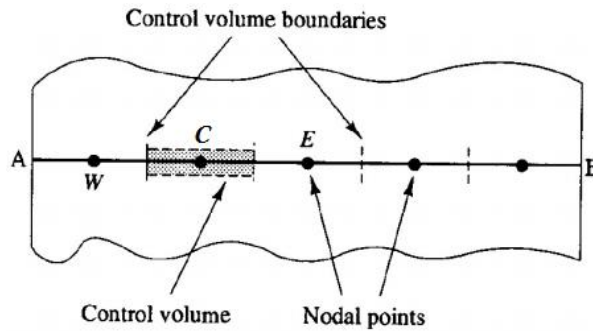- $gDiff_f = \dfrac{E_f}{d_{CF}}$

b. Convection Term Discretization

Figure 11: 1-Dorthogonal grid showing the central control volume and its neighbors [30]

In this section, a one dimensional (Figure 11) convection diffusion equation is used to proceed with the discretization.

$$\frac{d}{dx}(\rho u \emptyset) = \frac{d}{dx}\left(\Gamma^\phi \frac{d\emptyset}{dx}\right)$$
(3.37)

After discretization of equation (3.37) over a one dimensional element, we get

$$\int_{V_C} \nabla.\left(\vec{J}^{\emptyset,C} + \vec{J}^{\emptyset,D}\right) = 0$$
(3.38)

where $\vec{J}^{\emptyset,C} = \rho \vec{v} \emptyset$ and $\vec{J}^{\emptyset,D} = -\Gamma^\phi \frac{d\emptyset}{dx}$

Applying the divergence theorem on equation (3.38) yields

$$\int_{V_C} \nabla.\left(\vec{J}^{\emptyset,C} + \vec{J}^{\emptyset,D}\right) = \int_{\partial V_c}\left(\vec{J}^{\emptyset,C} + \vec{J}^{\emptyset,D}\right).d\mathbf{S} = 0$$
(3.39)

By switching the surface integral to summation, the equation becomes

$$\left[(\rho u \Delta y \emptyset)_e - \left(\Gamma^\emptyset \frac{d\emptyset}{dx}\Delta y\right)_e\right] - \left[(\rho u \Delta y \emptyset)_w - \left(\Gamma^\emptyset \frac{d\emptyset}{dx}\Delta y\right)_w\right] = 0$$
(3.40)

Since the continuity equation is satisfied and the problem is one dimensional, the convection diffusion equation is written after adopting the central difference scheme as preliminary derivation

$$a_C \emptyset_C + a_E \emptyset_E + a_W \emptyset_W = 0$$
(3.41)

with

- $a_E = -\frac{\Gamma_e^\phi}{\delta x_e} + \frac{(\rho u)_e}{2}$

- $a_W = -\frac{\Gamma_w^\phi}{\delta x_w} + \frac{(\rho u)_w}{2}$

- $a_C = -(a_E + a_W)$

The value of Ø at the cell faces is calculated according to the adopted scheme. Several spatial discretization schemes can be selected: Upwind, Downwind, Central Difference, Second Order Upwind, Quick Scheme, etc. [30].

c. Source Term Discretization

In order to discretize the source term, the general form of the conservation equation with an explicit source term is presented in equation (3.42) [24].

$$\mathbf{a_C \emptyset_C + \sum_{F \sim NB(C)} a_F \emptyset_F = Q_C^\emptyset V_C} \tag{3.42}$$

where $Q_C^\emptyset = Q(\emptyset_C)$ represents the source term in function of the scalar variable Ø.

Using Taylor-like series expansion to linearize $Q_C^\emptyset$, we get

$$\mathbf{Q(\emptyset_C) = Q(\emptyset_C^*) + \left(\frac{\partial Q}{\partial \emptyset_C}\right)^* (\emptyset_C - \emptyset_C^*) = \left(\frac{\partial Q}{\partial \emptyset_C}\right)^* \emptyset_C + Q(\emptyset_C^*) - \left(\frac{\partial Q}{\partial \emptyset_C}\right)^* \emptyset_C^*} \tag{3.43}$$

where "*" denotes the values of the previous iteration.

Thus, the source can be calculated as

$$\mathbf{Q_C^\emptyset V_C = \iint_{V_C} Q^\emptyset dV = \iint_{V_C} \left(\frac{\partial Q_C^*}{\partial \emptyset_C} \emptyset_C\right) dV + \iint_{V_C} \left(Q_C^* - \frac{\partial Q_C^*}{\partial \emptyset_C} \emptyset_C^*\right) dV}$$

$$\mathbf{= \left(\frac{\partial Q_C^*}{\partial \emptyset_C} V_C\right) \emptyset_C + \left(Q_C^* - \frac{\partial Q_C^*}{\partial \emptyset_C} \emptyset_C^*\right) V_C = FluxC_C \emptyset_C + FluxV_C} \tag{3.44}$$

The algebraic equation of the source term is written as

$$\mathbf{[a_C - FluxC_C]\emptyset_C + \sum_{F \sim NB(C)} a_F \emptyset_F = FluxV_C} \tag{3.45}$$

CHAPTER IV

COMPUTATIONAL APPROACH

A. Case Geometry
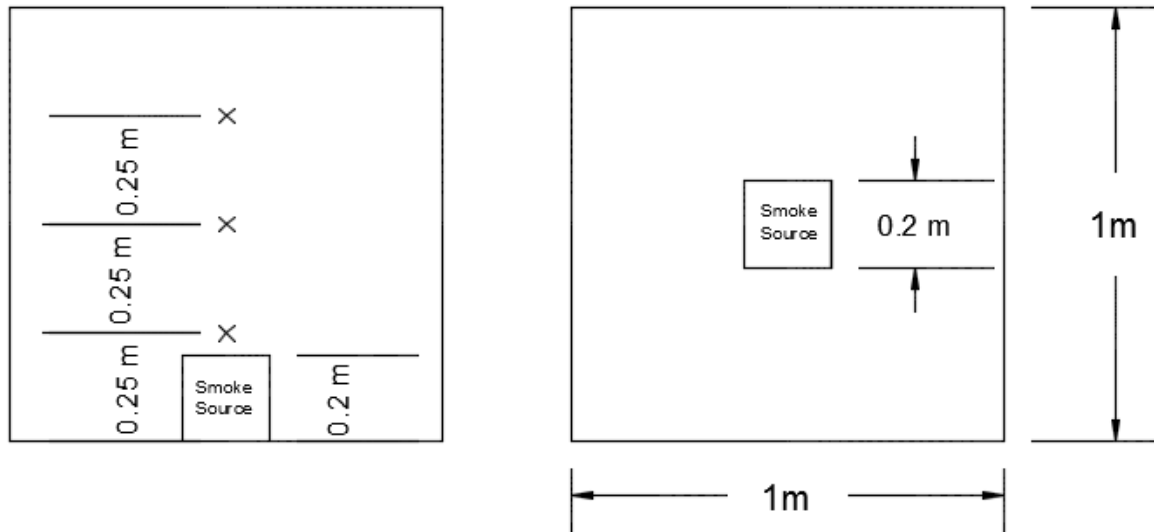
Figure 12 : Case Geometry

The test case is a room having the dimensions of 1x1x1 m^3 with a smoke source of 0.2x0.2x0.2 m^3 located at the floor in the middle of the space as shown in Figure 12. Three points having respectively the heights of 0.25m, 0.5 m and 0.75 m from the floor surface are selected to be studied in the analysis section.

B. Computational Setup

As a first step in the simulation, the drawing and the mesh generation of the case were done using fluent-ANSYS; then saved as ASCII mesh type in order to export it to ".msh" file. "FluentMeshtofoam" is the term used in OpenFOAM$^©$ to import and recognize the mesh of the drawing. Figure 13 shows the case and its mesh using the pre-processing software "paraFoam".
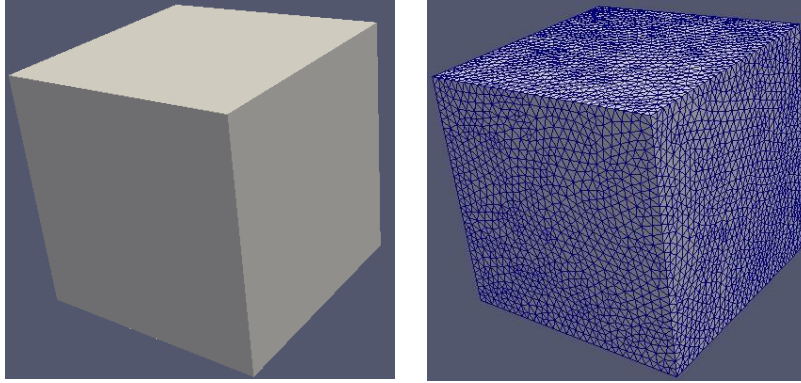
Figure 13: Scheme showing the test case (left) and its mesh (right) on OpenFOAM©

C. Numerical Scheme

OpenFOAM© offers a wide range of choices to assign a numerical scheme. In the simulations, first and second order numerical behavior were implemented and summarized in Table 10 which shows the scheme for each subcategory.

Table 10: Numerical behavior for each scheme used in the test case

Subcategory	1st Order Numerical Scheme	2nd Order Numerical Scheme
Transient	Crank Nicolson 0.5	Crank Nicolson 0.5
Gradient	Gauss linear	Gauss linear
Laplacian	Gauss linear limited 0.5	Gauss linear limited 0.5
Divergence	Gauss upwind	Gauss linear

D. Simulations Description

Three simulations were performed in both software (OpenFOAM & ANSYS) as summarized in Table 11. In all cases, the worst case scenario was applied with no smoke removal systems or sprinklers.

Simulation 1: Fast fire growth $\alpha = 47$ W/s^2 in a closed area with second order numerical scheme and k-epsilon turbulence model.

Simulation 2: Fast fire growth $\alpha = 47$ W/s^2 in a closed area where first order numerical scheme and k-epsilon turbulence model.

Simulation 3: Fast fire growth $\alpha = 12$ W/s^2 in a closed area where second order numerical scheme and k-epsilon turbulence model.

Table 11: Different numerical approach

	Numerical Scheme Order	Turbulence Model	Fire Growth Coefficient α
Simulation 1 (Base Case)	2nd Order	K-ε	47
Simulation 2	1st Order	K-ε	47
Simulation 3	2nd Order	K-ε	12

E. Mesh and Grid

The grid consists of 244,776 cells having tetrahedral and prism shape. Three boundary types are involved: walls, ceiling, and floor. Table 12 shows the number of faces and points for each boundary.

Table 12: List of boundaries with their number of faces and points

Boundary	Number of faces	Number of points
Wall	6780	3506
Floor	1692	905
Ceiling	1720	919

F. Peak Heat Release Rate Calculation

According to equation 17, peak heat release rate for fuel control fire is calculated as:

$$Q_{\text{max fuel}} = 250 \times A_f = 250 \times 0.04 = 10 \; kW$$

G. Initial Values

The initial values used in the code are shown in Table 13.

Table 13: Initial values for the test case

Properties	Value
Dynamic Viscosity (m^2/s)	1e-05
Beta (1/K)	3e-03
Reference Temperature (K)	293.15
Laminar Prandtl	0.713
Turbulent Prandtl	0.9
Laminar Schmidt	0.7
Turbulent Schmidt	0.81
Density (Kg/m^3)	1.2
Heat of Combustion (KJ/Kg)	19.5e3

CHAPTER V

RESULTS AND DISCUSSIONS

A. Simulation 1: Base Case

After simulating the base case in both OpenFOAM$^©$ and ANSYS, a set of comparative plots for velocity (m/s), temperature (K), concentration (kg_{smoke}/kg_{air}), visibility (m) and time to lose consciousness due to smoke and heat exposure (min) are presented at the plume centerline for three heights 0.25 m, 0.5 m, 0.75 m. Growth and fully developed regions are the two stages studied in the fire simulation; where the time to reach fully developed one is 14.589 seconds.

As shown in Figures 14 to 16, the results of the studied parameters on the plume centerline generated by the source using both programs are in close proximity. The comparison of temperature plots in Figure 14 gives similar trends with a maximum difference of 2.78% during the growth period and 3% in the fully developed regime. A slight difference is noticed at the height of 0.25m where the slope of the results acquired from ANSYS is higher than OpenFOAM$^©$. However, at the heights of 0.5m and 0.75m, the difference is mainly in the fully developed regime, where temperature values obtained by ANSYS are higher than the ones generated by OpenFOAM$^©$.

As for the concentration of smoke presented in Figure 15, the results obtained from the simulations give similar profile as the one obtained for temperature; where the highest percentage difference is equal to 10.47% at a height of 0.25m in the growth region.
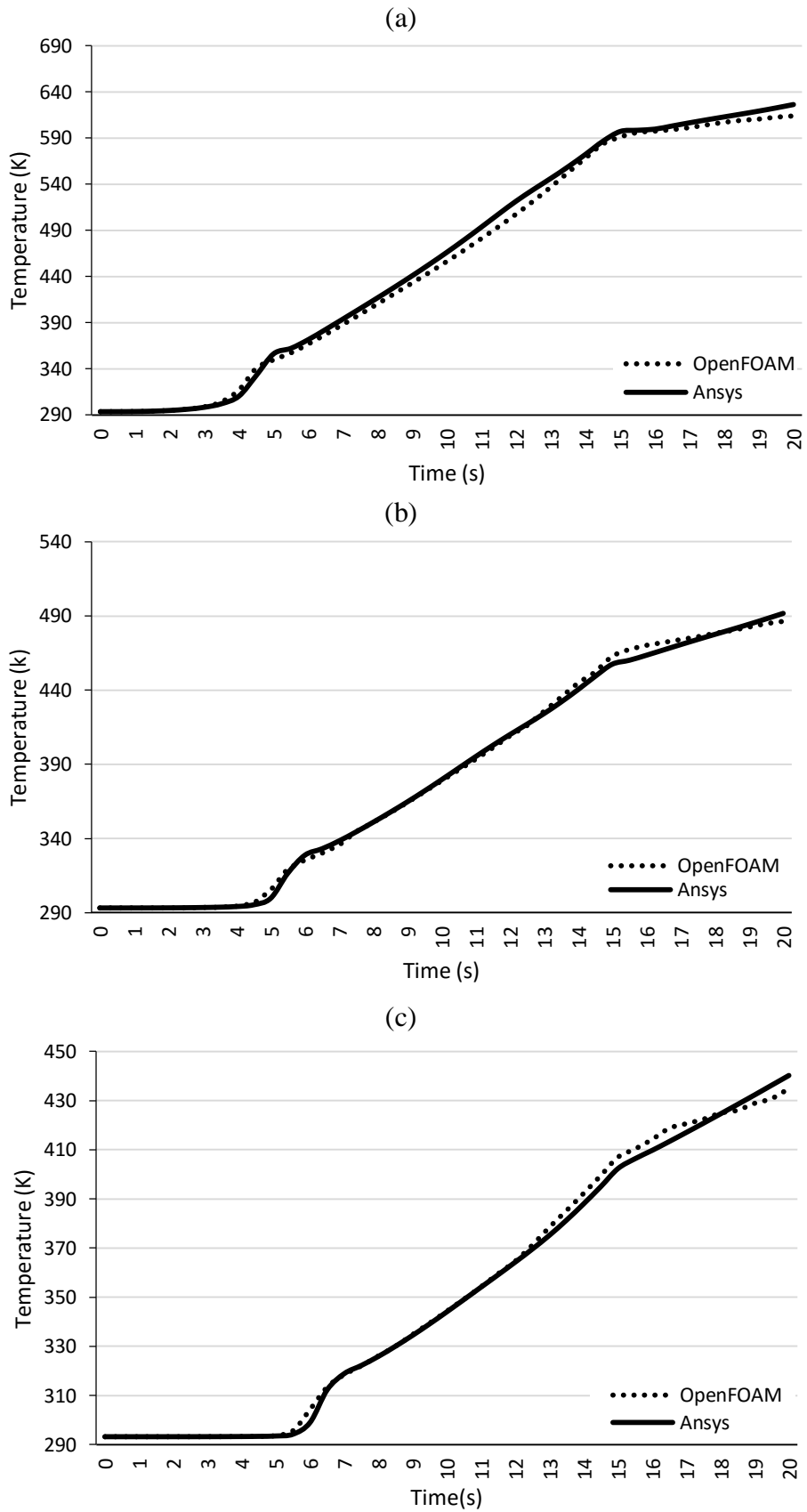
(a)

(b)

(c)

Figure 14: Temperature Variation at: (a) height 0.25m, (b) height 0.5m, and (c) height 0.75m for simulation 1
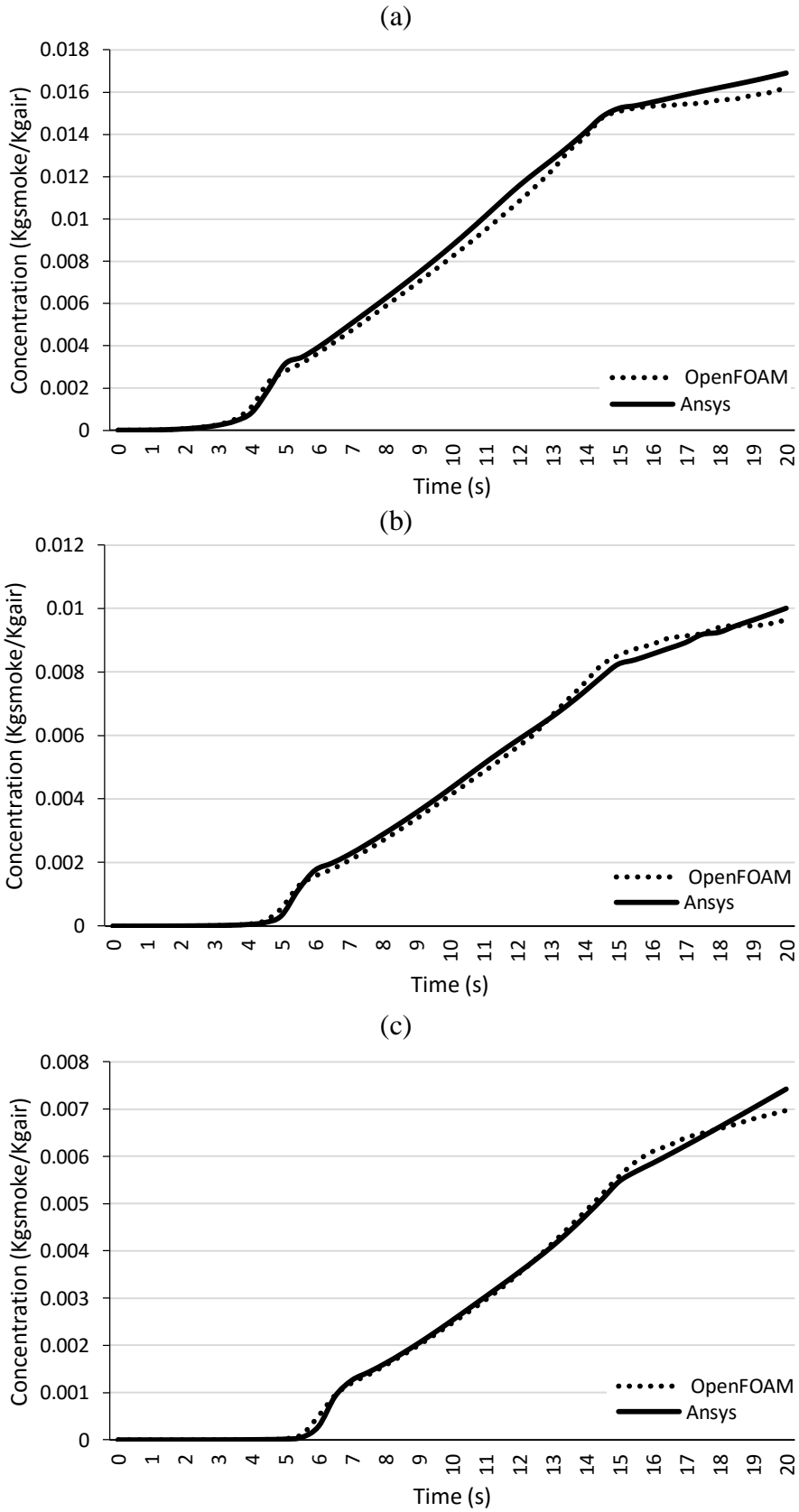
Figure 15: Concentration Variation at: (a) height 0.25m, (b) height 0.5m, and (c) height 0.75m for simulation1
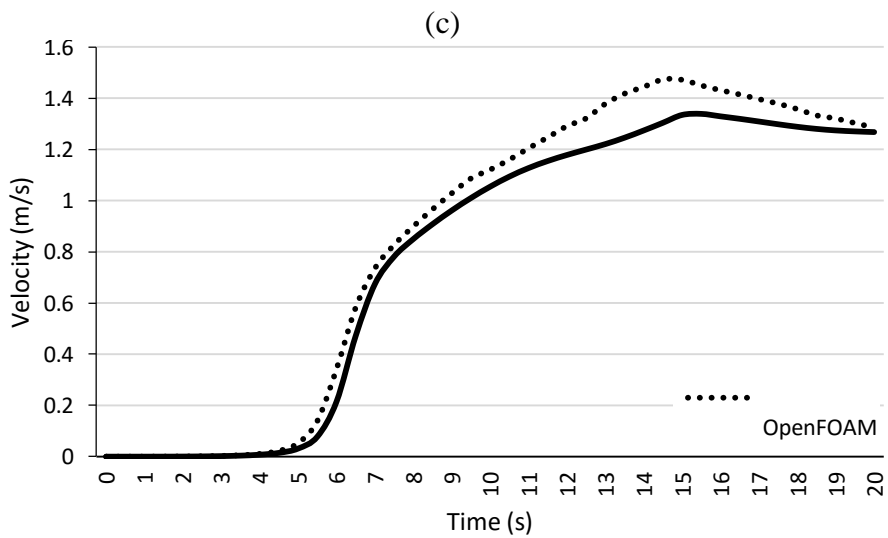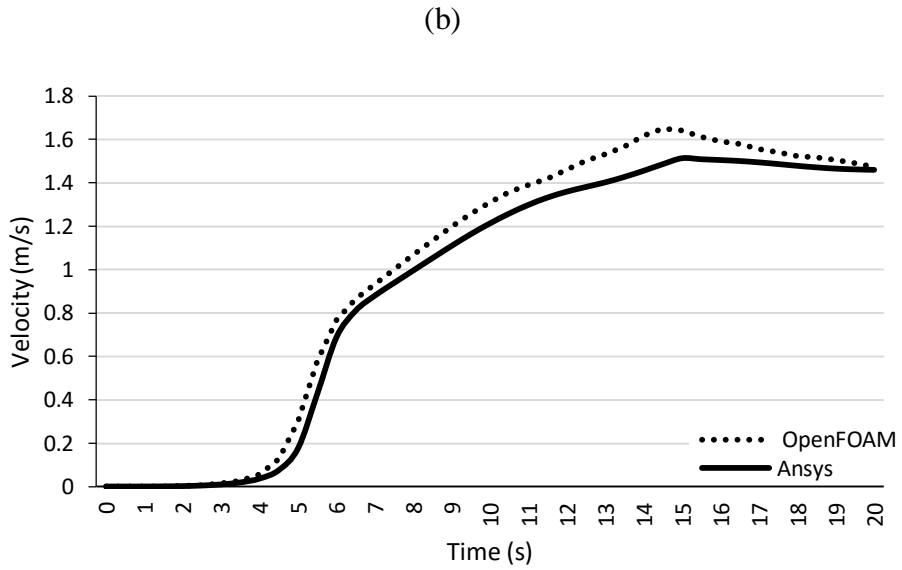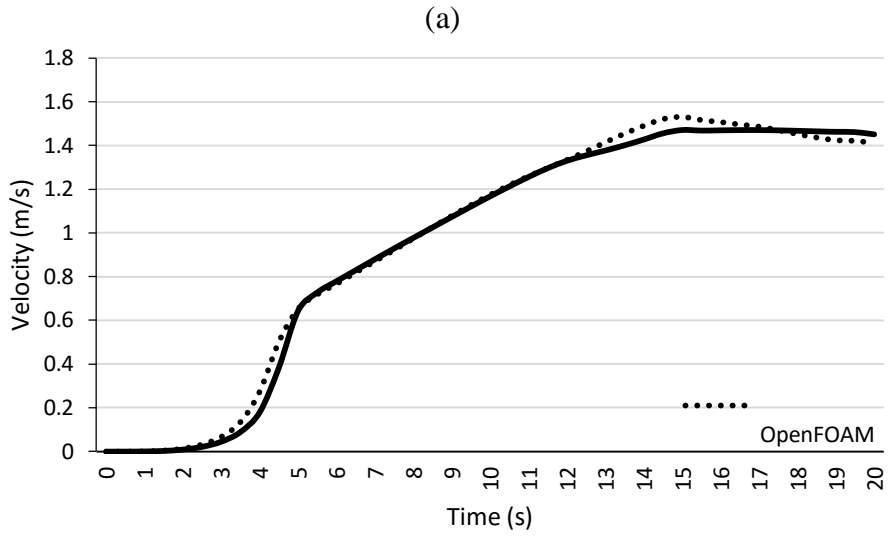
(a)

(b)

(c)

Figure 16: Velocity Variation at: (a) height 0.25m, (b) height 0.5m, and (c) height 0.75m for simulation 1

Regarding the velocity variation shown in Figure 16, the outcome of both software simulations are roughly the same with a parallel trend at a height near the source 0.25m. On the other hand, at higher heights of 0.5 m and 0.75 m, the calculated velocities using OpenFOAM$^{©}$ are higher than the ones predicted by ANSYS while following the same trend. As revealed in Figures 14 and 16, the growth phase is divided into two main periods, each possessing a different behavior. The first period extends over 5 s showing an exponential increase in the temperature and concentration field at the closest point to the fire location (located at 5 cm from top of the fire source). In this phase, the process is diffusion dominant, where the velocity field is not yet well established in the computational domain (Figure 16 depicts the variation of velocity magnitude at the closest point to the fire, y= 0.25 m). In the second period, both temperature and concentration, being majorly controlled by the convective process, continuously increases, however, at a lower rate as compared to the rate of increase during the first phase period.

In summary, results for temperature, smoke concentration and velocity obtained with both packages using a second order scheme were close. The reason behind the slight differences can be attributed to the manner by which each software is written. ANSYS is a black box that solves the same parameters and equations used in OpenFOAM$^{©}$ but in a way that makes it more efficient and less time consuming.
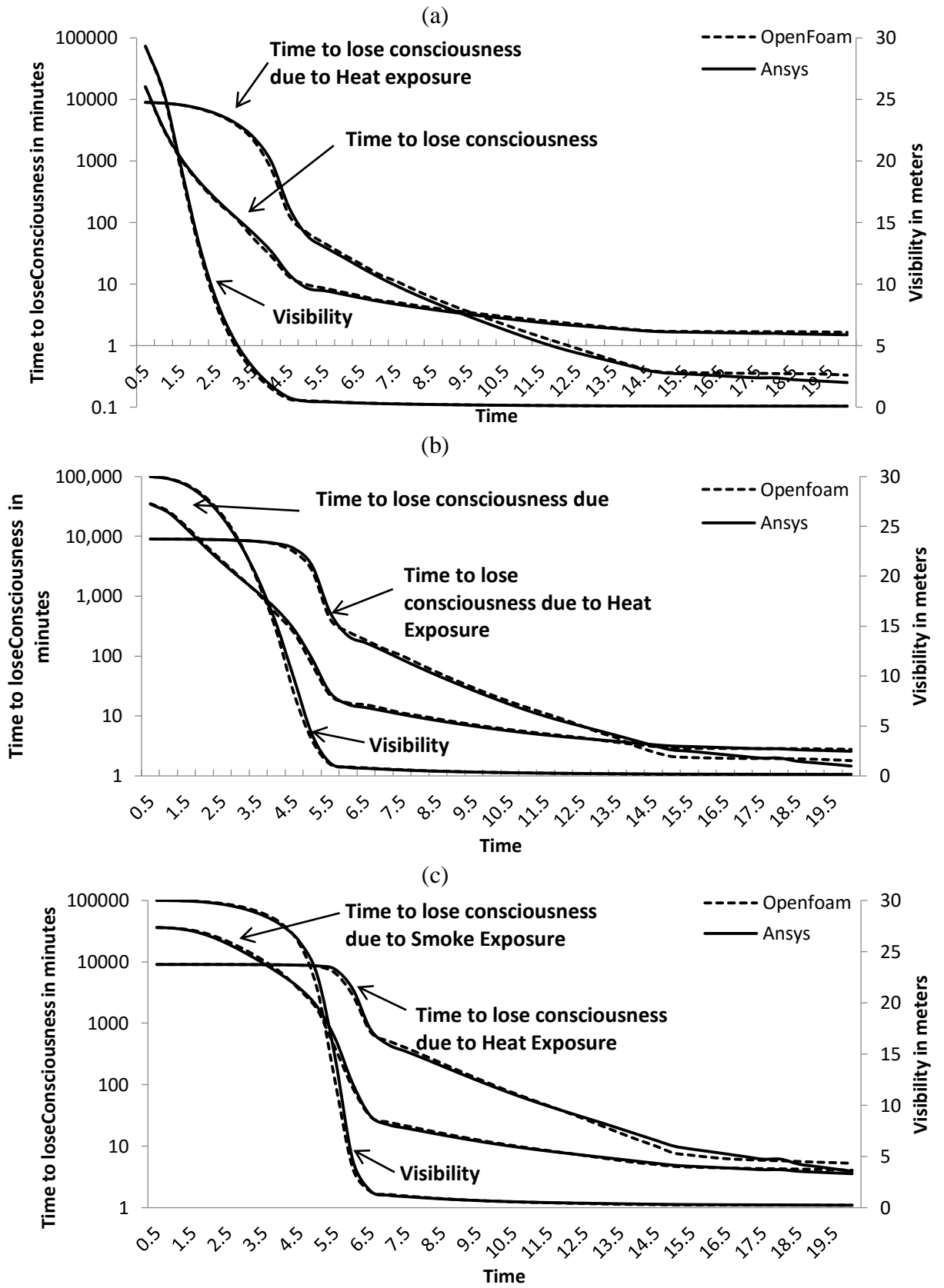
Figure 17: Effects of fire Variation at: (a) height 0.25m, (b) height 0.5m, and (c) height 0.75m for simulation1
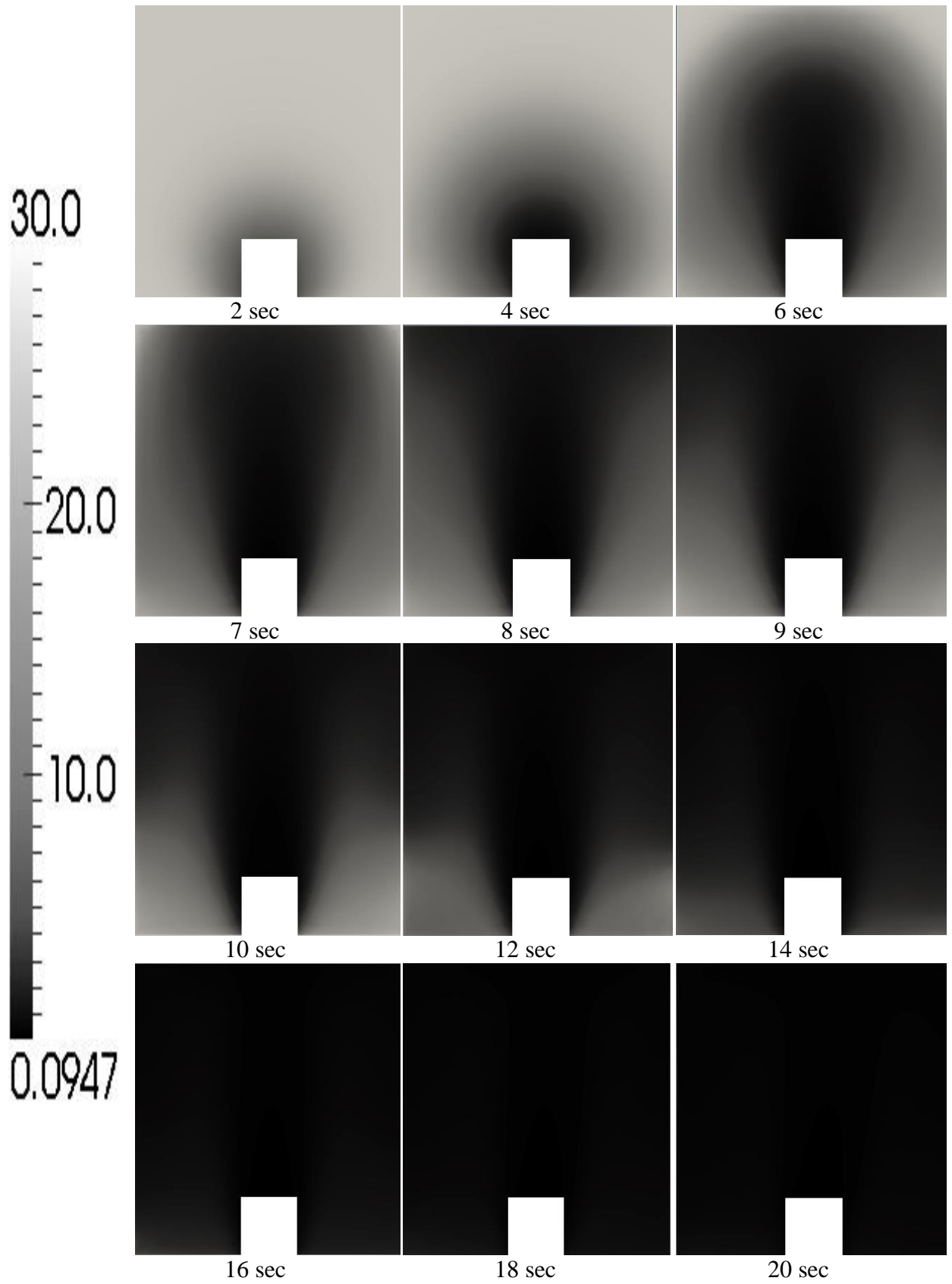
42

Figure 18: Middle section of the test case showing visibility in meters

As for the effects of fire, as related to visibility and time to lose consciousness due to heat and smoke exposure, results are presented in Figure 17; they are calculated according to Equations 22, 23 and 24. The increase in smoke production leads to a reduction in the visibility and to a higher rate in losing consciousness of the occupants in the area. These effects have higher values near the source and they decrease gradually with height. The comparison of the plots for both OpenFOAM© and ANSYS shows that there is minor difference between the values and maintain the same profile at all heights.

Visibility reduction caused by smoke intensification is illustrated at the middle section of the test room in Figure 18. The combustion process starts by the diffusion of smoke from the source where the visibility starts to decrease around the source. Then, the intensifying plume rises to reach the ceiling and starts to fill the room as described in the process elaborated earlier and in Figure 1. The different stages of the fire in the above figure clarifies the danger caused by fire.

The assessment of a safe design in any region should comply with the following restrictive values for a normal body height of 1.8 m as described by J. Schabacker [15]:

- The time of incapacitation should be less than 20 minutes.

- The range of vision should be greater than 5 m.

- The temperature should be less than 100°C or 373 K with an exposure time less than 10 min.

B. Simulation 2

For this case, plots for temperature, concentration and velocity at a height of 0.25m obtained from both packages are presented in Figure 19. Temperature and concentration have similar comparative profiles with the highest percentage difference equal to 4.13% and 12.55% respectively. Data from ANSYS has higher slope than OpenFOAM$^©$ in the fully developed region. Regarding velocity, the values predicted by OpenFOAM$^©$ are higher than the corresponding ones generated by ANSYS FLUENT while maintaining approximately the same profile variation.

Figure 20 shows the calculated results for 1st and 2nd order schemes implemented in OpenFOAM$^©$. The maximum percentage difference for temperature and concentration is equal to 2.63% and 4.88% respectively. Adopting the 1st order for the divergence scheme increases the error of the numerical discretization resulting in less accurate values than the ones obtained by high order schemes.

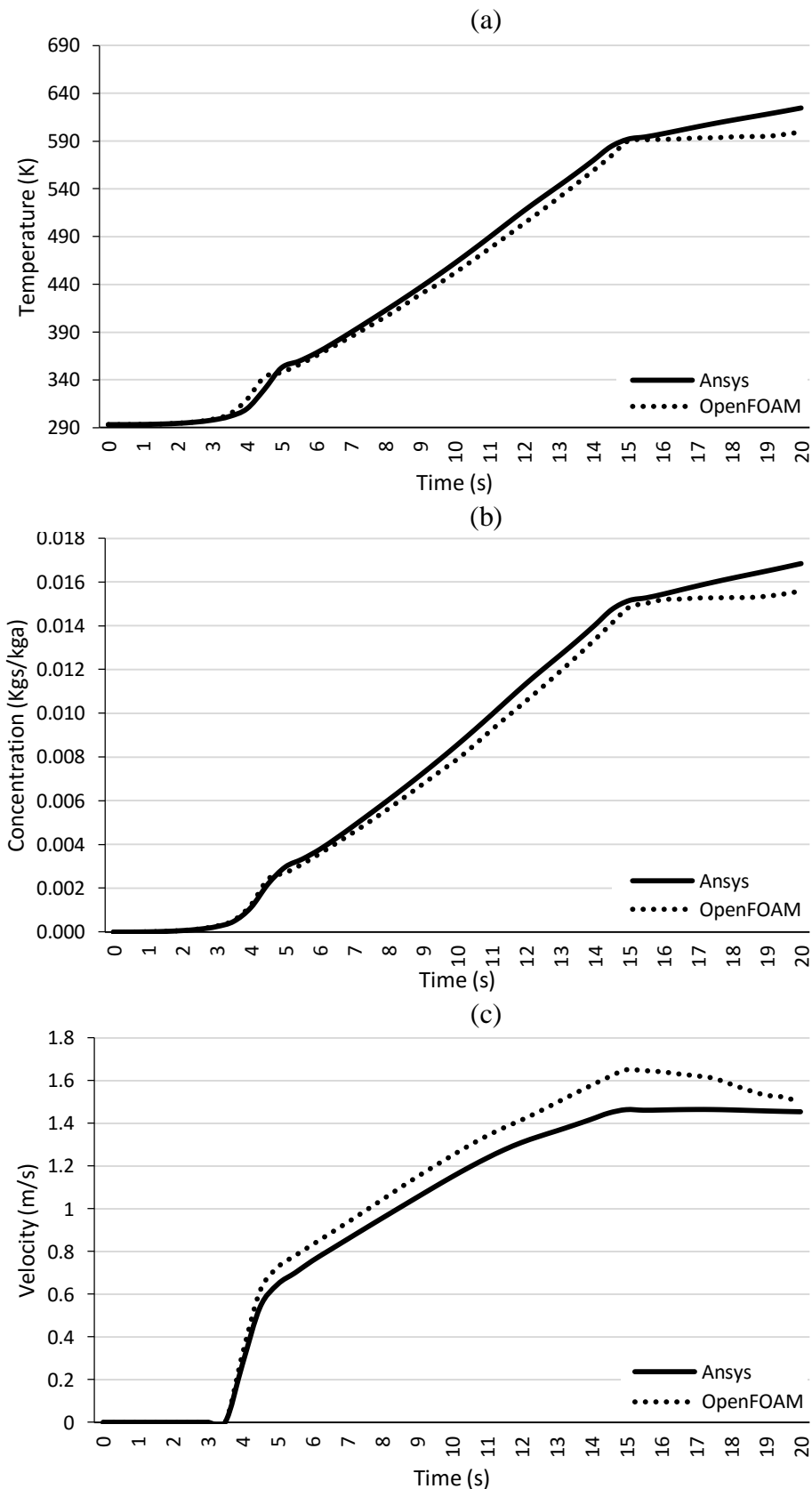Figure 19: At height 0.25m, the variation of: (a) Temperature, (b) Concentration, and (c) Velocity for simulation 2

Figure 20: At height 0.25m, the comparison of: (a) Temperature, (b) Concentration, and (c) Velocity for between simulations 1 and 2

C. Simulation 3

This case investigates the effects of the heat release rate on fire. For this purpose, the source term is reduced to $12\,\text{kW/s}^2$ to simulate a medium type fire. The first 20 seconds of the growth period are simulated using both software packages. The results for temperature and concentration obtained with OpenFOAM$^©$ and ANSYS are compared in Figure 21. Results show similar trends with the maximum relative differences in temperature and concentration values being 2 % and 8.9 %, respectively. The differences are approximately the same as for the base case (i.e. the differences were 2.78% for temperature and 10.47% for concentration). This indicates that even with different types of fire (different alphas), simulations using OpenFOAM$^©$ are expected to give appropriate results. Figure 22 compares results obtained in simulations 1 with $\alpha=47\,\text{W/s}^2$ that represents a fast fire with similar ones generated in simulation 3 with $\alpha=12\,\text{W/s}^2$ for medium fire. The comparison reveals as expected that both temperature and concentration reach higher values in case 1. In addition, the growth region in the medium fire takes more time to reach the fully developed one. For this reason, fire engineers should carefully assess the studied building to identify the different types of burned materials and the different scenarios expected to occur.

(a)

(b)

(c)

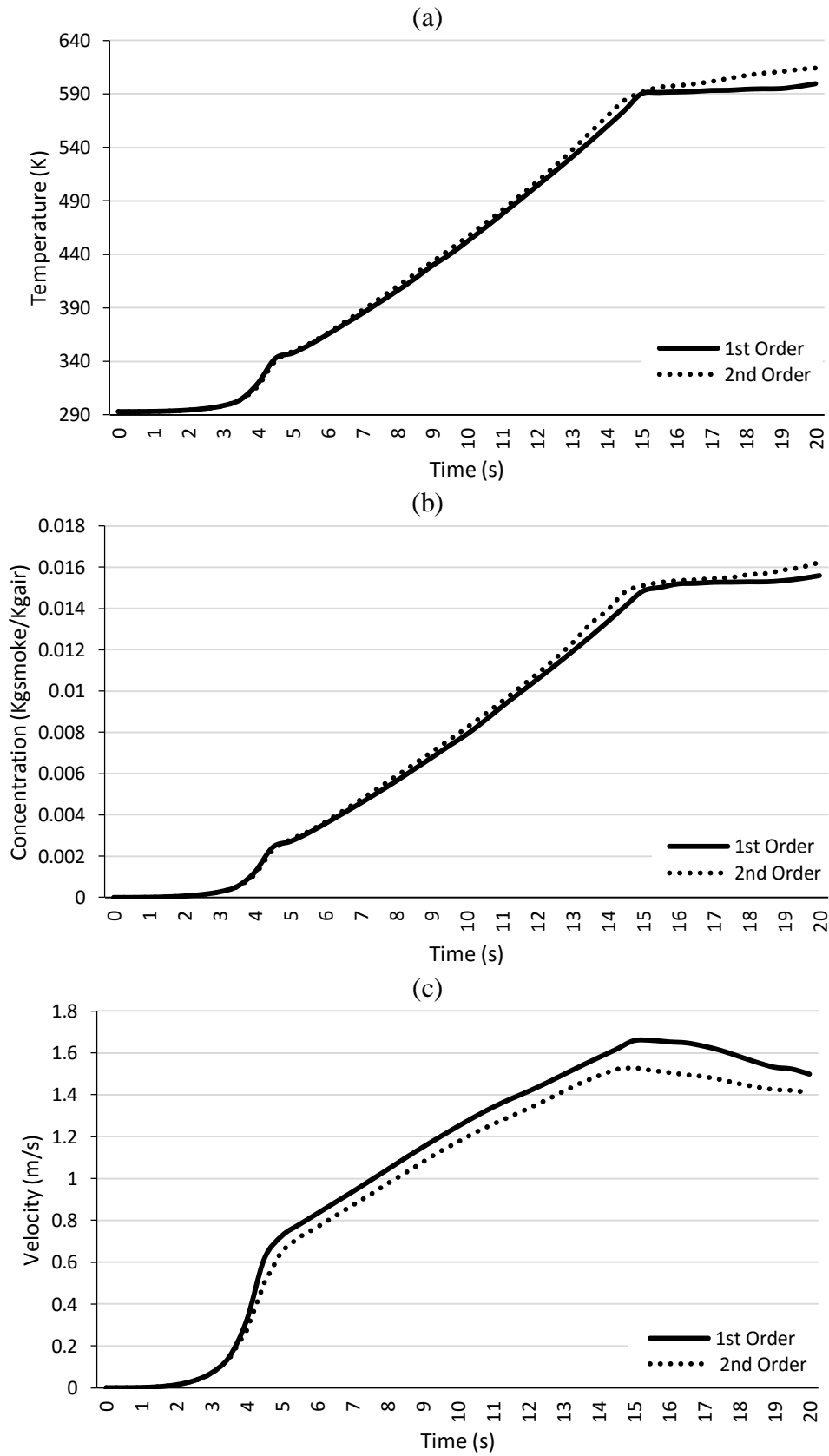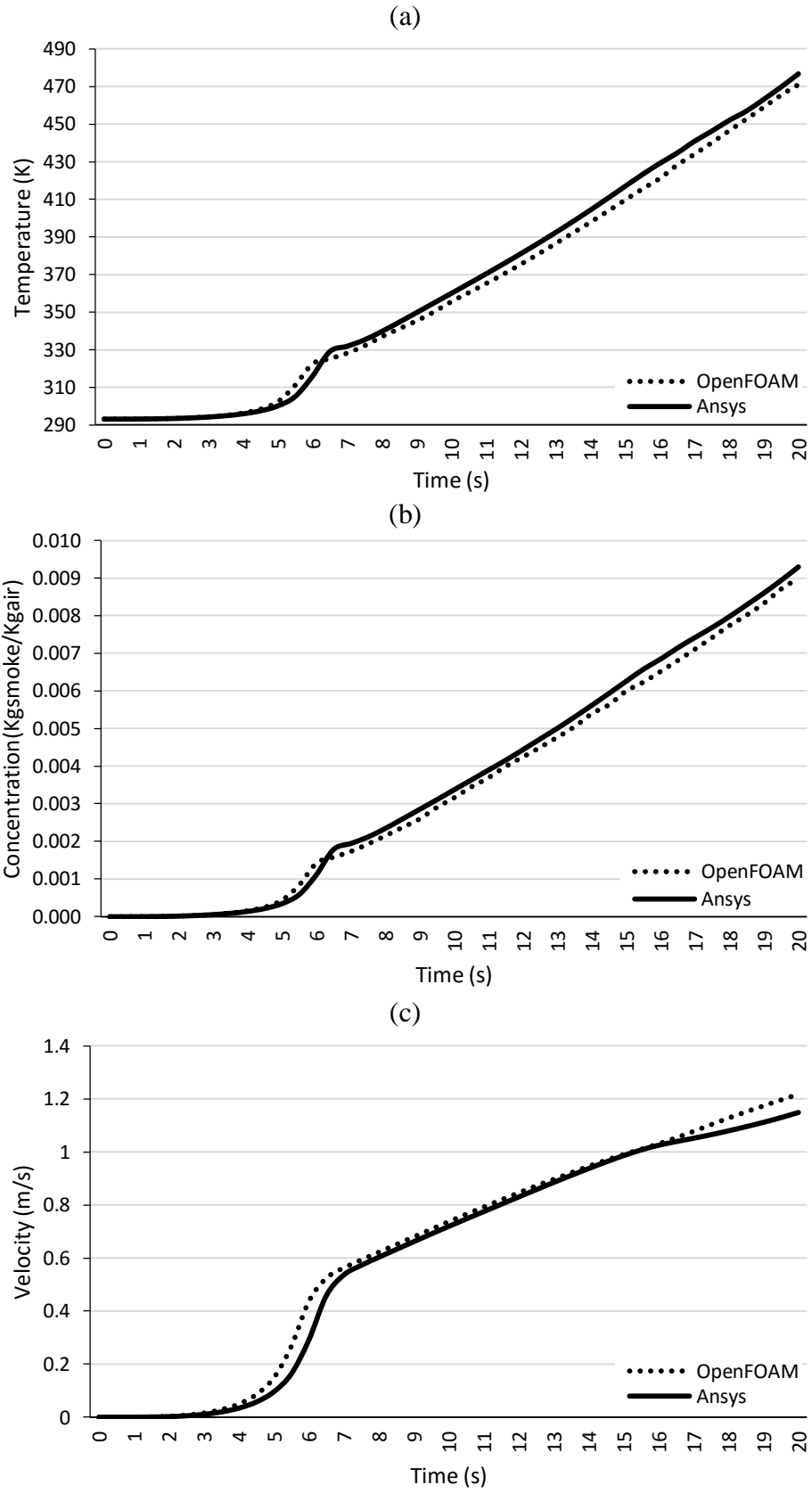Figure 21: At height 0.25m, the variation of: (a) Temperature, (b) Concentration, and (c) Velocity for simulation 3
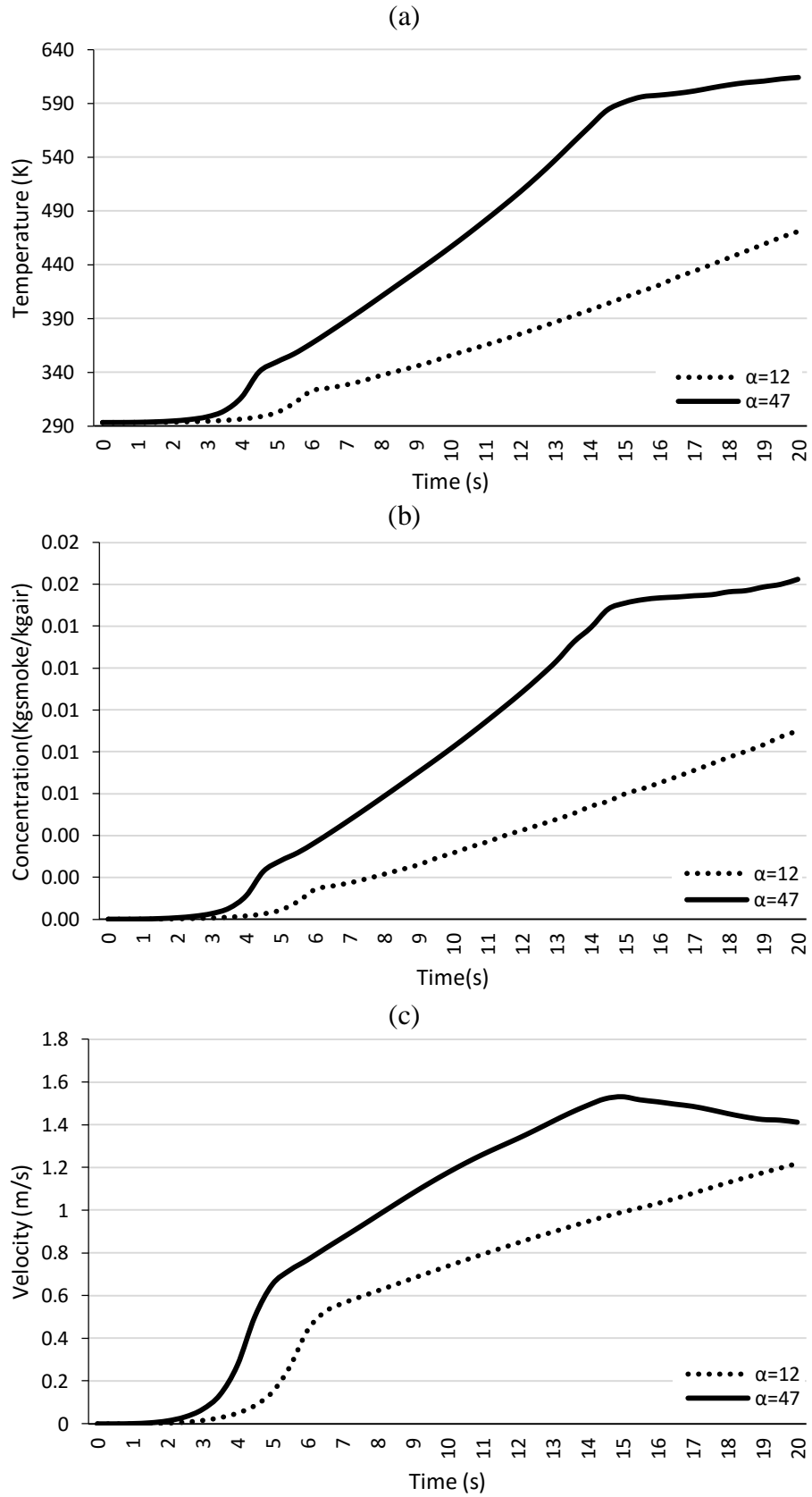
Figure 22: At height 0.25m, the comparison of: (a) Temperature, (b) Concentration, and (c) Velocity for between simulations 1 and 3

CHAPTER VI

CONCLUSION

A numerical study for smoke simulation inside buildings using the open source platform "OpenFOAM$^{©}$" was conducted. By using the SmokeFoam code, one can generate, after implementing the appropriate inputs, values of concentration and temperature of the smoke. These values can be used to calculate the visibility and time to lose consciousness from smoke and heat in order to produce a safe design of the studied building. The results obtained were compared to a commercial and reliable software called ANSYS FLUENT to validate its accuracy. The difference in results between the two software is minor and acceptable; this indicates that the OpenFOAM$^{©}$ code can be used in new designs and complex structures conducted by engineers to simulate smoke in case of fire. In addition, further improvement may be added to the code in order to expand its capabilities and enhance its efficiency.

APPENDIX

A. OpenFOAM© Code

1. Temperature Equation

```
{
    volScalarField tetaEff("tetaEff", turbulence->nu()/Pr + turbulence->nut()/Prt);
    fvScalarMatrix TEqn
    (
        fvm::ddt(T)
      + fvm::div(phi, T)
      + fvm::SuSp(-fvc::div(phi),T)
      - fvm::laplacian(tetaEff, T)
      - (alpha*runTime.time()*runTime.time()/(rho*Cp*V))*ss
    );
    TEqn.relax();
    solve(TEqn).print(Info);
}
```

2. Concentration Equation

```
{
    volScalarField zettaEff("zettaEff", turbulence->nu()/Sc+turbulence->nut()/Sct);
    fvScalarMatrix CEqn
    (
        fvm::ddt(C)
      + fvm::div(phi, C)
      + fvm::SuSp(-fvc::div(phi),C)
      - fvm::laplacian(zettaEff, C)
      - (alpha*runTime.time()*runTime.time()/(rho*HOC*V))*ss
    );
    CEqn.relax();
    solve(CEqn).print(Info);
}
```

3.Pressure Equation

```
volScalarField DU = 1.0/UEqn.A();
surfaceScalarField DUf("DUf",linearInterpolate(DU));
dimensionedScalar deltaT
("deltaT",dimensionSet(0,0,1,0,0,0,0),scalar(runTime.deltaTValue()));
surfaceScalarField DTf("DTf",DUf/deltaT);
surfaceVectorField U_avg_f = linearInterpolate(U);
```

```
surfaceVectorField U_avg_prevIter_f = linearInterpolate(U.prevIter());
surfaceVectorField U_old_f = linearInterpolate(U.oldTime());
surfaceScalarField& phi_old = phi.oldTime();
volVectorField gradP  = fvc::grad(p);
surfaceVectorField gradp_avg_f = linearInterpolate(gradP);
surfaceScalarField gradp_f = fvc::snGrad(p);
scalar URFU = mesh.equationRelaxationFactor("U");
surfaceVectorField B_reconstructed_f = linearInterpolate(B_reconstructed);
// Rhie-Chow interplation
phi = (U_avg_f & mesh.Sf())
    - DUf*( (gradp_f*mesh.magSf())-(gradp_avg_f&mesh.Sf()))
     + (scalar(1) - URFU)*(phi.prevIter() - (U_avg_prevIter_f & mesh.Sf()))
   + DTf*(phi_old - (U_old_f& mesh.Sf()))
   + DUf* ( (B_f - B_reconstructed_f)& mesh.Sf() )  ;
pp = scalar(0.0)*pp;
pp.correctBoundaryConditions();
fvScalarMatrix ppEqn
(
   - fvm::laplacian(DUf, pp, "laplacian(pDiff,pp)")
   + fvc::div(phi)
);
#include "continuityErrs.H"
ppEqn.setReference(pRefCell, pRefValue);
ppEqn.solve().print(Info);
phi += ppEqn.flux();
scalar URFPP = mesh.equationRelaxationFactor("pp");
p += URFPP*pp;
p.correctBoundaryConditions();
U -= fvc::grad(pp)*DU;
U.correctBoundaryConditions();
```

4. Velocity Equation

```
// Solve the Momentum equation
// B-Reconstructed
// Solve the Momentum equation
volVectorField B = -beta*(T - TRef)*g;
surfaceVectorField B_f = linearInterpolate(B);
volVectorField B_reconstructed (
                IOobject
                (
                "Breconstructed",
                runTime.timeName(),
                mesh,
                IOobject::NO_READ,
                IOobject::AUTO_WRITE
                ),
```

```
                    mesh,
                    dimensionedVector("zero", B.dimensions(), vector::zero),
                    "zeroGradient"

                    );
forAll(owner, facei)
{
    vector SfBfdf = (B_f[facei]&delta[facei])*Sf[facei];
    B_reconstructed[owner[facei]] +=  weights[facei]*SfBfdf;
    B_reconstructed[neighbour[facei]] += (1.0-weights[facei])*SfBfdf;
}


//Info<< "\n****** B_reconstructed  ******\n" <<B_reconstructed<< endl;

//Info<< "\n****** Momentum Equations 03 ******\n" << endl;

forAll(mesh.boundary(), patchi)
{
    const labelUList& pFaceCells = mesh.boundary()[patchi].faceCells();
        const vectorField& pSf = Sf.boundaryField()[patchi];
    const vectorField& pdeltaf = delta.boundaryField()[patchi];
    const vectorField& pB_f = B_f.boundaryField()[patchi];

    forAll(mesh.boundary()[patchi], facei)
    {
        vector SfBfdf =pSf[facei]*(pB_f[facei]&pdeltaf[facei]);
        B_reconstructed[pFaceCells[facei]] += SfBfdf;
    }
}

// divide by volume
B_reconstructed.internalField() /= mesh.V();
forAll(mesh.boundary(), patchi)
{
    const labelUList& pFaceCells = mesh.boundary()[patchi].faceCells();
    vectorField& pBf = B_reconstructed.boundaryField()[patchi];
    forAll(mesh.boundary()[patchi], facei)
    {
        pBf[facei] = B_reconstructed[pFaceCells[facei]];
    }
}
fvVectorMatrix UEqn
(
     fvm::ddt(U)
    + fvm::div(phi, U)
    + turbulence->divDevReff(U)
    + fvm::SuSp(-fvc::div(phi),U)
```

```
);
UEqn.relax();
solve
(
    UEqn == -fvc::grad(p) + B_reconstructed
).print(Info);
```

5. SmokeFOAM Code

```cpp
#include "fvCFD.H"
#include "singlePhaseTransportModel.H"
#include "RASModel.H"
#include "fvIOoptionList.H"
#include "pimpleControl.H"


// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
* * * * * * * * * //

int main(int argc, char *argv[])
{
#    include "setRootCase.H"
#    include "createTime.H"
#    include "createMesh.H"
#    include "createFields.H"
#    include "initContinuityErrs.H"
    pimpleControl pimple(mesh);
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
* * * * * * * * * //

    Info<< "\nStarting time loop\n" << endl;
    const unallocLabelList& owner = mesh.owner();
    const unallocLabelList& neighbour = mesh.neighbour();

    tmp<surfaceVectorField> tdelta = mesh.delta();
    const surfaceVectorField& delta = tdelta();

    const surfaceVectorField& Sf = mesh.Sf();

    tmp<surfaceScalarField> tweights = mesh.weights();
    const surfaceScalarField& weights = tweights();


    while (runTime.run())
    {
        #include "readTimeControls.H"
        #include "CourantNo.H"
        #include "setDeltaT.H"
```

```cpp
        runTime++;

        Info<< "Time = " << runTime.timeName() << nl <<
endl;

        scalar iter=0;
        while (pimple.loop())
        {
            iter++;
            U.storePrevIter();
            phi.storePrevIter();

            // Pressure-velocity SIMPLE corrector
            //Info << "Solving Ueq p\n" << endl;
            #include "UEqn.H"
            //Info << "Solving ppEqn p\n" << endl;
            #include "ppEqn.H"
            //
            #include "CEqn.H"
            // Info << "Solving Tqn p\n" << endl;
            #include  "TEqn.H"

            Info<< "------------------------------------"
<< endl << endl;
            if (pimple.turbCorr())
            {
                turbulence->correct();
            }
        }
        runTime.write();
                Info<< "ExecutionTime = " <<
runTime.elapsedCpuTime() << " s"
        << "  ClockTime = " << runTime.elapsedClockTime()
<< " s"
        << nl << endl;
    }
    Info<< "End\n" << endl;
    return(0);
}
```

BIBLIOGRAPGY

[1] ANSYS, Inc. (2015). *ANSYS*. Retrieved from http:// www.ansys.com

[2] Banjac, M. a. (2008). Numerical Study of Smoke Flow Control in Tunnel Fires Using Ventilation Systems. *FME Transactions , 36*, 145-150.

[3] Barakat, M. S. (1998). Smoke Data Determination for Various Types of Fuel . *Fire Safety J. , 30* , 293-306.

[4] B. Karlsson and J. G. Quintiere, Enclosure fire dynamics. Boca Raton, Florida: CRC Press, 2000, pp. 17-18, 39-43, 100, 121, 125, 129, 228-238, 244-253.

[5] B.S. Kandola and M. Morris, "Smoke hazard assessment using computational fluid dynamics(CFD) modeling", Institution of Chemical Engineers Symposium Series, No. 134, pp. 171-194 (1994).

[6] Boverket, *Boverkets allmänna råd 2011:xx - Vägledning i analytisk dimensionering av byggnaders brandskydd - Remiss. Karlskrona:* Boverket, 2010.

[7] British Standards Institution, BS 7974:2001, Application of fire safety engineering principles to the design of building - Code of practice. London: British Standards Institution, 2001, pp. 10, 15, 19-20.

[8] Cox, G. Chitty, R and Kumar, S. (1989). Fire Modeling and the King's Cross Fire Investigation. *Fire Safety J.* , Vol 15, pp. 103-106.

[9] D.J. den Boer, J.J. Mertens, N.J. van Oerle and J.F. Wijnia, "Validation of CFD calculations of full scale medium sized fires in a two lane road tunnel", 11th International Symposium on Aerodynamics and Ventilation of Vehicle Tunnels, Vol. 2, pp. 657-667 (2003).

[10] David A. Purser, 1995, 'Toxity Assessment of combustion products' in SFPE handbook of Fire Protection Engineering, 2nd edition, 1995, NFPE Publication 95-68247, Section 2 / Chapter 8.

[11] E.F.P. (2013, November 21). *eurofireprotection*. Retrieved from http://www.eurofireprotection.com

[12] European Standard, "Eurocode 1: Actions on structures - Part 1-2: General actions - Actions on structures exposed to fires," EN 1991-1-2, 2002.

[13] Hall, J.R., Jr., "Brief History of Home Smoke Alarms(Abridged)," Proceedings of the Research and Practice: Bridging the Gap Fire Suppression and Detection Research Application Symposium. Orlando, FL. February 7-9, 2001, National Fire Protection Research Foundation , Quincy, MA, 2001, pp.258-281.

[14] Institution, B. S. (2003). *Application of fire safety engineering principles to the design of buildings - Part 1: Initiation and development of fire within the enclosure of origin.* London: British Standards Institution.

[15] J. Schabacker et al. (n.d.). *CFD Study of Temperature and Smoke Distribution in a Railway Tunnel with Natural Ventilation System.* Switzerland: HBI Haerter AG.

[16] J. H. Ferziger and M. Peric, *Computational Methods for Fluid Dynamics*, 3rd, revised, Springer-Verlag, pp 9-10, 2002

[17] F. M. White, *Viscous Fluid Flow*, 2nd, McGraw-Hill, pp 73, 1991

[18] K.D. Steckler, J.G. Quintiere and W.J. Rinkinen, "Flow induced by fire in a compartment", Symposium (International) on Combustion, pp. 913-920 (1982).

[19] Kerrison, L., Mawhinneey, N., Galea, E.R., Hoffmann, N. and Patel, M.K. (1994a). A comparison of Two Fire Fields Models with Experimental Room Fire Dara, Fire Safety Science, pp. 161-172.

[20] Kevin McGrattan, Bryan Klein, Simo Hastikka, and Jason Floyd. Fire Dynamics Simulator (Version 5) User's Guide. July, 2007. NIST Building and Fire Research Laboratory. Gaithersburg Maryland USA . NIST Special Publication 1019-5.

[21] Kevin McGrattan, Simo Hastikka, Jason Floyd, Howard Baum, and Ronald Rehm. Fire Dynamics Simulator (Version 5) Technical Reference Guide. October, 2007. NIST Building and Fire Research Laboratory. Gaithersburg Maryland USA . NIST Special Publication 1018-5.

[22] Klote, J.H. and Milke, J.A. (2002). Principles of Smoke Management. American Society of Heating, Refrigerating and Air-Conditioning Engineers & Society of Fire Protection Engineers, Atlanta.

[23] Launder, B.E. and spalding, D.B., (1974), "The numerical computation of turbulent flows", Computer Methods in Applied Mechanics and Engineering, 3(2): 269-289.

[24] Moukalled, F., Mangani, L., & Darwish, M. (2015). *The Finite Volume Method in Comutational Fluid Dynamics.* Beirut: Springer.

[25] M. Tabarra, B. Kenrick and R.D. Matthews, "CFD validation of natural smoke movement in a model tunnel", American Society of Mechanical Engineers, Fluids Engineering Division (Publication) FED, Vol. 238, No. 3, pp. 543-546 (1996).

[26] M.P. Deng, P.C. Miclea and D. McKinney, "CFD modeling considerations for train fires in underground subway stations", American Society of Mechanical Engineers, Fluids Engineering Division (Publication) FED, Vol. 238, No. 3, pp. 547-555 (1996).

[27] *National Fire Protection Association.* (2014). Retrieved September 10, 2014, from National Fire Protection Association Website: http://www.nfpa.org/press-room/reporters-guide-to-fire-and-nfpa/consequences-of-fire.

[28] Novozhilov, V. (2001). Computational Fluid Dynamics Modeling of Compartment Fire, Prog. Energy Combust. Sci., Vol. 27, pp. 611-666.

[29] OpenFOAM Foundation. (2004, December 10). OpenFOAM Foundation. Retrieved from http:// www.openfoam.org.

[30] Patankar, SV 1980. *Numerical Heat Transfer and Fluid Flow.* McGraw-Hill, New York.

[31] Quintiere, B. K. (2000). *Enclosure fire dynamics.* Boca Raton, Florida: CRC Press.

[32] R.N. Mawhinney, E.R. Galea, N. Hoffmann andM.K. Patel, "Critical comparison of a PHOENICS based fire field model with experimental compartment fire data", Journal of Fire Protection Engineering, Vol. 6, No. 4, pp. 137-152 (1994).

[33] Standard, E. (2002). Eurocode 1: Actions on structures - Part 1-2: General actions- Actions on structures exposed to fires. *EN* , 1991-1-2.

[34] Society of Fire Protection Engineers, *The SFPE engineering guide to performance-based fire protection analysis and design,* 1st ed. Quincy, Massachusetts: National Fire Protection Association, 2000, pp. 14-18, 23-24, 27-29, 31-36, 41, 58.

[35] Simcox, S., Wilkes, N.S. and Jones, I.P. (1992). Computer Simulation of the Flows of Hot Gases from the Fire at King's Cross Underground Station, *Fire Safety J.,*Vol. 18, pp 49-73.

[36] T. Hertzberg, B. Sundström, and P. van Hees, "Design fires for enclosures - A first attempt to create design fires based on Euroclasses for linings," SP Fire Technology, *SP Swedish National Testing and Research Institute*, Borås, SP Report 2003:02, 2003, pp. 10-11.

[37] Versteeg, H. K., & Malalasekera, W. (1995). *An introduction to finite volume method*. London: Longman Scientific & Technical.

[38] V. Novozhilov, "Computational fluid dynamics modeling of compartment fires", Progress in Energy and Combustion Science, Vol. 27, No. 6, pp. 611-666 (2001).

[39] Y.L. Sinai, M.P. Owens and P. Smith, "Advances in CFD assessment of fire and smoke movement", Proceedings of the International Conference on Offshore Mechanics and Arctic Engineering - OMAE, Vol. 2, pp. 389-395 (1995).

[40] Yang, K.T. and Chang, J. C. (1977). UNDSAFE-I. A Computer Code for Buoyant Flow in an Enclosure, University of Notre Dame Technical Report TR 79002-77-1.

[41] Moghtaderi, B., Novozhilov, V., Fletcher, D., and Kent, J., A new correlation of bench-scale piloted ignition data of wood. *Fire Safety Journal* 1997, *29*, 41-59.

[42] B. Karlsson and J. G. Quintiere, Enclosure fire dynamics. Boca Raton, Florida: CRC Press, 2000, pp. 17-18, 39-43, 100, 121, 125, 129, 228-238, 244-253.

[43] Boverket, Boverkets allmänna råd 2011:xx - Vägledning i analytisk dimensionering av byggnaders brandskydd - Remiss. Karlskrona: Boverket, 2010.