



PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE
ESCUELA DE INGENIERÍA

FIREFOAM (CFD SOLVER) VALIDATION IN COMPARTMENT FIRE SCENARIO USING HIGH RESOLUTION DATA

RAFAEL ZAMORANO

Thesis submitted to the Office of Research and Graduate Studies
in partial fulfillment of the requirements for the degree of
Master of Science in Engineering

Advisor:

WOLFRAM JAHN

Santiago de Chile, April 2018

© 2018, RAFAEL ZAMORANO



PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE
ESCUELA DE INGENIERÍA

FIREFOAM (CFD SOLVER) VALIDATION IN COMPARTMENT FIRE SCENARIO USING HIGH RESOLUTION DATA

RAFAEL ZAMORANO

Members of the Committee:

WOLFRAM JAHN

ELWIN VAN'T WOUT

MARCELA CRUCHAGA

GONZALO CORTÁZAR

Thesis submitted to the Office of Research and Graduate Studies
in partial fulfillment of the requirements for the degree of
Master of Science in Engineering

Santiago de Chile, April 2018

© 2018, RAFAEL ZAMORANO

*Gratefully to my parents, sibling,
and teachers*

ACKNOWLEDGEMENTS

Special thanks to my parents Rafael and Zunilda, who always supported me in my education and constantly guided me to aim higher. Also thanks to my brother Rafael, who despite not being with me during the development of this Thesis, was constantly and unconditionally supporting me. Finally I want to acknowledge my school and university teachers, specially Wolfram Jahn, who apart from guiding me through this Thesis, taught me masterfully some of the most influential courses of my studies.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS	iv
LIST OF FIGURES	viii
LIST OF TABLES	xvi
ABSTRACT	xviii
RESUMEN	xix
1. CHAPTER 1: INTRODUCTION	1
1.1. Context	1
1.2. Computational Fluid Dynamics (CFD)	2
1.3. Motivation and Scope	3
1.4. FireFoam Validation: State of the Art	5
1.4.1. Validated CFD Numerical Models	6
1.4.2. Validation Techniques in Literature	7
1.4.3. Up-To-Date FireFOAM Performance	18
2. CHAPTER 2: THEORETICAL BACKGROUND	28
2.1. Navier-Stokes System and Turbulence	28
2.2. LES-filtering	30
2.3. Favre-averaging	31
2.4. Gas-Phase Governing Equations	32
2.5. Radiation in participating media	37
2.6. Turbulence modeling	39
2.7. Finite Volumes (FV) Discretization	40
3. CHAPTER 3: METHODOLOGY	43
3.1. Experimental Setup: ETFT E-I	43
3.1.1. Compartment Topology and Materials	43

3.1.2.	Measuring Devices	48
3.1.3.	Fire Conditions	54
3.2.	Validation Structure for This Exercise	57
3.2.1.	Solution Quality Testing	57
3.2.2.	Runtime Testing	60
3.3.	Numerical Setup	61
3.3.1.	Computational Mesh: Spatial Scheme	61
3.3.2.	Temporal Scheme	70
3.3.3.	Boundary and Initial Conditions	71
3.4.	Physical Models Implemented in FireFOAM	81
3.5.	Data Management	83
3.5.1.	Time-averaged Data	84
3.5.2.	Simulated Thermocouple Temperatures	85
3.5.3.	Temperature Contours	90
3.5.4.	Streamline Maps	91
3.5.5.	Incident Radiation Contours	91
3.5.6.	Energy Balance Calculations	94
3.5.7.	Runtime Data	96
4.	CHAPTER 4: RESULTS	97
4.1.	Runtime Sensitivity	97
4.1.1.	Mesh Size	97
4.1.2.	Number of Processors	99
4.1.3.	Number of Solid Angles	100
4.2.	Theory-Based Validation	100
4.2.1.	Energy Balance	101
4.2.2.	Streamline Mesh-Convergence	107
4.3.	Experimentally-Based Validation	109
4.3.1.	Temperature Contours	109
4.3.2.	Incident Radiation Contours	111

4.3.3.	Interior Temperature Profiles	113
4.3.4.	Opening Plane Temperature and Velocity Profiles	118
4.3.5.	Transient Temperature Profiles: Central Tree	120
5.	CHAPTER 5: DISCUSSION	122
5.1.	FireFOAM Strengths	122
5.1.1.	Real-Time Computing Capabilities	122
5.1.2.	Convergence Capabilities	123
5.2.	FireFOAM Limitations	124
5.2.1.	Irregular-Tetrahedron Meshes	124
5.2.2.	Steady-State Boundary Conditions	125
5.2.3.	Sensitivity to Domain Construction	126
5.3.	Simulated Data Agreement with Experimental Measurements	127
6.	CONCLUSIONS	130
6.1.	Relevant findings	130
6.2.	Methodology limitations	132
6.3.	Future work	133
	REFERENCES	135
	APPENDIX	138
A.	Elaborating an OpenFOAM case using Gmsh	139
A.1.	Case Setup	139
A.2.	Creating the Mesh	140
A.3.	Exporting the Mesh to OpenFOAM	151
A.4.	Viewing and Post-Processing Results	155

LIST OF FIGURES

1.1	Left: Stream-wise variation of mean enthalpy flow rates, $HRR = 58$ kW. Right: Centerline mean temperature rise —black symbols represent simulation results and gray dotted line experimental correlation from McCaffrey (1979) experiments—. Source: Wang, Y. et al. (2011).	8
1.2	Left: Time trace of centerline stream-wise velocity at a height of 0.5 m. Right: Power spectrum of stream-wise velocity on the centerline at height of 0.5 m. These authors test FireFOAM's and FDS's performance with and without SGS models, against experimental measurements. Source: G. Maragkos et. al. (2012).	10
1.3	Wall-normal variations of mean temperature at $z=77$ m. Left: Measured (radiation-corrected) and simulated (LES) gas temperature; Right: Measured (raw) and simulated thermocouple temperatures. Source: Ren, N. et. al. (2016).	11
1.4	Surface temperature simulated in FireFOAM; experiment (left), model (middle), and IR (right) screen shots from the 245 rack storage configuration. Source: Ren, N. et. al. (2017).	13
1.5	Comparison of flame snapshots at different times; flame shape is illustrated by an iso-contour of mixture fraction at the stoichiometric ratio, simulated in FireFOAM. Source: Ren, N. et. al. (2015).	14
1.6	Normalized (against \dot{Q}_{fire}) heat distribution for ETFT E-I: fully-developed fire with fully open static segments. Source: Maluk, C. et. al. (2017).	17
1.7	Wall-normal variations of mean properties at $z=77$ cm: (a) mean vertical flow velocity; (b) mean temperature. Grid convergence study: $\Delta y_w = 1.5, 2, 3, 5, 10$ and 15 mm. Source: Ren, N. et. al. (2016).	18
1.8	Comparison between simulations (symbols) and an empirically-based correlations proposed by Heskestad (solid line) and McCaffrey (dashed line). Left: Vertical	

	variations of time averaged vertical mass flow rate. Elevation is made non-dimensional by the flame height; mass flow rate is made non-dimensional by the air mass flow rate required for stoichiometric complete combustion. Right: Vertical variations of time-averaged centerline vertical flow velocity. Elevation is scaled by the heat release rate raised to the power (2/5); flow velocity is scaled by the heat release rate raised to the power (1/5). Source: Trouvé & Wang (2010).	20
1.9	Warehouse scenarios for large scale experimental-based validation. Above: Illustration of 2x4x3 rack storage experimental configurations; (a) elevation view; (b) plan view showing the two ignition locations for the two different experiments: Cases 1 and 2. Below: Illustration of dimensions of 1/6 scale warehouse and H ₂ sensor locations.	21
1.10	Left: Chemical HRR for two experimental rack-storage cases. Right: Vertical velocity time history at three elevation locations above the igniter for Case 1. Source: Ren, N. et. al. (2015).	22
1.11	Comparison of experiment (two repeated experiments) and simulation data (three mesh sizes) for different sensor locations. Source: Bauwens et. al. (2014). . . .	23
1.12	Left: Computational domain with multi-level mesh refinement. The Fuel port supplies pure gaseous methane, while the oxidizer port supplies a O ₂ -N ₂ mix with variable composition. Right: Combustion efficiency resulting from different oxidizer composition. Red area represents contribution to combustion from flame ignition, and blue area from flame re-ignition. Source: Valfayeau et. al. (2016).	25
1.13	Temperature and radial velocity under the ceiling for the 2 6 5 water-mist suppressed case. Source: Ren, N. et. al. (2017).	26
1.14	Vertical variations of the mean total wall heat flux for variable fuel mass flow rate. LES data is compared to experimental measurements. The vertical dashed line marks the boundary between laminar and turbulent flame regions. Source: Ren et. al. (2016).	26

3.1	The drawing illustrates the characteristic internal dimensions of the compartment as well as the directions and origin of the global coordinate system used to define it and its contents. The global origin was aligned with the back left corner of the platform, when looking into the compartment from the front face, and leveled with the finished compartment floor. Source: J.P. Hidalgo et. al. (2017).	43
3.2	Images of the front (open) face of the compartment. Left: Layout and dimensions of the 15 openings along the front (open) face. Right: Roller shutter system, which for ETFT E-I is not used (all shutters opened). Source: J. P. Hidalgo et. al. (2017).	46
3.3	ETFT roof composition. Source: J. P. Hidalgo (2017).	47
3.4	Plan layout of the compartment, showing gas burners (large red squares; located at floor level), internal thermocouple trees (blue triangles), floor- and ceiling-level TSCs (red circles), gas-species sampling points just below ceiling level (small black squares), and obscuration gauges (large green squares; also close to ceiling level). All values are in m and reference the global coordinate system. Source: J. P. Hidalgo et. al. (2017).	50
3.5	Location of sensors in the plane of the facade (global $x=5.05$ plane) and beyond, as indicated. Red circles represent outward facing TSCs above and on the same plane as the compartment openings. Blue squares represent downward facing TSCs 0.175 m out from the plane of the opening. Green triangles represent outward facing TSCs 0.4 m out from the plane of the opening. Blue crosses represent thermocouple locations 0.4 m out from the opening plane. Black circles represent velocity probes 0.05 m towards the compartment interior from the opening plane. Source: J. P. Hidalgo et. al. (2017).	51
3.6	Back wall ($x=0.1$ plane) as viewed from the front of the compartment. Red circles represent TSCs. Coordinates shown are in m. Source: J. P. Hidalgo et. al. (2017).	52

3.7	Right wall ($y=17.9$ plane) as viewed from the front of the compartment. Red circles represent TSCs. Left side wall ($y=0.1$ plane) is a mirror image of this. All values are in m.. Source: J. P. Hidalgo et. al. (2017).	52
3.8	Left: Relative positions of three gas burners inside the compartment. Right: Pilot flame above the box burners. Source: J. P. Hidalgo et. al. (2017).	55
3.9	Above: Mayor event timeline in ETFT Test I. Bottom: Propane flow rate from the Large Mass Flow Controller (solid line) and corresponding HRR assuming 100% combustion efficiency within the compartment (horizontally dashed line). The vertical dashed lines mark three times at which mass flow is varied. Source: J. P. Hidalgo et. al. (2017).	56
3.10	Top-view of key locations to be used for data plotting. Thermocouples are shown as hollow light blue triangles, and roof TSCs as hollow red circles. Note that the roof section will be used, but it is not marked for visualization limitations.	59
3.11	Front-view of key locations to be used for data plotting. Velocity probes in each opening are shown as solid black circles, and are located 0.5m towards the compartment interior from the opening plane $x=4.9$ m.	59
3.12	Gas-phase domains. Left: Built for tetragonal meshing. The 12 prism-shaped holes inside the domain correspond to the 12 propane burners. The inside of these prisms are later to be unmeshed. Right: Built for cubic meshing. For both constructions, the left structure corresponds to the compartment interior, and the right structure to the secondary entrainment/escape zone.	62
3.13	Gas-phase domain patches for the 20 cm cubic mesh.	63
3.14	X-Z plane containing the sub-surfaces (C1, C2, ..., C24) that form the master surface. Each sub-surface is extruded in the Y-axis to form the final 3D domain.	65
3.15	Front view of the gas-phase domain meshed with the baseline case M20. Note that the hanger is grid-size in the X-axis and therefore 1-cell width.	66

3.16	Top view of the meshed baseline M20 case.	67
3.17	Front view of the different cubic meshes. From coarsest to finest: M40, M30, M20, M10 and M7.5.	68
3.18	Top view of the different cubic meshes. From coarsest to finest: M40, M30, M20, M10 and M7.5.	69
3.19	'ControlDict' file for each FireFOAM case.	71
3.20	Total prescribed mass flow rate (advective plus diffusive) at burner inlet patches, built as a piece-wise function of linear functions.	74
3.21	Temperature boundary conditions in each domain patch.	79
4.1	Runtime for different meshes. Log10[] scale is applied for runtimes as to capture all magnitude orders. The red dashed line marks the real time limit, i.e. Log10[750s]=2.88.	97
4.2	Runtime for cases M40, M30, M20, M10 and M7.5 represented as number of elements rather than average cell size. The red dashed line marks the real time limit, i.e. Log10[750s]=2.88.	98
4.3	Runtime for different number of parallel processors. Parallel efficiency, expressed as a percentage, is calculated as $\epsilon = \frac{T_1}{p \cdot T_p}$ (where T_p is the computing time for p processors) and placed adjacent to each case. The red dashed line marks the real time limit, i.e. 750s. 20 cm cubic mesh and 144 solid angles for all cases.	99
4.4	Runtime for different number of solid angles. 20 cm cubic mesh for all cases. The red dashed line marks the real time limit, i.e. 750s.	100
4.5	Mass fluxes considering the compartment interior as a control volume. Above: M20. Below: M10. Fuel mass flow rate is negligible compared to air entrainment, as expected.	101
4.6	Simulated compartment energy fluxes. Above: M20. Below: M10.	103

4.7	Normalized energy fluxes with respect to the prescribed heat release rate ('QfirePrescribed'). Above: M20. Below: M10.	104
4.8	Time-averaged streamline maps in section 15 for the steady state period of 74% HRR_{max} . Velocity modulus in [m/s]. From top to bottom: M40, M30, M20, M10 and M7.5.	108
4.9	Time-averaged thermocouple temperature contours in section B for the steady state period of 74% HRR_{max} . Temperatures in [°C]. From top to bottom: Experimental, M40, M30, M20, M10 and M7.5.	109
4.10	Time-averaged thermocouple temperature contours in section 15 for the steady state period of 74% HRR_{max} . Temperatures in [°C]. From left to right and descending: Experimental, M40, M30, M20, M10 and M7.5.	110
4.11	Time-averaged back wall incident radiation contours for the steady state period of 74% HRR_{max} . Scale in [kW/m ²]. Note that for the experimental case the scale is from 0 to 10 kW/m ² , while for simulations from 0 to 20 kW/m ² . This change was done for visual purposes, as the M20 case does not reproduce well the experimental value of G. From top to bottom: Experimental, R4 (nPhi=nTheta=1), R16 (nPhi=nTheta=2), R64 (nPhi=nTheta=4), R144 (nPhi=nTheta=6), R256 (nPhi=nTheta=8) and R400 (nPhi=nTheta=10).	112
4.12	Time-averaged back wall incident radiation contours for the steady state period of 74% HRR_{max} . Top to bottom: M20 and M10 (144 s.a.).	113
4.13	Time-averaged thermocouple temperatures in section B. 74% HRR_{max} steady state period. From top to bottom: 600mm, 1200mm, 1600mm and 1950mm. . .	114
4.14	Time-averaged thermocouple temperatures in section 15. 74% HRR_{max} steady state period. From top to bottom: 600mm, 1200mm, 1600mm and 1950mm. . .	115

4.15	Time-averaged thermocouple temperatures for the 74% HRR steady state period. Left: Central tree D15, located in the far field. Right: Tree B2, located along the flame-plume region of Burner 6a.	116
4.16	Time-averaged thermocouple temperatures at openings for the 50% (left) and 74% (right) HRR_{max} steady state periods. Results correspond to an average between openings 1, 8 and 15.	118
4.17	Time-averaged inflow (negative) and outflow (positive) velocities at openings for the 50% (left) and 74% (right) steady state HRR periods. Simulation results correspond to an average between openings 1, 8 and 15. Experimental results are averages according to Hidalgo et. al. (2017).	119
4.18	Transient thermocouple temperatures at four heights along the central tree D15. Begins at the opening of propane supply to the burners, and ends at the ending of the 99% HRR steady state period.	121
5.1	Compartment X-Z plane cut. Thin black lines represent wall boundaries, and thick red lines entrainment zones. These constructions are defined through the boundary condition set. Chosen cuts, for simplicity, do not include propane burners. Images not scaled to real dimensions.	126
A.1	Characteristic domain and mesh dimensions defined in Gmsh.	140
A.2	Geometry and mesh definition. Part I.	141
A.3	Geometry and mesh definition. Part II.	142
A.4	X-Z view of the Domain, with Lines and Points numerated using functions located in the Tools-Options-Geometry Menu. Lines and Points can also be viewed in the graphic interphase by locating the cursor on top. C12 and C14 dimensions are those of the burners.	143
A.5	X-Z plane mesh, constructed by regular quadrangles. Mean grid size: 20cm. . .	145

A.6	Definition of extrusions on the '.geo' file. Part I.	147
A.7	Definition of extrusions on the '.geo' file. Part II.	148
A.8	Extrusions of the X-Z layer through the Y direction. Surface labels removed for simplicity. Label visibility can be activated in the Tools-Options-Geometry Menu, as to facilitate the extrusion task.	149
A.9	Defining the domain volume and surfaces. Surface labels are later used for boundary conditions.	150
A.10	Saving the '.msh' file	151
A.11	Exporting the mesh into the OpenFOAM case directory.	152
A.12	'/constant/polyMesh' directory, after exporting the mesh.	152
A.13	Edited 'boundary' script for the 20cm mesh.	154
A.14	'Allrun' script for the ETFT Test I cases.	155
A.15	Case selection, patch/volume selection, and field extraction in ParaView.	157
A.16	'.csv' file being extracted for the 'rearAndSideWalls' patch. Here no slice was used, as this patch was directly chosen. To perform a Slice, the 'internalField' part has to be selected. Data is being printed as a list of cells contained in the patch, with its respective fields including its surface area (created with the Mesh Quality Filter).	158

LIST OF TABLES

1.1	Most common CFD models used in FireFOAM validation exercises.	7
1.2	Listed simulated properties are plotted against measurements.	9
1.3	Techniques with their respective physical properties used for internal software validation.	15
3.1	Floor composition and thermal properties. Order goes from outside to inside the compartment.	44
3.2	Wall composition and thermal properties. Order goes from outside to inside the compartment.	45
3.3	Roof composition and thermal properties. Order goes from outside to inside of the compartment.	48
3.4	List of sensors used for characterization of fire-related variables during ETFT. Source: J. P. Hidalgo (2017)	49
3.5	Solution quality testing cases. 'B' stands for baseline case.	58
3.6	Validation plots for solution quality testing.	60
3.7	Runtime testing cases. 'P' stands for augmentation/reduction of parallel processors running the simulation, and the number after the number of processors.	61
3.8	Elements and vertices reported post-meshing by Gmsh. All numbers approximated to the third significant figure.	70
3.9	First part: Boundary Conditions in OpenFOAM.	72
3.10	Second Part: Boundary Conditions in OpenFOAM.	73
3.11	Initial domain conditions set in OpenFOAM.	80

3.12	Main events regarding HRR changes. Times are reported relative to the time where all the burners were lit.	84
3.13	Used time-steps for time averaging in each HRR steady state period. Times are reported relative to the simulation beginning.	85
4.1	Energy distribution analysis for both steady state periods.	106

ABSTRACT

This work presents a LES-based CFD study for FireFOAM validation in a large-scale compartment fire scenario. A recreation of the Edinburgh Tall Building Fire Test I (2017) was elaborated to perform this evaluation, which aimed in testing both runtime and result quality. Different mesh geometries (cubic and tetrahedral) and sizes (7.5cm, 10cm, 20cm, 30cm and 40cm) were tested. Performed validation was both theory-based and experimentally-based. Theory-based validation mainly consisted in an energy balance to test how conservative the implemented Finite Volumes method is, as well as to perform an energy distribution analysis for comparison with the Maluk et al. (2017) study for Test I. Results indicated a non-conservative solution, with 0 to 20% of the heat release rate unaccounted for, which was attributed to the solver's admissible errors when numerically solving the Navier-Stokes system, and sub-grid scale modeling incorporation. No convergence was achieved in the tetrahedral mesh. Experimentally-based validation consisted in a comparison of simulated data with experimental measurements of temperature, compartment inflow/outflow velocities, and incident radiation. Well reproduced phenomena were flow patterns at the compartment interior, neutral plane height, and temperatures outside the roof boundary layer. Poorly reproduced phenomena were incident radiation in the hot-gas zone, thermal boundary layer transient temperatures, and compartment inflow/outflow velocities. Mesh convergence in the far-field temperatures and flow patterns were achieved for the 10cm cubic mesh, while no mesh convergence was achieved at opening plane velocities and near-flame temperatures. Systematical errors were attributed to deficient heat transfer boundary conditions, underestimated air entrainment, and uneven distribution of propane among the experimental Test I burners. It was found that runtime could be reduced in 60-70% when implementing parallel processing, or reducing solid angles in FVDOM. Only the 30cm and 40cm meshes managed to run faster than real time.

Keywords: Full-scale compartment fire, LES, FireFOAM, validation, runtime.

RESUMEN

Se realizó una validación de software CFD, en donde se evaluó el desempeño del solver FireFOAM en un escenario de incendio en compartimento de escala real. La evaluación de desempeño, enfocada en calidad de resultados y tiempo de cálculo, se realizó recreando el Test I de la serie *Edinburgh Tall Building Fire Tests* (2017), o ETFT. Se probaron mallas de distinta geometrías (cúbicas y tetrahédricas) y tamaños (7,5cm, 10cm, 20cm, 30cm y 40cm). La validación basada en teoría constó en un balance de energía tanto para determinar qué tan conservativa es la discretización por Volúmenes Finitos, como para realizar un análisis de distribución de energía basado en el balance experimental desarrollado por Maluk et al. (2017). Se obtuvieron resultados no conservativos, con 0 a 20% de la tasa de liberación de calor no simulada, lo que se atribuyó tanto a la sensibilidad del solver al resolver el sistema Navier-Stokes, como a imprecisiones en los modelos de escala sub-malla. La malla tetrahédrica mostró divergencia. La validación basada en experimentos constó en una comparación entre simulaciones y mediciones de temperatura, velocidad e irradiación. Fenómenos bien captados fueron patrones de flujo al interior de compartimento, altura del plano neutro, y temperaturas fuera de la capa límite del techo. Fenómenos mal captados fueron irradiación en la capa humo, temperaturas transientes en la capa límite térmica, y ventilación. Convergencia de malla se logró tanto en temperaturas lejos de la llama como en campos de velocidad al interior del compartimento, para la malla cúbica de 10 cm, mientras que no se consiguió convergencia de malla en velocidades del plano de apertura ni en temperaturas de llama. Errores sistemáticos se atribuyen a condiciones de borde de transferencia de calor deficientes, sub-estimación de la tasa de ventilación, y distribución no uniforme de propano entre los quemadores del Test I. Se encontró que el tiempo de cálculo puede reducirse entre 60 y 70% al implementar procesamiento paralelo. Solo las mallas de 30 cm y 40 cm lograron correr más rápido que tiempo real.

Palabras Claves: Incendio, escala real, LES, FireFOAM, validación, tiempo de cálculo.

1. CHAPTER 1: INTRODUCTION

1.1. Context

Fire modeling can take different approaches depending on the complexity of the required analysis. First approaches consisted in flame height estimations through experimental correlations. Correlations for flame height dependence on heat release rate were developed by Thomas (1961), Steward (1970), Terai & Nitta (1975), McCaffrey (1979) and Zukoski (1981), among others. These experimental correlations are mostly limited to well-ventilated fully-developed flames, where fire dimensions are negligible compared to the compartment dimensions. Alongside with experimental correlations, the first compartment fire models¹—also known as Zone Models—were developed around the 1970s by authors such as Petersson, Magnusson & Thor (1976) and Babrauskas & Williamson (1978). Zone Models consist in dividing the compartment in zones and imposing mass and energy conservation equations among them, where each zone is assumed to have uniform conditions represented by a single smoke layer temperature. The 1970s models correspond to single-zone models for fully-developed fires—also known as post-flashover fires—and deliver as a solution only the plume height and its average temperature. Zone Models are a useful tool only for a simplified analysis, as no momentum conservation is considered and conservation equations are imposed on the whole zone control volume, giving a zero dimensional result—i.e. no spatial distributions are calculated. Given this, the Zone Model technique does not capture velocity distributions along compartment openings, spatial temperature stratification, and propagation/extinction phenomena, therefore a different approach must be used for applications that require more detailed results. Applications of this nature include fire-safe building design, building performance evaluation in fire hazard scenarios, and flame/smoke domain predictions for firefighter operation assistance, among others.

¹Compartment fires, also known as enclosure fires, are understood as fire conditions that feature limited ventilation and smoke accumulation (Vilfayeau, 2015). These conditions usually promote internal stratification of two layers: a lower cold air entrainment layer, and an upper hot smoke layer.

1.2. Computational Fluid Dynamics (CFD)

Detailed analysis of compartment fires is achieved through the Computational Fluid Dynamics (CFD) method. Herein, the whole domain is sub-divided in Finite Volumes —also called cells— forming a meshed configuration, and conservation equations —now including mass, energy and momentum— are imposed over each volume. The system is numerically solved by a computational routine, and results are delivered in a spatial resolution equal to the cell size. CFD is therefore more calculation-extensive than both flame height correlations and Zone Models, but it delivers results with a precision and resolution vital for most engineering applications. Given the significant extent of this fire modeling technique, validation of CFD combustion tools is fundamental for fire hazard analysis, active and passive protection design, and equipment dimensioning (Almeida, Lage, & Silva, 2015). CFD tools are adequate for combustion modeling, since they are based on the solution of conservation equations with strong physical basis and considerable experimental support. Because of this, combustion simulation presents a high level of difficulty due to the strong coupling between chemistry, turbulence and thermal radiation, requiring a detailed model to simulate them (Almeida et al., 2015). Particularly where transient large-scale motions and intermittency are inherent to turbulent buoyant flames (Almeida et al., 2015), the transient Large Eddy Simulation (LES) method is used for turbulence modeling. This means additional computational resource demand compared to the steady-state turbulence model counterpart, known as the Reynolds Averaged Navier-Stokes (RANS) approach.

Precision gained by using CFD as a fire modeling tool comes at the expense of increased computing time. As reference, a LES simulation of a real scale fire involves a computing time three magnitude orders bigger than the simulated flame time, considering an appropriate cell size is used as to achieve consistent results. For example, a one-second flame simulation takes approximately one-thousand seconds of executing time. Extended executing time can be a problem in applications where time is of the essence, as good quality results are needed in brief time scales.

1.3. Motivation and Scope

This Thesis' line of work focuses on reducing CFD computing time in fire simulations, potentially reaching real-time simulations. Real-time fire simulations can be essential in the fire hazard analysis field, as flame propagation can be predicted before it actually happens. Having the possibility to predict fire can assist in critical tasks, such as firefighter operations and wildfires suppression, as more inquired field-decisions can be done. As computing time is mostly given by the mesh resolution, it can be reduced by increasing the cell size at the expense of getting poor-quality results. Reducing the number of cells while obtaining good-quality results can be achieved with the Mesh Refinement technique, now widely used in both the research and industry fields. Mesh Refinement consists in refining cells in the flame domain—where increased resolution is needed for modeling turbulent mixing and chemical reaction rate—and coarsening in the far field, leaving a high mesh resolution only where needed. In the fire hazard analysis field, where flame propagation and extinction through the solid-phase fuel (e.g. wood) controls the heat release rate, the Dynamic Mesh Refinement technique must be implemented in order to make the refined zone follow the flame structure. As not all the available solvers support the use of irregular meshing, a selection among the available CFD software must then be made.

The Fire Dynamics Simulator (FDS) from the National Institute of Standards and Technology (NIST) is the most popular LES-based solver for fire hazard simulation in the industry, whose mayor drawback is the strict use of regular rectangular prism numerical meshes. Therefore, FDS cannot take advantage of either Mesh Refinement and/or Dynamic Mesh Refinement techniques, making it a poor choice when fast simulations are required. FireFOAM—an OpenFOAM solver for turbulent diffusion flames—has the capacity for implementing irregular meshes (e.g. tetrahedral finite volumes with different levels of local refinement), which makes possible both local refinement and dynamic meshing. Given this, there is a need to test it's performance through validation exercises, as to evaluate its capabilities for dynamic meshing implementation. Up to date numerous validation exercises have been done, but all of them fall into at least one of the following limitations:

- Only global fire parameters are tested, e.g. evolution of global heat release rate (Ren, de Vries, Chaos, & Yi, 2015), therefore there is a need for more aggressive evaluations of the LES-based capabilities (Trouvé & Wang, 2010).
- Validation is done based on low spatial resolution measurements and/or literature correlations (Y. Wang, Chatterjee, & de Ris, 2011).
- Validation is done for small-medium scale fires, e.g. < 1 m in diameter pool-fire configurations (Almeida et al., 2015).

It is important to clarify that all validation exercises are valuable, since they provide guidelines and experience for more polished and complete validation. Given this, new validation exercises need to take these previous guidelines and new published data in order to deliver improved CFD performance evaluations. The publishing of the high-resolution multi-variable measurements from the "Edinburgh Tall Building Fire Tests" (ETFT) compartment fire experiments (J. Hidalgo et al., 2017) presents a great opportunity to validate FireFOAM without incurring into the previously stated limitations. With multiple variables measured — gas-phase temperature, incident radiant flux, gas flow velocity, $O_2/CO_2/CO$ concentrations, soot volume fraction, and thermal video imaging— in high spatial resolution —203 thermocouple trees with 8 thermocouples per tree, with thermocouples distributed in approximately 50 cm intervals— on a large scale fire —5 x 18 x 2 m compartment—, these experiments present a demanding validation scenario for evaluating FireFOAM's capabilities. In LES applications for large-scale industrial fires, where scale separation is large and therefore small cell-size cannot be afforded, Sub Grid Scale (SGS) modeling becomes more challenging and important. Following the previous logic, future studies are needed to examine the adequacy of the current model for very large fires (Y. Wang et al., 2011) like the ETFT experiments. According to Y. Wang et al. (2011), there is the possibility of numerical challenges when going to such significantly different situations.

Therefore, this Thesis's scope will include the validation of FireFOAM's capabilities based on the ETFT experimental measurements, particularly Test I². This work's final goal

²In the ETFT series, Test I consists in a fully-developed fire with complete and static ventilation.

is to prove that FireFOAM can successfully simulate real scale fires, delivering results with appropriate precision for fire hazard analysis purposes. This is the beginning on a line of work focused on a particular goal: implement Dynamic Mesh Refinement for LES-based combustion in FireFOAM, to finally achieve good quality simulations that run in real time. Therefore, the scope also contemplates the evaluation of FireFOAM's performance when running coarse meshes with real-time computing time scales. As to study potential discrepancies between simulated results and experimental measurements, a compartment global energy balance will be performed over the simulation results, which will be in contrast compared to the experimental balance performed by Maluk et al. (2017) on ETFT Test I. A global energy balance adds additional value to this validation exercise, as it will also reveal how conservative the Finite Volume Method behaves in FireFOAM.

1.4. FireFoam Validation: State of the Art

A diversity of authors have attempted to test FireFOAM's performance in LES-based combustion simulations. Validation techniques can be classified as n-order statistical moments (Trouvé & Wang, 2010). First-order statistical moments include but are not limited to: time-averaged centerline temperature, time-averaged centerline vertical flow velocity, and time-averaged vertical mass flow rate. These are most commonly used in 3D pool fire scenarios (Trouvé & Wang, 2010), where both experimental measurements and experimental correlations are used as validation data for numerical models. Second-order statistical moments include: centerline energy conservation (e.g. time-averaged vertical centerline enthalpy distribution), and centerline fluctuations (e.g. velocity vector components, and mixture fraction). These moments are for example used in small-scale square-burner configurations (Y. Wang et al., 2011), where additionally the LES-filtered species mass fraction is computed from a Beta-distribution function for the centerline mixture fraction. Both previously referenced validations are based on experimental correlations for small-scale buoyant

diffusion flames (McCaffrey, 1979), i.e. 10 - 60 kW Heat Release Rates (HRRs). More sophisticated 3D Pool-Fire configurations have been used to test additional FireFOAM capabilities —e.g. flame extinction models (Vilfayeau, White, Sunderland, Marshall, & Trouvé, 2016)— employing similar statistical moment orders as the previous studies.

Further extension of the CFD model validation to high-order statistics calls for new experimental data with greater resolution (Trouvé & Wang, 2010) —e.g. ETFT Measurements (2017)—, therefore there is a need to first explore FireFOAM’s validation state of the art to appropriately select which statistical orders are needed to validate implemented models against the experimental measurements in ETFT E-I.

1.4.1. Validated CFD Numerical Models

CFD Combustion Modeling corresponds to a multi-physics simulation where multiple phenomena are simulated in an integrated methodology. These phenomena include: gas-phase turbulence, combustion chemistry, heat transfer (conduction, convection and radiation), solid-phase pyrolysis, and soot formation/oxidation. OpenFOAM has a wide variety of models for each phenomena (e.g. radiation models include: NoRadiation, Finite Volume Discrete Ordinates —or fvDOM—, and P1), each one with its respective advantages and disadvantages. Table 1.1 shows a summary of the most popular OpenFOAM models in the validation field.

Table 1.1. Most common CFD models used in FireFOAM validation exercises.

Combustion Modeling				
Gas-phase turbulence	Combustion Chemistry	Thermal Radiation	Solid-phase Pyrolysis	Soot formation/oxidation
→ LES	→ EDC: Infinitely fast single-step chemistry	→ fvDOM	→ No pyrolysis	→ PaSR-based laminar smoke point
→ RANS	→ n-th order Arrheinus	→ P1	→ n-th order Arrheinus	→ Constant soot fraction

1.4.2. Validation Techniques in Literature

Validation techniques can be divided in two groups: experimental-based and theoretically-based. Experimental-based validation compares experimental measurement results with simulation results, while theoretically-based validation internally verifies the conservation of physical properties and convergence in the solver. Figure 1.1 shows examples of both techniques mentioned, performed in a pool-fire validation exercise (Y. Wang et al., 2011).

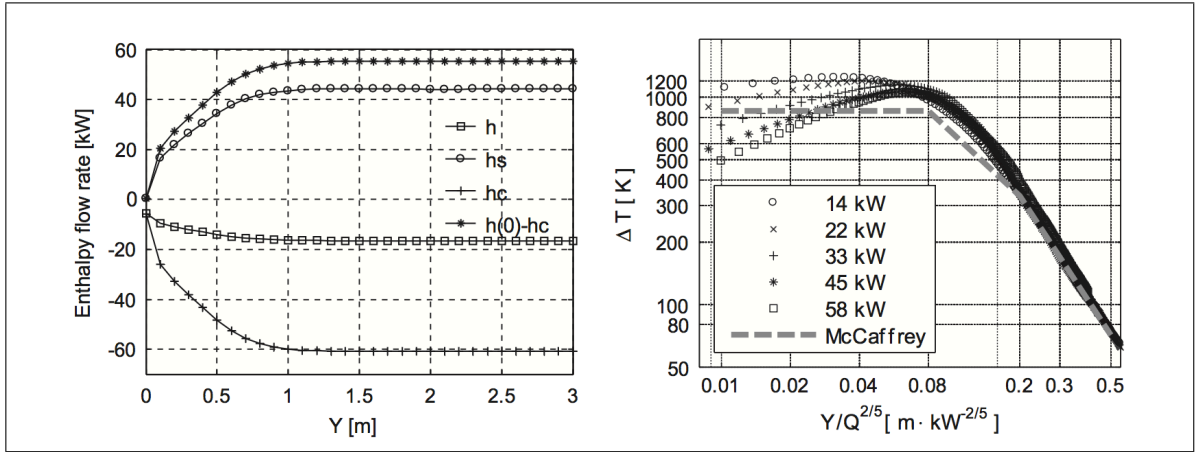


Figure 1.1. Left: Stream-wise variation of mean enthalpy flow rates, $HRR = 58 \text{ kW}$. Right: Centerline mean temperature rise —black symbols represent simulation results and gray dotted line experimental correlation from McCaffrey (1979) experiments—. Source: Wang, Y. et al. (2011).

In this exercise both conservation of properties within the solver (i.e. vertical mean enthalpy flux) and experimental measurement agreement with numerical results (i.e. centerline mean temperature rise) are verified. Tables 1.2 and 1.3 show a summary of experimental-based and theoretically-based validation techniques.

1.4.2.1. Experimental-based validation

The FireFOAM validation field has mostly used the experimental-based validation techniques shown in Table 1.2. Note that HRR refers to Heat Release Rate from now on. Local HRR refers to Global HRR per unit area or volume, e.g. $[\text{kW}/\text{m}^2]$ or $[\text{kW}/\text{m}^3]$.

Table 1.2. Listed simulated properties are plotted against measurements.

Experimental-based		
Global parameter interactions	Time series (interval-averaged)	Spatial distribution (mean, RMS or instantaneous)
→ Flame height vs. O ₂ molar fraction → Combustion efficiency vs. O ₂ molar fraction → Puffing frequency → Flame height vs. global HRR → Flame tilt/height/SEP vs. pool diameter (pool fire configurations)	→ Global HRR → Local HRR → Fuel mass loss rate (pool fire configurations) → Temperature → Velocity components → Molar fraction → Flame height	→ 1D and 2D profiles: - Temperature - Velocity components - Molar fractions - Convective and radiative heat fluxes - Air-mass entrainment - Vorticity vector map - Velocity component fluctuation - Temperature fluctuations - Mixture fraction fluctuations → 2D and 3D iso-contours: - Stoichiometric mixture fraction - Local HRR - Lower Flammability Limit (LFL) - Upper Flammability Limit (UFL)

The techniques mentioned in the first column —i.e. Global parameter interactions— are typically used in correlation-based validation exercises. For example, a plume’s puffing frequency —i.e. frequency at which large turbulent structures pass through the flame domain— is calculated by using a Fourier Transform of the time signal of the centerline stream-wise velocity at a certain point, in which case the puffing frequency is the plot’s peak. Figure 1.2

shows this procedure implemented in FireFOAM (Maragkos, Rauwoens, & Merci, 2012), which results in a puffing frequency of 1.38 Hz with good agreement with the 1.34 Hz suggested by Cetgen and Casper’s correlation (2008).

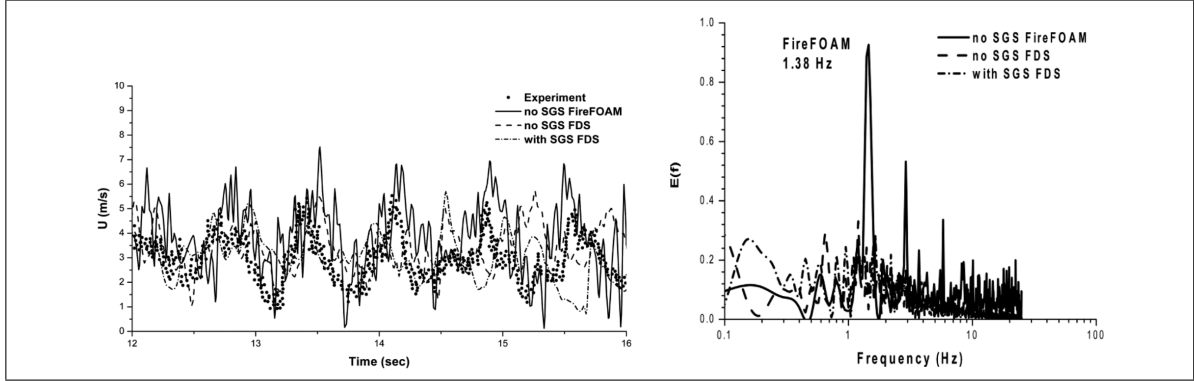


Figure 1.2. Left: Time trace of centerline stream-wise velocity at a height of 0.5 m. Right: Power spectrum of stream-wise velocity on the centerline at height of 0.5 m. These authors test FireFOAM’s and FDS’s performance with and without SGS models, against experimental measurements. Source: G. Maragkos et. al. (2012).

Table 1.2’s second column lists the parameters usually plotted in time series for experimental validation. Fig. 1.2 (Left) shows an example of FireFOAM simulations results (solid line) plotted against experimental measurements (dots) for comparison. These series are also named ”interval-averaged series”, due to the fact that each simulation plotted point is an time-interval-average of several other points. This technique is quite common, since comparing instantaneous data can be misleading given the considerable property fluctuations in turbulent-transient modeling.

Finally, Table 1.2’s third column lists some physical quantities used in spatial mapping for validation. Figure 1.3 (Left) shows an example of a 1D mean temperature profile for a vertical wall fire validation scenario (Ren, Wang, Vilfayeau, & Trouvé, 2016).

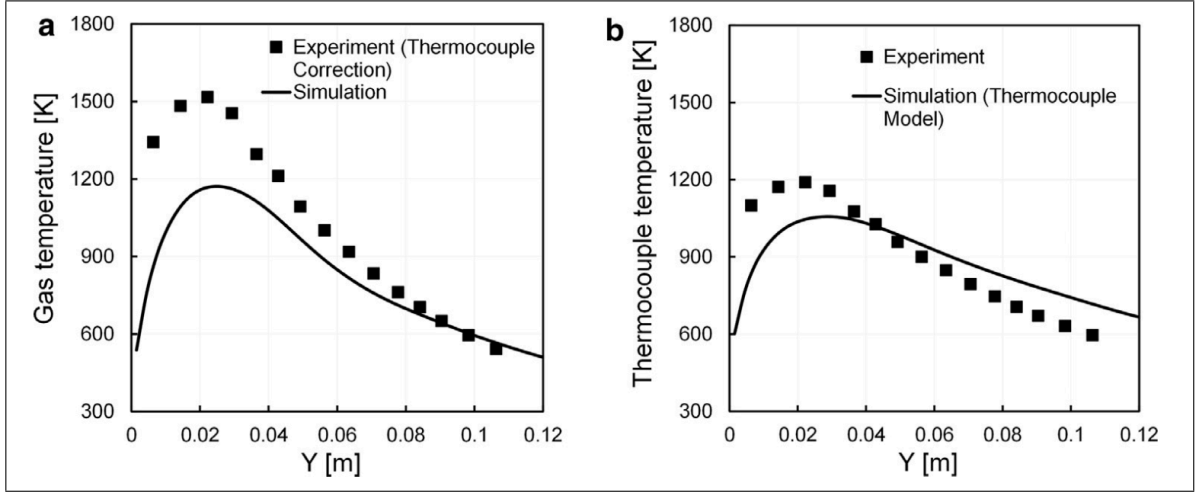


Figure 1.3. Wall-normal variations of mean temperature at $z=77$ m. Left: Measured (radiation-corrected) and simulated (LES) gas temperature; Right: Measured (raw) and simulated thermocouple temperatures. Source: Ren, N. et. al. (2016).

As shown above, it is necessary to distinguish between the "raw" and "radiation-corrected" thermocouple temperatures. Thermocouple raw measurements correspond to the bead temperature, not to the gas-phase temperature profiles that result from the LES simulations. Therefore, there is heat transfer between the gas-phase and the bead, which states that the bead's internal energy rate of change is equal to the convective plus radiative heat transfer from the gas-phase (Ren et al., 2016):

$$\rho_{tc} c_{tc} \frac{V_{tc}}{A_{tc}} \frac{dT_{tc}}{dt} = \varepsilon_{tc} (G - \sigma T_{tc}^4) + h_{tc} (T_g - T_{tc}) \quad (1.1)$$

Where ρ_{tc} , c_{tc} , V_{tc} , A_{tc} and ε_{tc} designate the mass-density, heat capacity, volume, surface area and emissivity of the thermocouple bead, respectively. T_{tc} is the thermocouple bead temperature, T_g the local gas-phase temperature, G the irradiation received by the thermocouple bead, and h_{tc} the convective heat transfer coefficient. By using eq. (1.1), in Figure

1.3 (Left) the raw measurements are corrected (black squares) to be compared to the simulation results (solid line). Also using this model, in Figure 1.3 (Right) the LES-simulated gas-phase measurements are converted into thermocouple temperatures (solid line) for comparison with the raw measurements (black squares). Discrepancies are justified in the fact that the mesh size is not converged³. Section 3.5. Data Management describes in more detail how this thermocouple correction will be used in the present validation exercise.

Also common in experimental-based spatial distribution validation techniques, are contour maps. These can be 2D or 3D, and are usually complemented with experimental IR thermography and flame snapshots. Figure 1.4 (Ren et al., 2017) shows FireFOAM-simulated 3D surface temperature maps (indicated as Model) validated against flame snapshots (indicated as Experiment) and experimental IR thermography images (indicated as IR Camera). As to further exemplify, Figure 1.5 (Ren et al., 2015) shows simulated stoichiometric mixture fraction iso-contours validated against flame snapshots.

When using IR thermography data for validation, the simulated gas-phase temperatures need to be corrected to superficial temperatures. On the other hand, validating simulated stoichiometric mixture fraction iso-contours against flame snapshots only gives a hint on how the LES-simulation captures the flame spread and tilt, given the fact that turbulence modeling has a aleatory component which can cause local discrepancies between the iso-contour and the snapshot.

³A grid size is denominated "converged" —or equivalently, a result is denominated "grid-converged"— when further refinement practically does not change the simulation results. The minimum percentage change in results to consider them grid-converged is defined by the analyst's criteria.

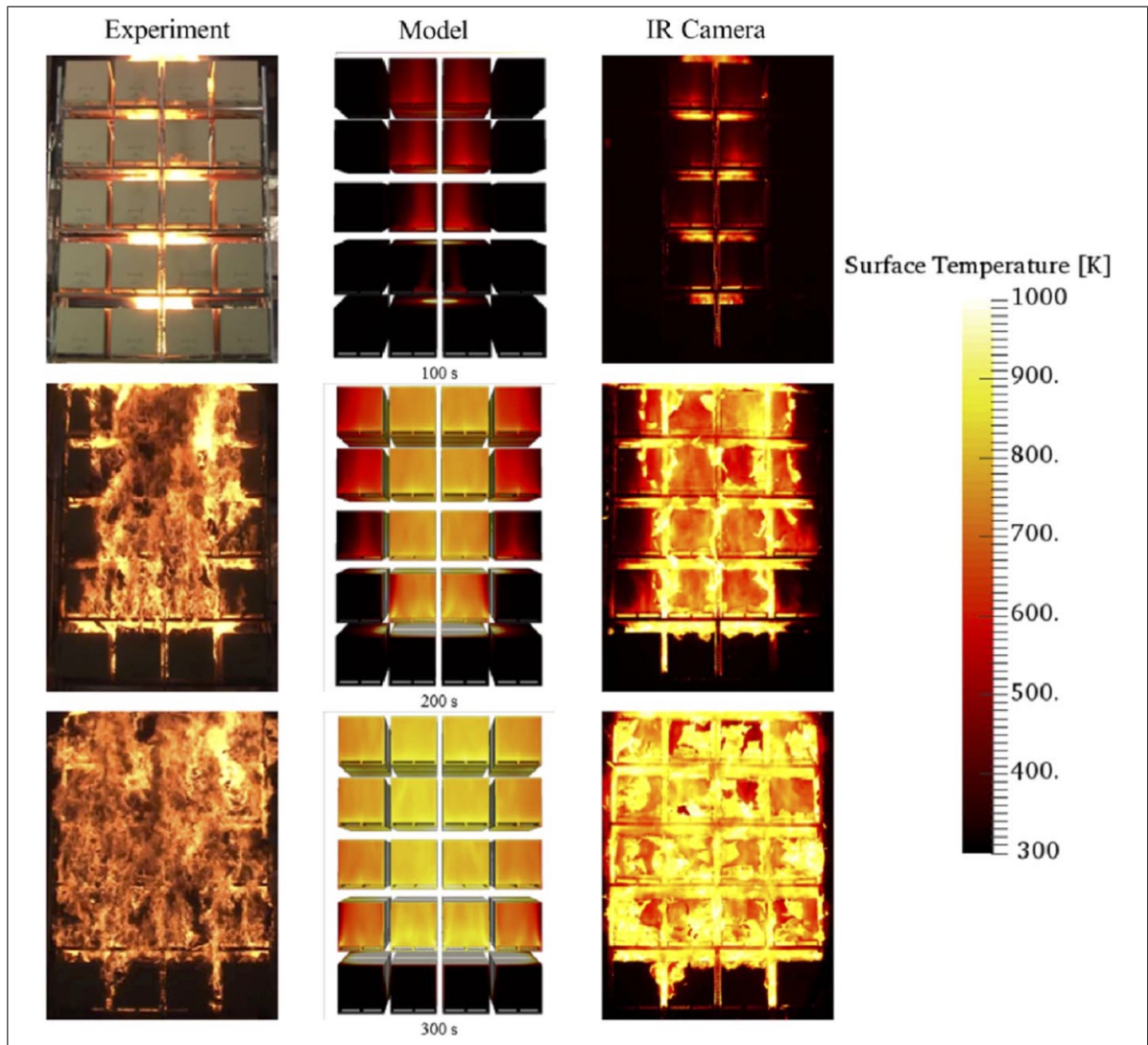


Figure 1.4. Surface temperature simulated in FireFOAM; experiment (left), model (middle), and IR (right) screen shots from the 245 rack storage configuration. Source: Ren, N. et. al. (2017).

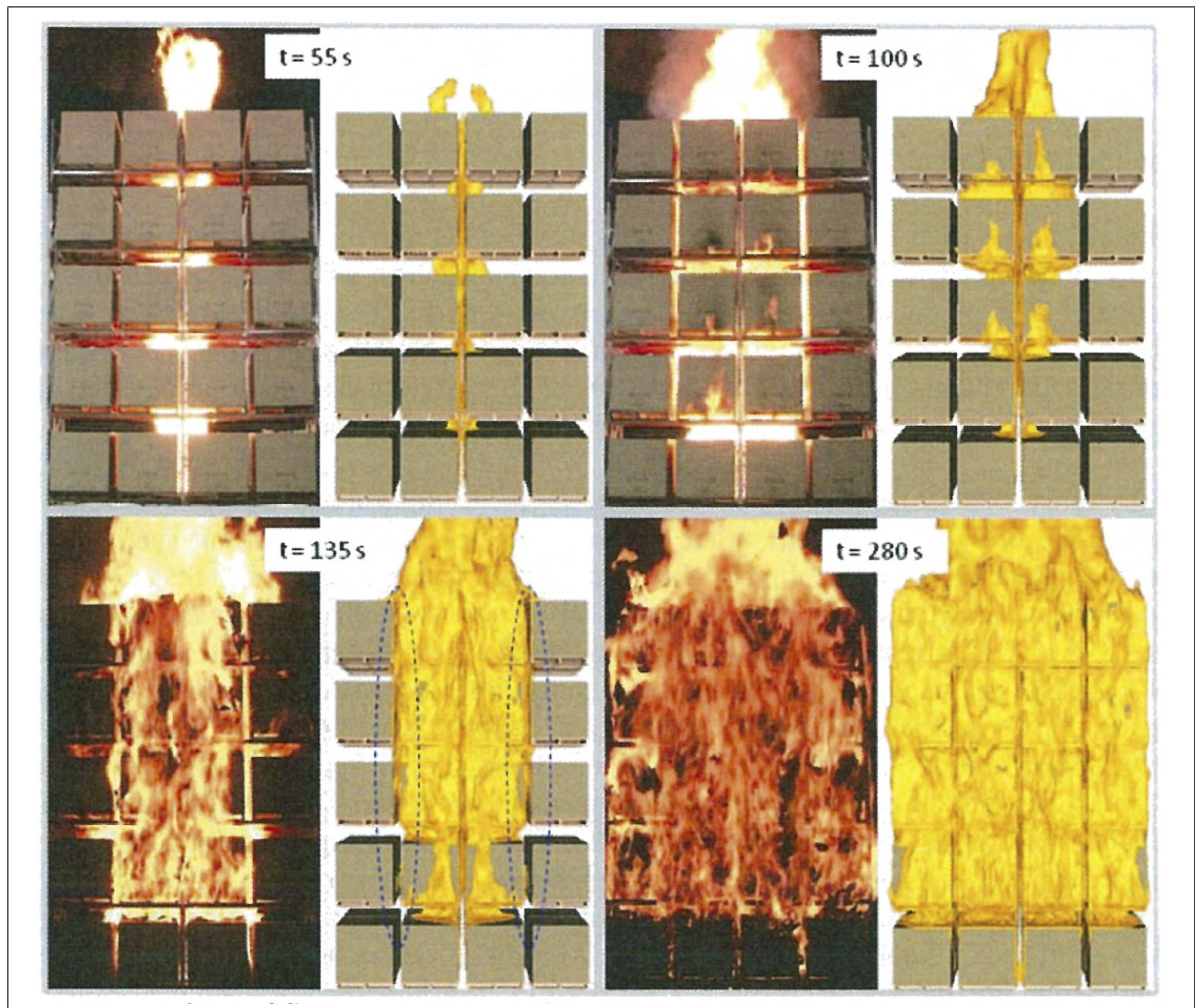


Figure 1.5. Comparison of flame snapshots at different times; flame shape is illustrated by an iso-contour of mixture fraction at the stoichiometric ratio, simulated in FireFOAM. Source: Ren, N. et. al. (2015).

1.4.2.2. Theory-based validation

Table 1.3 shows typical theoretical-based validation techniques. These techniques are complementary to the experimental-based and help verify the internal consistency of the solver, e.g. conservation of physical properties implicitly present in Navier-Stokes relations.

Solvers that perform Direct Numerical Simulation (DNS from now on) show the best performance in theoretically-based validation⁴. DNS simulations are limited to small-scale laminar flames and cannot currently be used for simulations of the large-scale wall fire configurations that are relevant to fire safety engineering applications (Ren et al., 2016), therefore LES-based solver validation must include theoretically-based techniques that account for the multi-physics modeling capabilities and limitations regarding the conservation of properties and convergence.

Table 1.3. Techniques with their respective physical properties used for internal software validation.

Theoretically-based	
Time series (interval-averaged)	Spatial distribution (mean, RMS or instantaneous)
→ Global energy conservation → Grid convergence studies: - Temporal series: Velocity, temperature, heat fluxes (convective or radiative)	→ Grid convergence studies: - Mean or RMS fields: Velocity, temperature, heat fluxes (convective or radiative)

The global energy conservation test is an indicator of how the solver's models and discretization attain to the Fluid Continuum governing equations, also known as the Navier-Stokes formulation. This formulation is by construction conservative, therefore any misalignment from this tendency shows failure in either modeling and/or discretization methods.

⁴In this work DNS will be used as a generic term that refers to fully-resolved, first-principles, unsteady reactive Navier-Stokes simulations, including detailed simulations of laminar flame-flow phenomena. These methodologies use a finite rate, global chemistry approach in which pyrolysis and combustion processes are described using semi-empirical Arrhenius-like expressions (Ren et. al., 2016)

FireFOAM uses the Finite Volume Method for spatial discretization, which by construction is also conservative, therefore the global energy conservation test in this case will provide insight on multi-physics modeling quality.

In the case of ETFT Test I, an energy conservation study based on the experimental measurements was performed by Maluk et al. (2017). This exercise consisted in an energy balance inside the compartment volume for four scenarios in the ETFT series of experiments, including ETFT Test I, which stated that the fire's HRR (\dot{Q}_{fire}) was equal to the sum of the energy loss in the opening ($\dot{Q}_{out,opening}$), the heat loss in the boundaries ($\dot{Q}_{boundaries}$), and the change in the gas phase internal energy (\dot{Q}_{gas}):

$$\dot{Q}_{fire} = \dot{Q}_{out,opening} + \dot{Q}_{boundaries} + \dot{Q}_{gas} \quad (1.2)$$

Each term was calculated using experimental measurements and physical properties. The results for ETFT Test I are shown on Figure 1.6. In this Figure, \dot{Q}_{total} corresponds to the right hand-side of eq. (1.2), and neglects radiation through openings in the term $\dot{Q}_{out,opening}$. Fresh air inflow enthalpy at openings is also neglected, as it is stated that is negligible compared on the hot gas outflow enthalpy. As an example to illustrate the authors' analysis, it is stated that the fact that the sum of the different components is lower than the total heat release from the burner (of the order of 10 to 20%) is indicative of radiant losses not being negligible, and apparently larger as the HRR increases (Maluk et al., 2017). This type of analysis clarifies the fact that the energy balance has the purpose of validating measurement quality based on a theoretical basis. As this work will perform a theory-based energy balance using FireFOAM simulation data, more insight on the methodology for calculating each term of eq. (1.2) will be introduced in Section 3.5 Data Management.

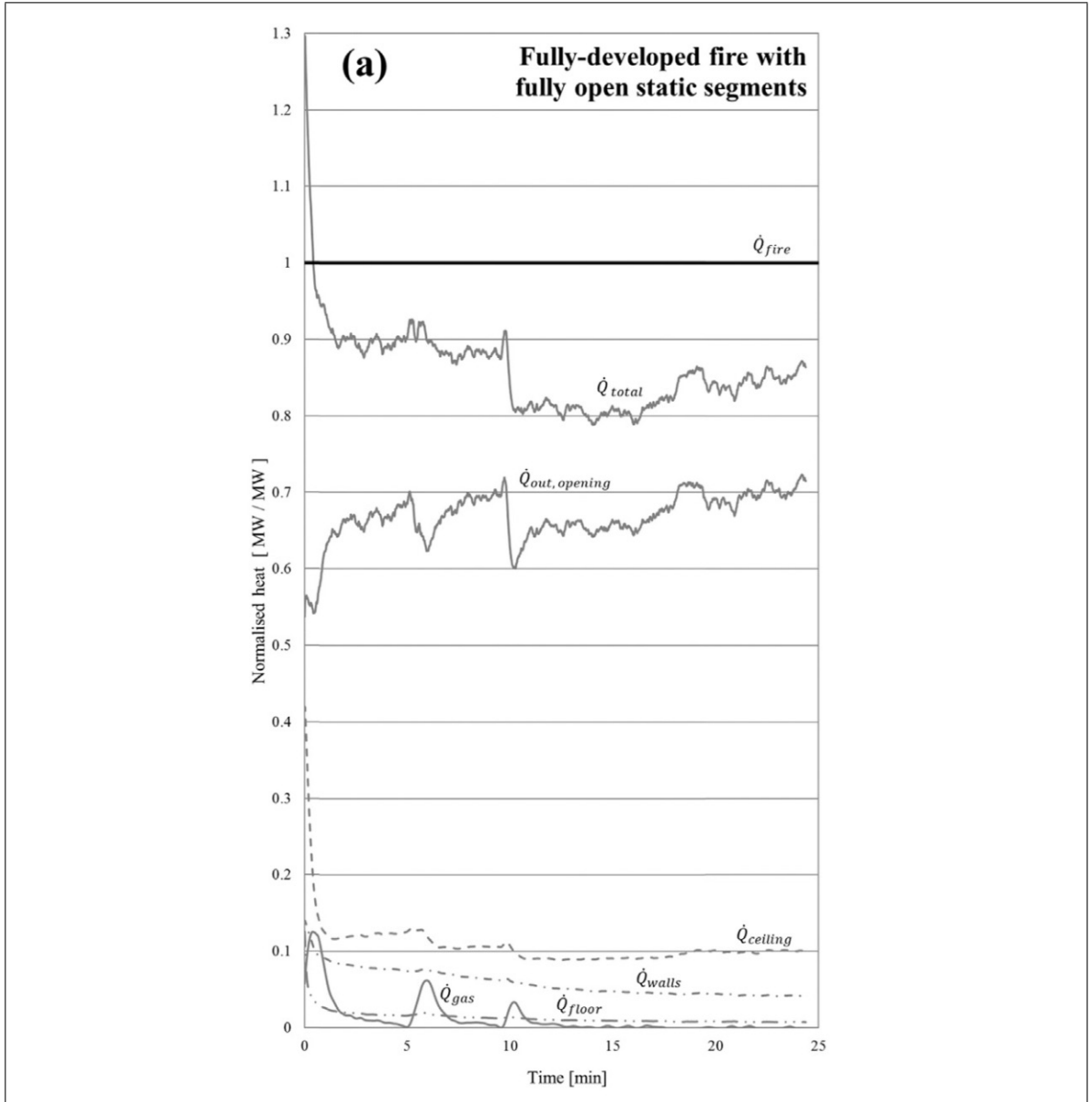


Figure 1.6. Normalized (against \dot{Q}_{fire}) heat distribution for ETFT E-I: fully-developed fire with fully open static segments. Source: Maluk, C. et. al. (2017).

Going back to Table 1.3, apart from global energy conversion techniques, there are grid convergence studies. These studies plot simulation results (e.g. time-series or spatial distribution 1D plots) for different grid sizes. As the grid size decreases, a well formulated

solver should deliver practically the same results past a specific grid size, in other words the simulations should deliver grid-convergent results. Different types of solver have different grid-convergence sizes which depend on the simulation scale and implemented models. For example, in a 145 cm vertical wall-resolved fire simulated in FireFOAM the authors conclude that grid convergence is achieved with a 3 mm computational mesh (Ren et al., 2016). This example is shown in Fig. 1.7.

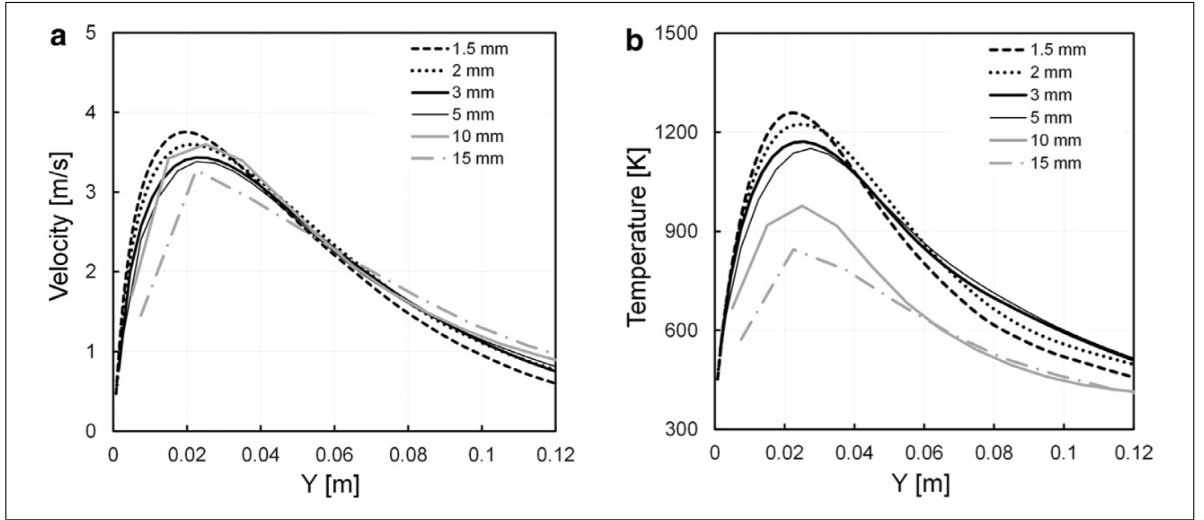


Figure 1.7. Wall-normal variations of mean properties at $z=77$ cm: (a) mean vertical flow velocity; (b) mean temperature. Grid convergence study: $\Delta y_w = 1.5, 2, 3, 5, 10$ and 15 mm. Source: Ren, N. et. al. (2016).

In this same work, a grid-convergence study using RMS temperature and velocity (wall-normal, and wall-vertical components) is performed, therefore the physical property type (e.g. temperature or velocity) and presentation method (e.g. mean or RMS) is the author's choice.

1.4.3. Up-To-Date FireFOAM Performance

Validation exercises are case-specific, this implies that if FireFOAM—or any numerical solver—shows good agreement with experimental data it does not necessarily mean that FireFOAM will behave satisfactory in all engineering design scenarios, hence numerous

validation exercises for different scenarios are available in literature⁵. There are three factors used to assess experimental-based validation exercises:

- Experimental scale⁶.
- Measurement resolution.
- Variables measured: Physical and derived quantities.

In fire safety engineering design, an ideal experimental-based validation exercise will include a large experimental scale, high measurement resolution, and all types of variables measured, including both physical (e.g. temperature, molar concentrations, etc.) and derived (e.g. combustion efficiency, global heat release rate, etc.) quantities. ETFT E-I has all these characteristics, therefore it presents an ideal opportunity to test FireFOAM's capabilities when simulating real scale fires. As a base to elaborate the Validation Structure for this exercise —later presented in Section 2.2—, an overview of FireFOAM's performance up to date will be shown.

First FireFOAM validation exercises (Trouvé & Wang, 2010) involve fuel-supply controlled⁷ small scale pool-fire configurations, i.e. 30 cm x 30 cm pool fire measurements by McCaffrey (1979). As Figure 1.8 shows, these first validation exercises mostly centered on experimental correlations, which were calculated from approximately 5 cm data resolution of physical properties by McCaffrey (1979).

⁵Up to date (Sept. 2017), approximately 20 formal FireFOAM validation exercises have been published in recognized Journals/Books.

⁶From now on, "large experimental scale" will be referred to experimental domains with the same spatial scale as building/house plane floors or rooms, i.e. for characteristic dimensions of 3 m and above.

⁷In fire characterization literature, there are two types of fire regimes: fuel-supply controlled and ventilation controlled. In the first type, combustion is controlled by the fuel supply (i.e. pool size and evaporation rate), while in the second type combustion is controlled by oxidant supply (i.e. ceiling height and openings size and distribution).

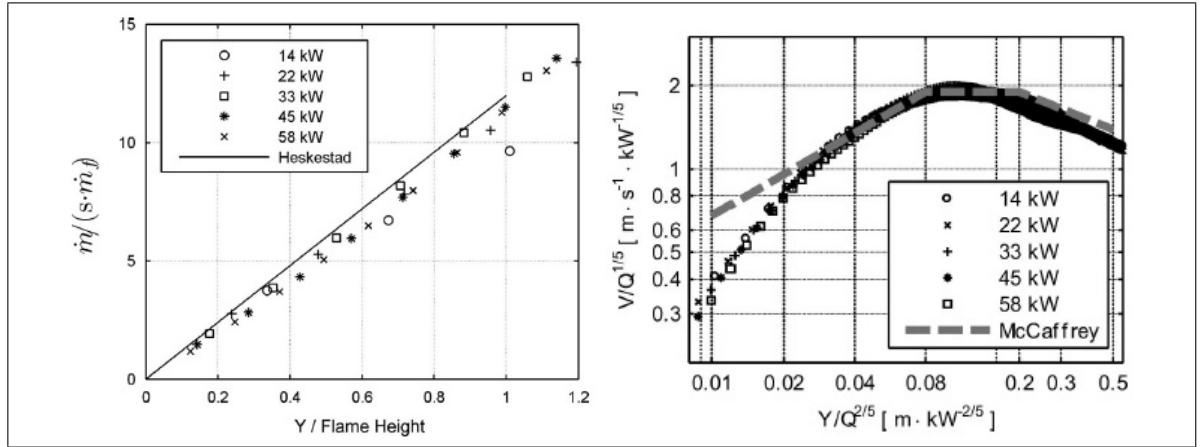


Figure 1.8. Comparison between simulations (symbols) and an empirically-based correlations proposed by Heskestad (solid line) and McCaffrey (dashed line). Left: Vertical variations of time averaged vertical mass flow rate. Elevation is made non-dimensional by the flame height; mass flow rate is made non-dimensional by the air mass flow rate required for stoichiometric complete combustion. Right: Vertical variations of time-averaged centerline vertical flow velocity. Elevation is scaled by the heat release rate raised to the power (2/5); flow velocity is scaled by the heat release rate raised to the power (1/5). Source: Trouvé & Wang (2010).

Good agreement is seen between correlations and simulation data, but the authors insist that further progress in fire modeling will require the development of high-resolution experimental databases corresponding to well-controlled laboratory configurations (Trouvé & Wang, 2010).

Later experimental-based validation exercises started testing FireFOAM's capabilities in large scale experimental warehouse scenarios, like rack-storage fires of cardboard boxes resting on top of wooden wood pallets (Ren et al., 2015) and accidental Hydrogen releases (Bauwens & Dorofeev, 2014). Both scenarios are shown in Fig 1.9 for reference. The rack-storage fires validation was based on vertical velocity measured at 3 heights and global HRR, while the accidental Hydrogen release validation was based on molar Hydrogen concentration measurements —see Figure 1.9 (Below) for sensor locations—. Figures 1.10 (Ren et al.,

2015) and 1.11 (Bauwens & Dorofeev, 2014) show both validation results for these large-scale exercises. Note that the rack-storage fire results in Figures 1.10 are the same as in Figure 1.5.

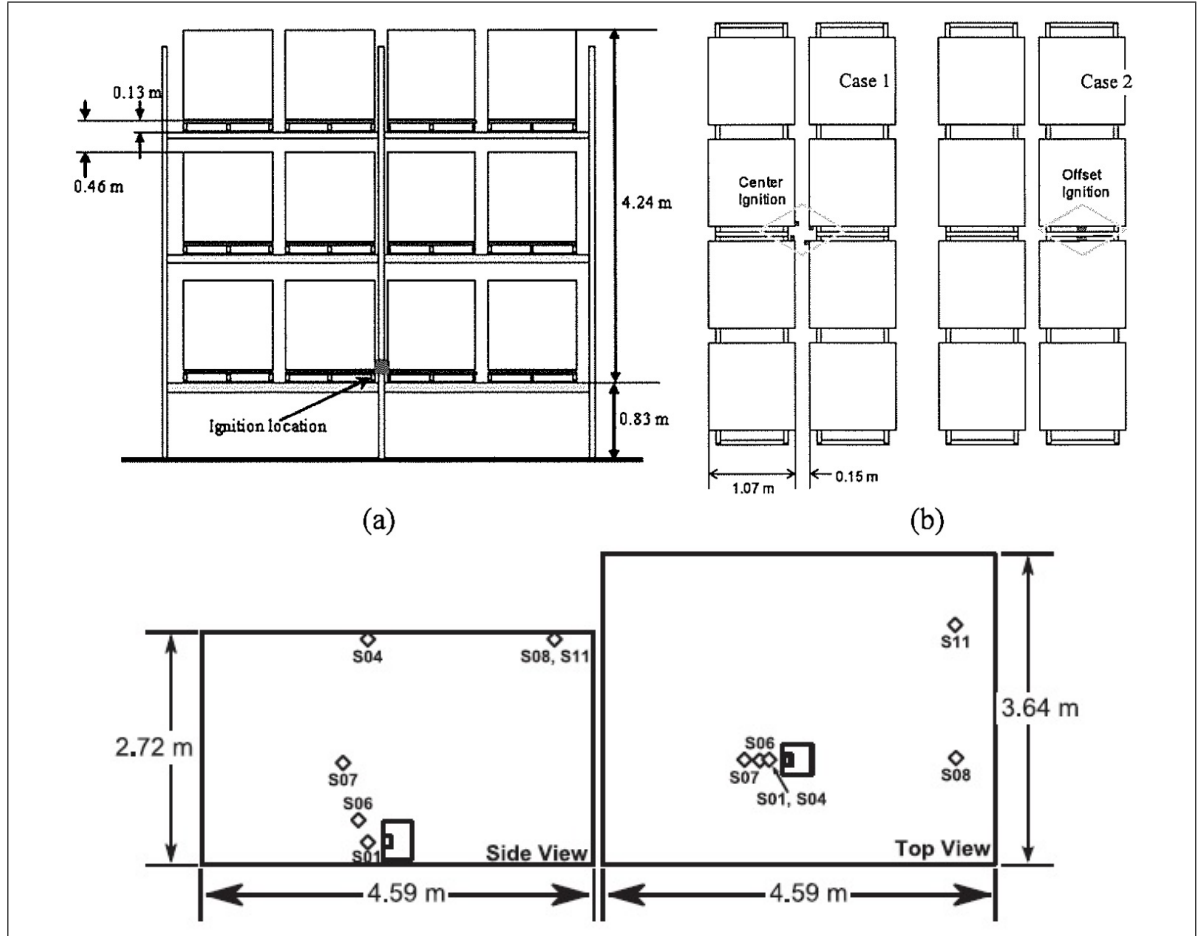


Figure 1.9. Warehouse scenarios for large scale experimental-based validation. Above: Illustration of 2x4x3 rack storage experimental configurations; (a) elevation view; (b) plan view showing the two ignition locations for the two different experiments: Cases 1 and 2. Below: Illustration of dimensions of 1/6 scale warehouse and H₂ sensor locations.

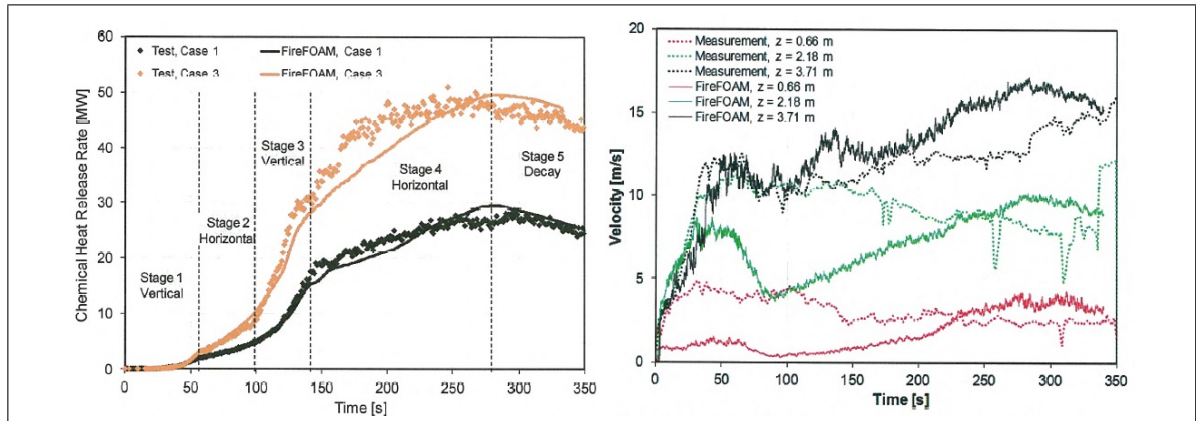


Figure 1.10. Left: Chemical HRR for two experimental rack-storage cases. Right: Vertical velocity time history at three elevation locations above the igniter for Case 1. Source: Ren, N. et. al. (2015).

The rack-storage fire validation exercise (see Figure 1.10) shows good agreement between simulation results and experimental measurements. The Global Heat Release Rate (HRR) time-series shows how FireFOAM captures different fire spread stages, as calculated and measured HRR time-series slopes closely match (Ren et al., 2015). With respect to vertical flow velocity, good agreement is seen for the top sensor (top tiers), while under prediction is seen at the beginning for medium and low tier measurements. This is because the flow mixing is insufficient at lower elevations due to lack of mesh resolution, therefore lower velocities result from the simulation. At top tier measurements the flow is turbulent and well mixed, therefore velocities are in good agreement (Ren et al., 2015). Fine meshes near the mixing zone (e.g. ETFT Test I burner outlet) are a necessity, specially in laminar-turbulent transition zones. Conclusions like this are valuable for big scale validation, where fine meshes cannot be afforded in the whole domain and therefore careful assessment for mesh refinement location has to be made.

The hydrogen release study (see Figure 1.11) shows good global agreements of H_2 molar concentrations. Local discrepancies are more notorious after 10s and at lower concentrations —at the highest sensors: S04, S08 and S11—, where Hydrogen in the coarsest mesh

disperses faster (Bauwens & Dorofeev, 2014). Again, conclusions like this are valuable and account for the solver's performance under different grid sizes in big scale validation.

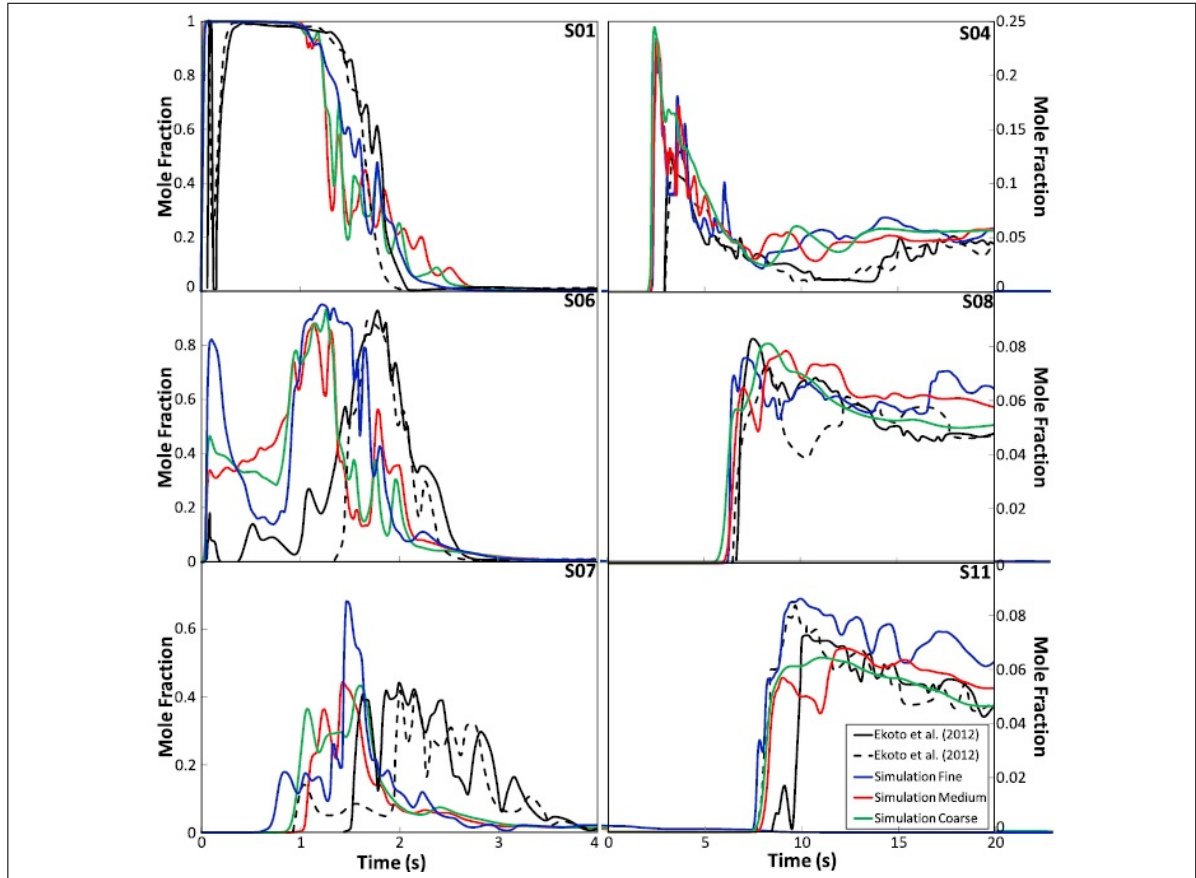


Figure 1.11. Comparison of experiment (two repeated experiments) and simulation data (three mesh sizes) for different sensor locations. Source: Bauwens et. al. (2014).

Even though these exercises validate FireFOAM for large-scale scenarios, measurement resolution is not big enough for testing LES capabilities in the whole domain with directly-measured data (and not interpolated), and the limited number of measured variables does not allow testing FireFOAM's ability to predict all relevant quantities for fire hazard evaluation, including but not limited to: smoke dynamics (e.g. dispersion and composition, plume interface height, and entrained vertical mass flow), temperature map evolution, and flame height. Despite of the above, these exercises are necessary as they leave important lessons

for further research. These particular cases provide useful insight on FireFOAM’s turbulence model (i.e. LES) performance in different mesh sizes, and fire spread models (i.e. pyrolysis).

Most recent FireFOAM experimental-based validation exercises are focused on testing new implemented capabilities —e.g. flame extinction models (Vilfayeau et al., 2016) and spray droplet evaporation with thin liquid superficial film models (Ren et al., 2017)—, and the solver’s DNS capabilities —e.g. wall-resolved heat transfer in vertical wall fires (Ren et al., 2016)—. This recent work’s experimental measurements are centered on specific validation of new features, i.e. only variables associated with the specific tested model are measured and analyzed in depth. Usually, experimental thermocouple measurements are first included as a validation technique —mainly to verify the global agreement with the experimental scenario—, but its measurement resolution is far less from the one needed for high quality physical-magnitude validation.

For the flame extinction models tested by Vilfayeau et al. (2016), mostly combustion efficiency vs. oxidant concentration in air flow supply is validated against experimental data (see Figure 1.12). Good agreement between experimental data and FireFOAM simulations is seen. Flame extinction occurs for an oxidizer composition of $X_{O_2} < 0.13$.

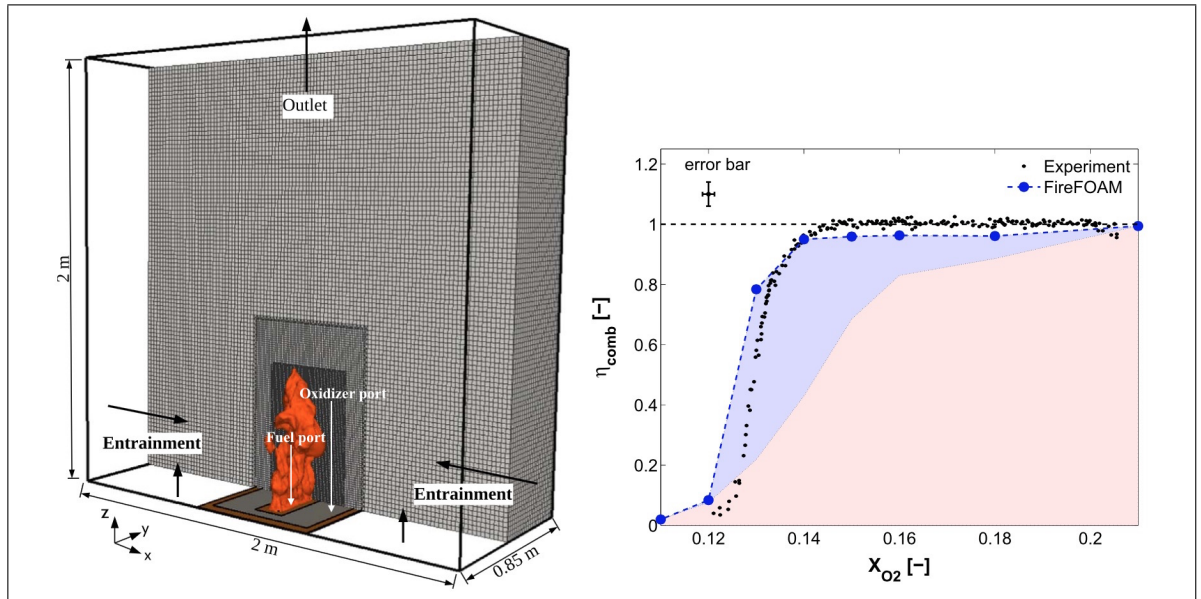


Figure 1.12. Left: Computational domain with multi-level mesh refinement. The Fuel port supplies pure gaseous methane, while the oxidizer port supplies a $\text{O}_2\text{-N}_2$ mix with variable composition. Right: Combustion efficiency resulting from different oxidizer composition. Red area represents contribution to combustion from flame ignition, and blue area from flame re-ignition. Source: Valfayeau et. al. (2016).

In the spray droplet evaporation and thin liquid superficial film models tested by Ren et al. (2017), only temperature and velocity parameters were validated against experimental data (see Figure 1.13). Other parameters, such as global HRR are compared to unsuppressed fire (no suppression water mist injection) simulations, hence does not correspond to experimentally-based validation.

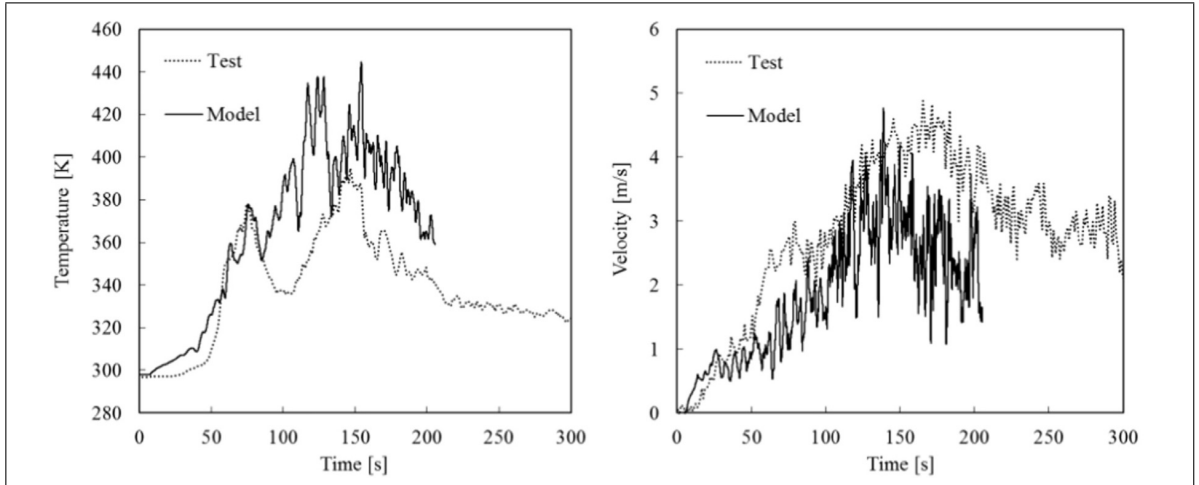


Figure 1.13. Temperature and radial velocity under the ceiling for the 2 6 5 water-mist suppressed case. Source: Ren, N. et. al. (2017).

In DNS of heat transfer on wall-resolved fires (Ren et al., 2016), mainly global heat transfer along the vertical direction of the wall (see Figure 1.14) and temperature profiles are used as a validation tool. The just mentioned temperature validation profiles are the ones shown in Figure 1.3 as an example of experimental-based validation.

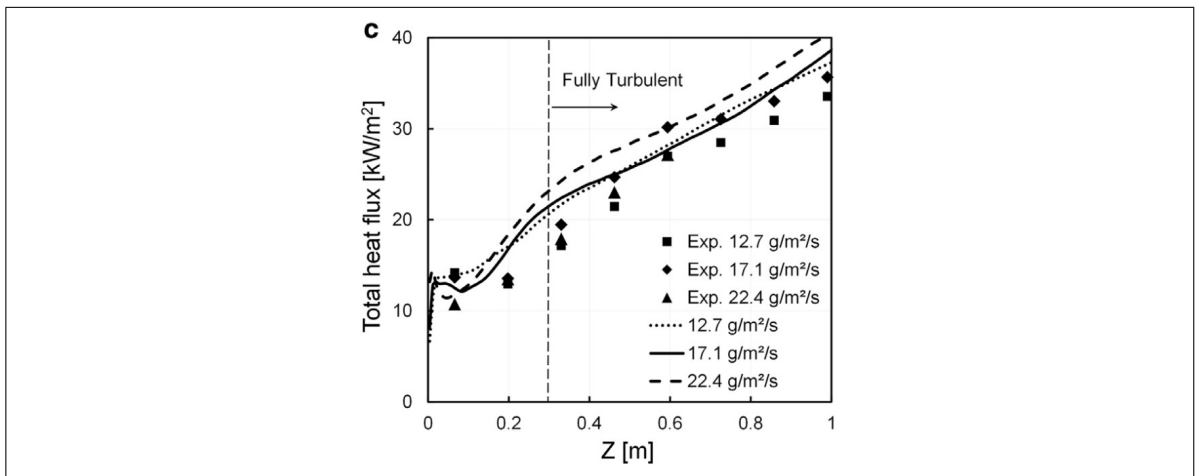


Figure 1.14. Vertical variations of the mean total wall heat flux for variable fuel mass flow rate. LES data is compared to experimental measurements. The vertical dashed line marks the boundary between laminar and turbulent flame regions. Source: Ren et. al. (2016).

It is important to note that these works' scope is new model validation, hence one cannot ask for complete validation of a series of physical/derived quantities through multiple spatial locations and temporal series. Concluding Chapter I, there is clearly a need for FireFOAM validation in big scales fires with high resolution measurement data of multiple variables, both physical and derived quantities. In other words, as stated by J. Hidalgo et al. (2017), there is a clear need to supplement existing frameworks and models to characterize the full range of potential fire dynamics and energy transport for fires in compartments representative of real buildings.

2. CHAPTER 2: THEORETICAL BACKGROUND

This chapter presents the physical models implemented in FireFOAM, alongside with some considerations and assumptions for this particular validation exercise.

2.1. Navier-Stokes System and Turbulence

CFD modeling consists of solving a set of conservation equations in a domain of interest, obtaining as a result relevant physical fields such as pressure, temperature, and velocity, among others. The set of conservation equations form the well-known Navier-Stokes system, consisting of a five Partial Differential Equation (PDE) system¹ that describes the behaviour of a fluid continuum². The Navier-Stokes system is derived by applying mass, momentum and energy conservation in a differential material volume contained in the fluid continuum. In a few words, Navier-Stokes is Newton's second law applied to a fluid, usually combined with mass continuity and energy conservation. In this work, internal forces derived from internal friction are modeled considering a Newtonian Fluid³. The general fluid dynamics problem to be solved is:

Find the velocity vector (\mathbf{u}), mass density (ρ), pressure (p), internal energy (E_i) and temperature (T), for $t \in [t_0, t_f]$, in a domain $\Omega \subset \mathbb{R}^3$, given conditions (2.1) to (2.7). Fluid properties —such as dynamic viscosity (μ), thermal conductivity (k) and constant-volume specific heat (c_v) — are usually calculated using expressions that depend on local temperature. R represents the gas constant, calculated as the universal gas constant (R_u) divided in the mixture molar weight. $\overline{\mathbf{S}}_M$ corresponds to the body force field (e.g. gravitation), S_E to the work done by this body force field, and Φ to the viscous dissipation term which accounts for conversion of mechanical energy to heat.

¹Five PDEs conforming the Navier-Stokes system: mass conservation (1), momentum conservation in each direction (3), and energy conservation (1).

²A fluid continuum is defined as a medium that continuously deforms under the effect of shear stress.

³A Newtonian Fluid is defined as a continuum whose internal stresses are proportional to the velocity gradients in the shear-stress perpendicular direction. The constant of proportionality is known as dynamic viscosity, μ .

$$\textbf{Mass conservation (1):} \quad \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0 \quad (2.1)$$

$$\textbf{Momentum conservation (3):} \quad \rho \frac{D\mathbf{u}}{Dt} = -\nabla p + \nabla \cdot (\mu \nabla \mathbf{u}) + \overline{\mathbf{S}_M} \quad (2.2)$$

$$\textbf{Energy conservation (1):} \quad \rho \frac{DE_i}{Dt} = -p(\nabla \cdot \mathbf{u}) + \nabla \cdot (k \nabla T) + \Phi + S_{E_i} \quad (2.3)$$

$$S_{E_i} = S_E - \overline{\mathbf{S}_M} \cdot \mathbf{u}$$

$$\begin{aligned} \Phi = \mu \left[2 \left(\left(\frac{\partial u}{\partial x} \right)^2 + \left(\frac{\partial v}{\partial y} \right)^2 + \left(\frac{\partial w}{\partial z} \right)^2 \right) + \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right)^2 + \left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} \right)^2 \right. \\ \left. + \left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \right)^2 \right] - \frac{2}{3} \mu (\nabla \cdot \mathbf{u})^2 \end{aligned}$$

$$\textbf{Ideal gas state equation (1):} \quad p = \rho RT \quad (2.4)$$

$$\textbf{Internal energy ideal gas (1):} \quad E_i = c_v T \quad (2.5)$$

$$\textbf{Boundary conditions:} \quad \text{Every field over } \partial\Omega \quad \forall t \quad (2.6)$$

$$\textbf{Initial conditions:} \quad \text{Every field for } t = t_0 \text{ in } \Omega \quad (2.7)$$

It can be noted that this is a seven equation system for seven variables, therefore numerical solving seems straight forward. In practice this is not the case, given that at Reynolds numbers⁴ present in most engineering problems turbulence becomes an issue. Turbulence in free flows occurs when inertial forces within the fluid are almost two magnitude orders bigger than viscous forces —effect usually seen at high speeds—, which causes the appearance of vortex structures of diverse spatial magnitudes⁵, which develop chaotic and random movements. These movements generate considerable velocity fluctuations within small scales, which makes numerical simulation of engineering problems computationally unfeasible. Typical effects of turbulence are energy dissipation and increase in mass diffusivity

⁴The Reynolds adimensional number, Re , quantifies the inertial forces over the viscous forces. Its definition is $Re_L = \text{Inertial Forces} / \text{Viscous Forces} = \rho \mathbf{u}^2 L^2 / (\mu \mathbf{u} L) = \rho \mathbf{u} L / \mu$. In free flows, like combustion in open compartments, local turbulence begins at $Re_L \geq 500$, where L is the vertical distance from the burning fuel to the laminar-turbulent transition zone.

⁵There are three scales defined in literature: Integral (order of hundreds of km), Taylor micro-scale (order of m to cm) and Komolgorov micro-scale (order of mm downwards).

due to mixing, which becomes specially relevant in multi-species mixtures typical of combustion. Vortex structures can reach the lower spatial order of micro-meters —e.g. a 0.1 m x 0.1 m x 0.1 m domain can contain $\sim 10 \mu\text{m}$ structures—, a size which is computationally unfeasible for engineering-level numerical meshes of at most centimeters to millimeters.

The above reasons explain why the Navier-Stokes system cannot be directly solved for typical engineering problems. As reference, Direct Numerical Simulations (DNS) —where all eddy scales are solved through the unsteady Navier-Stokes system— can be performed for problems with Re_L below the order of 100, a Reynolds scale well below the current engineering problems⁶. This is why FireFOAM uses turbulence models at sub-grid scales. The LES spatial filter cuts off sub-grid vortex structures, directly simulating only large eddies and modeling small eddies. Favre-averaging mitigates the effects of continuous velocity-pressure changes due to the chaotic and random states induced by turbulence. The following sections explain both LES-filtering and Favre-averaging, followed by the application of these to the Navier-Stokes system.

2.2. LES-filtering

As mentioned in Chapter 1: Introduction, transient large-scale motions and intermittency are intrinsic to turbulent diffusion flames (Almeida et al., 2015). This is why FireFOAM treats turbulence through the transient method of Large Eddy Simulation (LES), where eddies bigger than the grid size —or 'Large Eddies'— are simulated, and sub-grid eddies are modeled. The filtering process for modeling only Sub-Grid-Scale (SGS) structures is done by working with spatially-filtered velocity values, where the LES-filter filters out small eddies. The LES-filter is defined as the convolution of the filtering function and the physical property:

⁶For reference, a cigarette smoke line becomes turbulent after $\text{Re}_L \geq 500$, with L being the laminar line length (which is in the order of centimeters).

$$\overline{\psi}(\mathbf{x}, t) \equiv \int_{\mathbb{R}^3} G(\mathbf{x} - \mathbf{x}', \Delta) \psi(\mathbf{x}', t) d\mathbf{x}' \quad (2.8)$$

Where $\psi(\mathbf{x}, t)$ corresponds to a physical magnitude, and $\overline{\psi}(\mathbf{x}, t)$ its LES-filtered value. Herein $G(\mathbf{x}, \mathbf{x}', \Delta)$ corresponds to the Box Filter function, which cuts off values of the physical property function $\psi(\mathbf{x}, t)$ beyond a half filter width, being Δ the filter width. The Box Filter function is defined as:

$$G(\mathbf{x}, \Delta) = \frac{1}{\Delta} H\left(\frac{1}{2}\Delta - |\mathbf{x}| \right) \quad (2.9)$$

Where H is the Heaviside function.

2.3. Favre-averaging

According to CFD Online (2005) and what was stated in section 2.1, at Reynolds numbers typically present in most engineering problems the Navier-Stokes system usually has highly chaotic and turbulent solutions. Therefore, it is not possible to solve the instantaneous Navier-Stokes equations, and hence it is necessary to model the influence of the smallest scales. Most turbulence models are based on one-point averaging of instantaneous equations (CFD Online, 2005), where in the case of FireFOAM is according to the Favre-averaging procedure.

Classical time-averaging (Reynolds averaging) of a physical magnitude Φ is done according to:

$$\overline{\Phi} \equiv \frac{1}{T} \int_T \Phi(t) dt \quad (2.10)$$

Where $\bar{\Phi}$ corresponds to the averaged time-dependant variable $\Phi(t)$ over a T interval. The fluctuation (Φ') is defined as:

$$\Phi' = \Phi - \bar{\Phi} \quad (2.11)$$

Density weighted time-averaging (Favre averaging) of the same physical magnitude Φ is done according to:

$$\tilde{\Phi} \equiv \frac{\overline{\rho\Phi}}{\bar{\rho}} \quad (2.12)$$

Where $\tilde{\Phi}$ is the Favre-filtered variable. The fluctuation here is defined as:

$$\Phi'' = \Phi - \tilde{\Phi} \quad (2.13)$$

The above definitions of fluctuation present the following properties: $\overline{\Phi'} = 0$; $\overline{\Phi''} \neq 0$.

2.4. Gas-Phase Governing Equations

This section explains the gas-phase model implented in FireFOAM, as described by Almeida et al. (2015). After applying the LES-filter and using density-averaged mean Favre variables, the continuity equation for compressible flows is:

$$\text{Mass conservation: } \frac{\partial \bar{\rho}}{\partial t} + \nabla \cdot (\bar{\rho} \tilde{\mathbf{u}}) = 0 \quad (2.14)$$

Where $\bar{\rho}$ and $\tilde{\mathbf{u}}$ correspond to the gas-phase density and velocity vector, respectively. In this and the following equations, $\overline{\psi}$ corresponds to the filtered ψ quantity and $\tilde{\psi}$ the filtered density-weighted ψ quantity. Given these definitions, the following equivalence is established:

$$\tilde{\psi} = \frac{\overline{\rho\psi}}{\bar{\rho}} \quad (2.15)$$

FireFOAM treats the gas-phase as ideal, therefore the following state equations are used:

$$\bar{\rho} = \frac{\bar{p}M}{R\tilde{T}}; \quad M = \left[\sum_{i=1}^N \frac{\tilde{Y}_i}{M_i} \right]^{-1} \quad (2.16)$$

Where p , M , R , T and Y_i corresponds to the absolute pressure, molar weight, universal gas constant (8.314 kJ/kmol/K), absolute temperature, and mass fraction of the i -th specie (C_3H_8 , O_2 , CO_2 , H_2O), respectively.

The gas-phase total enthalpy, later to be used in the energy conservation equation, is calculated as a mixture considering each species property:

$$\tilde{h}_s = \sum_{i=1}^N \tilde{Y}_i h_{s,i}(\tilde{T}); \quad h_{s,i}(T) = h_{s,i}^{ref} + \int_{T_{ref}}^T c_{p,i}^0 dT \quad (2.17)$$

Where each species total enthalpy ($h_{s,i}$) is calculated as the sum between its formation enthalpy ($h_{s,i}^{ref}$) and its sensible enthalpy. The specific heat capacity —at constant reference atmospheric pressure— of the i -th specie ($c_{p,i}^0$) was treated as the same polynomial expressions used in the FireFOAM tutorial cases. As Favre mean variables were directly used in equations (2.16) and (2.17), cross relations between mass fractions and powers of temperature are ignored, therefore this may lead to numerical errors (Almeida et al., 2015).

The momentum transfer equation, considering a Newtonian fluid and the Boussinesq hypothesis —introduced and explained in the following paragraph— for the sub-grid scale tensor, takes the form:

$$\begin{aligned} \textbf{Momentum conservation: } \quad & \frac{\partial \bar{\rho} \tilde{\mathbf{u}}}{\partial t} + \nabla \cdot (\bar{\rho} \tilde{\mathbf{u}} \tilde{\mathbf{u}}) = -\nabla \bar{p}_m - (\mathbf{g} \cdot \mathbf{x}) \nabla \bar{\rho} \\ & + \nabla \cdot \left[\mu_{eff} \left(\nabla \tilde{\mathbf{u}} + (\nabla \tilde{\mathbf{u}})^T - \frac{2}{3} (\nabla \cdot \tilde{\mathbf{u}}) \mathbf{I} \right) \right] \end{aligned} \quad (2.18)$$

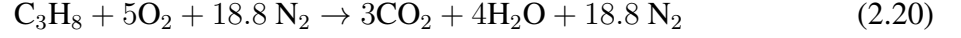
Where μ_{eff} is the effective viscosity, calculated as the sum between the sub-grid scale viscosity (μ_{sgs}) and the dynamic viscosity (μ). The dynamic viscosity was calculated as temperature-dependant through Sutherland's law. \mathbf{I} corresponds to the identity matrix. The Boussinesq eddy viscosity assumption postulates that the momentum transfer caused by turbulent eddies can be modeled with an eddy viscosity, which is analogous to the modeling of momentum transfer in molecular motion through a molecular viscosity. Note that in eq. (2.18) the Favre-averaged pressure \bar{p}_m is not the Favre-averaged thermodynamic pressure (\bar{p}), but a modified pressure calculated as:

$$\bar{p}_m = \bar{p} - \bar{\rho}(\mathbf{g} \cdot \mathbf{x}) \quad (2.19)$$

This modification is done as to improve the numerical solution (Almeida et al., 2015). In equation (2.18) the isotropic part of the sub-grid scale tensor, $(2/3)\rho k$ — where k is the turbulent kinetic energy—, was neglected as it is considered a valid assumption for Mach numbers under 0.4 (Almeida et al., 2015).

The reaction model consists in a single-step, infinitely fast, complete combustion reaction that neglects soot formation. This is a good assumption for a compartment-scale propane diffusion flame, as soot formation/oxidation is usually relevant only in highly sooting fuels, or in large pool fires with crosswinds (e.g. large-scale LNG sea pool fires), where limited air

entrainment at the center of the pool and winds promote uneven soot distribution (C. Wang, Wen, & Chen, 2014). Each kmol of propane reacts according to the following stoichiometry:



The gas composition is determined by solving the continuity —or mass conservation— equations for each of the four reacting species:

$$\text{Species mass conservation: } \frac{\partial \bar{\rho} \tilde{Y}_i}{\partial t} + \nabla \cdot (\bar{\rho} \tilde{\mathbf{u}} \tilde{Y}_i) = \nabla \cdot (\bar{\rho} D_{eff} \nabla \tilde{Y}_i) + \bar{\omega}_i \quad (2.21)$$

$$i = \text{C}_3\text{H}_8, \text{O}_2, \text{H}_2\text{O}, \text{CO}_2$$

Where the effective mass diffusivity (D_{eff}) is calculated as the sum of the species individual diffusivities in the mixture (D_i), and the sub-grid scale mass diffusivity (D_{sgs}). The nitrogen mass fraction was calculated from the following closure relation rather than through a transport equation:

$$\sum_{i=1}^5 Y_i = 1 \quad (2.22)$$

The propane reaction rate was calculated through the Eddy Dissipation Concept proposed by Magnussen and Hjertager (1977), considering an infinitely fast chemical reaction:

$$\bar{\omega}_{\text{C}_3\text{H}_8} = -C_r \bar{\rho} \frac{\varepsilon}{k} \min \left[\widetilde{Y_{\text{C}_3\text{H}_8}}, \frac{\widetilde{Y_{\text{O}_2}}}{s_t} \right] \quad (2.23)$$

Where C_r is the model constant equal to 4 (Magnussen & Hjertager, 1977), s_t the oxygen-propane mass stoichiometric ratio equal to 3.6363, ε the rate of dissipation of turbulent kinetic energy, and k the turbulent kinetic energy. Once calculated the propane reaction rate ($\overline{\omega_{C_3H_8}}$), the other species' reaction rate is obtained through the proportions established in reaction (2.20).

The energy conservation equation is solved in terms of the total enthalpy:

$$\textbf{Energy conservation: } \frac{\partial \bar{\rho} \tilde{h}_s}{\partial t} + \nabla \cdot (\bar{\rho} \tilde{u} \tilde{h}_s) = \frac{D\bar{p}}{Dt} + \nabla \cdot (\bar{\rho} \alpha_{eff} \nabla \tilde{h}_s) + \overline{Q_c} - \nabla \cdot \overline{\mathbf{q}_R} \quad (2.24)$$

Where the effective thermal diffusivity (α_{eff}) is the sum between the mixture thermal diffusivity (α) and the sub-grid scale diffusivity (α_{sgs}). The sub-grid scale thermal diffusivity (α_{sgs}) was calculated assuming a unity Prandtl number: $\alpha_{sgs} = \nu_{sgs} = \mu_{sgs}/\rho$. The mixture thermal diffusivity (α) is calculated as:

$$\alpha = \frac{\gamma}{\bar{\rho} c_p}; \quad c_p = \sum_{i=1}^N \tilde{Y}_i c_{p,i}^0(\tilde{T}) \quad (2.25)$$

Where the gas-phase thermal conductivity (γ) is determined through the modified Eucken method (Almeida et al., 2015). The heat generated by combustion ($\overline{Q_c}$) is obtained through the propane reaction rate:

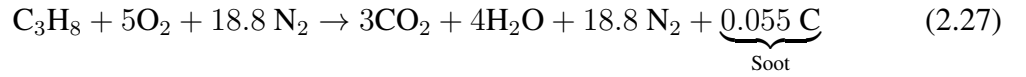
$$\overline{Q_c} = \Delta H_c^0 \cdot \overline{\omega_{C_3H_8}} \quad (2.26)$$

Where ΔH_c^0 is the enthalpy variation in the complete iso-thermic propane combustion—also known as the inverse additive of the heat of combustion—, with a value of -46,450

kJ/(kg C₃H₈) according to J. Hidalgo et al. (2017). The computation of the divergence of thermal radiation flux, $\nabla \cdot \overline{\mathbf{q}_R}$, is explained in the next section.

2.5. Radiation in participating media

Only C₃H₈, H₂O and CO₂ participate in radiation, as O₂ and N₂ are diatomic molecules and hence do not present strong emission bands in the infrared region of the electromagnetic spectrum (Almeida et al., 2015). Soot is only considered to participate in radiation scattering—and not in gas composition—as a constant soot fraction, according to the chemical reaction:



As no reference values were available for propane diffusion flames, the 0.055 constant was taken from the methane-fueled 'smallPoolFire2D' tutorial case. As propane is a medium-sooting fuel, it was deemed better to include an under-estimated soot fraction—corresponding to the value for methane—rather than no soot fraction.

The spectral Radiative Transfer Equation (RTE) is given by the expression:

$$\begin{aligned} \nabla \cdot (\hat{\mathbf{s}} I_\lambda) ds &= \hat{\mathbf{s}} \cdot \nabla I_\lambda ds \\ &= \underbrace{dI_\lambda}_{\text{Variation in spectral intensity}} = \underbrace{\kappa_\lambda I_{b\lambda} ds}_{\text{Augmentation due to emission}} - \underbrace{\kappa_\lambda I_\lambda ds}_{\text{Attenuation due to absorption}} \end{aligned} \quad (2.28)$$

Where κ_λ is the spectral absorption coefficient for radiation wavelength λ , with SI units of m⁻¹. I_λ is the spectral intensity, with SI units of watts per square-meter per steradian⁷ per

⁷Steradian, noted as 'sr' in the SI system despite of being adimensional, corresponds to the tri-dimensional equivalent of a radian to measure a solid angle Ω . It is calculated based on a sphere of radius r that subtends an S surface, according to $\Omega = \frac{S}{r^2}$. A whole sphere represents a solid angle of $\Omega = 4\pi r^2/r^2 = 4\pi$ sr.

unit wave-number —or $\text{W m}^{-2} \text{sr}^{-1} \text{m}^{-1}$ equivalently—. The first and second terms in the rightmost hand side of equation (2.28) correspond to the augmentation and attenuation of radiation by emission and absorption, respectively, when the radiation ray propagates along direction $\hat{\mathbf{s}}$ (Almeida et al., 2015). Assuming a grey medium, i.e. the spectral absorption coefficient is wavelength independent and quantified as a mean adimensional Planck absorption coefficient κ_p , equation (2.28) can be integrated in the entire electromagnetic spectrum resulting in:

$$\begin{aligned} \int_{\lambda} \nabla \cdot (\hat{\mathbf{s}} I_{\lambda}) d\lambda &= \kappa_p \int_{\lambda} (I_{b\lambda} - I_{\lambda}) d\lambda \\ \rightarrow \nabla \cdot (\hat{\mathbf{s}} I) &= \kappa_p (I_b - I) \end{aligned} \quad (2.29)$$

The divergence of the total radiative heat flux $\nabla \cdot \overline{\mathbf{q}_R}$ —term associated with radiation loss in the energy conservation equation (2.24)— is calculated by integrating eq. (2.29) in the entire spherical solid angle:

$$\begin{aligned} \nabla \cdot \overline{\mathbf{q}_R} &= \int_{4\pi} \nabla \cdot (\hat{\mathbf{s}} I) d\Omega = \kappa_p \left(4\pi I_b - \int_{4\pi} I d\Omega \right) = \kappa_p (4\sigma T^4 - G) \\ \rightarrow \nabla \cdot \overline{\mathbf{q}_R} &= \kappa_p \left(\underbrace{4\sigma T^4}_{\text{Emission term}} - \underbrace{G}_{\text{Absorption term}} \right) \end{aligned} \quad (2.30)$$

Where the irradiation (G), defined as the radiation received from all directions with SI units of watts per square meter, is calculated as the radiation intensity (I) integrated along the entire spherical solid angle. For each chemical specie, the Planck mean absorption coefficient is obtained through the gas-phase temperature according to the polynomial expressions contained in the FireFOAM tutorial cases. The mixture coefficient (κ_p) is obtained by weighting the individual coefficients ($\kappa_{p,i}$) according to the partial pressures (p_i):

$$\kappa_p = \sum_{i=1} \kappa_{p,i} p_i; \quad p_i = \frac{\tilde{Y}_i M}{M_i} \bar{p} \quad i = \text{CH}_4, \text{H}_2\text{O}, \text{CO}_2 \quad (2.31)$$

The irradiation term (G) is obtained through the numerical solution of the RTE through the Finite Volume Discrete Ordinates Method (FVDOM), a method widely used to solve irradiation in CFD combustion problems and implemented in OpenFOAM. This method consists in splitting the radiation field in a user-defined number of solid angles, and then calculating the radiation intensity (I_i) within the i solid angles in the mean directions. Finally, irradiation is obtained through a numerical integration of radiation intensities along the spherical solid angle:

$$G = \int_{4\pi} I d\Omega \approx \sum_i I_i \cdot \Delta\Omega \quad (2.32)$$

For a full comprehension of the FVDOM implementation in OpenFOAM, the user is referred to the work performed by Mishra and Roy (2007).

2.6. Turbulence modeling

As mentioned in section 2.2, turbulence in FireFOAM is modeled throughout the Large Eddy Simulation (LES) method. The described Box Filter was used for the spatial filtering operation, where the filter cut-off width (Δ) was calculated taking the cubic root of the grid cell volume (Almeida et al., 2015).

The sub-grid scale stress tensor was modeled throughout the one equation eddy viscosity model, which solves turbulent kinetic energy conservation:

$$\frac{\partial \bar{\rho} k}{\partial t} + \nabla \cdot (\bar{\rho} \tilde{\mathbf{u}} k) = \nabla \cdot (\mu_{eff} \nabla k) - \bar{\rho} \tau_{sgs} : \tilde{\mathbf{S}} - C_e \bar{\rho} \frac{k^{3/2}}{\Delta} \quad (2.33)$$

Where $\tilde{\mathbf{S}}$ is the strain rate tensor and C_e is a constant equal to 1.048 (Almeida et al., 2015). The term $-\bar{\rho}\tau_{sgs} : \tilde{\mathbf{S}}$ corresponds to the production rate of turbulent kinetic energy, while $-C_e\bar{\rho}k^{3/2}/\Delta$ is responsible for the dissipation rate of this energy (Almeida et al., 2015). The sub-grid scale stress tensor (τ_{sgs}) is modeled through the Boussinesq approximation:

$$\tau_{sgs} = -2\mu_{sgs}\text{dev}(\tilde{\mathbf{S}}) + \frac{2}{3}\bar{\rho}k\mathbf{I}; \quad \text{dev}(\tilde{\mathbf{S}}) = \frac{1}{2}\left[\nabla\tilde{\mathbf{u}} + (\nabla\tilde{\mathbf{u}})^T\right] \quad (2.34)$$

The Boussinesq method also uses the following approximation for the sub-grid scale viscosity (Almeida et al., 2015):

$$\mu_{sgs} = \rho C_v \Delta \sqrt{k} \quad (2.35)$$

Where the constant C_v is equal to 0.094 (Almeida et al., 2015).

2.7. Finite Volumes (FV) Discretization

FireFOAM performs a numerical solution for the compressible Navier-Stokes equations described in section 2.4 Gas-Phase Governing Equations, based on the Finite Volumes (FV) method. Equations in the five-PDE system are integrated over each finite volume—or mesh cell equivalently—, where differential calculus relations—like the Divergence Theorem—are applied in order to transform complex volume integrals into simple surface integrals. Fluxes through cell faces are approximated with adjacent node information. Cell-average variables can be approximated to cell-center values, following a first order scheme. As an example, the mass-conservation equation (2.14) is integrated according to the Finite Volumes Method along the cell volume V , through to the procedure:

$$\begin{aligned}
\textbf{Differential mass conservation:} \quad & \frac{\partial \bar{\rho}}{\partial t} + \nabla \cdot (\bar{\rho} \tilde{\mathbf{u}}) = 0 \\
\stackrel{f_V()}{\longrightarrow} \int_V \frac{\partial \bar{\rho}}{\partial t} dV + \int_V \nabla \cdot (\bar{\rho} \tilde{\mathbf{u}}) dV = 0 & \quad (2.36)
\end{aligned}$$

Regarding the first integral, the density can be considered as a cell-average and therefore the integral can be discretely computed. Regarding the second volume integral, the Divergence Theorem can be applied as to transform it into a surface integral which is easier to compute. This results in the following expression:

$$\frac{\partial}{\partial t} \left(\bar{\rho}_a \cdot \Delta V \right) + \oint_S \bar{\rho} (\tilde{\mathbf{u}} \cdot d\vec{S}) = 0 \quad (2.37)$$

Where a corresponds to the cell-averaged value, ΔV to the cell volume, and S to the complete cell outer surface. Considering f as the set of six faces that conform surface S , the second integral can be re-written as:

$$\begin{aligned}
\textbf{Discrete mass conservation:} \quad & \underbrace{\frac{\partial}{\partial t} \left(\bar{\rho}_a \cdot \Delta V \right)}_{\text{Cell mass accumulation}} + \underbrace{\sum_{f=1}^6 \bar{\rho}_f (\tilde{\mathbf{u}}_f \cdot \hat{\mathbf{n}}_f) S_f}_{\text{Mass fluxes through cell faces}} = 0 \quad (2.38)
\end{aligned}$$

Where S_f corresponds to the surface area of face f , and $\hat{\mathbf{n}}_f$ to its normal vector. The cell-average value of density ($\bar{\rho}_a$) can be evaluated as the cell-center value or throughout a n -order quadrature considering n values in the cell. The temporal derivative can be evaluated through Finite Differences. The face-average density ($\bar{\rho}_f$) and velocity vector ($\tilde{\mathbf{u}}_f$) can be interpolated from neighbour nodes through Upwind Interpolation (UDS), Linear Interpolation (CDS), or Quadratic Upwind Interpolation (QUICK), among others. This whole process involves therefore three levels of approximation: Interpolation, Differentiation and Integration. After applying the FV method to all conservation equations in a similar way as

shown for mass conservation, a discrete linear Navier-Stokes system is obtained, which can be solved through computational methods based on linear algebra.

Finally, the discrete Navier Stokes set of equations are solved based on the PIMPLE algorithm, which is a combination of both Pressure-Implicit Split-Operator (PISO) and the Semi-Implicit Method for Pressure-Linked Equations (SIMPLE). These algorithms are iterative procedures for coupling equations for momentum and mass conservation, PISO and PIMPLE being used for transient problems and SIMPLE for steady-state (OpenCFD, 2007). Within time-steps, both algorithms solve a pressure equation, to enforce mass conservation, with an explicit correction to velocity to satisfy momentum conservation. They optionally begin each step by solving the momentum equation –the so-called momentum predictor (OpenCFD, 2007)–, which is activated in FireFOAM by default. For more information on numerical methods implemented in FireFOAM, the reader is referred to the OpenFOAM User Guide (OpenCFD, 2007).

3. CHAPTER 3: METHODOLOGY

3.1. Experimental Setup: ETFT E-I

3.1.1. Compartment Topology and Materials

A description of the experimental setup of ETFT E-I will be developed as a first step of this FireFOAM validation exercise. Figure 3.1 shows the experimental compartment of the ETFT experiments, which is 17.8 m long 4.9 m deep 2.0 m high. The same coordinate system in Figure 3.1 will be used from now on to describe the numerical setup and results.

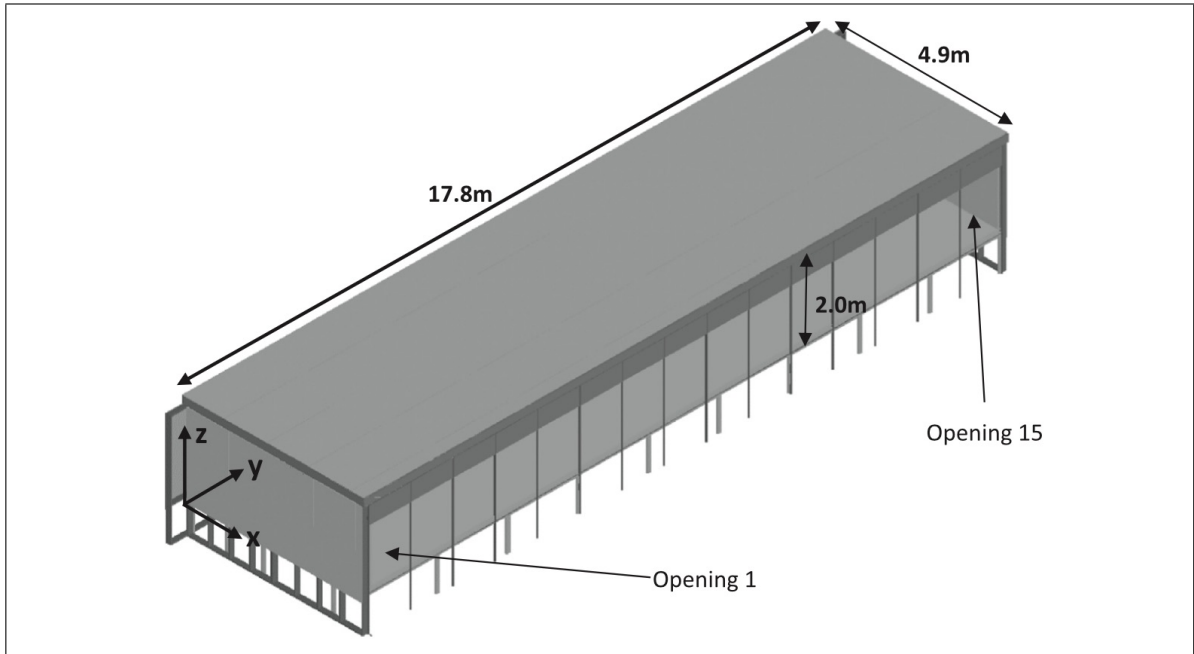


Figure 3.1. The drawing illustrates the characteristic internal dimensions of the compartment as well as the directions and origin of the global coordinate system used to define it and its contents. The global origin was aligned with the back left corner of the platform, when looking into the compartment from the front face, and leveled with the finished compartment floor. Source: J.P. Hidalgo et. al. (2017).

The compartment sat on a 1 m high platform to enable ease of access from underneath. The platform was 18 m long x 5 m wide. The compartment counted with 15 frontal openings,

each 1.1 m wide and 1.5 m high, and separated from each other by 0.1 m thick squared-profile columns which run along the compartment's frontal face (see Figure 3.2). Each opening extended from the compartment floor-level 1.5 m up to the base of a 0.5 m high hangar. Below there is a brief description of the compartment components' materials, dimensions, and thermal properties relevant for Section 2.3. Numerical Setup.

3.1.1.1. Floor

The floor sits on a 1 m high stage, consisting of a steel frame supporting an 18 mm thick birch plywood board. A 1 mm steel sheet was attached to the top surface of the board, and protected with insulation. The floor was insulated with a 140 mm thick ROCKWOOL FLEXI[®], except underneath the gas burners themselves where 25 mm thick ROCKWOOL BEAMCLAD[®] was used. The global heat transfer coefficient U for the floor is $0.23 \text{ W m}^{-2} \text{ K}^{-1}$ (J. Hidalgo et al., 2017). Table 3.1 shows the floor composition with the respective thermal conductivities.

Table 3.1. Floor composition and thermal properties. Order goes from outside to inside the compartment.

Position	Material	Thickness / mm	Thermal Conductivity / $\text{W m}^{-1} \text{ K}^{-1}$
1	Birch plywood board	18	0.13
2	Steel	1	43
3	ROCKWOOL FLEXI [®]	140	0.038

3.1.1.2. Rear and side walls

The inside of these walls consisted in a protective 50 mm layer of concrete aerated blocks. Behind these bricks was a 50 mm thick layer of ROCKWOOL FLEXI[®] held in place by

50 mm wide, 0.5 mm thick steel C-section profiles. These were attached to a 140 mm thick timber frame with vertical studs at 600 mm centers. Finally, a 15 mm thick plasterboard layer was attached to the rear to hold the insulation in place. The global heat transfer coefficient U of wall arrangement is $0.18 \text{ W m}^{-2} \text{ K}^{-1}$ (J. Hidalgo et al., 2017). Five panels in the rear wall contained a range of alternative insulation products as part a sub-experimental project. However, these panels were designed to maintain the same U value and the interior 50 mm thick aerated brick lining. Table 3.2 shows the wall composition with the respective thermal conductivities.

Table 3.2. Wall composition and thermal properties. Order goes from outside to inside the compartment.

Position	Material	Thickness / mm	Thermal Conductivity / $\text{W m}^{-1} \text{ K}^{-1}$
1	Plasterboard	15	0.17
2	ROCKWOOL FLEXI®	50	0.038
3	Aerated concrete blocks	50	0.110 - 0.135 (0.1225)

3.1.1.3. Front (open) face

Columns located at the compartment's front face forming the openings (see Figure 3.2) consisted of 50 mm square sections, made with 3 mm thick mild steel. These sections were protected on all sides by 25 mm thick ROCKWOOL BEAMCLAD®, forming the 100 mm squared-profile columns indicated in Figure 3.2 (Left).

The 0.5 m high hangers at the openings (see Figure 3.2) consisted in two layers of 25 mm thick ROCKWOOL BEAMCLAD® supported by a 50 mm wide steel C-section, bolted on either side to the squared-section columns. A thin layer of ROCKWOOL FLEXI® was

sandwiched between the hanger top and the compartment roof to create a seal, as shown in Figure 3.3.

As some mesh sizes to be used will exceed the columns' 0.1 m width, these will not be included in the computational domain as they are sub-grid structures. Hence, the compartment will be frontally fully opened for simulation purposes and therefore the column materials and properties are not relevant for this work. As for the hangers, their 0.1 m deep dimension will as well be sub-grid for some meshes, therefore this dimension will be modeled as grid-size. Hangers cannot be left out of the modeling as done with the columns, due to the fact that their 0.5 m height is relevant compared to the 1.5 m vertical opening dimension. As no perpendicular heat transfer will be modeled through the hangers —because both sides will be exposed to gas at similar temperature, hence no thermal gradient—, its thermal properties are not relevant for this work.

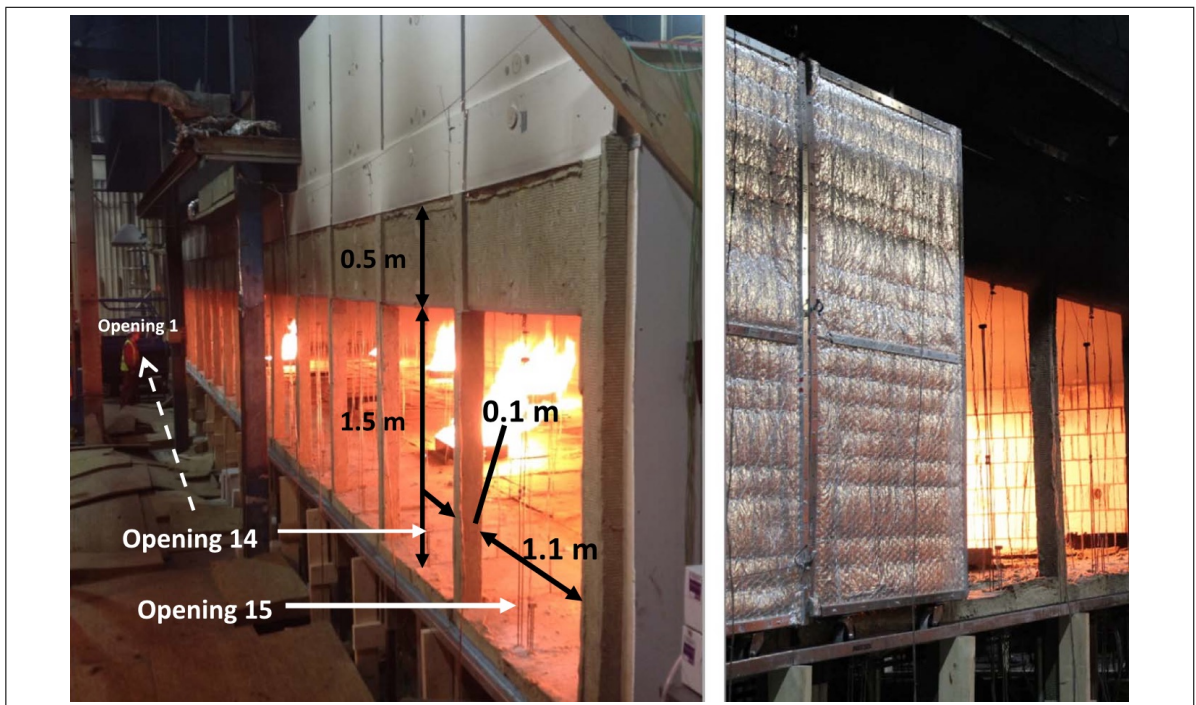


Figure 3.2. Images of the front (open) face of the compartment. Left: Layout and dimensions of the 15 openings along the front (open) face. Right: Roller shutter system, which for ETFT E-I is not used (all shutters opened). Source: J. P. Hidalgo et. al. (2017).

3.1.1.4. Roof

The roof's structure (see Fig. 3.3) consisted in 220 mm deep timber beams going from the rear timber frame to the mild steel hollow section beams at the front face. A 25 mm thick plasterboard was screwed to the top of these beams, and 50 mm deep steel C-section was screwed to the underside of the beams with a further 25 mm layer of plasterboard attached to the underside of these C-sections. The 270 mm cavity enclosed by the plasterboards was filled with a 270 mm deep layer of ROCKWOOL FLEXI®. The final inner layers consisted of a 0.5 mm thick black steel sheet covered by a 25 mm thick layer of ROCKWOOL BEAMCLAD®. The global heat transfer coefficient U is $0.15 \text{ W m}^{-2} \text{ K}^{-1}$ (J. Hidalgo et al., 2017). Table 3.3 resumes the roof composition along its thermal properties.



Figure 3.3. ETFT roof composition. Source: J. P. Hidalgo (2017).

Table 3.3. Roof composition and thermal properties. Order goes from outside to inside of the compartment.

Position	Material	Thickness / mm	Thermal Conductivity / $\text{W m}^{-1} \text{K}^{-1}$
1	Plasterboard	25	0.17
2	ROCKWOOL FLEXI®	270	0.038
3	Plasterboard	25	0.17
4	Steel	0.5	43
5	ROCKWOOL BEAMCLAD®	25	–

3.1.2. Measuring Devices

Table 3.4 shows an overview on the measuring instrumentation used in ETFT. The following sections provide a description of the relevant aspects for this work of the instrumentation characteristics.

Table 3.4. List of sensors used for characterization of fire-related variables during ETFT. Source: J. P. Hidalgo (2017)

Parameter Measured	Sensor type	Location in compartment	Number of sensors
Gas-phase temperature	Thermocouple	Interior gas-phase	1629
		Openings	75
		Exterior near openings	165
Incident radiant heat flux	Thin Skin Calorimeter (TSC)	Interior walls, ceiling, and floor	165
		Exterior faade	94
		Exterior facing openings	15
Gas flow velocity	Bi-directional velocity probe	Openings	30
O ₂ , CO ₂ , CO	Gas analyser	Interior gas-phase	5
Obscuration	Obscuration meters	Interior gas-phase	5
Propane flow	Mass flow controller	Exterior	2
Mass loss	Load cells	Exterior	8
Thermal imaging	IR camera	Exterior	1
Video imaging	Video cameras	Interior and exterior	~ 6

3.1.2.1. Gas-phase temperature

Inconel sheathed type K thermocouples with a 1.5 mm bead were used to create spatial and temporal temperature maps in the compartment interior and openings. Seven rows — in the x-direction, refer to Figure 3.1 coordinate system— of 29 thermocouple trees each, with each tree containing 8 thermocouples, were evenly spaced throughout the compartment interior at 0.7 m intervals in the x-direction and 0.6 m intervals in the y-direction. Each tree

had thermocouples at heights: 0.3, 0.6, 0.9, 1.2, 1.4, 1.6, 1.8 and 1.95 above the floor. Rows were labeled A-G in the x-direction and 1-29 in the y-direction. Figure 3.4 shows describes the location of internal trees and other instrumentation to be described. At the center of the back wall ($y=9050$ mm, $z=985$ mm) in plane $x=10$ mm, five thermocouples were placed at $x=10, 15, 20, 25$ and 30 mm from the back wall to measure gas-phase temperature variation close to this boundary.

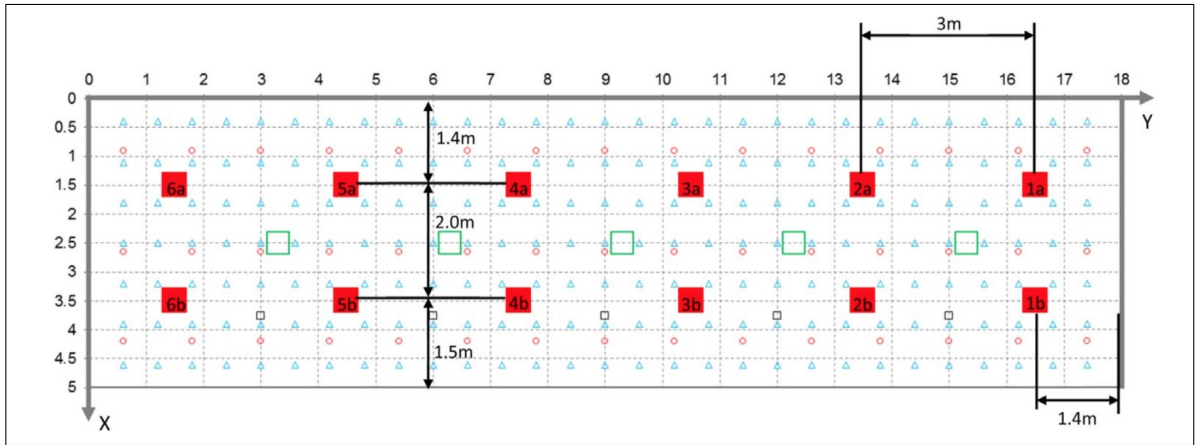


Figure 3.4. Plan layout of the compartment, showing gas burners (large red squares; located at floor level), internal thermocouple trees (blue triangles), floor- and ceiling-level TSCs (red circles), gas-species sampling points just below ceiling level (small black squares), and obscuration gauges (large green squares; also close to ceiling level). All values are in m and reference the global coordinate system. Source: J. P. Hidalgo et. al. (2017).

Apart from the internal trees, there was a tree at the center line of each of the 15 openings with five thermocouples at locations above the compartment floor: 0.18, 0.43, 0.68, 0.93 and 1.18 m. Additionally to this row of trees, there was a final one located 0.4 m out from each opening, in line with the center of the openings as the previous row. Relative to the compartment floor, thermocouples in each of these trees were located at: -0.39, -0.14, 0.11, 0.36, 0.61, 0.86, 1.11, 1.36, 1.61, 1.86 and 2.11 m. The locations of these external thermocouples relative to the façade of the compartment are indicated in Figure 3.5.

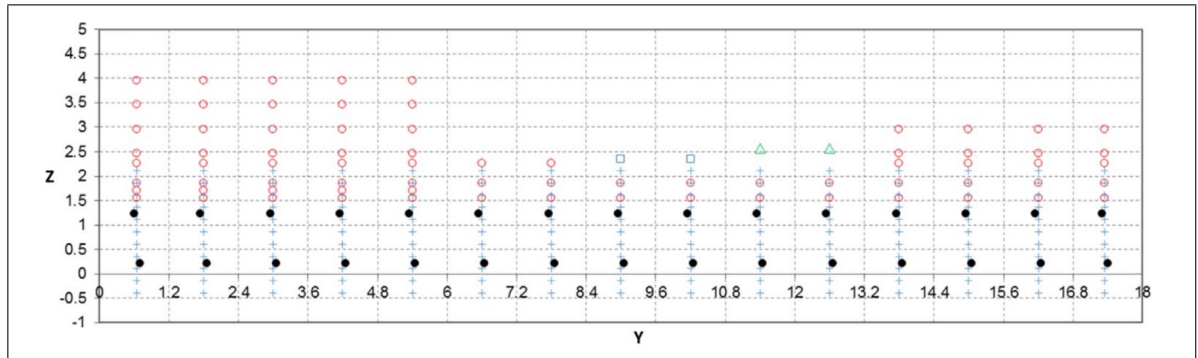


Figure 3.5. Location of sensors in the plane of the facade (global $x=5.05$ plane) and beyond, as indicated. Red circles represent outward facing TSCs above and on the same plane as the compartment openings. Blue squares represent downward facing TSCs 0.175 m out from the plane of the opening. Green triangles represent outward facing TSCs 0.4 m out from the plane of the opening. Blue crosses represent thermocouple locations 0.4 m out from the opening plane. Black circles represent velocity probes 0.05 m towards the compartment interior from the opening plane. Source: J. P. Hidalgo et. al. (2017).

3.1.2.2. Incident radiant heat flux

Thin Skin Calorimeters (TSCs) —custom designed and calibrated by the procedure referenced by J. Hidalgo et al. (2017)— were used for estimating incident radiant heat flux. These were placed in all five internal surfaces of the compartment. 45 TSCs were located in the roof of the compartment, arranged in 3 rows of 15 gauges in a mirror-like configuration with the ones in the floor indicated in Figure 3.4. 45 other gauges were located on the back wall, in a grid of 3 rows of gauges in height with 15 along each row in the length of the compartment, as indicated in Figure 3.6. The right and left walls, corresponding to planes $y=17.9$ m and $y=0.1$ m respectively, had 15 gauges each, with 3 rows of 5 gauges as figure as Figure 3.7 shows. Other 15 gauges were located outside, opposite to the front face of the compartment, at variable distances. Finally, 94 TSCs were arranged in the façade above the openings according to Figure 3.5.

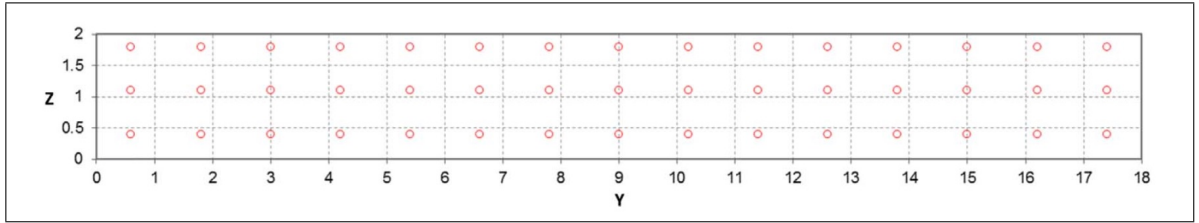


Figure 3.6. Back wall ($x=0.1$ plane) as viewed from the front of the compartment. Red circles represent TSCs. Coordinates shown are in m. Source: J. P. Hidalgo et. al. (2017).

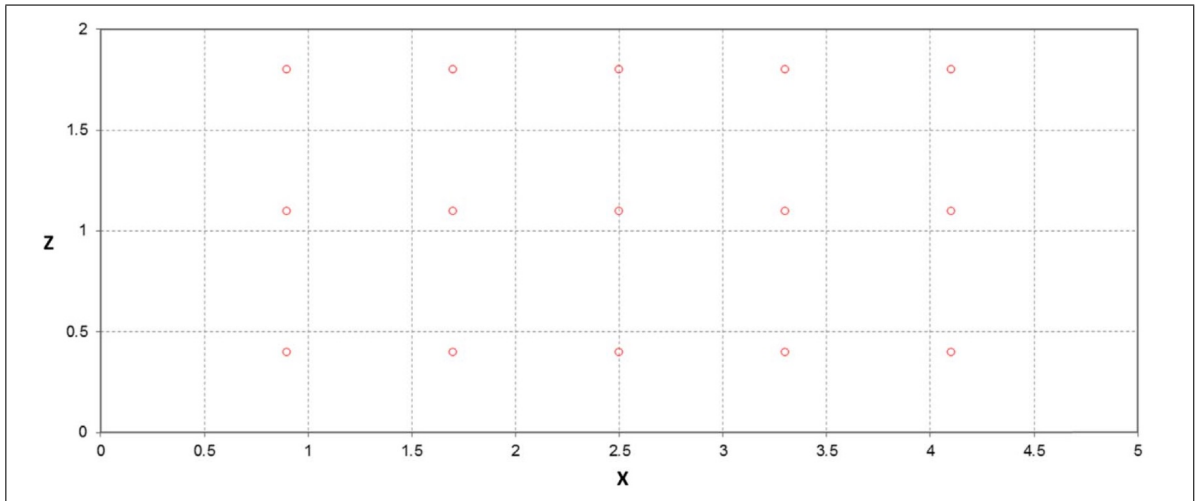


Figure 3.7. Right wall ($y=17.9$ plane) as viewed from the front of the compartment. Red circles represent TSCs. Left side wall ($y=0.1$ plane) is a mirror image of this. All values are in m.. Source: J. P. Hidalgo et. al. (2017).

3.1.2.3. Gas flow velocity

Each of the 15 openings contained two bi-directional velocity probes, one 0.22 m from the base of the opening to capture fresh air inflow velocity, and another 1.23 m from the openings' base to capture combustion gases outflow velocity. To avoid interference in the measurement due to the presence of thermocouples trees, the upper probes were off-set 50 mm to the left of the openings' centerline, and the lower probe 50 mm to the right as viewed

from the front to the inside of the compartment. Figure 3.5 indicates the probes' locations in black circles.

3.1.2.4. O₂, CO₂ and CO concentrations

Five gas sampling points, referenced as GSP A-E, were located along the plane $x=3.75$ m as indicated in Figure 3.4 (small black squares). Sampling points A-E are located along this plane in $y=15, 12, 9, 6,$ and 3 m. Locations A, C and E measured O₂, CO₂ and CO concentrations, while B and D only measured O₂ concentrations.

Each sampling point's instrumentation consisted in a 1 m, 5mm internal diameter stainless steel tube lowered 0.1 m below the roof level, through a roof hole. Gas was suctioned through the tube, passing into a connecting plastic tube attached to the analyzer. Therefore, there was a delay of less than 5 s between the gas aspiration and measurement registering.

3.1.2.5. Obscuration

Soot volume fraction was measured using a diode laser light absorption system at the five locations, as indicated in Figure 3.4 as large green squares. The obscuration gauges' working principle, design, and operation are described by J. Hidalgo et al. (2017). As the laser goes through a length of smoke before reaching the receiver, measured soot volume fraction corresponded to an average over a 0.28 m distance. Located along the compartment centerline (see Figure 3.4), readings from gauges 1, 3 and 5 are located 0.2 m below the ceiling, and from gauges 2 and 4 0.5 m below the ceiling.

3.1.2.6. Thermal video imaging

An infrared thermal imaging camera and a regular handheld video camera were located on tripods at the left of the compartment, as viewed from the outside facing towards the frontal openings. Both filmed along the line of openings (y -direction) from approximately mid-height of the openings. Additionally, five network cameras were used in ETFT E-I, two

located in each side wall ($y=0$ m and $t=17.8$ m), about 0.5 m above the floor facing down the length of the compartment. The other three were located opposite to the openings, looking into the compartment.

3.1.3. Fire Conditions

Twelve gas burners were arranged in pairs inside the compartment, as indicated in Figure 3.4. Propane flow was supplied by two mass flow controllers: a fine one supplying the pilot flame—with a peak delivered flow of 0.64 kg/min—and a coarse one supplying the square-outlet burners—with a peak delivered flow of 3.2 kg/min—. Burners are labeled according to a pair number (1-6 starting from the far right) and position ('a' located at the back closest to the global origin, and 'b' at the front), as shown in Figure 3.4. The coarse mass flow controller fed the burner pairs in series, from the center of the compartment.

Burners consisted in 0.35 m high metal boxes, with a squared open-top fuel outlet with dimensions 0.5 m x 0.5 m in plan. Gas flowed into the box just above the base through the side facing its burner pair. Gas then permeated through a gravel bed, which filled the box up to 25 mm below its top. Figure 3.8 shows the burners.

The main events in ETFT Test I are summarized in Figure 3.9. This experiment corresponded to a fire where the spread rate was approximately instantaneous, which in experimental terms means that all the burners were lit simultaneously as the system allowed and the experiment was deemed to have begun (J. Hidalgo et al., 2017). Note that in Figure 3.9 (Below) 100% combustion efficiency is assumed—considering a heat of combustion for propane of 46.45 MJ/kg (J. Hidalgo et al., 2017)—, which is accurate as the flames are fully ventilated. Due to functionality issues, in ETFT Test I the pilots were replaced by small pots of heptane placed in the corner of each box burner.

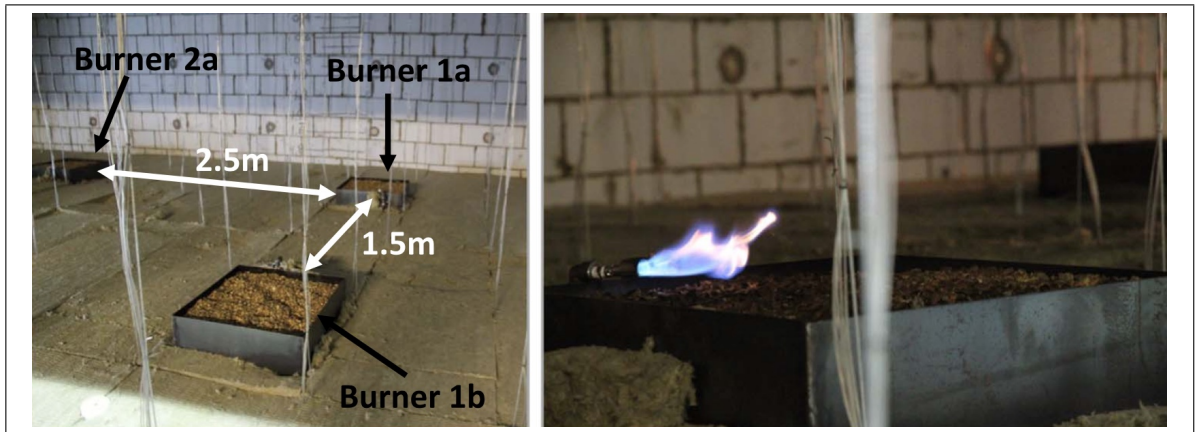


Figure 3.8. Left: Relative positions of three gas burners inside the compartment. Right: Pilot flame above the box burners. Source: J. P. Hidalgo et. al. (2017).

Event	Local Time	Time t Relative to Test Start [s]
Start Lighting Pilots	12:04:00	-790
All Pilots Lit	12:08:20	-530
Open Propane Supply to Burners	12:14:00	-190
Start of HRR Plot (CREF)	12:14:35	-155
Only Burners 2B & 3B Not Lit	12:16:00	-70
All Burners Lit	12:17:10	0
50% of Max HRR	12:17:54	44
Begin HRR Change	12:22:28	318
74% of Max HRR	12:23:20	370
Begin HRR Change	12:27:03	593
99% of Max HRR	12:29:10	720
Begin HRR Change	12:34:43	1053
88% of Max HRR	12:35:50	1120
Begin HRR Change	12:39:20	1330
86% of Max HRR	12:39:40	1350
Close Propane Supply to Burners	12:42:00	1490

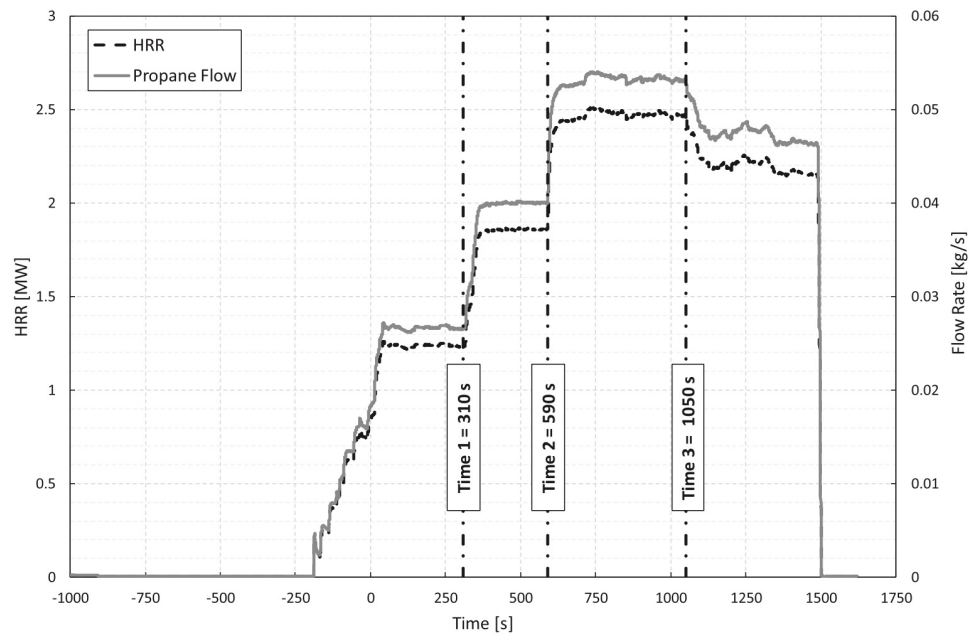


Figure 3.9. Above: Mayor event timeline in ETFT Test I. Bottom: Propane flow rate from the Large Mass Flow Controller (solid line) and corresponding HRR assuming 100% combustion efficiency within the compartment (horizontally dashed line). The vertical dashed lines mark three times at which mass flow is varied. Source: J. P. Hidalgo et. al. (2017).

3.2. Validation Structure for This Exercise

This validation exercise will be focused on testing FireFOAM’s performance in both solution quality and computing time. Solution quality will be tested according to both experimentally-based and theoretically-based techniques, while computing time will be tested by running the solver in different conditions and comparing runtime within these different CPU-demand scenarios. In solution quality testing, experimental thermocouple, velocity probes and Thin Skin Calorimeters (TSCs) data will be plotted against simulation results.

3.2.1. Solution Quality Testing

Scenarios for solution quality testing will include different mesh sizes, different mesh geometries, and different radiation sub-model complexity. A 20 cm cubic mesh with 144 solid angles in FVDOM will be the baseline scenario. Table 3.5 summarizes the solution quality testing scenarios. In each scenario, solution quality will be evaluated by comparing numerical solution and experimental data. Figures 3.10 and 3.11 show in red the key locations to be used for data plotting. Note that sections B and 15 pass along thermocouple trees marked as light blue triangles, and openings 1, 8 and 15 along velocity probes marked as solid black circles. The back wall plane marked in Fig. 3.10 contains a mesh configuration of TSCs (refer to Fig. 3.6) as to validate the radiation model capabilities throughout the construction of contours. Experimental-based validation techniques used in shown key locations are summarized in Table 3.6.

Data is plotted along sections B and 15 to capture the solution quality along both Y-Z and X-Z spans in the compartment, respectively. Tree B2 will be used to validate the solver’s vertical temperature stratification quality near the burners —burner 6a in this case—, while the central tree far away from the burners. As for tree B2, the solver steady state capabilities for heat release and buoyant acceleration modeling will be tested in the flame region, while modeling of turbulent mixing with quiescent air will be tested in the plume region. According to Trouvé and Wang (2010), the successful representation of these phenomena —heat

release and bouyant acceleration for flame region, and turbulent mixing with ambient air for the plume region— are key to achieve reliable results. As for the central tree, modeling of heat transfer due to turbulent mixing will be tested along the compartment height, for both steady and transient states. Three openings will be used to validate the solver’s capabilities to reproduce the neutral-plane height, understanding the neutral plane as the division between the air entrainment layer and the smoke outlet layer. Herein, both temperature and velocity are important as validation variables, as the air entrainment layer is mostly at ambient temperature and with inflow regime, while the smoke layer is mostly at plume temperature and outflow regime. Both temperature and inflow velocity profiles will be calculated as an average of the three characteristic openings shown in Figure 3.11, as to account for length-wise variations. Finally, the FVDOM radiation model capabilities will be tested using steady state incident radiation contours at the compartment back wall. This variable is exceptionally important when simulating thermocouple data, therefore it can also account for correction errors.

Table 3.5. Solution quality testing cases. 'B' stands for baseline case.

Case	Mesh type	Mesh size / cm	Number of solid angles
M20 (B)	Cubic	20	144
M10	Cubic	10	144
M7.5	Cubic	5	144
M30	Cubic	30	144
M40	Cubic	40	144
T20	Tetragonal	20	144
R256	Cubic	20	256
R400	Cubic	20	400
R064	Cubic	20	64
R016	Cubic	20	16

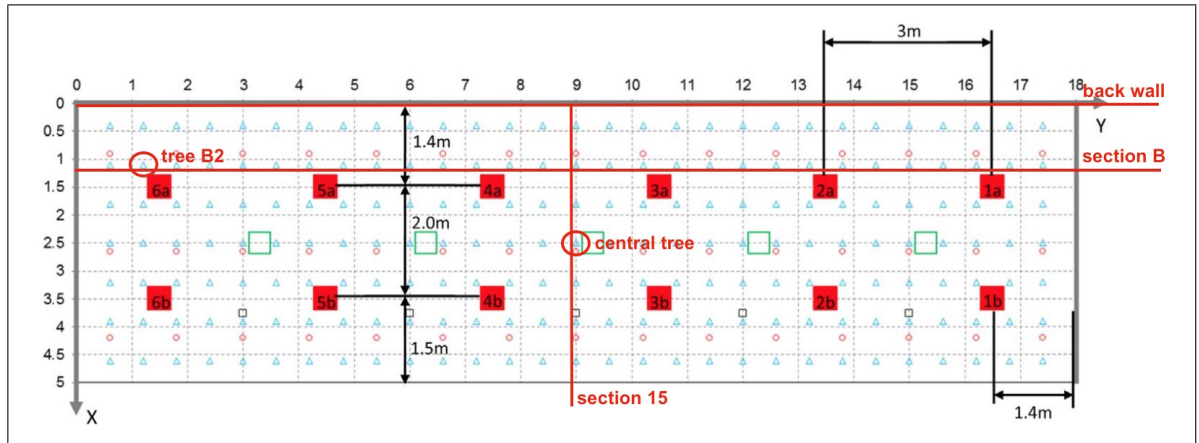


Figure 3.10. Top-view of key locations to be used for data plotting. Thermocouples are shown as hollow light blue triangles, and roof TSCs as hollow red circles. Note that the roof section will be used, but it is not marked for visualization limitations.

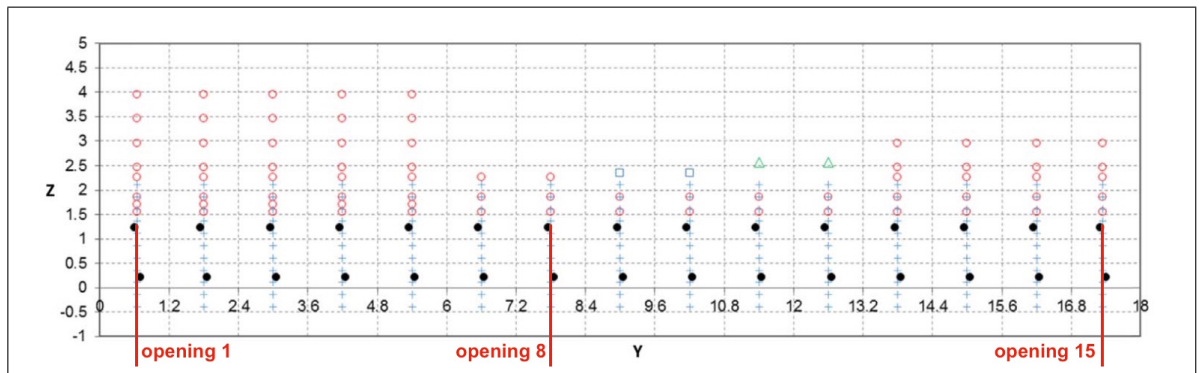


Figure 3.11. Front-view of key locations to be used for data plotting. Velocity probes in each opening are shown as solid black circles, and are located 0.5m towards the compartment interior from the opening plane $x=4.9$ m.

Table 3.6. Validation plots for solution quality testing.

Location	Plotted Variables	Data representation	Temporal representation
Section B	Temperature	Contour	Steady state 74% HRR
Section B	Temperature	Profiles at four heights	Steady state 74% HRR
Section 15	Temperature	Contour	Steady state 74% HRR
Section 15	Temperature	Profiles at four heights	Steady state 74% HRR
Tree B2	Temperature	Profile	Steady state 74% HRR
Central tree	Temperature	Profile	Steady state 74% HRR
Central tree	Temperature	Profile	Transient from propane supply opening until end of steady state 99% HRR
Openings	Temperature	Profile	Steady states 50% and 74% HRR
Openings	Velocity	Profile	Steady states 50% and 74% HRR
Back wall	Incident radiation	Contour	Steady state 74% HRR

3.2.2. Runtime Testing

Scenarios for runtime testing will include different mesh sizes, number of solid angles, and different number of parallel processors performing the simulation. All cases shown in Table 3.5 will be part of the runtime tests. Additionally, the baseline case 'B' will be run with different number of processors, taking advantage of OpenFOAM's parallel processing capabilities. Table 3.7 summarizes the different parallel processing simulation cases.

Runtime is important in fire hazard assessment applications, as simulation results need to predict how the fire damage will evolve before it does. As this application is part of this work's motivation, FireFOAM runtime needs to be tested in different scenarios as to deliver a database for further work in this area. Therefore, the idea of this section is to quantify the

runtime cost at which the numerical solution improves when refining the mesh or increasing the number of solid angles, and the runtime cost when reducing the number of computational resources.

Table 3.7. Runtime testing cases. 'P' stands for augmentation/reduction of parallel processors running the simulation, and the number after the number of processors.

Case	Mesh type	Mesh size / cm	Number of solid angles	Number of processors
B	Cubic	20	144	4
PF	Cubic	20	144	6
PFF	Cubic	20	144	8
PB	Cubic	20	144	2
PBB	Cubic	20	144	1

3.3. Numerical Setup

This section describes the numerical setup used in the simulation of ETFT Test I. This includes the construction of the different mesh structures, the temporal discretization, and the initial and boundary conditions for the internal field and patches. The idea of this section is to keep a registry of how the simulation was constructed, as to account for possible limitations in the results and as well provide feedback for future work in FireFOAM. Annex A explains in detail the process from mesh construction to its incorporation to the FireFOAM case.

3.3.1. Computational Mesh: Spatial Scheme

Both geometry and mesh were constructed in the software Gmsh (Geuzaine & Remacle, 2009). For this study, only the gas-phase domain was constructed and meshed, as the temperature distribution in the solid-phase (i.e. compartment walls) is outside FireFOAM's main solving scope. Because of this, heat transfer through the compartment walls will be modeled through a heat balance in the surface conducted by the temperature boundary condition.

Figures 3.12 and 3.13 show the modeled gas-phase domain and the corresponding patches in Gmsh, respectively. Main compartment dimensions correspond to the ones shown in Figure 3.1, as described by J. Hidalgo et al. (2017). Note that for the cubic mesh construction (Fig. 3.12, right hand side), an additional block was added to the external sub-domain —located below the compartment floor level—, as to capture air entrainment from the compartment bottom. For simplicity, this subdomain was not added in the tetragonal mesh construction (Fig. 3.12, left hand side), given that irregular meshing usually has convergence issues in CFD. Therefore, divergence before final time-step is expected in the T20 case.

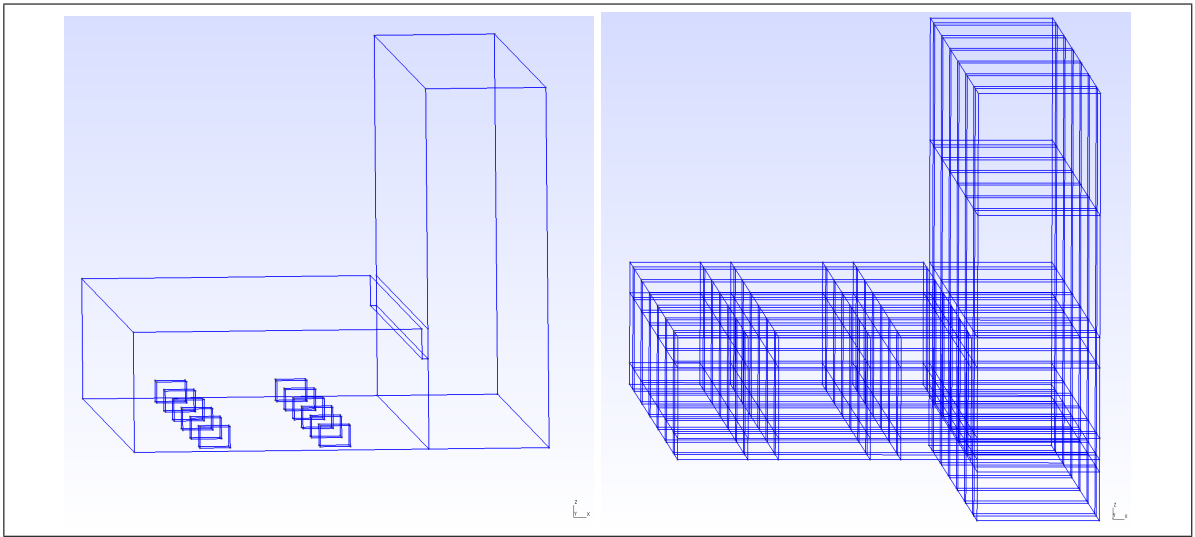


Figure 3.12. Gas-phase domains. Left: Built for tetragonal meshing. The 12 prism-shaped holes inside the domain correspond to the 12 propane burners. The inside of these prisms are later to be unmeshed. Right: Built for cubic meshing. For both constructions, the left structure corresponds to the compartment interior, and the right structure to the secondary entrainment/escape zone.

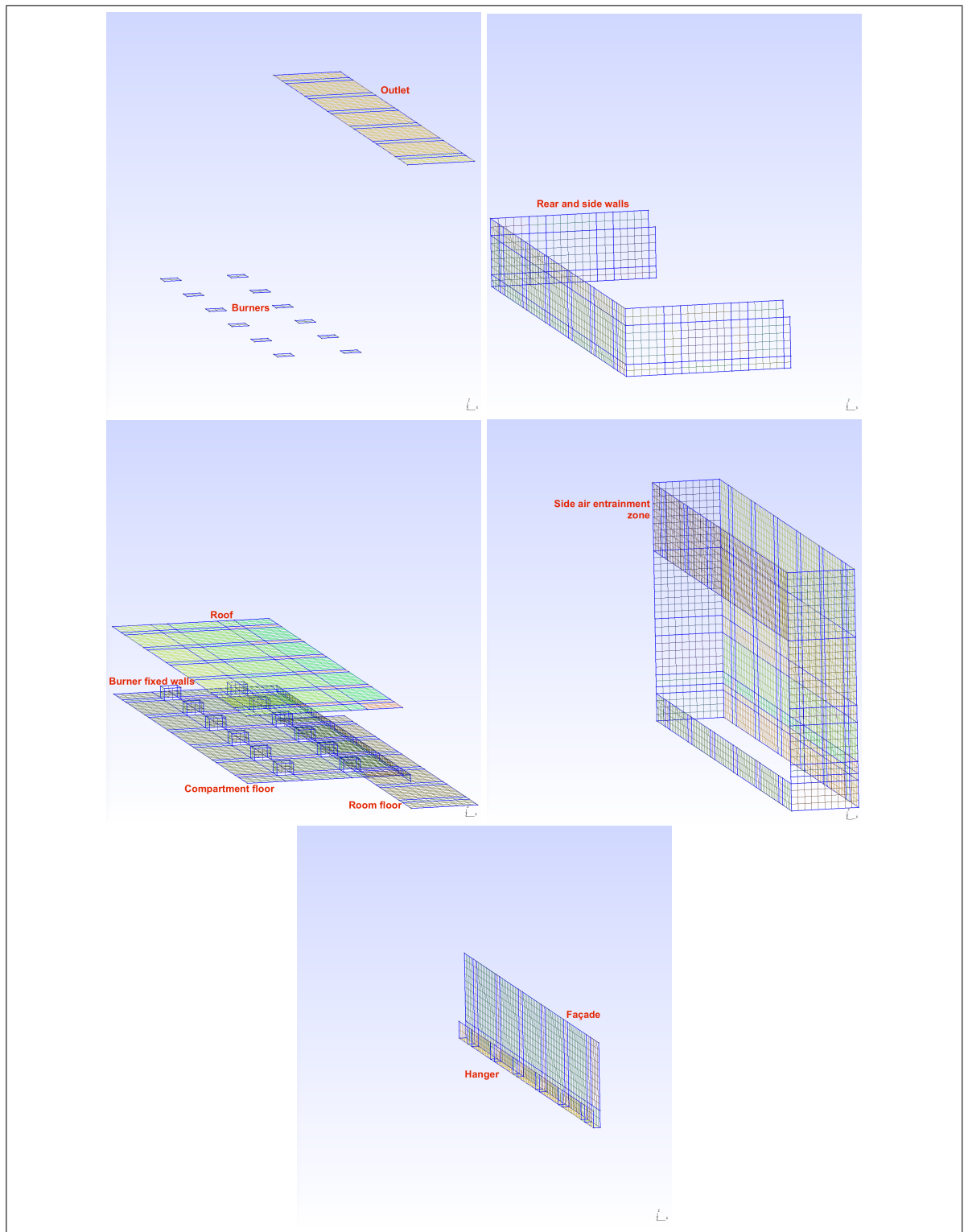


Figure 3.13. Gas-phase domain patches for the 20 cm cubic mesh.

Opening spacings —1.5 m high, 10 cm square section profiles— were not included as they are sub-grid structures for the 20, 30 and 40 cm meshes and are not expected to cause mayor disturbances on the velocity fields near the openings' center lines —where probe measurements are made—. The hanger, shown in Figure 3.13, contains a 10 cm dimension (x-wise, entering the compartment) that can be in principle neglected for being a sub-grid structure, as done with the spacings. Now, neglecting this structure drastically changes the smoke dynamics inside the compartment, as its 50 cm vertical dimension controls smoke accumulation and therefore the smoke layer height. Because of this, the hanger's 10cm-width will be set to be grid-size for the 20, 30 and 40 cm meshes. Finally, note that a secondary zone was constructed adjacent to the compartment openings, as to capture the air entrainment/smoke escape from/to the rest of the room. The dimensions of this secondary zone, selected as to prevent turbulent structures in boundaries, were: 2 m in x-span, 17.8 m in y-span and 4 m in z-span. As mentioned before, the cubic mesh construction extends the secondary entrainment zone additionally 1 m below the compartment floor level. This way, entrainment from below the compartment is captured, and it is not deemed necessary for the tetragonal construction as it is expected to diverge anyways.

As seen in Figure 3.12, the tetragonal meshing domain has a simpler construction than the cubic meshing domain, as the meshing algorithm in the former creates 3D stacked irregular tetrahedrons that adapt their geometry independent from the domain dimensions compared to the desired mesh dimension. For cubic meshing, the domain needs to be divided in 3D prism sub-structures, each one constructed by extrusion from four-sided surfaces. Figure 3.14 shows how the domain's main extruded surface (contained in plane X-Z and extruded in the Y-axis) was divided in sub-surfaces.

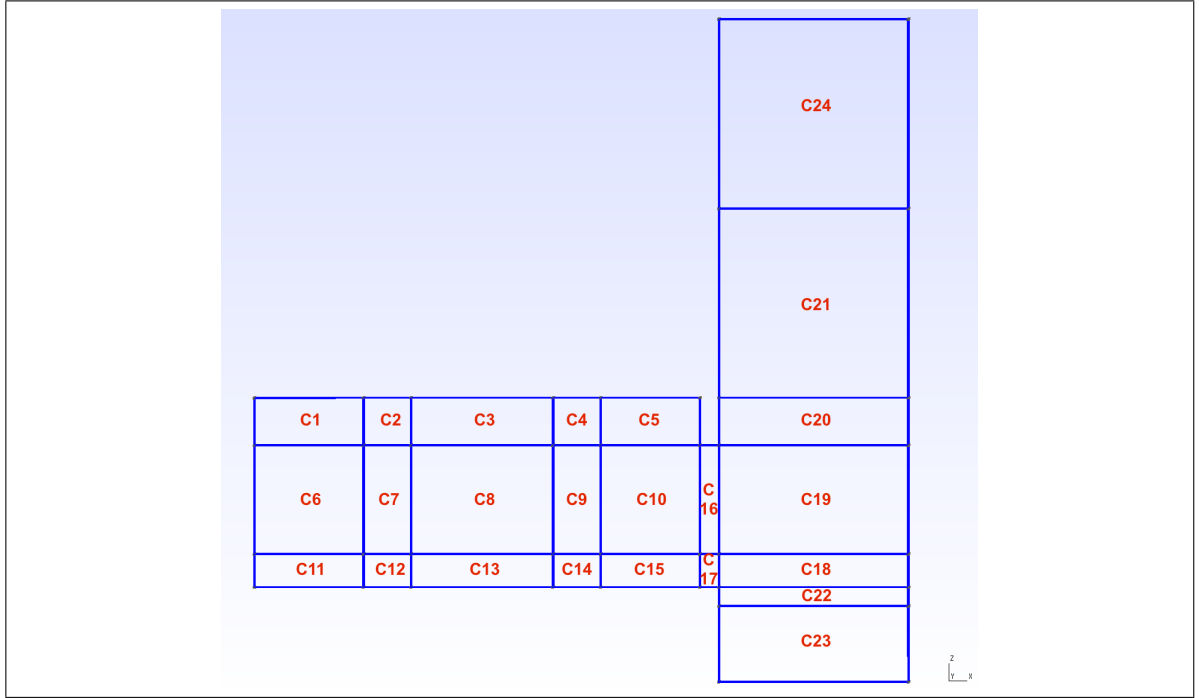


Figure 3.14. X-Z plane containing the sub-surfaces (C1, C2, ..., C24) that form the master surface. Each sub-surface is extruded in the Y-axis to form the final 3D domain.

The meshing algorithm begins meshing each sub-surface—from C1 to C24—with a 2D pattern, and then each 2D pattern is extruded in layers each one with grid-size width. Not all lines that form the sub-surfaces have a length equal to a whole multiple of the chosen grid size, therefore by construction not all elements will be 20 cm cubes. To simplify this, each line was at first meshed with a 1D algorithm which calculates the number of nodes within the line, called Transfinite Line in Gmsh. For a line with length L , the number of nodes n within it was programmed to be the truncated value of:

$$n = \frac{L}{\text{gridsize}} + 1 + \text{deltaNode} \quad (3.1)$$

In equation (3.1), 'deltaNode' was a parameter set to 0.25, as to prevent bigger than expected cell dimensions. For example in the baseline 20 cm mesh, applying equation (3.1) to the burners' 0.35m vertical lines —i.e. vertical lines of faces C12 and C14— the 1D algorithm meshes them with 3 nodes, which corresponds to two 17.5 cm elements. If 'deltaNode' would have been zero, the 1D algorithm would have calculated 2 nodes (the $n=2.75$ would have been truncated to 2) and therefore one 35 cm element, which is coarse for a 20 cm expected mesh. Note that for a length equal to a multiple of the grid-size, the number of nodes corresponds to exactly that multiple plus one as 'deltaNode' is truncated —as expected. Finally, a 2D Transfinite Surface algorithm with Recombine algorithm joins the nodes forming regular 2D prisms. For the extrusion, the number of extruded layers is set to be simply the extrusion length divided in the grid size, truncating any decimal result in the division. Therefore, the mesh is expected to contain prism elements that in average represent a cubic grid size mesh. Figure 3.15 shows an X-Y front view of the baseline 3D mesh.

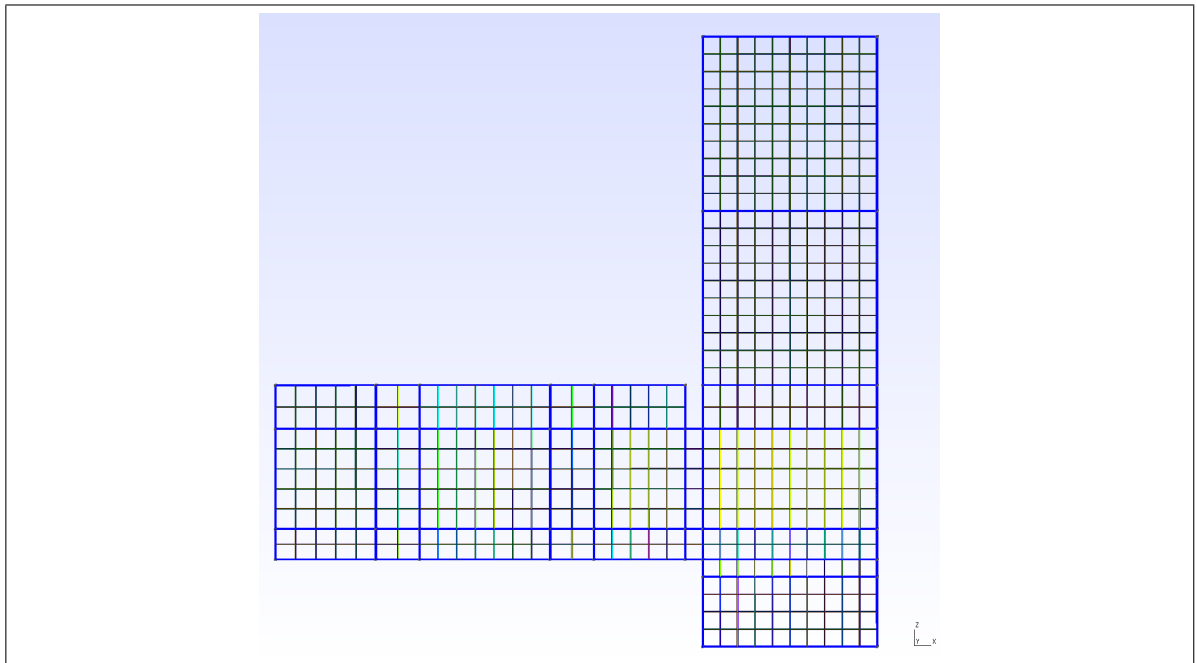


Figure 3.15. Front view of the gas-phase domain meshed with the baseline case M20. Note that the hanger is grid-size in the X-axis and therefore 1-cell width.

Figure 3.16 shows a top view of the meshed baseline case. Here the consistency of the 1D meshing algorithm can be verified: as the burners are in plan 0.5m x 0.5m each line should contain three nodes after truncation — $0.5/0.2+1.25=3.75$ for the X-span line, and $0.5/0.2+1=3.5$ for the Y-span line— and therefore two 25 cm elements. It is then concluded that in the baseline '20cm cubic mesh' case the burner is formed by four $0.25\text{m} \times 0.25\text{m} \times 0.175\text{m}$ prisms. Even though this limits the desired mesh size, it allows to conserve the original experimental dimensions obtaining in average the defined mesh size.

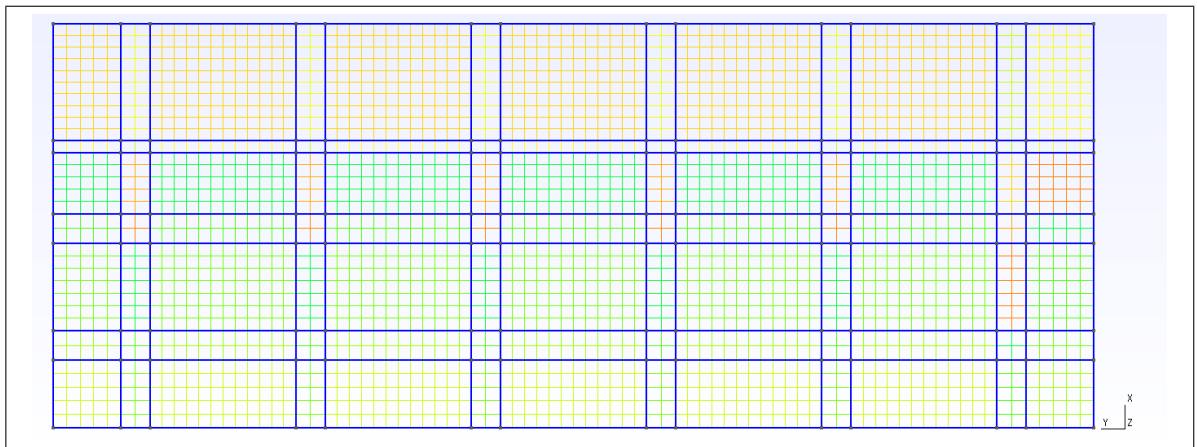


Figure 3.16. Top view of the meshed baseline M20 case.

For reference, Figures 3.17 and 3.18 show the front and top-views (respectively) of the meshed domain with all five cases: M7.5, M10, M20, M30 and M40. Table 3.8 summarizes the fundamental mesh information in each case.

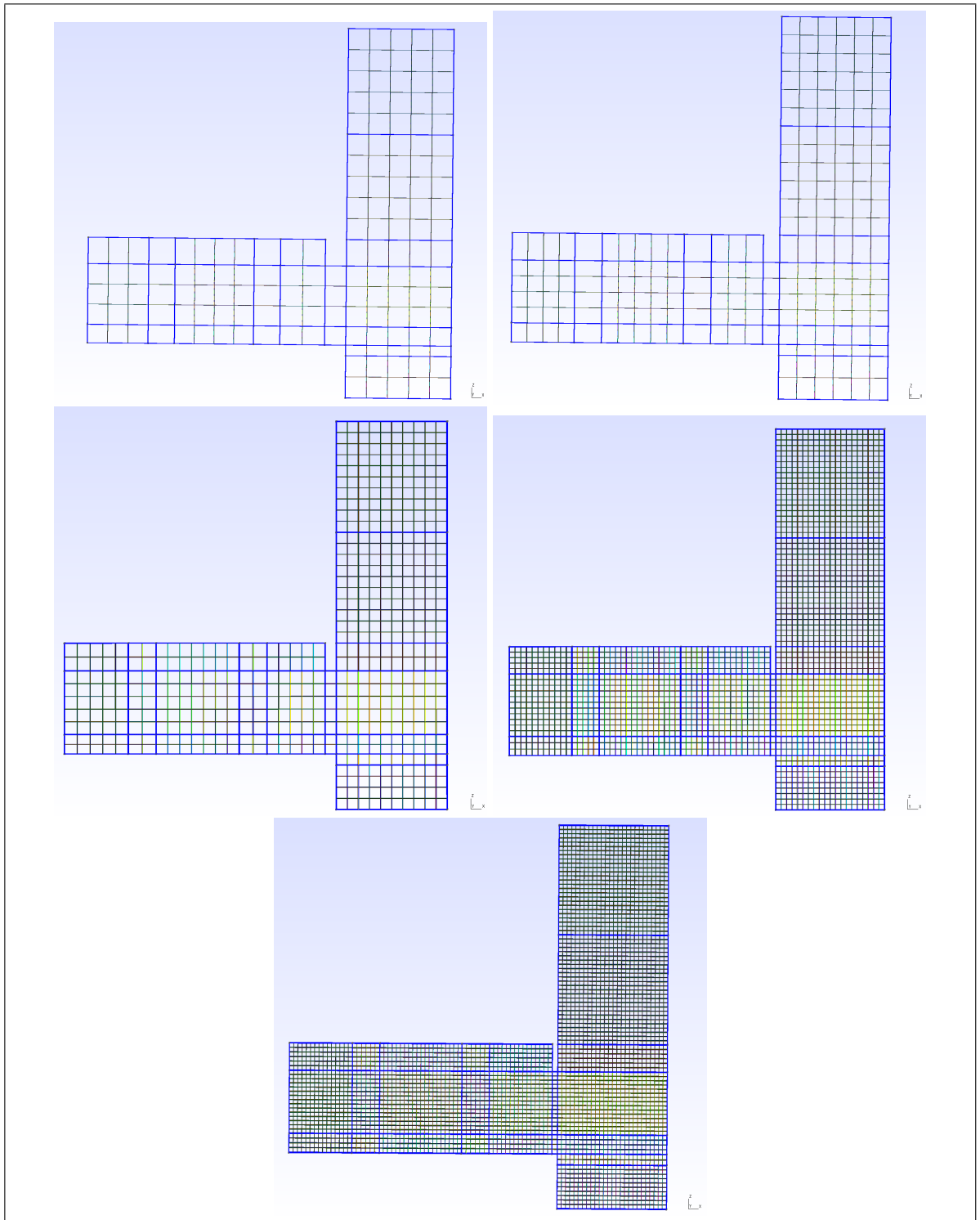


Figure 3.17. Front view of the different cubic meshes. From coarsest to finest: M40, M30, M20, M10 and M7.5.

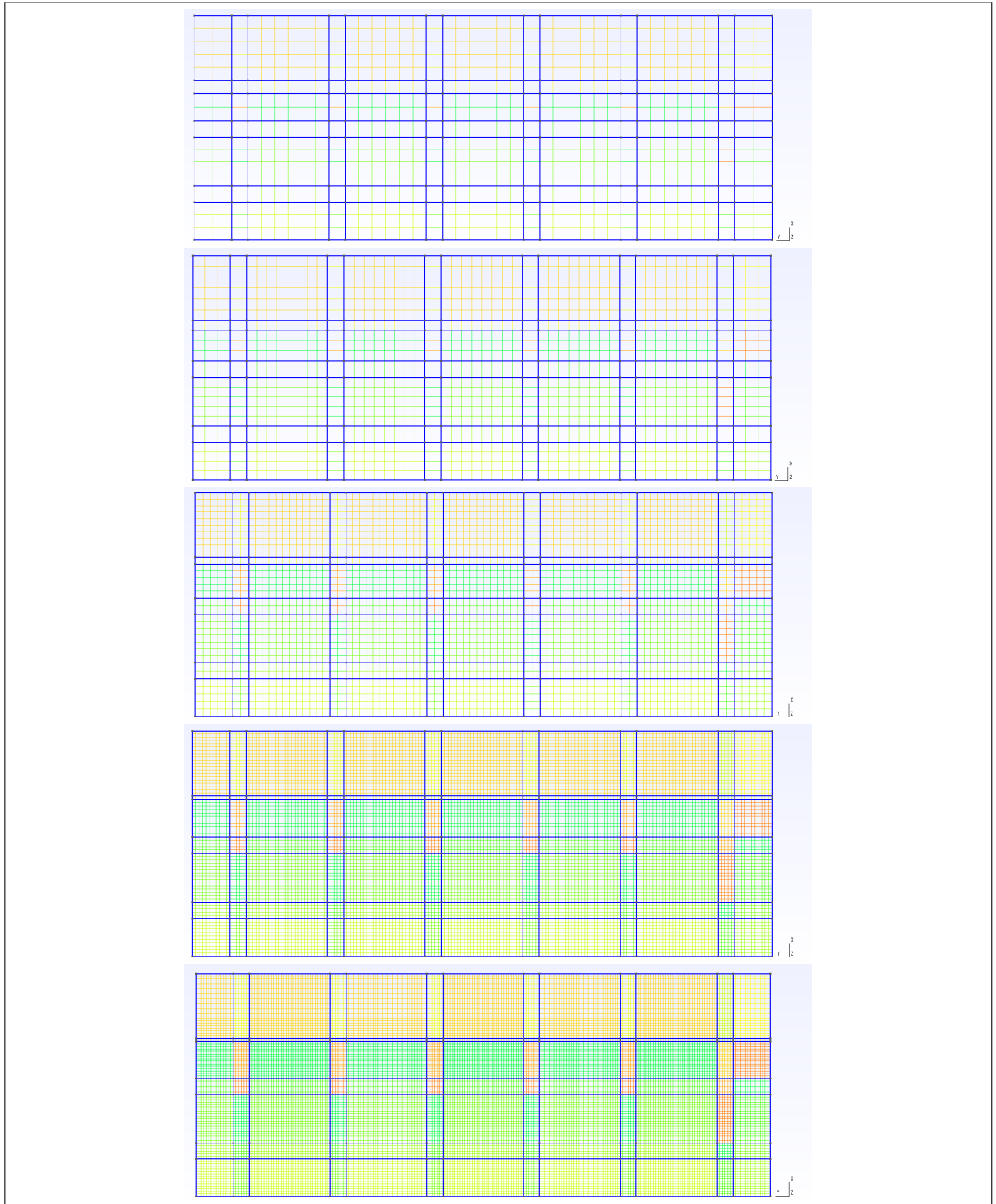


Figure 3.18. Top view of the different cubic meshes. From coarsest to finest: M40, M30, M20, M10 and M7.5.

Table 3.8. Elements and vertices reported post-meshing by Gmsh. All numbers approximated to the third significant figure.

Case	Number of vertices	Number of elements
M40	7 380	18 100
M30	13 700	29 000
M20 (B)	50 200	82 900
M10	432 000	558 000
M7.5	943 000	1 150 000
T20	85 700	501 000

3.3.2. Temporal Scheme

Using Figure 3.9 as reference, the baseline scenario was run for 1250 s starting from the event 'Open Propane Supply to Burners' (12:14:00 local time) until the event 'Begin HRR Change' (12:34:43) after the 99% HRR steady state period. In this way, all the steady state periods of 50% HRR, 74% HRR and 99% HRR are captured as well as the in between transients. An initial time-step (Δt) of one second was used, but as the velocity field changes throughout the domain, FireFOAM adjusts Δt as to maintain the Courant number inferior to unity thus guaranteeing numerical stability. This is done through the function 'adjust-TimeStep' in the 'ControlDict' file, shown in Figure 3.19. Note from the same file that results are written out every 25 s.

```

/*----- C++ -----*/
|=====|
| \ \ \ \ | F i e l d | OpenFOAM: The Open Source CFD Toolbox
| \ \ \ \ | O p e r a t i o n | Version: 4.1
| \ \ \ \ | A n d | Web: www.OpenFOAM.org
| \ \ \ \ | M a n i p u l a t i o n |
/*-----*/

FoamFile
{
    version      2.0;
    format       ascii;
    class        dictionary;
    object       controlDict;
}
// *****

application      fireFoam;

startFrom         latestTime;

startTime         0.0;

stopAt            endTime;

endTime           1250.0;

deltaT            1;

writeControl      adjustableRunTime;

writeInterval     25.0;

purgeWrite        0;

writeFormat       binary;

writePrecision    6;

writeCompression  uncompressed;

timeFormat        general;

timePrecision     6;

graphFormat       raw;

runTimeModifiable yes;

adjustTimeStep    yes;

maxCo             1;

maxDeltaT         5.0;

// *****

```

Figure 3.19. 'ControlDict' file for each FireFOAM case.

3.3.3. Boundary and Initial Conditions

Boundary and initial conditions are summarized in Tables 3.9, 3.10 and 3.11.

Table 3.9. First part: Boundary Conditions in OpenFOAM.

	Outlet	Side ent	Burner	Burner fixed walls	Hanger	Floor/ roof/ walls	Facade/ room floor
α_t	zeroGrad	'...'	'...'	'...'	'...'	'...'	'...'
C3H8	inletOutlet inletVal=0	'...'	totalFlow RateAdv Diff=1	zero Grad	'...'	'...'	'...'
G	Marshak Rad $\varepsilon = 1$	'...'	Marshak Rad $\varepsilon = 0.9$	'...'	'...'	'...'	Marshak Rad $\varepsilon = 1$
IDef	greyDiff Rad $\varepsilon = 1$	'...'	greyDiff Rad $\varepsilon = 0.9$	'...'	'...'	'...'	greyDiff Rad $\varepsilon = 1$
k	inletOutlet inletVal= 10^{-4}	'...'	fixedValue value= 10^{-4}	'...'	'...'	'...'	'...'
N2	inletOutlet inletVal=0.767	'...'	totalFlow RateAdv Diff=0	zeroGrad	'...'	'...'	'...'
ν_t	zeroGrad	'...'	'...'	'...'	'...'	'...'	'...'
O2	inletOutlet inletVal=0.233	'...'	totalFlow RateAdv Diff=0	zeroGrad	'...'	'...'	'...'
p	calculated	'...'	'...'	'...'	'...'	'...'	'...'
p- rgh	prghTotal HydroPress=0	'...'	fixedFlux Pressure	'...'	'...'	'...'	'...'

Table 3.10. Second Part: Boundary Conditions in OpenFOAM.

	Outlet	Side ent	Burner	Burner fixed walls	Hanger	Floor/ roof/ walls	Facade/ room floor
ph_ rgh	fixedValue value=0	fixedFlux Pressure	'...'	'...'	'...'	'...'	'...'
soot	calculated	'...'	'...'	'...'	'...'	'...'	'...'
T	inletOutlet inletVal=283.65	'...'	fixedValue value=283.65	zeroGrad	'...'	HT	zeroGrad
U	PressureInlet OutletVelocity	'...'	flowRate InletVelocity	noSlip	'...'	'...'	'...'
YDef	inletOutlet value=0	'...'	totalFlow RateAdv Diff=0	zeroGrad	'...'	'...'	'...'

Default solver variables for a pool fire scenario were here used. Logically, the corresponding mesh, temporal scheme, and initial/boundary conditions are set to the ETFT Test 1 conditions. Some particular temperature, pressure, velocity, k , incident radiation and radiation intensity ray boundary conditions correspond to the ones recommended by Almeida et al. (2015).

In species inlet at the burner (C_3H_8 , O_2 , N_2 , CO_2 , H_2O), the boundary condition 'totalFlowRateAdvectionDiffusive' was used with mass fraction equal one for propane and zero for the rest. Note that the 'YDefault' field accounts for combustion products: CO_2 and H_2O . Combined with the velocity boundary condition 'flowRateInletVelocity', which defines mass flow rate as a function of time (obtained from Figure 3.9), both assure that the prescribed mass flow rate accounts for both advection and diffusion fluxes. Prescribing a fuel mass flow rate which does not account for diffusion will cause overestimation of the heat release

rate particularly important in turbulent buoyant diffusion flames, as the fuel gradients near the inlet are steep, inducing an additional flow (Y. Wang et al., 2011). This methodology is used both conventional diffuser burner (Almeida et al., 2015) and porous burner (Ren et al., 2016) configurations, the latter corresponding to ETFT Test 1. As the propane flow rate was directly measured in Test 1, the replication of this boundary condition —and hence the total HRR— is expected to be quite accurate. Figure 3.20 shows both experimental and boundary condition prescribed mass flow rates. For all species a 'inletOutlet' boundary condition was established at the 'outlet' and 'side entrainment' patches, which imposes 'zeroGradient' for outflow and 'inletValue' for inflow. As seen in Table 3.9, in the case in inflow through the 'outlet' or 'side entrainment' surfaces, a volume mixture of 0.767 for N₂ and 0.233 for O₂ was prescribed as air is entrained into the domain.

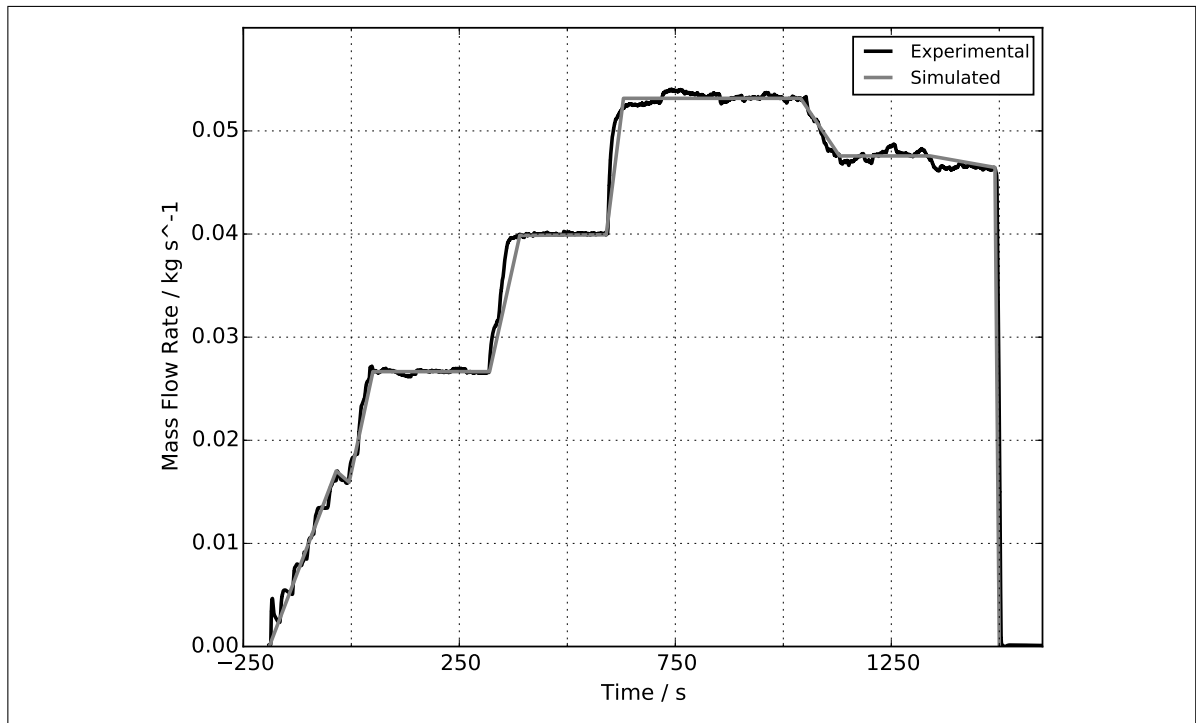


Figure 3.20. Total prescribed mass flow rate (advective plus diffusive) at burner inlet patches, built as a piece-wise function of linear functions.

Considering the implementation of the radiation FVDOM model, the corresponding 'MarshakRadiation' and 'greyDiffusiveRadiation' boundary conditions were established for incident radiation (G) and radiation ray intensity (IDefault). The outlet and side entrainment zones were considered as black bodies (i.e. $\varepsilon=1$), typical assumption for construction materials. All the constructive materials at the compartment interior were considered to have an emissivity equal to 0.9, following the recommendation proposed by Maluk et al. (2017) for the ETFT series.

Air entrainment through the 'outlet' and 'side entrainment' surfaces can be modeled through the combination of the boundary conditions 'pressureInletOutletVelocity' and 'totalPressure' for velocity and the pressure variable 'p_rgh', respectively. Here, the term 'p_rgh' is defined as:

$$p_rgh = p - p_{\text{ref}} + \rho \left| \vec{g} \cdot \vec{h} \right| \quad (3.2)$$

Where \vec{g} and \vec{h} stand for the gravity and cell-center vectors, respectively. By using p_{ref} , the specified pressure boundary conditions in both p_rgh and ph_rgh fields are relative to the reference of 101 325 Pa to avoid rounding errors. This combination of 'U' and 'p_rgh' conditions establishes 'zeroGradient' for velocity at all times, except on the tangential component which is set to zero for inflow. The pressure variable p_rgh is finally calculated from the velocity field through the 'totalPressure' condition as (OpenCFD, 2007):

$$p_rgh = \begin{cases} (p_rgh)_0 & \text{for outflow} \\ (p_rgh)_0 - \frac{1}{2}\rho \left| \mathbf{U} \right| & \text{for inflow} \end{cases} \quad (3.3)$$

This 'totalPressure' boundary condition was used by FireFOAM in OpenFOAM-3.x and below. OpenFOAM-4.x has the newly implemented —and here used— 'prghTotalHydrostaticPressure' as a substitute to the 'totalPressure' boundary condition for p_rgh. According to H. G. Weller (CFD Direct Ltd), this 'p_rgh' boundary condition uses the hydrostatic pressure field ('ph_rgh') as the reference state for the far-field, which provides much more accurate entrainment in large open domains, typical of many fire scenarios. Regarding the wall patches, OpenCFD (2007) recommends using the boundary condition 'fixedFluxPressure' for pressure. This guideline was followed for the 'p_rgh' and 'ph_rgh' fields. Finally, as stated in Table 3.9, the 'p' field is calculated from the solved 'p_rgh' field through equation (3.2).

Soot is calculated by the radiation model as a constant soot fraction in a one-step chemical reaction. Using the reaction rate computed by the chemistry solver (EDC model), the radiation model calculates the soot distribution along the entire internal field and patches, hence the 'calculated' boundary condition. This calculation is done according to the one-step reaction:



Where ν_{soot} is taken equal to 0.055 kmol as it is explained in section 2.4. Physical Models Implemented in FireFOAM.

Finally for the temperature boundary conditions, fuel has a prescribed inlet temperature equal to ambient conditions —283.65 K (10.5 °C)—, considering the fuel supply is at thermal equilibrium with the atmosphere, and neglecting inlet fuel heating due to flame radiation—. Heat dissipation through walls was modeled through the 'externalWallHeatFluxTemperature' boundary condition, which in this case works as a one-dimensional steady-state heat transfer model that considers internal convection, internal wall conduction and external convection with ambient air at 283.65 K (10.5 °C). The radiation field 'Qr'

was not considered in the energy balance, as it caused numerical instability dependent on the solver's 'relaxation' factor. The former was solved considering an equivalent internal convection coefficient h'_w that includes radiation, which was calculated using the original average convection coefficient from the compartment interior walls, $h_w = 9.5 \text{ W/m}^2/\text{K}$ (Maluk et al., 2017), and experimental measurements of temperature and incident radiation (J. Hidalgo et al., 2017). This methodology begins performing a heat balance in the compartment interior walls, which states that the heat entering the wall equals the sum on the incident radiation and the convective gain, minus the emitted radiant flux:

$$\dot{q}'_{net,in} = \varepsilon G + h_w(T_g - T_w) - \sigma \varepsilon T_w^4 \quad (3.4)$$

Where $\dot{q}'_{net,in}$ is the heat flux per unit area entering the wall, ε the surface emissivity, G the wall per unit area incident radiant flux, h_w the average wall convective coefficient, T_g the adjacent gas temperature, T_w the surface wall temperature, and σ the Stefan-Boltzmann constant ($5.67 \cdot 10^{-8} \text{ W m}^{-2} \text{ K}^{-4}$). Considering an equivalent convective coefficient which accounts for radiation (h'_w), eq. (3.4) can be re written as:

$$\begin{aligned} \varepsilon G + h_w(T_g - T_w) - \sigma \varepsilon T_w^4 &= h'_w(T_g - T_w) \\ \rightarrow h'_w &= h_w + \frac{\varepsilon}{T_g - T_w} \left(G - \sigma T_w^4 \right) \end{aligned} \quad (3.5)$$

Therefore h'_w can be calculated from equation (3.5) knowing average $h_w = 9.5 \text{ W m}^{-2} \text{ K}^{-1}$. Considering the 74% HRR (1800 kW) steady state period as an intermediate case between the 50% HRR (1220 kW) and 99% HRR (2410 kW), h'_w will be estimated as the average value of Test 1. In 74% HRR, the average steady state incident radiation (G) on the back wall and gas phase temperature (T_g) on section D (x=2.5 plane) are 3000 W m^{-2} and $250 \text{ }^\circ\text{C}$ respectively (J. Hidalgo et al., 2017). Therefore, only an estimation of the steady

state wall temperature (T_w) is needed to calculate h'_w from equation (3.5). This estimation can be done using equation (3.4). According to an energy balance study made in the ETFT series (Maluk et al., 2017), for Test I the heat loss through walls is approximately 30% of the total HRR. Therefore, the average heat loss through the compartment walls is equal to $0.3 \cdot 1800 = 540$ kW. Considering the compartment interior walls, roof and floor, the loss per unit area is $q'_{net,in} = 2351.51 \text{ W m}^{-2}$. Substituting these values on equation (3.4):

$$2351.52 = 0.9 \cdot 3000 + 9.5 \cdot ((250 + 273.15) - T_w) - \sigma \cdot 0.9 \cdot T_w^4$$

$$\rightarrow T_w = 409 \text{ K} = 135 \text{ }^\circ\text{C}$$

Substituting the estimations T_w , G and T_g for the 74% HRR period on equation (3.5), the Test I average equivalent convection coefficient is $21 \text{ W m}^{-2} \text{ K}^{-1}$. This value is approximately 2.2 times higher than the original convection coefficient, indicating that radiation accounts for 55% of the total wall heat loss (being convection the other 45%). These numbers are in good agreement with vertical wall fire heat distribution studies by Ren et al. (2015), where it was found and validated with literature and correlations that in the inert wall region convection and radiation accounted for approximately 40% and 60%, respectively. The slightly higher radiation proportion can be justified in the fact that in vertical wall fires, the flame is attached to the wall and therefore the entire wall surfaces faces the flame.

Thermal conduction coefficients and material composition for the floor, internal walls, and roof were obtained from Tables 3.1, 3.2 and 3.3 respectively. Figure 3.21 corresponds to the 'T' boundary condition script. As the script shows, conductivities and material thicknesses are included in the respective patches, and the equivalent internal heat transfer coefficient ($h'_w = 21 \text{ W m}^{-2} \text{ K}^{-1}$) is included in the calculation of the equivalent overall convective resistance ($1/h$). This equivalency considers both internal ($h'_w = 21 \text{ W m}^{-2} \text{ K}^{-1}$) and external ($h_{ext} = 20 \text{ W m}^{-2} \text{ K}^{-1}$) convective resistances:

$$\frac{1}{h} = \frac{1}{h'_w} + \frac{1}{h_{ext}} \longrightarrow h = 10.24 \text{ W m}^{-2}\text{K}^{-1}$$

Where the external convection coefficient was taken from the Chilean HVAC Standard NCh 853/2007 for both vertical and horizontal surfaces exposed to ambient conditions.

```

/*----- C++ -----*/
// =====
//      F i e l d
//      O p e r a t i o n
//      A n d
//      M a n i p u l a t i o n
// =====
//      OpenFOAM: The Open Source CFD Toolbox
//      Version: 4.1
//      Web: www.OpenFOAM.org
// =====
FoamFile
{
    version      2.0;
    format       ascii;
    class        volScalarField;
    location     "0";
    object       T;
}
// =====

dimensions      [0 0 0 1 0 0 0];
internalField   uniform 283.65;
boundaryField
{
    "(outlet|sideEnt)"
    {
        type      inletOutlet;
        inletValue $internalField;
        value      $internalField;
    }

    rearAndSideWalls
    {
        type      externalWallHeatFluxTemperature;
        Ta        uniform 283.65;
        h          uniform 10.24;
        thicknessLayers (0.015 0.050 0.050);
        kappalayers (0.17 0.038 0.1225);
        value      uniform 283.65;
        Qr         none;
        kappaMethod fluidThermo;
        kappa      fluidThermo;
    }

    floor
    {
        type      externalWallHeatFluxTemperature;
        Ta        uniform 283.65;
        h          uniform 10.24;
        thicknessLayers (0.018 0.001 0.140);
        kappalayers (0.13 43 0.038);
        value      uniform 283.65;
        Qr         none;
        kappaMethod fluidThermo;
        kappa      fluidThermo;
    }

    roof
    {
        type      externalWallHeatFluxTemperature;
        Ta        uniform 283.65;
        h          uniform 10.24;
        thicknessLayers (0.025 0.270 0.025 0.0005 0.025);
        kappalayers (0.17 0.038 0.17 43 0.038);
        value      uniform 283.65;
        Qr         none;
        kappaMethod fluidThermo;
        kappa      fluidThermo;
    }

    "(hanger|burnerFixedWalls|facade|roomFloor)"
    {
        type      zeroGradient;
    }

    burner
    {
        type      fixedValue;
        value      uniform 283.65;
    }
}
// =====

```

Figure 3.21. Temperature boundary conditions in each domain patch.

Finally, Table 3.11 summarizes the initial values of the internal fields, i.e. initial conditions. For air composition and pressure, a standard 101.3 kPa static atmosphere composed of 23 %vol/vol oxygen and 77 %vol/vol nitrogen was established. The initial internal field temperature was considered to be 283.65 K (10.5 °C), obtained as the average thermocouple measurements before Test 1 begun.

Table 3.11. Initial domain conditions set in OpenFOAM.

	internalField
α_t	uniform 0 kg/m/s
C3H8	uniform 0
G	uniform 0 W/m ²
IDefault	uniform 0 W/m ²
k	uniform 10 ⁻⁴ m ² /s ²
N2	uniform 0.767
ν_t	uniform 0 m ² /s
O2	uniform 0.233
p	uniform 101325 Pa
p_rgh	uniform 0
ph_rgh	uniform 0
soot	uniform 0
T	uniform 283.65 K
U	uniform (0 0 0) m/s
YDefault	uniform 0

With the Numerical Setup well defined, the only aspects left to begin simulating are the selection of Physical Models Implemented in FireFOAM.

3.4. Physical Models Implemented in FireFOAM

Based on the FireFOAM sample cases, literature cases—which concluded “good-agreement” with experimental data and moderate use of computational resources (Almeida et al., 2015)—, and technical reasons explained below, the following models were chosen for the ETFT Test I simulations:

- **Gas-phase turbulence:** Large Eddy Simulation (LES).
- **Combustion Chemistry:** Eddy Dissipation Concept (EDC), with $C_{EDC} = 4$ (Magnussen & Hjertager, 1977).
- **Radiation Heat Transfer:** Finite Volume Discrete Ordinates Model (FVDOM). Variable number of solid angles, ranging from 16 to 400 (baseline case: 144).
- **Solid-phase pyrolysis:** No pyrolysis.
- **Soot formation/oxidation:** ‘mixtureFractionSoot’ model with a soot fraction equal to 0.21% vol/vol, only participating in the radiation model.

LES-based turbulence modeling was chosen over RANS, given the fact that a transient solution is needed as it captures the transient large-scale motion and intermittency of buoyant fires (Almeida et al., 2015). Also, as the simulation aims to capture the solver’s transient capabilities during variable propane mass flux conditions, a solver that admits variable boundary conditions—and therefore solutions—is indispensable. FireFOAM solves the LES-filtered Navier Stokes set of equations based on the PIMPLE algorithm, which is a combination of both Pressure-Implicit Split-Operator (PISO) and the Semi-Implicit Method for Pressure-Linked Equations (SIMPLE). These algorithms are iterative procedures for coupling equations for momentum and mass conservation, PISO and PIMPLE being used for transient problems and SIMPLE for steady-state (OpenCFD, 2007). Within time-steps, both algorithms solve a pressure equation, to enforce mass conservation, with an explicit correction to velocity to satisfy momentum conservation. They optionally begin each step by solving the momentum equation—the so-called momentum predictor (OpenCFD, 2007)—, which is activated in FireFOAM by default. OpenCFD (2007) states that the amount of times

the solver solves the pressure equation and momentum corrector in each step is typically set to 'nCorrectors=2' or 'nCorrectors=3' (this is done in the 'fvSolution' directory inside the 'system' folder in OpenFOAM). In this exercise, 'nCorrectors' is set to 4, given the fact that velocity discontinuities were detected in some time steps when using 'nCorrectors=2'. The amount of loops over the entire system of equations in each time step can be chosen in the entry 'nOuterCorrectors', located in the same file as 'nCorrectors'. This parameter was set to one, as this value is deemed appropriate for pool fire configurations according to the FireFOAM tutorial cases.

The EDC model for combustion rate is chosen over the n-th order Arrhenius model due to several reasons. Apart from the fact that EDC is computationally less demanding than n-th order Arrhenius modeling, the chemical reaction in buoyant diffusion flames is controlled by the rate at which oxidant and fuel molecules encounter, therefore it behaves similar as an infinitely-fast reaction as described by EDC. As no solid-phase fuel is present in ETFT Test I, no pyrolysis chemical reaction model is needed.

Regarding radiation, the Finite Volume Discrete Ordinate Method (FVDOM from now on) has been chosen over P1, because even though fvDOM is more CPU-intensive as it is more comprehensive —i.e. it accounts for scattering, semi-transparent medium, specular surfaces, and wavelength-dependent transmission—, P1 is most useful for optically thin situations that are uncommon in fire applications (Runefors, Anderson, Wahlqvist, Huang, & Husted, 2016).

Soot formation is only considered in radiation modeling as a constant fraction simplified model, given the fact that propane is a weak-medium sooting fuel, therefore soot radiation ideally must not be neglected. The soot fraction value was chosen based on the methane-fueled 'smallPoolFire2D' FireFOAM case, which considers that each kmol of reacting CH₄ forms 0.055 kmol of soot (noted as 'nuSoot'=0.055 in the 'radiationProperties' script located at the 'constant' directory), modeled as C. As no reference values for 'nuSoot' were found for propane diffusion flames regarding the 'mixtureFractionSoot' model (declared in the 'constant/radiationProperties' script), the same soot quantity per burned fuel-unit was

chosen ($\nu_{\text{Soot}}=0.055$). As propane is expected to be more soot intensive than methane, radiation losses are under-estimated by this choice, which is either way considered better than neglecting soot radiation. As Test I involves a fully ventilated compartment, soot is not expected to be a mayor part of radiation losses —as minor soot formation is expected—, therefore errors produced by this choice are expected to be negligible. Complete soot formation/oxidation modeling is only justifiable in scenarios with highly sooting fuels, or with large pool fire diameters with crosswinds (e.g. large-scale LNG sea pool fires) where limited air entrainment at the center of the pool and wind distributions promote uneven soot distribution (C. Wang et al., 2014).

Given the fact that a coarse mesh will be used near the walls, the simulation will not be "wall-resolved". Simulations are "wall-resolved" when these are performed with sufficient grid resolution to capture the wall gradients and do not require a wall heat transfer model (Ren et al., 2017). Therefore, as explained in previous sections, a wall heat transfer model will be used in these simulations integrated as the OpenFOAM boundary condition 'externalWallHeatFluxTemperature', which performs a steady-state 1D heat balance in the wall surface. This will have an influence in the heat flux through the wall, specially in transient fire growth periods.

3.5. Data Management

The process of generating results required both experimental (raw) and simulation data management. Raw data was obtained from the Data Loggers generated during ETFT Test I, while simulation data was generated using the ParaView post-processing tool compatible with OpenFOAM. This section, created for the readers to understand the theoretical basis of calculations and the associated assumptions, describes the different techniques used on both raw and simulated data to generate final results. Both raw and simulated data were contained in '.csv' files, which were treated using Python and its associated mathematical and scientific packages.

3.5.1. Time-averaged Data

Simulation results such as simulated thermocouple temperatures, velocity fields and incident radiation will be presented in some cases as time-averaged data. As the mesh gets finer, local time fluctuations increase drastically, therefore time averaged results must be used. These fields will be averaged over the steady-state HRR periods, i.e. when propane flow is kept constant. Using the data from Figure 3.9, Table 3.12 summarizes the main events relevant to time-averaging.

Table 3.12. Main events regarding HRR changes. Times are reported relative to the time where all the burners were lit.

Event	Time relative to test start / s
Open propane supply to burners (FireFOAM simulation begins)	-190
Test start with all burners lit	0
50% of maximum HRR	44
Begin of HRR change	318
74% of maximum HRR	370
Begin of HRR change	593
99% of maximum HRR	720
Begin HRR change	1053

As established in the 'controlDict' script (inside the 'system' directory), time-steps in the baseline case are written each 25 s for a total of 1250 s of simulation, therefore a total of 51 steps are written. Table 3.13 shows which and how many time steps are used for averaging during each steady state HRR period.

Table 3.13. Used time-steps for time averaging in each HRR steady state period. Times are reported relative to the simulation beginning.

Steady state period	Simulation beginning time	Considered beginning time/ time-step	Simulation ending time	Considered ending time/ time-step	Number of averaged time-steps
50% HRR	190+44=234s	300 s / 13	190+318=508s	500 s / 21	9
74% HRR	190+370=560s	600 s / 25	190+593=783s	750 s / 31	7
99% HRR	190+720=910s	950 s / 39	190+1053=1243s	1250 s / 51	13

Note that there is a characteristic heating time for ETFT Test I, quantified as 2 min (120 s) by Maluk et al. (2017), defined as the time taken for the gas-phase to reach the steady state after a steady state HRR is established. For example, once reached the 50% HRR period—44s after the test start—the gas phase needs 120 s additional seconds to reach steady state temperatures. As seen in Table 3.13, the considered beginning times for time-averaging purposes are less than 120 s after the steady HRR is reached, therefore time averaging begins where temperatures are still transient. Now, as the chosen averaging intervals are the same for simulated and experimental data, the transient effect on the averages is included for both data sets. Furthermore, as the simulated data is corrected to thermocouple data including the transient term (refer to section 2.5.2 Simulated Thermocouple Temperatures), thermocouple heating inertia is also considered, hence both simulated and experimental data averages are submitted to the same transient effects.

3.5.2. Simulated Thermocouple Temperatures

To analyze the solver's temperature simulation capability, simulated temperatures will be compared to thermocouple measurements. In turbulent diffusion flames, where thermal radiation is not negligible, the thermocouple temperature (raw measurements) will present discrepancies with respect to the gas-phase temperatures (solver output). It is expected that

at high elevations —where the compartment accumulates hot gases and receives considerable amounts of flame radiation— the radiation levels will be high and therefore the thermocouples will read above gas-phase temperature, with expected differences of 10-100 °C (J. Hidalgo et al., 2017). This previous range is referred as "radiation error" by J. Hidalgo et al. (2017). For validation purposes the correction can be done to either side: calculate the thermocouple temperature from the solver's gas-phase temperature —in order to compare it with the raw measurement—, or calculate the gas-phase temperature from the raw thermocouple measurement —in order to compare it to the gas-phase temperature calculated by the solver. For this exercise the former was chosen, given the fact that incident radiation —needed in the thermocouple correction model— is calculated by the solver at a higher resolution compared to the experimental measurements. From now on, the corrected simulated gas-phase temperature will be referred as "simulated thermocouple temperature". In this subsection the correction process and model will be described.

As anticipated in Section 1.2.2.1. Experimental-based validation, the relationship between the gas-phase temperature (T_g) and the thermocouple temperature (T_{tc}) can be obtained performing an energy balance in the thermocouple bead (Ren et al., 2016):

$$\rho_{tc}c_{tc}\left(\frac{V_{tc}}{A_{tc}}\right)\frac{dT_{tc}}{dt} = \varepsilon_{tc}(G - \sigma T_{tc}^4) + h_{tc}(T_g - T_{tc})$$

Where ρ_{tc} , c_{tc} , V_{tc} , A_{tc} and ε_{tc} are thermocouple properties: mass density, specific heat capacity, volume, surface area and emissivity respectively. Considering thermocouples as 1.5 mm Nickel spheres (J. Hidalgo et al., 2017), a density and specific heat capacity of 9000 kg/m³ and 440 J/kg/K (Welch, Jowsey, Deeny, Morgan, & Torero, 2007) were respectively considered. The volume and surface area can be calculated from the radius considering a spherical geometry. The thermocouple surface emissivity will be taken as 0.9 (Welch et al., 2007). This equation allows to calculate the simulated thermocouple temperature (T_{tc}) from

the known simulated gas-phase temperature (T_g), incident radiation (G) and correlation-calculated convective heat transfer coefficient (h_{tc}). Reorganizing the previous equation:

$$\rho_{tc}c_{tc}\left(\frac{V_{tc}}{A_{tc}}\right)\frac{dT_{tc}}{dt} + T_{tc}^4(\sigma\varepsilon_{tc}) + T_{tc}(h_{tc}(T_{tc})) - (\varepsilon G + h_{tc}(T_{tc})) = 0 \quad (3.6)$$

The time-derivative term will be calculated using finite differences. The time discretization will correspond to the simulation write interval (Δt), which in all cases is 25 s. This discretization is appropriate as for ETFT Test I the characteristic time of the gas-phase heating—time until steady state temperatures are reached after a HRR change—is approximately 120 s (Maluk et al., 2017). This way, the time derivatives will be calculated as follows:

$$\begin{aligned} \text{Time step } n = 0: \quad \frac{dT_{tc}}{dt} &= \frac{T_{tc,0} - T_{tc,-1}}{\Delta t} \xrightarrow{T_{tc,0}=T_{tc,-1}} \frac{dT_{tc}}{dt} = 0 \\ \text{Time step } n = i: \quad \frac{dT_{tc}}{dt} &= \frac{T_{tc,i} - T_{tc,i-1}}{\Delta t} \end{aligned}$$

Substituting the above discretization in eq. (3.6), the simulated thermocouple temperature for time step i will be calculated as:

$$\begin{aligned} \rho_{tc}c_{tc}\left(\frac{V_{tc}}{A_{tc}}\right)\frac{T_{tc,i} - T_{tc,i-1}}{\Delta t} + T_{tc,i}^4(\sigma\varepsilon_{tc}) + T_{tc,i}(h_{tc}(T_{tc,i})) \\ - (\varepsilon_{tc}G + h_{tc}(T_{tc,i})) = 0 \end{aligned} \quad (3.7)$$

Therefore when calculating the thermocouple temperature in time step i , the previous value ($i - 1$) will be needed. As the derivative term in the first time step is zero, the thermocouple temperature at the beginning ($T_{tc,0}$) can be directly calculated from the known values, and based on this value the thermocouple temperatures in further time steps are recursively calculated.

The convective heat transfer coefficient (h_{tc}) can be calculated from the Nusselt number with characteristic dimension D (thermocouple diameter) as:

$$h_{tc} = \frac{k \cdot \text{Nu}_D}{D} \quad (3.8)$$

Where k is the fluid thermal conductivity, estimated with the 'thermo.mixture' package from Python considering the fluid to be air at the temperature and pressure given by the simulation. The air assumption is valid in the context of compartment post-flashover fires, as the fuel mass flow is negligible compared to the air entrainment in the plume region (Torero, Majdalani, Abecassis-Empis, & Cowlard, 2014). In the simulations, the fuel mass flow rate was less than 1% of the air entrainment, therefore the assumption is valid in this particular case. The Nusselt number is calculated through correlations of natural convection—using the Raleigh and Prandtl numbers— or forced external convection—using the Reynolds and Prandtl numbers— for a sphere depending on the modulus of the local velocity, also given by the solver:

$$\text{Forced convection (Mills): } \text{Nu}_D = 2 + \left(0.4 \text{Re}_D^{1/2} + 0.06 \text{Re}_D^{2/3} \right) \text{Pr}^{0.4} \quad (3.9)$$

$$0.7 \leq \text{Pr} \leq 380 \quad 3.5 \leq \text{Re}_D \leq 8 \cdot 10^4$$

$$\text{Natural convection (Mills): } \text{Nu}_D = 2 + \frac{0.589 \text{Ra}_D^{1/4}}{\left[1 + (0.469/\text{Pr})^{9/16} \right]^{4/9}} \quad (3.10)$$

$$\text{Ra}_D < 10^{11} \quad \text{Pr} > 0.5$$

Where each number was calculated as:

$$\text{Ra}_D = \frac{g D^3 |T_p - T_g|}{T_p (\mu/\rho) \alpha} \quad \text{Re}_D = \frac{|\vec{U}| D}{\nu} \quad \text{Pr} = 0.7$$

The Prandtl number is generally taken as a constant value of 0.7, following studies of non-buoyant flows (Welch et al., 2007). T_p corresponds to the mean film temperature—or limit layer mean temperature—, calculated as the average between the thermocouple surface temperature and adjacent gas-phase temperature:

$$T_p = \frac{T_{tc} + T_g}{2}$$

All fluid properties are estimated at the mean film temperature (T_p). The fluid's dynamic viscosity (μ) can be calculated from Sutherland's Law (Welch et al., 2007), and the kinematic viscosity (ν) and thermal diffusivity (α) with their definition following the μ calculation:

$$\mu = \frac{C_1 T^{1.5}}{T + C_2} \rightarrow \nu = \frac{\mu}{\rho} \rightarrow \alpha = \frac{\nu}{\text{Pr}}$$

Where the constants C_1 and C_2 are respectively defined as $1.458 \cdot 10^{-6} \text{ kg/m/s/K}^{0.5}$ and 110.4 K.

The iterative process used for the thermocouple correction at time-step i is as follows:

- (i) Begin with an initial guess of the thermocouple surface temperature $T_{tc,I}$, taken as the local gas phase temperature T_g —calculated by the solver at that same location—.
- (ii) Calculate the convective heat transfer coefficient h_{tc} using the initial guess. This involves the calculation of the mean film temperature T_p , the adimensional numbers, and the Nusselt number by applicable correlations. If $\text{Re}_D > 3.5$ correlation (3.9) applies, if not natural convection hence correlation (3.10) must be used.
- (iii) With the calculated h_{tc} and the previous time-step thermocouple surface temperature, calculate the actual thermocouple surface temperature through eq. (3.7), obtaining a second guess noted as $T_{tc,II}$.

- (iv) Calculate the change in the thermocouple temperature guess as: $|T_{tc,II} - T_{tc,I}|$. If it exceeds 1°C , then return to step (i) considering as an initial guess $T_{tc,I} = T_{tc,II}$ and calculate the new guess $T_{tc,II}$. When the error is less than 1°C , use $T_{tc,II}$ as the simulated thermocouple correction.

A 1°C sensitivity is considered sufficient as the thermocouple measurement error is $\pm 2.5^\circ\text{C}$ (J. Hidalgo et al., 2017). Throughout this procedure the thermocouple temperatures were simulated through the time-steps using the gas phase temperature fields the solver delivered. As stated at the beginning, this procedure was programmed as a Python routine.

3.5.3. Temperature Contours

Temperature contours consist of thermocouple temperature maps along the compartment's sections B and 15 (refer to Fig. 3.10). Experimental thermocouple temperatures were directly obtained as the Data Logger entries, while for the simulated gas-phase temperatures the radiation correction procedure explained in the previous section was used.

Once both raw and simulated thermocouple temperatures were obtained, contours were built from a regular temperature interpolated virtual mesh, using Python's Matplotlib 'griddata' function. For section B for example, consisting of a rectangular slice 2 m in the z-direction and 17.8 m in the y-direction, a regular 2m x 17.8m mesh with 200 nodes in the z-direction and 1780 nodes in the y-direction was used—hence with a 1 cm spacing between nodes—. Interpolation from the registered temperature points—thermocouple locations for raw data, and cell locations for the simulated data—to the virtual mesh nodes was done through a 'linear' interpolation algorithm implemented in the 'griddata' function. Finally, the meshes were set up on the Matplotlib 'contourf' function, plotting the temperature fields across the section using 25 levels of resolution on the color bar. The same 1 cm resolution for interpolation was used for section 15.

3.5.4. Streamline Maps

Streamline maps —elaborated with the purpose of verifying the velocity field mesh convergence— were also plotted along sections B and 15, and only for simulated data as no probes were placed inside the compartment. Regular interpolated velocity grids were generated the same way —and in the same resolution— as explained on the previous section for temperature contours. Interpolated velocity components corresponded to the ones parallel to the section plane, e.g. for section B (plane $x=1.1$) interpolated meshes were generated for y and z velocity components. For streamline generation, the function 'streamplot' from the 'Pyplot' utility was used. Stream lines were colored according to the plane-parallel velocity modulus.

3.5.5. Incident Radiation Contours

Simulated incident radiation contours are directly plotted from the solver output volume field G [kW/m²], corresponding to the sum of all radiation contribution from all the rays, and its calculated as the multiplication of the different ray intensities by its solid angle. The G field contour is constructed the same way as described in section 2.5.2. Temperature Contours.

Raw data from Thin Skin Calorimeters (TSCs) is reported as the calorimeter's disc temperature, therefore data processing as indicated by J. P. Hidalgo et al. (2017) is needed. As the procedure indicates, the incident radiant flux on the TSC is calculated from an energy balance in its disc (the absorbed radiant heat flux is equal to the stored internal energy plus the radiation and convection emissions):

$$\dot{q}_{inc}''(T_{disc}) = \frac{1}{\alpha_{TSC}(1 - C)} \left[\dot{q}_{stor,TSC}'' + E_{TSC} + \dot{q}_{conv}'' \right] \quad (3.11)$$

Where the disc's absorptivity (α_{TSC}) is 1 and the correction factor C is calculated as a linear function of the temperature (J. P. Hidalgo et al., 2017):

$$C = -0.000018 \cdot T_{disc} + 0.637731 \quad (3.12)$$

The heat rate stored by the disc ($\dot{q}''_{stor,TSC}$), the emitted heat flux by radiation (E_{TSC}) and by convection (\dot{q}''_{conv}) are calculated as (J. Hidalgo et al., 2017):

$$\dot{q}''_{stor,TSC} = \frac{m_{disc}}{A_{disc}} \cdot c_p(T_{disc}) \cdot \frac{dT_{disc}}{dt} \quad (3.13)$$

$$E_{TSC} = \varepsilon_{TSC} \sigma T_{disc}^4 \quad (3.14)$$

$$\dot{q}''_{conv} = h_c(T_{disc} - T_{\infty}) \quad (3.15)$$

For the calculation of the heat rate stored by the disc ($\dot{q}''_{stor,TSC}$), TSC mass per unit area (m_{disc}/A_{disc}) is equal to 9.53272 kg/m² and specific heat capacity ($c_p(T_{disc})$) was calculated according to a 3rd-degree polynomial as indicated by J. P. Hidalgo et al. (2017).

The heat flux emitted (E_{TSC}) by radiation is calculated according to the measured disc temperature extracted from the Data Loggers, the Stefan-Boltzmann constant (σ) and the disc emissivity reported as 0.3 by J. P. Hidalgo et al. (2017).

Finally, for calculating the convection-dissipated heat flux, both the disc's convective heat transfer coefficient (h_c) and the far field gas-phase temperature (T_{∞}) are needed. The convective heat transfer coefficient, following the procedure indicated by J. P. Hidalgo et al. (2017), can be calculated for a vertical plate—as the TSC disc is placed with its round plane parallel to gravity—with the Churchill & Clue correlation:

Natural convection (Churchill & Clue):

$$\text{Nu}_L = 0.68 + \frac{0.670 \cdot \text{Ra}_L^{1/4}}{[1 + (0.492/\text{Pr})^{9/16}]^{4/9}} \quad \text{Ra}_L \leq 10^9 \quad (3.16)$$

Where both adimensional numbers, Ra_L and Pr , are calculated the same way as described in Section 2.5.2. Simulated Thermocouple Temperatures. For the disc only natural convection was considered, given the fact that no velocity probes were placed inside the compartment field and hence no experimental Reynolds number can be evaluated. An 80mm diameter was considered as the characteristic length L (J. Hidalgo et al., 2017).

Finally, the far-field gas-phase temperature (T_∞) —needed to quantify the convection losses from the disc— can be calculated from the measured thermocouple data using the radiation-correction methodology proposed by Welch et al. (2007). The 'A' thermocouple tree line, i.e. the first tree array facing the back wall (where the contour will be built) and located 40 cm away from it in the orthogonal direction, will be used for the far-field estimation. According to J. P. Hidalgo et al. (2017) the discs' boundary layer is no more than 10 cm thick, therefore tree line 'A' is outside the layer, as expected. For simplicity, the thermocouple measured temperatures will be considered as the far-field gas-phase temperatures, given that radiation near the wall region is expected not to cause mayor discrepancies between raw thermocouple and real gas-phase temperature measurements. The TSC configuration in the back wall is shown in Fig. 3.6, consisting of a 3 (z-span) x 15 (y-span) node mesh, where each node contains a TSC. The far field temperature for each TSC is considered to be the adjacent thermocouple bead measurement from tree line 'A', where a vertical interpolation was made for each TSC height. With both the disc temperature (T_{disc}) and far-field gas-phase (T_∞) known, the convective heat transfer coefficient can be calculated in one step following the explained procedure.

With this procedure, incident radiation for each TSC in the 3 x 15 back wall mesh (shown in Fig. 3.6) was directly calculated from eq. (3.11). Using the same procedure explained for

the temperature contours, i.e. generating an incident radiation interpolated virtual mesh from the TSC measurement points (using Matplotlib's 'griddata' function), experimental incident radiation contours at the back wall were constructed.

3.5.6. Energy Balance Calculations

An energy balance similar to the one performed by Maluk et al. (2017) on ETFT Test I will be developed in this work. The previous authors developed it based on experimental measurements, while this one will be based on simulated data as its purpose is to verify that FireFOAM is both conservative and capable of reproducing the experimental energy flows analyzed by Maluk et al. (2017). As discussed in section 1.2.2.2 Theoretically-based validation, a simplified energy balance was performed—as experimental data is more limited in resolution than simulated data—as stated in eq. (1.2). For this exercise, the heat release rate (\dot{Q}_{fire}) will be expected to be equal to the sum of the heat rate stored by the gas-phase (\dot{Q}_{gas}), the net enthalpy flux leaving the opening plane ($\dot{H}_{out,opening} - \dot{H}_{in,opening}$), the heat dissipation through the walls (\dot{Q}_{walls}) and the radiative flux leaving through the compartment opening plane ($\dot{Q}_{rad,opening}$):

$$\dot{Q}_{fire} = \dot{Q}_{gas} + (\dot{H}_{out,opening} - \dot{H}_{in,opening}) + \dot{Q}_{walls} + \dot{Q}_{rad,opening} \quad (3.17)$$

The calculation of each term was done through a Python routine. Data given by the post-processing tool ParaView was extracted as comma separated values (".csv" file) for each time-step. Extracted ".csv" fields are given as a list of cell data—where each row represents a cell and each column a property—, therefore each term in eq. (3.17) will be computed in a discrete way considering the original mesh:

$$\dot{Q}_{gas} = \sum_{cells} V_{cell} \cdot \rho(T_{g,cell}) \cdot c_p(T_{g,cell}) \cdot \frac{\Delta T_{g,cell}}{\Delta t} \quad (3.18)$$

$$\dot{H}_{out,opening} - \dot{H}_{in,opening} = \sum_{opening\ patches} \rho(T_{g,p}) \cdot A_{patch} \cdot u_x \cdot c_p(T_{g,p}) \cdot T_{g,p} \quad (3.19)$$

$$\begin{aligned} \dot{Q}_{walls} = & \sum_{wall\ patches} U_{walls}(T_{g,i} - T_{amb}) + \sum_{roof\ patches} U_{roof}(T_{g,i} - T_{amb}) + \\ & \sum_{floor\ patches} U_{floor}(T_{g,i} - T_{amb}) \end{aligned} \quad (3.20)$$

$$\dot{Q}_{rad,openings} = \sum_{opening\ patches} G_p \cdot A_{patch} \quad (3.21)$$

Thermodynamic properties —only specific heat capacity in this case— were estimated considering air in Python's 'thermo.mixture' library package, evaluating the temperature and pressure fields at the cell/patch center. Other solver fields such as density, normal velocity and incident radiation, were also taken as the central value in the patch/volume.

The global heat transfer coefficient (U) was calculated considering the internal equivalent convective heat transfer coefficient ($h'_w = 21 \text{ W/m}^2/\text{K}$), which as explained in section 2.2.3 Boundary and Initial Condition accounts for both convection and radiation dissipation through the wall. Internal solid conductive resistances were taken from material databases and the ROCKWOOL[®] manufacturers' website. The external wall heat transfer coefficient taken from the Chilean HVAC Standard NCh 853/2007 for vertical and horizontal external surfaces ($h_{ext} = 20 \text{ W/m}^2/\text{K}$), as previously mentioned in Section 3.3.3. Boundary and Initial Conditions. Either for the internal walls, roof or floor, the global heat transfer coefficient was calculated considering the j internal solid resistances:

$$U = \left[\frac{1}{h'_w \cdot A_{patch}} + \sum_j \left[\frac{L_j}{k_j \cdot A_{patch}} \right] + \frac{1}{h_{ext} \cdot A_{patch}} \right]^{-1} \quad (3.22)$$

Where L_j and k_j are the length and thermal conductivity of thermal layer j , respectively. For the calculation of \dot{Q}_{gas} , the original mesh was used for the compartment interior discretization. The transient term associated to temperature changes in time ($\frac{\Delta T_{g,v}}{\Delta t}$) was calculated using Finite Differences considering a time-step of 25 s (solver's write interval)—which as previously stated is expected to be reasonable given that the characteristic gas-phase heating time is 120 s (Maluk et al., 2017)—. Depending on the time-step, it was computed as:

$$\begin{aligned} \text{First time step (n=0): } \quad \frac{dT_{g,v}}{dt} &\approx \frac{T_0 - T_{-1}}{\Delta t} = 0 \\ \text{Intermediate time steps (n=i): } \quad \frac{dT_{g,v}}{dt} &\approx \frac{T_{i+1} - T_{i-1}}{2\Delta t} \\ \text{Last time step (n=N): } \quad \frac{dT_{g,v}}{dt} &\approx \frac{T_N - T_{N-1}}{\Delta t} \end{aligned}$$

For energy flux calculations through surfaces —i.e. roof, floor, walls and compartment opening plane—, the original mesh patches were used for surface discretization.

3.5.7. Runtime Data

Runtime data —quantified in seconds— was extracted from the 'log.fireFoam' file, generated in the case directory. The variable 'clockTime' gives the simulation's wall clock running time, as its defined as: 'the elapsed wall clock time from the beginning of the simulation until the latest time step.' (OpenCFD, 2007). Therefore, the total runtime taken by the simulation is the value of 'clockTime' for the final time-step.

4. CHAPTER 4: RESULTS

4.1. Runtime Sensitivity

Results from the runtime studies are shown in Figures 4.1, 4.2 (Mesh Size), 4.3 (Number of Processors) and 4.4 (Number of Solid Angles). The computing unit consisted in eight Intel(R) Core(TM) i7-4820k CPU @3.70 GHz processors, and 64 GB RAM memory. Note that T20 showed numerical divergence.

4.1.1. Mesh Size

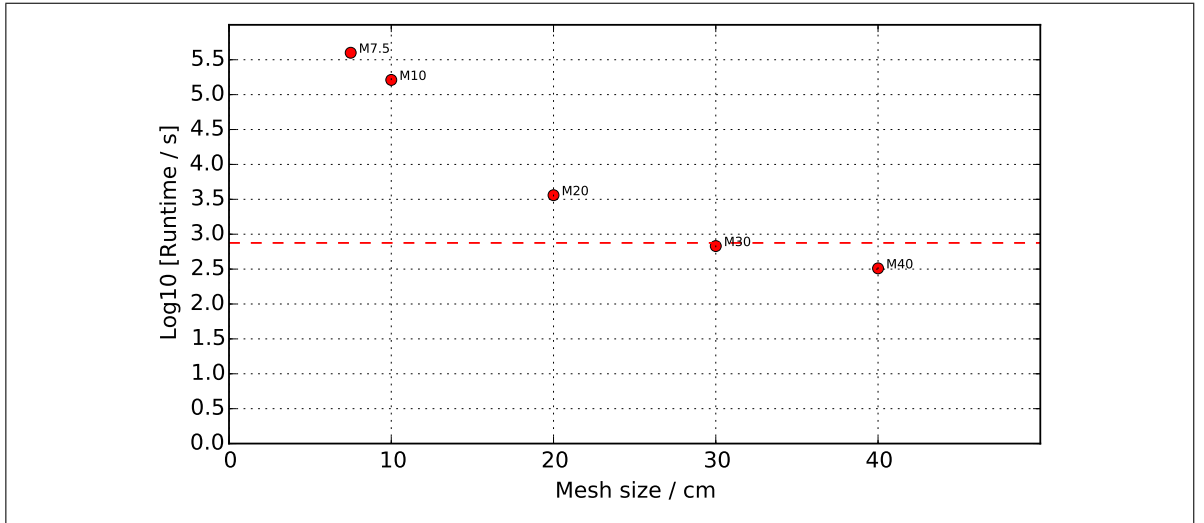


Figure 4.1. Runtime for different meshes. Log10[] scale is applied for run-times as to capture all magnitude orders. The red dashed line marks the real time limit, i.e. $\text{Log}_{10}[750\text{s}] = 2.88$.

Regarding the mesh runtime study, a logarithmic scale was applied as to capture all of its magnitude orders. Therefore, as expected, runtime is extremely sensible to the mesh size. As the mesh size decreases, not only the solver has to compute a bigger amount of cells, but it has to deal with smaller time-steps as the Courant number has to be kept lower than unity. Considering this, only cases M30 and M40 run faster than real time, where M30 has a small advantage of a few seconds (approximately 70s).

It is important to remember that, due to the mesh construction algorithm which keeps the original compartment dimensions intact, the mesh size only represents an approximation of the average cell-dimension. As the cell dimension approximates the structural characteristic dimensions, e.g. 50 cm for the burner, coarser meshes imply a small change in the number of elements. For example refinement from M40 (18 k elements) to M30 (29 k elements) only increases the number of elements by less than two times, while from M20 (83 k elements) to M10 (560 k elements) the number of elements increases in almost seven times. This explains the big increase seen in runtime when passing from M20 to M10. The effect of the number of elements in the runtime is seen in Figure 4.2.

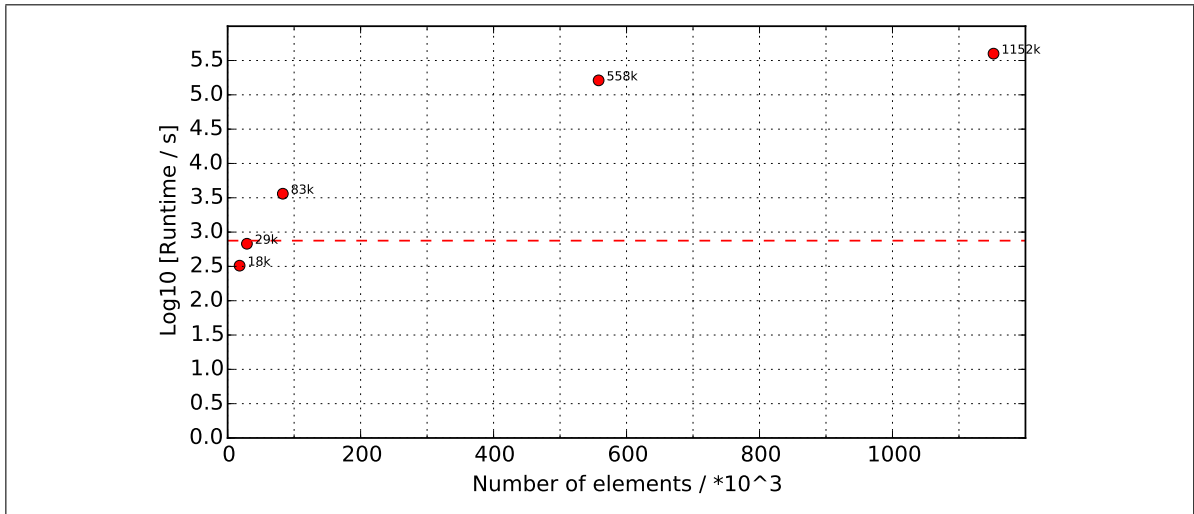


Figure 4.2. Runtime for cases M40, M30, M20, M10 and M7.5 represented as number of elements rather than average cell size. The red dashed line marks the real time limit, i.e. $\text{Log}_{10}[750\text{s}]=2.88$.

Here it is graphically seen that further refining high resolution meshes, e.g. M10 to M7.5, involves large increases in the number of elements as these are much smaller than compartment structures. This is a characteristic of the designed meshing algorithm, which has to be considered when interpreting results.

4.1.2. Number of Processors

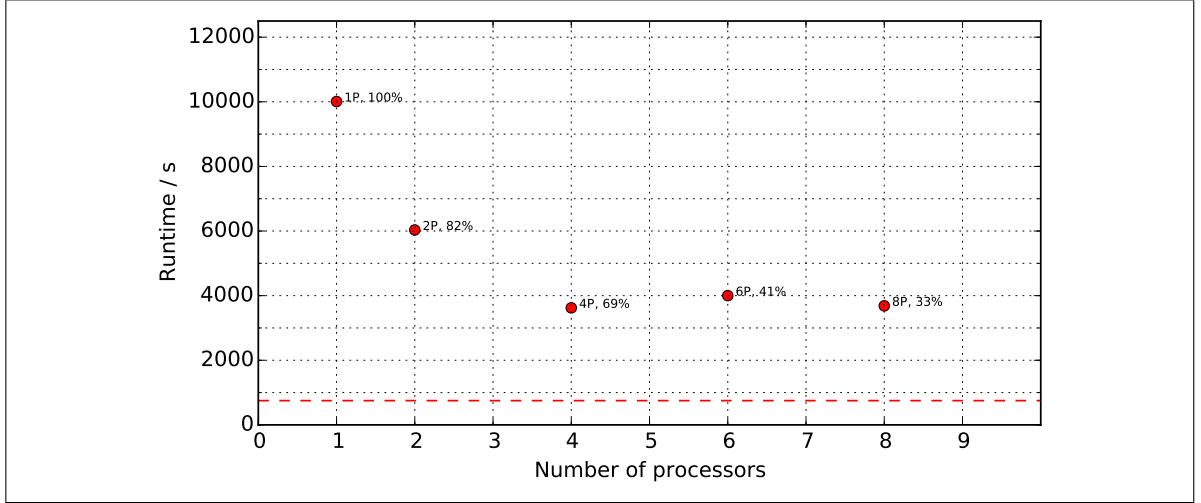


Figure 4.3. Runtime for different number of parallel processors. Parallel efficiency, expressed as a percentage, is calculated as $\epsilon = \frac{T_1}{p \cdot T_p}$ (where T_p is the computing time for p processors) and placed adjacent to each case. The red dashed line marks the real time limit, i.e. 750s. 20 cm cubic mesh and 144 solid angles for all cases.

Processor division was done as follows. The first addition of processors, i.e. from 1P to 2P, involves a domain division in the z-axis in two zones, each for one processor. Further addition of processors involves further divisions in the domain along the y-axis. For example, the case 6P was accomplished by a division of two zones in the z-axis, and three zones in the y-axis.

Changes in runtime due to parallel processing are significant, specially when adding the first parallel processors. When increasing from 1P to 2P, runtime decreases in 4000 s (nearly more than an hour), while form 2P to 4P it decreases in 2400 s (40 minutes). Further addition of parallel processors is not beneficial past the 4P case —as no runtime is reduced—, given that additional processors slow down the message passing protocols. The parallel efficiency parameter quantifies how message passing gets highly interrupted past this point: with 6 processors, less than half of the total runtime (41%) is used for computing results, while the rest (59%) is used in message passing.

4.1.3. Number of Solid Angles

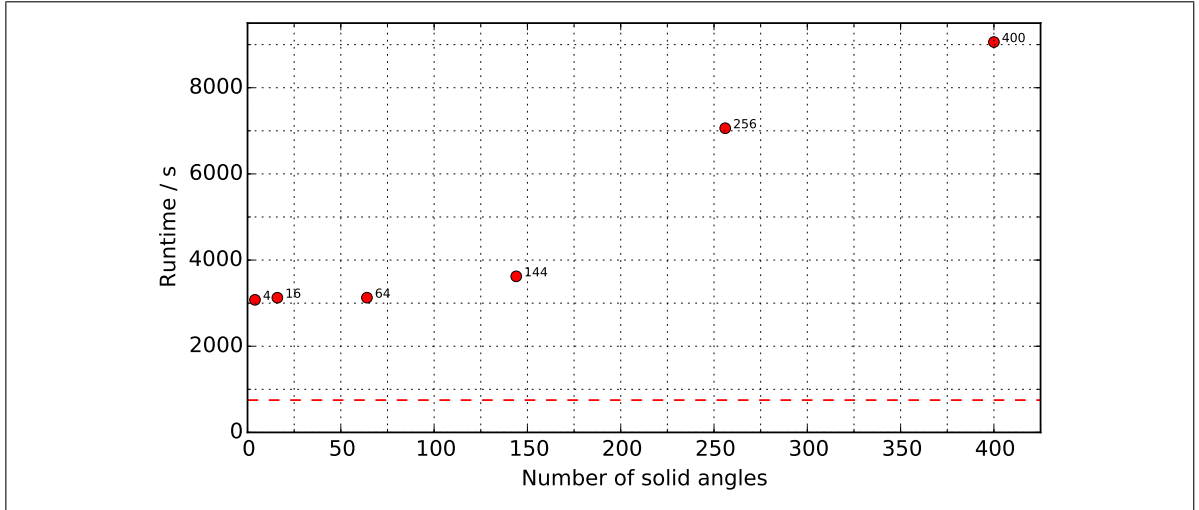


Figure 4.4. Runtime for different number of solid angles. 20 cm cubic mesh for all cases. The red dashed line marks the real time limit, i.e. 750s.

In the range of 4 to 144 solid angles, the increase in runtime is insignificant —approx. 550s (9 minutes) of real time— compared to the increase in the range of 144 to 400 solid angles. From 144 to 256 solid angles a 3400 s (55 minutes) increase in runtime is observed, while from 256 to 400 solid angles a smaller increase of 2000 s (33 minutes) is observed. As later results indicate, incident radiation convergence with solid angles is achieved at approximately 16 solid angles for the M20 case back wall, therefore in principle using from 16 to 144 solid angles is reasonable for the baseline. This number is expected to be mesh-dependant, as it will be later discussed.

4.2. Theory-Based Validation

This section was designed for testing the solver's internal consistency and convergence properties with different mesh sizes.

4.2.1. Energy Balance

Previous to the energy balance, a mass balance was performed as shown in Figure 4.5 for both M20 and M10 cases.

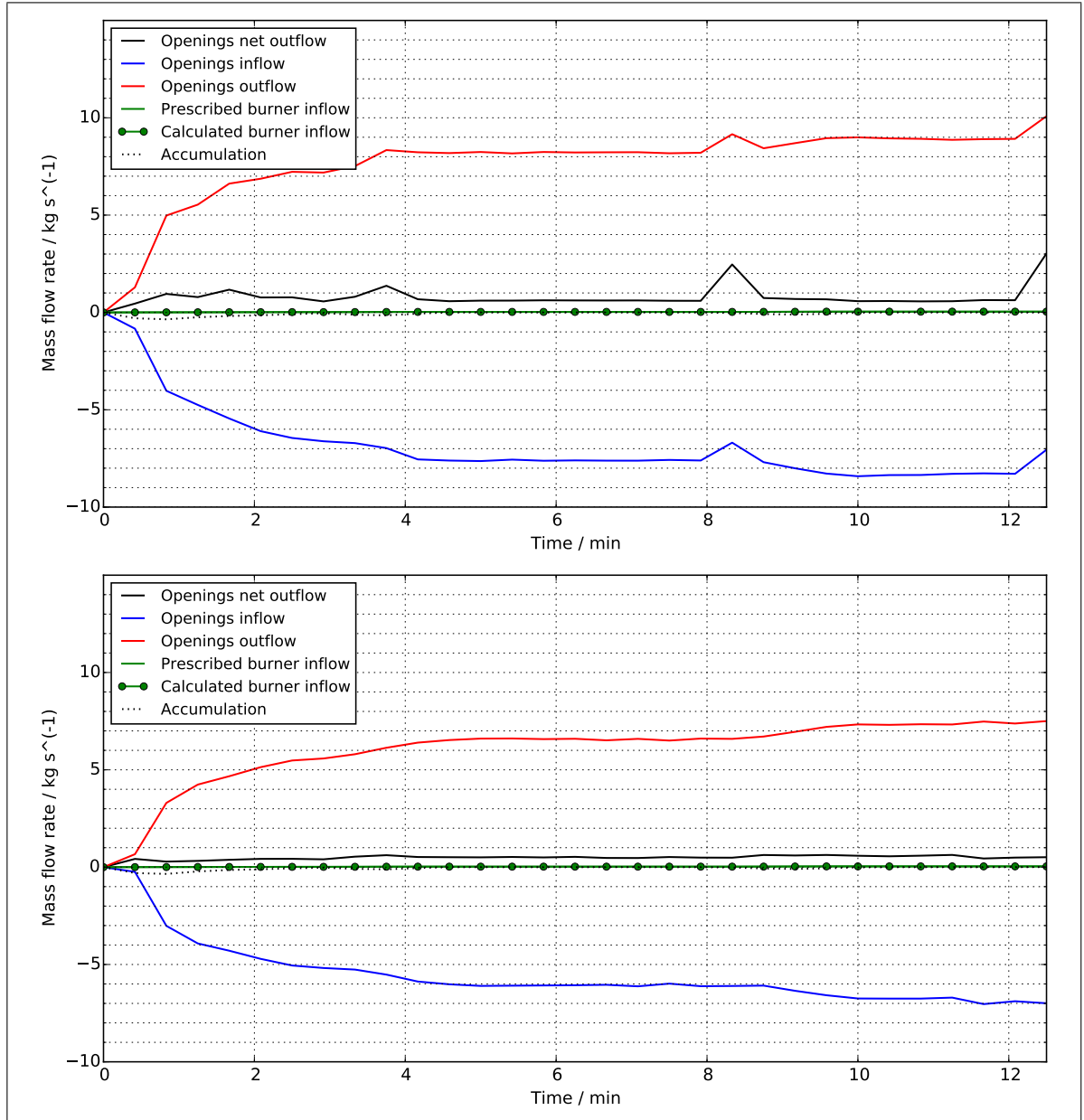


Figure 4.5. Mass fluxes considering the compartment interior as a control volume. Above: M20. Below: M10. Fuel mass flow rate is negligible compared to air entrainment, as expected.

It is expected that the net mass outflow is zero during all steady state periods, considering that during these the compartment does not loose/gain gas-phase mass and the fuel flow rate is negligible compared to the air entrainment. This was not accomplished for either M20 nor M10 cases, as a net outflow of 0.5 kg/s is seen in both cases. This 0.5 kg/s discrepancy represents approximately 7% of the maximum outflow of 7.5 kg/s, which is above the expected calculation error¹ of approximately $\pm 5\%$. As the 0.5 kg/s discrepancy is equal for both M10 and M20 cases, it is mostly attributed to systematical errors when imposing models on governing equations (e.g. sub-grid scale modeling). As shown in Chapter 2: Theoretical Background, the introduction of sub-grid scale modeling promotes deviations from the conservative form of the Navier-Stokes system. Inflow and outflow were calculated at the compartment opening plane, as all other internal patches were either walls or burners.

Discontinual solutions are seen within certain time steps in the M20 case (refer to Fig. 4.5), referred as time steps where calculated mass flow rates experience sudden increases. These discontinual time steps are contained in periods of fuel mass flow rate change. As it does not occur in the M10 case, these internal inconsistencies are attributed to poor transient modeling in coarse meshes. Although not noticeable on the plot, both prescribed and calculated burner mass flow rates are the same in all time steps ($\leq 1\%$ difference), including the M20 discontinual time steps. This implies that the boundary condition at the burners worked perfectly in terms of continuity and prescribed values, for both steady state and transient periods.

Finally, both cases show consistency regarding mass loss inside the compartment during transient periods. Figure 4.5 captures this phenomena as negative accumulation (black dotted curves) is seen when the heat release rate increases. This is physically consistent: as the heat release rate increases, internal temperatures also increase and therefore the compartment volume holds less mass (due to mean density decrease). Negative mass accumulation periods account for this extra mass leaving the compartment.

¹Calculation error is understood as deviations from expected values, due to the summed effect of interpolation, differentiation and integration errors.

Figure 4.6 shows the energy balance study results. To visualize the different proportions of energy dissipation in the compartment, Figure 4.7 shows the normalized energy balance. To test the solver's capabilities in reproducing the experimental energy fluxes quantified by Maluk et al. (2017), following discussions will be made mainly regarding the normalized fluxes.

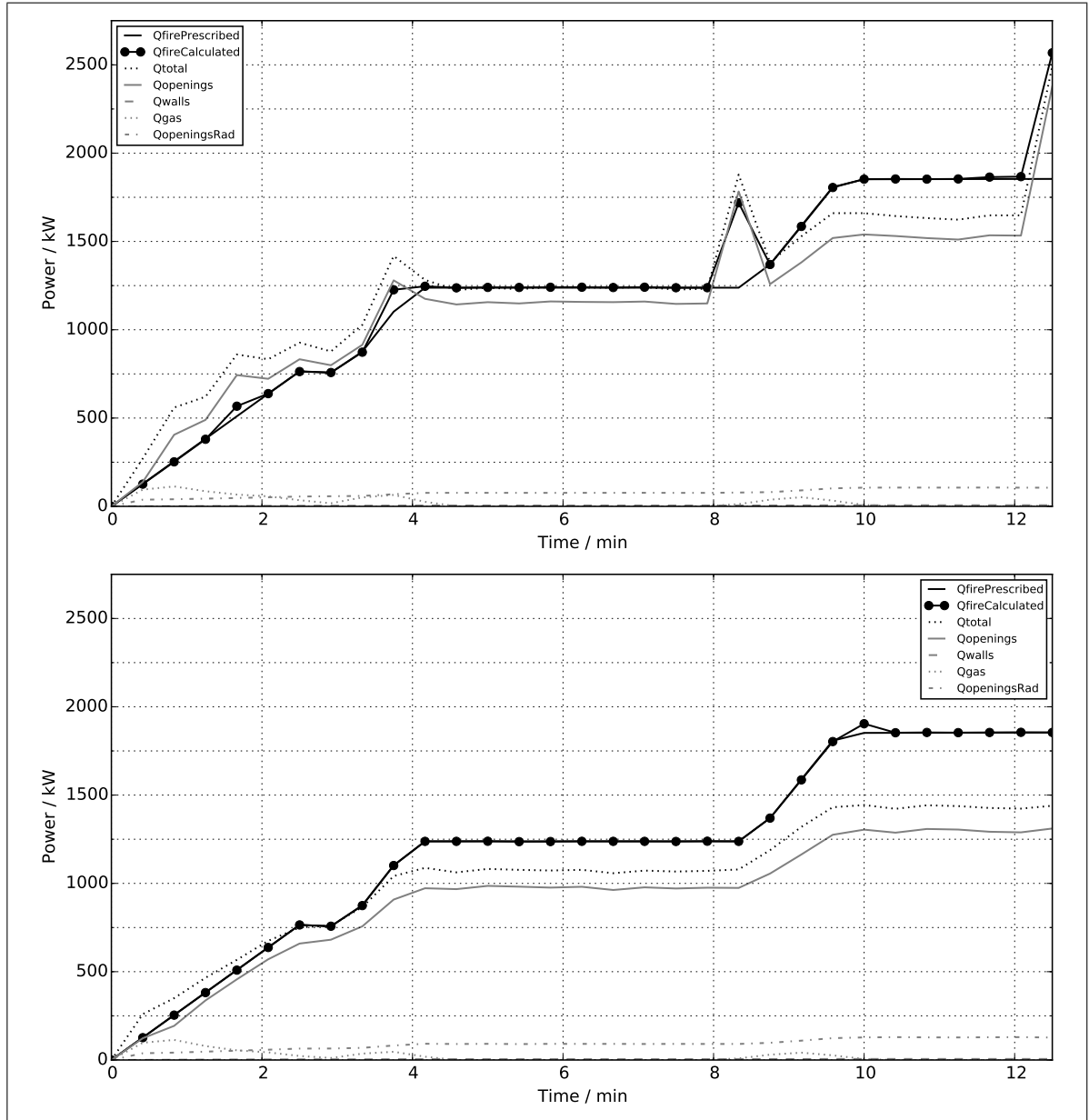


Figure 4.6. Simulated compartment energy fluxes. Above: M20. Below: M10.

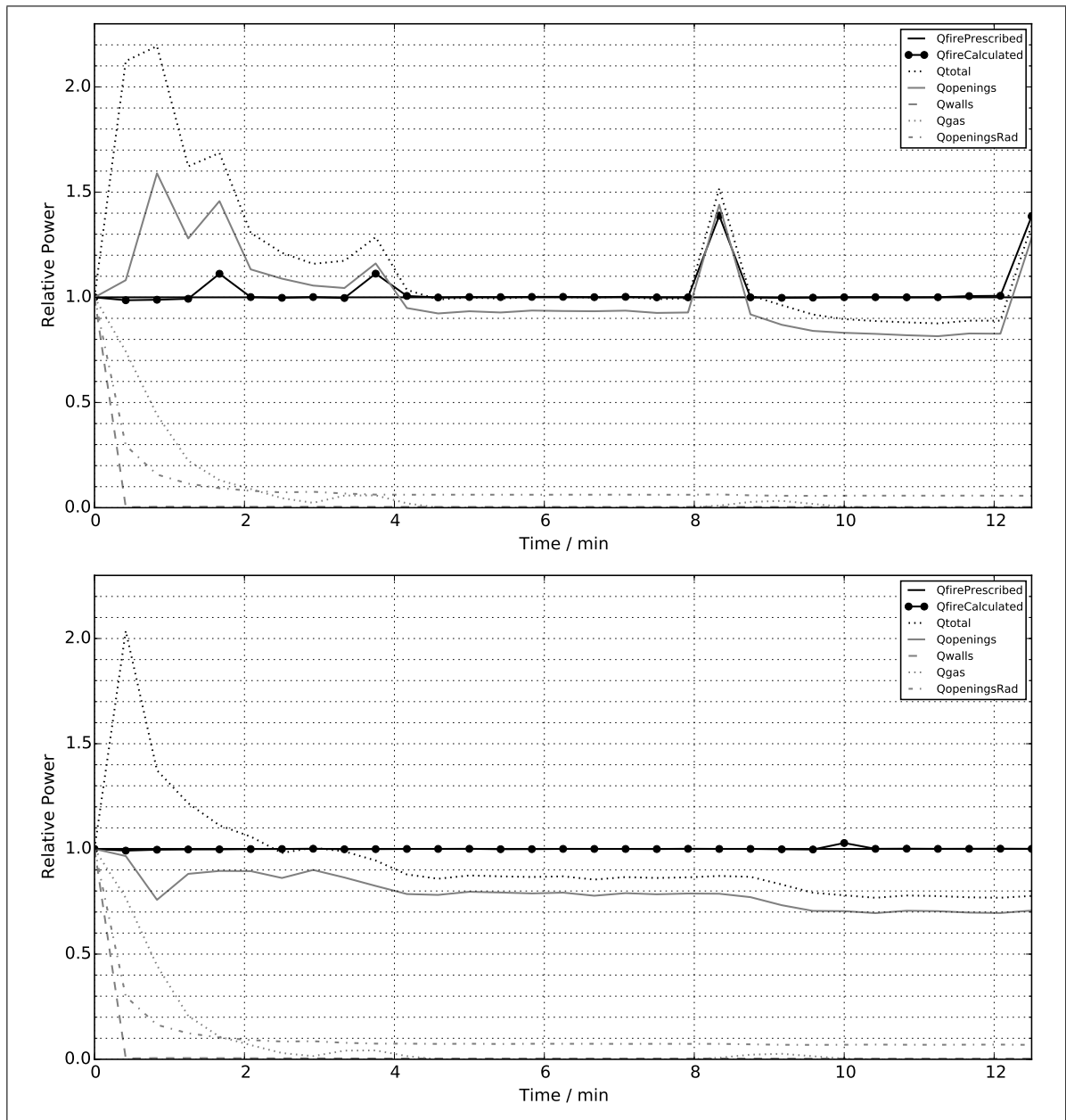


Figure 4.7. Normalized energy fluxes with respect to the prescribed heat release rate ('QfirePrescribed'). Above: M20. Below: M10.

Figures 4.6 and 4.7 quantify both the prescribed and calculated heat release rates, noted as 'QfirePrescribed' and 'QfireCalculated' respectively. The prescribed heat release rate was calculated as the prescribed propane mass flow rate times its specific heat of combustion

(46.45 MJ/kg), while the calculated heat release rate as the sum of the heat released in each compartment cell as calculated by the EDC reaction model. A consistent result would be if both prescribed and calculated values are equal for all time steps, which occurs only in the M10 case. The M20 case shows discontinuities in transient time-steps —same effect seen on the M20 mass balance—, reaching calculated heat release rates from 10% to 40% higher than the prescribed values. Given these inconsistencies in the M20 case, further discussions regarding the energy balance will refer to the M10 case, as here the solver shows internal consistency regarding prescribed heat release rates.

The curve 'Qtotal' —sum of energy dissipation through walls, openings (due to both net enthalpy outflow and radiation) and accumulation as internal energy— should ideally be equal to the heat release rate for all time-steps, considering an approximate $\pm 5\%$ of calculation error. Regarding the first 2 minutes of the M10 case, consisting of a linear fuel mass flow rate ramp-up, 'Qtotal' almost doubled the total heat release rate, mainly because of high values in both 'Qgas' and 'QopeningsRad'. From minutes 2 to 4, consisting in a brief propane flow rate hold period followed by a ramp-up, 'Qtotal' becomes similar to 'Qfire' considering the $\pm 5\%$ calculation error. From 4 to 8.5 minutes, corresponding to the 50% HRR_{max} steady state period, 'Qtotal' was aprox. 85-90% of the total heat release rate, decreasing to almost 80% during the steady state 74% HRR_{max} period (minutes 9.5 to 12.5). Therefore, as the heat release rate increases, the unaccounted proportion of energy fluxes becomes higher. This could be for either a flux not accounted for (which gets higher with increased HRR), or deviations from the conservative form of Navier-Stokes due to the introduction of models and assumptions (e.g. sub-grid scale modeling and Boussinesq hypothesis, among others).

Characteristic data for comparing how the simulated energy fluxes assimilate to the experimental fluxes are: characteristic heating times and proportions of each dissipation type. The characteristic heating time can be quantified in the HRR ramping from 50% HRR_{max} to 74% HRR_{max} . As seen for the M10 case in Figure 4.7, this characteristic time is equal to 2 minutes and occurs between minutes 8 and 10, as 'Qgas' rises to a peak of approx. 3-4%

of 'Qfire' and then goes back to zero. The experimental energy balance study performed by Maluk et al. (2017) also reports this characteristic time as 2 minutes, with 'Qgas' peaking at approx. 5% of 'Qfire' for the 50-74% HRR_{max} transition. Table 4.1 summarizes the simulated (M10) and experimental energy distributions for both 50% and 74% steady states periods.

Table 4.1. Energy distribution analysis for both steady state periods.

	Steady state HRR period	'Qtotal'	'Qopenings'	'QopeningsRad'	'Qwalls'
Simulated	50%	0.88	0.80	0.08	<0.01
	74%	0.78	0.70	0.08	<0.01
Experimental	50%	0.90	0.67	N/A	0.23
	74%	0.88	0.70	N/A	0.18

According to the experimental analysis performed by Maluk et al. (2017), an increase in the heat release rate from 50% to 74% HRR_{max} increases the proportion of energy dissipated due to hot gas enthalpy leaving the compartment—from 0.67 to 0.70—, lowering heat dissipation through walls—from 0.23 to 0.18—. The simulation does not capture this experimental observation, because even though the hot gas enthalpy flux increases, its proportion relative to 'Qfire' decreases from 0.80 to 0.70. Heat dissipation through internal walls is quantified as less than one percent in both simulations, and given that is quantified as nearly 20% in the experimental case, the heat transfer model—which performs a steady-state one-dimensional heat transfer problem through each boundary patch—is clearly not representative. Calculated U-values for internal walls, roof and floor were: 0.52 W/m²/K, 0.12 W/m²/K and 0.26 W/m²/K respectively. U-values reported by J. Hidalgo et al. (2017) for internal walls, roof and floor were: 0.18 W/m²/K, 0.15 W/m²/K and 1.02 W/m²/K respectively. As U-values in simulations are of similar magnitude as U-values reported by J. Hidalgo et al. (2017), specially for the roof where most of the dissipation occurs, thermal

properties of the boundaries were well reproduced. As Maluk et al. (2017) could not experimentally estimate the radiation dissipation through openings, the unaccounted 10% energy flux (refer to Experimental in Table 4.1) was attributed to this dissipation mechanism. This experimental estimation of 'QopeningsRad' is almost the same as the simulated value of 8%, which gives positive feedback on radiation modeling.

4.2.2. Streamline Mesh-Convergence

To test FireFOAM's performance in flow pattern convergence with mesh refinement, Figure 4.8 shows steady state streamline maps for the compartment interior. All cases capture the physical phenomena of air entrainment through the lower opening section and hot gas escape from the upper section. Refining from case M40 to M30 and from M30 to M20 showed no convergence, as the central vortex seen in M40 tends to disappear for M30 and reappear in M20. Convergence is achieved in the M10 case, as further refinement to M7.5 does not significantly change the flow pattern consisting of two steady state vortex structures. A similar outflow velocity modulus of approximately 1.5 m/s is seen in both M10 and M7.5, which also supports the fact that the M10 case is flow-converged.

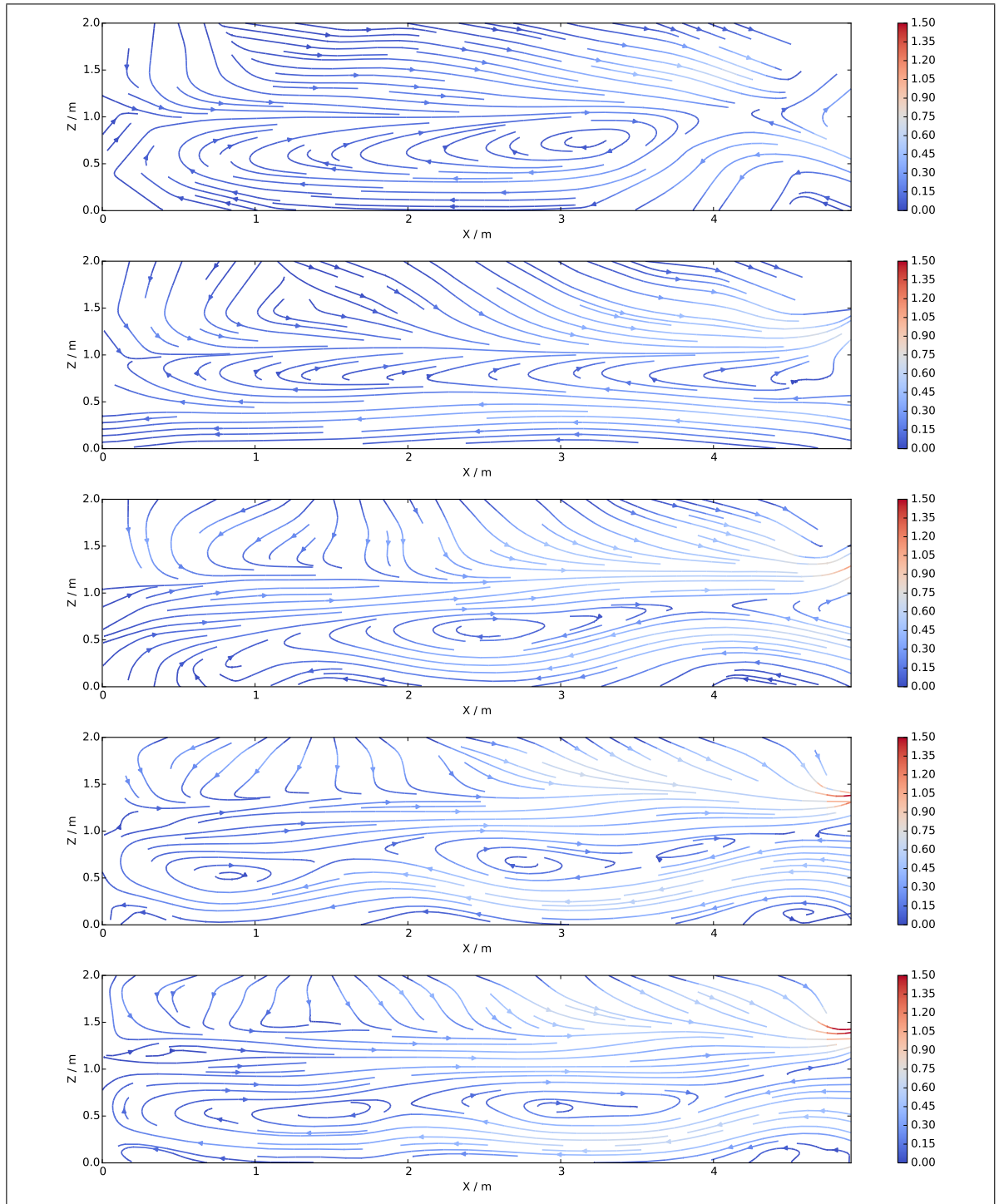


Figure 4.8. Time-averaged streamline maps in section 15 for the steady state period of 74% HRR_{max} . Velocity modulus in [m/s]. From top to bottom: M40, M30, M20, M10 and M7.5.

4.3. Experimentally-Based Validation

Results shown in this section include all simulated cases, with the exception of T20 (tetrahedral 20 cm mesh) where no convergence was achieved.

4.3.1. Temperature Contours

Figures 4.9 and 4.10 show experimental and simulated compartment temperature maps.

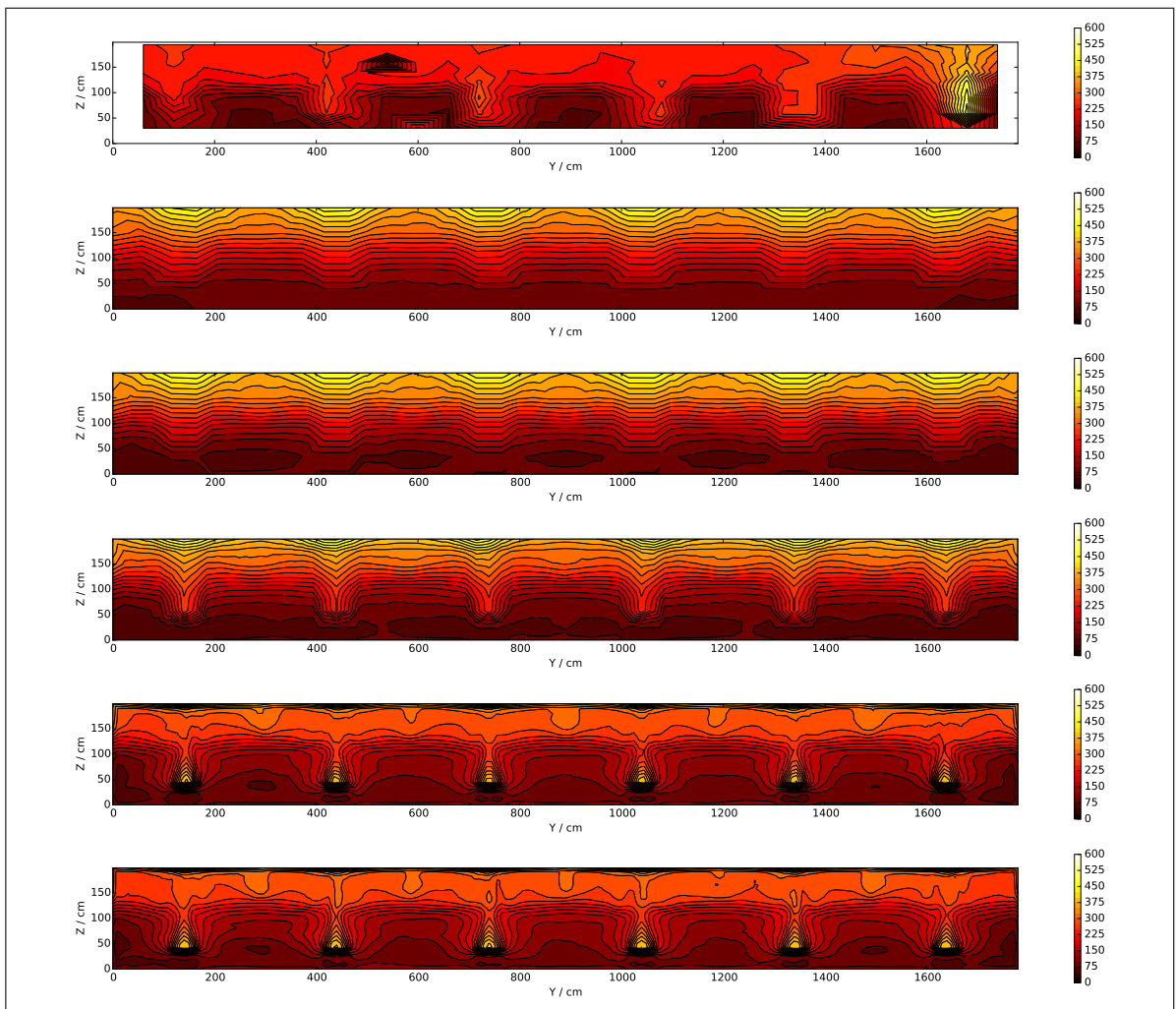


Figure 4.9. Time-averaged thermocouple temperature contours in section B for the steady state period of 74% HRR_{max} . Temperatures in [°C]. From top to bottom: Experimental, M40, M30, M20, M10 and M7.5.

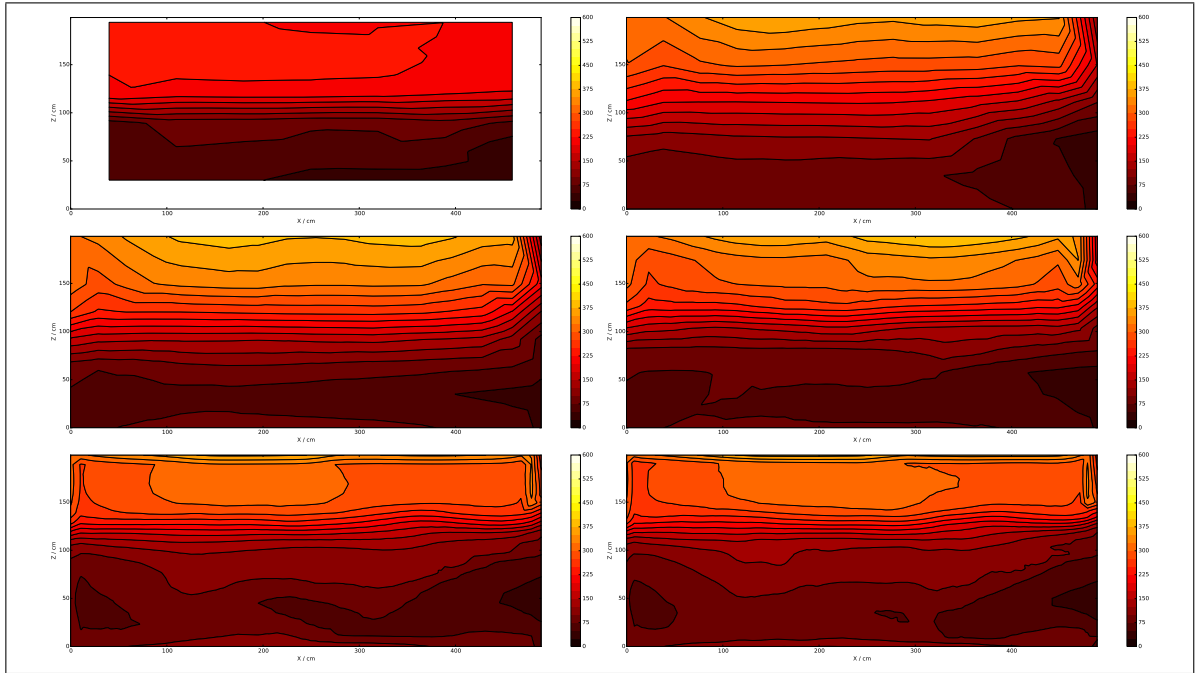


Figure 4.10. Time-averaged thermocouple temperature contours in section 15 for the steady state period of 74% HRR_{max} . Temperatures in [$^{\circ}C$]. From left to right and descending: Experimental, M40, M30, M20, M10 and M7.5.

As seen in both section B and 15 temperature contours (refer to Fig. 3.10 for locations at the compartment interior), the solver manages to capture the experimental neutral plane height of 1-1.2m (J. Hidalgo et al., 2017) —considered as the plane that divides the air entrainment zone from the hot combustion gas zones— from the M20 to the M7.5 case. This interface is created by the presence of the hanger (J. Hidalgo et al., 2017). Coarser meshes tend to over-estimate simulated thermocouple temperatures at the ceiling level, mainly due to high estimations of incident radiation (G field).

As seen in the experimental section B contour, uneven distribution of propane among the burners caused elevated temperatures near the far-right pair of burners. This condition is backed up by J. Hidalgo et al. (2017), who explicitly states that Burner 1a exhibited a higher output than the others. This behavior could not be reproduced in the simulations, as the magnitude of the imbalance is unknown. This considered, cases M10 and M7.5 tend to capture

better the 200°C hot layer temperature for the 74% HRR_{max} steady state period, with approximate overestimated values of 250 to 300 °C. Uneven propane distribution can account for some over-estimation error. Considering the section B experimental contour, temperatures in zones aligned with Burner pairs 6-2 are approximately 200°C, while the zone aligned with Burner pair 1 exhibits approximately 450°C. Weight-averaging these temperatures considering each burner pair as a weight gives an estimate of 240°C in the hot-gas layer if even propane distribution would have been present in Test I. Therefore, the measured smoke layer temperature from Burners 6-2 falls back approximately 40°C with respect to an even propane distribution scenario. This 40°C underestimation in measurements will be referred from now on as a systematical "fall-back error".

4.3.2. Incident Radiation Contours

Figure 4.11 shows the results for incident radiation (G) convergence with increasing number of solid angles in FVDOM, considering the M20 case. Note that the simulated cases use a wider scale than the experimental case, indicating an important limitation seen in the M20 case: incident radiation is highly overestimated in the hot-gas layer. Regarding convergence with solid angles, the M20 case is converged with only 16 solid angles. This value is much smaller than the aprox. 300 solid angle-convergence reported by Vilfayeau et al. (2016) for FireFOAM. This discrepancy is attributed to the difference in mesh sizes, as the cited study uses 2-12 mm cubic meshes, compared to the cubic 200mm baseline mesh used here. Therefore it is probable that the number of solid angles needed for G convergence increases with finer meshes. It was found that further improvement in the G estimation is achieved through finer meshes rather than more solid angles, as Figure 4.12 shows.

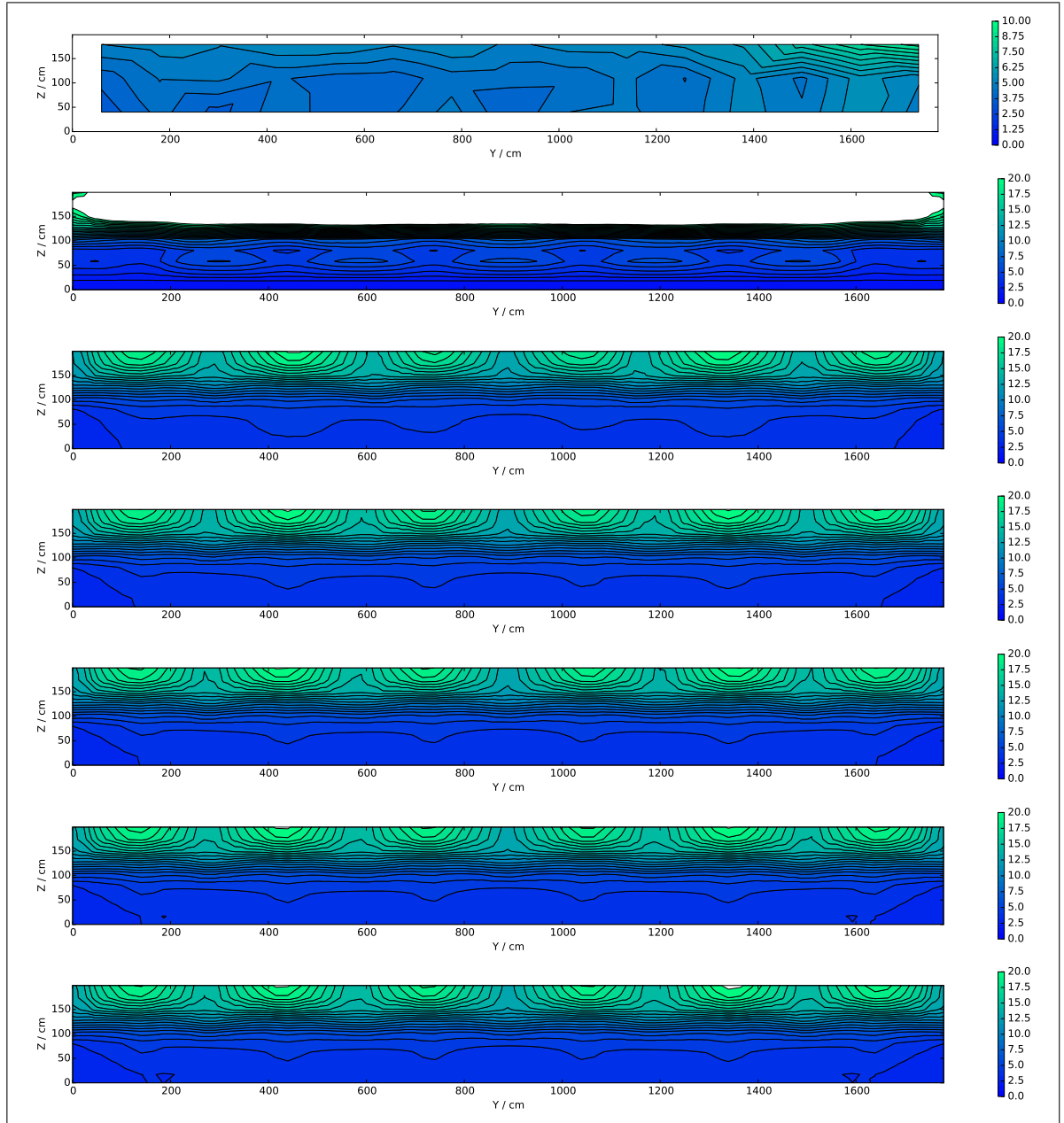


Figure 4.11. Time-averaged back wall incident radiation contours for the steady state period of 74% HRR_{max} . Scale in $[kW/m^2]$. Note that for the experimental case the scale is from 0 to 10 kW/m^2 , while for simulations from 0 to 20 kW/m^2 . This change was done for visual purposes, as the M20 case does not reproduce well the experimental value of G . From top to bottom: Experimental, R4 ($nPhi=nTheta=1$), R16 ($nPhi=nTheta=2$), R64 ($nPhi=nTheta=4$), R144 ($nPhi=nTheta=6$), R256 ($nPhi=nTheta=8$) and R400 ($nPhi=nTheta=10$).

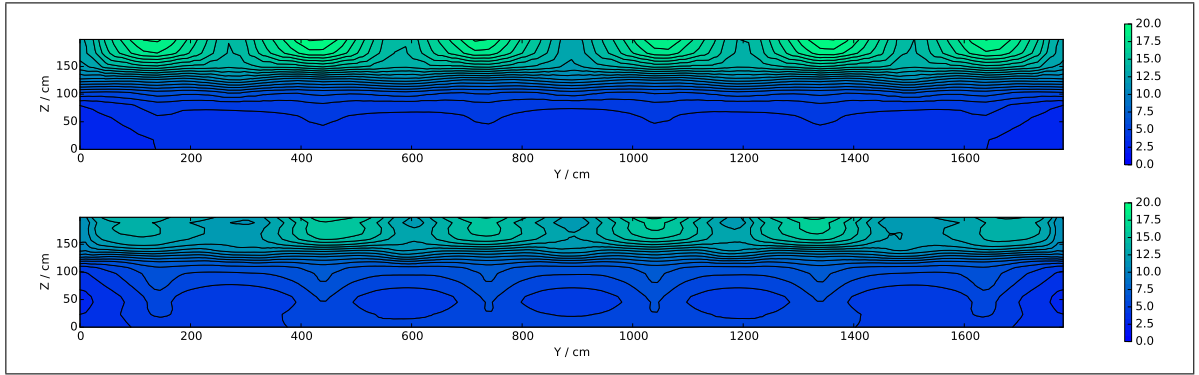


Figure 4.12. Time-averaged back wall incident radiation contours for the steady state period of 74% HRR_{max} . Top to bottom: M20 and M10 (144 s.a.).

4.3.3. Interior Temperature Profiles

To study Y-wise and X-wise temperature distributions, Figures 4.13 and 4.14 show thermocouple temperature profiles in sections B and 15, respectively (refer to Fig. 3.10 locations within the compartment). Temperature profiles are plotted along four heights (600mm, 1200mm, 1600mm and 1950mm), where each height represents a stratification zone at the compartment interior. These zones are: air entrainment layer (600mm), transition layer between air and smoke layers (1200mm), smoke layer (1600mm) and roof boundary layer (1950mm). At each height, profiles for the different meshes are included for studying mesh-convergence. Note that uneven propane distribution rises experimental temperatures on the far-right burners, an effect that becomes accented at lower altitudes.

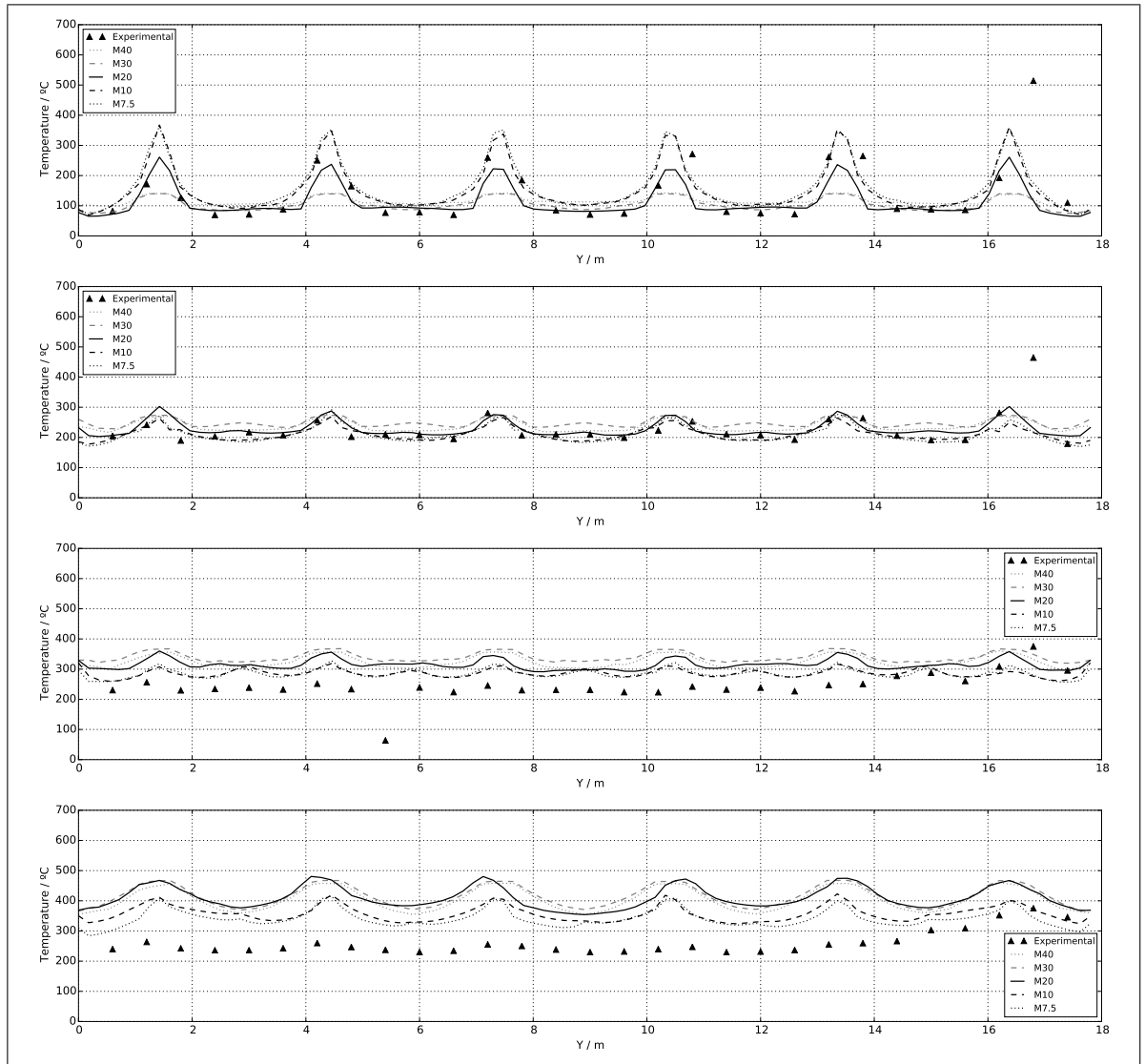


Figure 4.13. Time-averaged thermocouple temperatures in section B. 74% HRR_{max} steady state period. From top to bottom: 600mm, 1200mm, 1600mm and 1950mm.

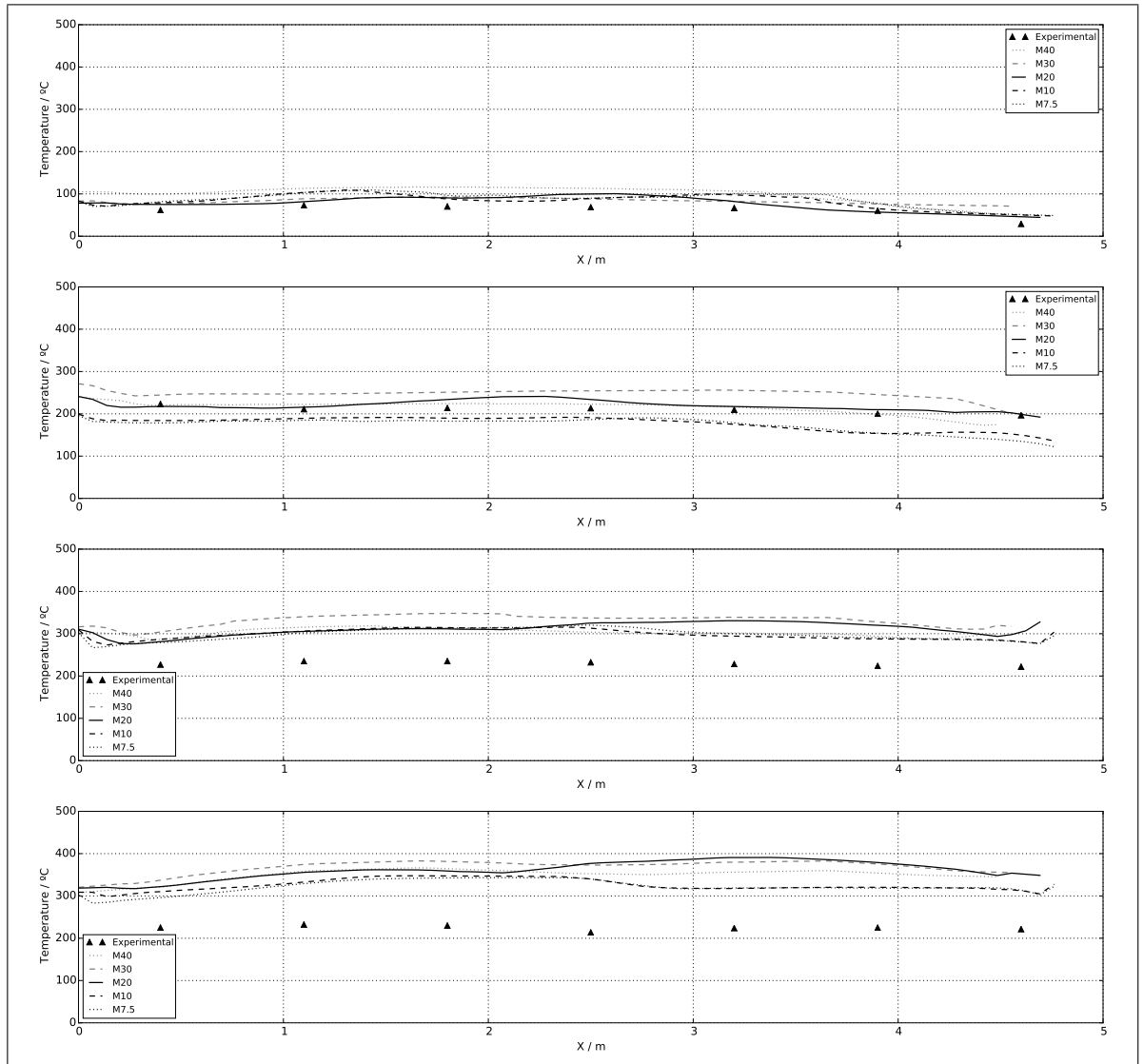


Figure 4.14. Time-averaged thermocouple temperatures in section 15. 74% HRR_{max} steady state period. From top to bottom: 600mm, 1200mm, 1600mm and 1950mm.

Mesh convergence at 600mm, 1200mm and 1600mm is achieved in the M10 case, as further refinement no M7.5 does not show significant improvement in temperature estimations. At 600mm and 1200mm above ground level, cases M10 and M7.5 show good agreement with thermocouple measurements. At 1600mm above the floor, temperature over-estimations of +25 to +50°C start occurring for all cases with no significant improvement from M10 to

M7.5. This error can be attributed to the 40°C experimental fall-back in the smoke layer due to uneven propane distribution among burners. At 1950mm above the ground level, over estimations for the M10 and M7.5 cases move from +50 to +100°C, now depending on the mesh size as the M7.5 over-estimations are smaller. Note that for the previous analysis the anomalous data points near Burner 1a (higher propane supply) are not considered.

With the previous said, a global analysis indicates that for heights 600mm, 1200mm and 1600mm M10 is mesh-converged, while for 1950mm no mesh convergence is achieved, as refinement from M10 to M7.5 improves the temperature estimations. Good data agreement is seen for heights 600mm and 1200mm. Considering the 40°C experimental fall-back, good data agreement is also seen at 1600mm.

For the study of the near-flame and far-field temperature simulation capabilities, Figure 4.15 shows two thermocouple trees temperature vertical profiles.

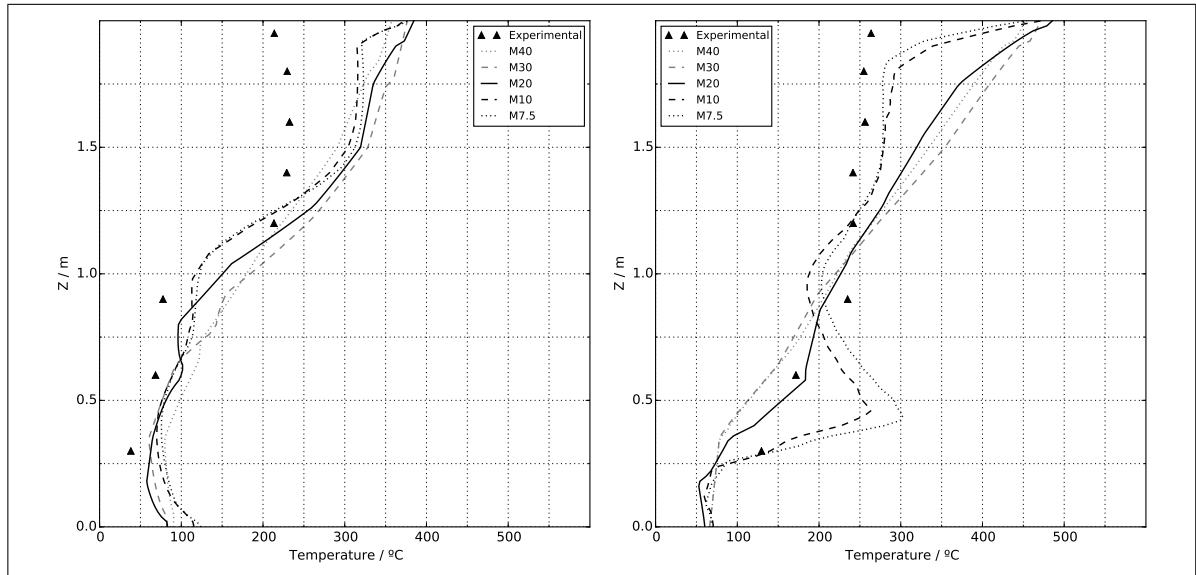


Figure 4.15. Time-averaged thermocouple temperatures for the 74% HRR steady state period. Left: Central tree D15, located in the far field. Right: Tree B2, located along the flame-plume region of Burner 6a.

Regarding the central tree (D15), good agreement between simulated and experimental data is seen between floor-level and 1200 mm. At higher altitudes, i.e. fully inside the hot gas layer, temperatures are over-estimated by 75°C to 150°C with increased errors at bigger heights. Some of this error can be explained by the 40°C experimental fall-back due to uneven propane distribution. For this far-field study, mesh convergence is achieved in the M10 case, while for the near-flame study (tree B2) no mesh convergence is observed. This is consistent, given that temperature distributions in the far field are dominated by the mixing of hot gases with quiescent air rather than chemical reactions and radiative heat transfer—phenomenon that demands major mesh refinement—. Over-estimation of the simulated thermocouple temperatures in the hot-gas zone is attributed to high values of simulated incident radiation in this sector (see Figure 4.11). As the G -field is considerably over-estimated (more than double of the experimental values), the thermocouple model delivers excessive thermocouple temperatures. The finer-meshed case M10 delivers less intensive G -fields near the roof (see Fig. 4.12), therefore thermocouple temperature over-estimations are lower.

As seen in the tree B2 profile, the near-flame temperature estimations are more sensitive to different mesh sizes, as in this zone heat release rate and buoyant acceleration dominates fire dynamics (Trouvé & Wang, 2010). Here it can be seen how finer meshes show better agreement with experimental data, especially in the hot gas zone near the roof. No converged results were achieved, as refinement from M10 to M7.5 still shows considerable improvements of approximately 25°C. Thermocouple temperature over estimation is seen at 600mm, with increasing error for finer meshes. Coarse meshes deliver temperatures averaged over a bigger cell than fine meshes, yielding lower reaction-zone temperatures as coarse averaging considers zones outside this hot layer. Some of the error in the hot-gas layer can be explained by the 40°C experimental fall-back caused the propane flow weighting towards Burner 1a, affecting Burner 6a which is adjacent to tree B2.

4.3.4. Opening Plane Temperature and Velocity Profiles

Results for the outflow study are shown in Figures 4.16 and 4.17. Simulated results show global agreement to what is physically expected: at higher heat release rates, both temperatures and outflow velocities increase at the compartment outlet.

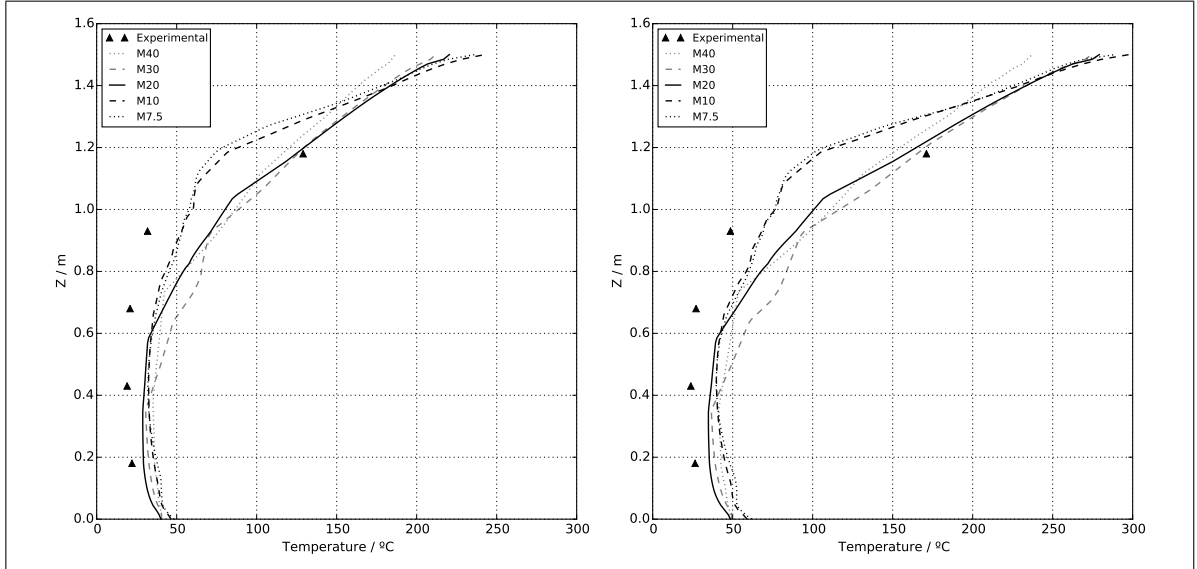


Figure 4.16. Time-averaged thermocouple temperatures at openings for the 50% (left) and 74% (right) HRR_{max} steady state periods. Results correspond to an average between openings 1, 8 and 15.

Regarding temperature profiles, it is seen that the M10 case is mesh-converged as refinement to M7.5 does not significantly improve the estimations. This is consistent with the observation done for the central tree: mesh convergence in the far field is achieved with coarser meshes than in the flaming zone, as the dominant phenomena in the former region does not demand mayor mesh resolution. As no turbulent mixing causing chemical heat release rate is present in the opening plane, the LES-filter does not need to capture small eddies relevant for thermal processes. It is also seen that temperature estimations show less experimental agreement at the 1.18m-high thermocouple when refining the mesh. These errors increase with higher heat release rates: for cases M10 and M7.5 error at 50% HRR_{max} is of approx. 50°C (under-estimation), while for the 74% HRR_{max} period of approx. 75°C (under-estimation).

Even though finer meshes (cases M10 and M7.5) deliver a poor estimation of temperatures at thermocouple $z=1.18\text{m}$, they reproduce the steep temperature gradients better at the neutral plane location seen in the experimental data. Cases M10 and M7.5 show good agreement with fresh air entrainment temperatures —measured by the lower four thermocouples— and hence with the neutral plane height, with a height over estimation of no more than 0.1 m. The previous, considering experimental and simulated neutral plane heights of 1.1 m —between 1.0 m and 1.2 m according to J. Hidalgo et al. (2017)— and 1.2 m, respectively

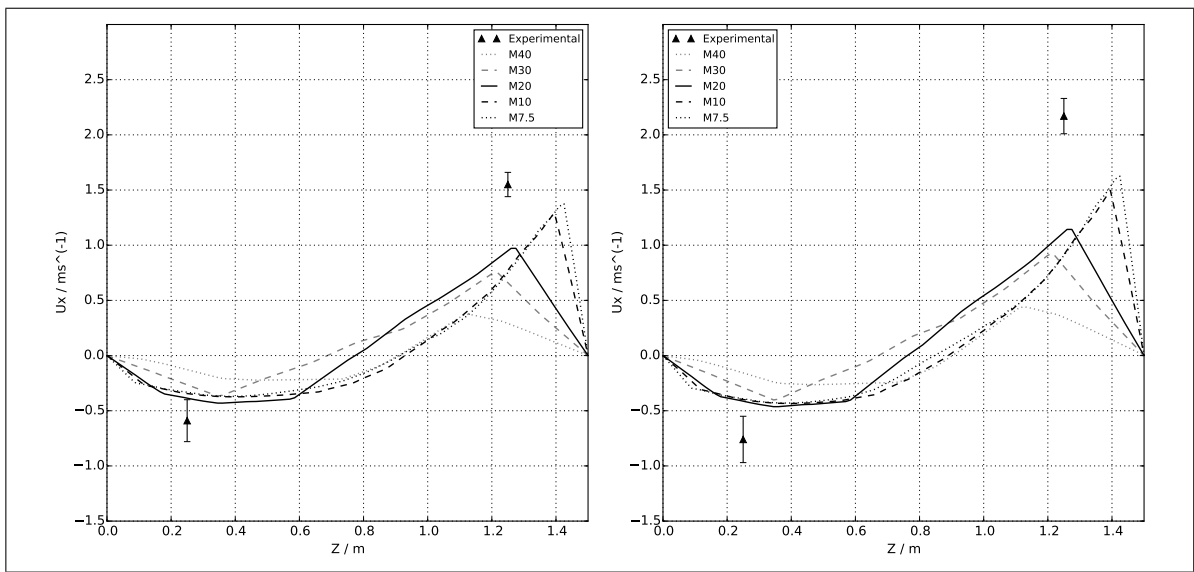


Figure 4.17. Time-averaged inflow (negative) and outflow (positive) velocities at openings for the 50% (left) and 74% (right) steady state HRR periods. Simulation results correspond to an average between openings 1, 8 and 15. Experimental results are averages according to Hidalgo et. al. (2017).

Figure 4.17 shows simulated velocity profiles at openings plotted against the average inflow and outflow velocities reported by J. Hidalgo et al. (2017). These values correspond to the averaged probe measurements at the 15 openings excluding anomalous data. Considerable underestimation of outflow velocities is seen, specially for the coarse cases M40, M30 and M20. As simulated outflow velocities increase when refining the mesh, with considerable changes even when refining from M10 to M7.5, no mesh-converged cases are accomplished. Regarding inflow velocities, convergence is achieved in the M30 case, with good

agreement for the 50% HRR_{max} scenario and slight underestimation for the 74% HRR_{max} case. Most probably, the tested cases do not provide enough mesh resolution to correctly capture hot gas escape dynamics, as small eddies adjacent to the hanger tip pass-through the LES-filter and are therefore approximately modeled rather than directly simulated. This is evidenced in the fact that smaller LES-filters —i.e. finer meshes— deliver better agreement with experimental data, specially near the hanger tip.

4.3.5. Transient Temperature Profiles: Central Tree

Results for the solver transient study are shown in Figure 4.18. As already discussed for Figure 4.15, the baseline case M20 does not show good agreement with measured thermocouple temperatures, specially for 1.4m-high thermocouples and above. Given the former, only transient behavior will be of interest in this sub-section. As temperature ramps in both experimental and simulated data occur in the same time intervals, heating times associated with thermocouple thermal inertia are well captured by the transient thermocouple model. This good agreement is specially seen in 1.6m-high thermocouples and below. Temperatures measured by the 1.95m-high thermocouple do not reach steady state conditions, an effect not captured by the simulated values which actually reach steady state temperatures during steady state HRR periods. As the 1.95m-high thermocouple is located in the roof boundary layer, its temperature is controlled by heat dissipation through the roof. As simulated values use the steady state heat transfer 'externalWallHeatFluxTemperature' boundary condition, which by construction cannot model the observed transient boundary layer, it is expected that the simulated boundary layer temperatures are not representative.

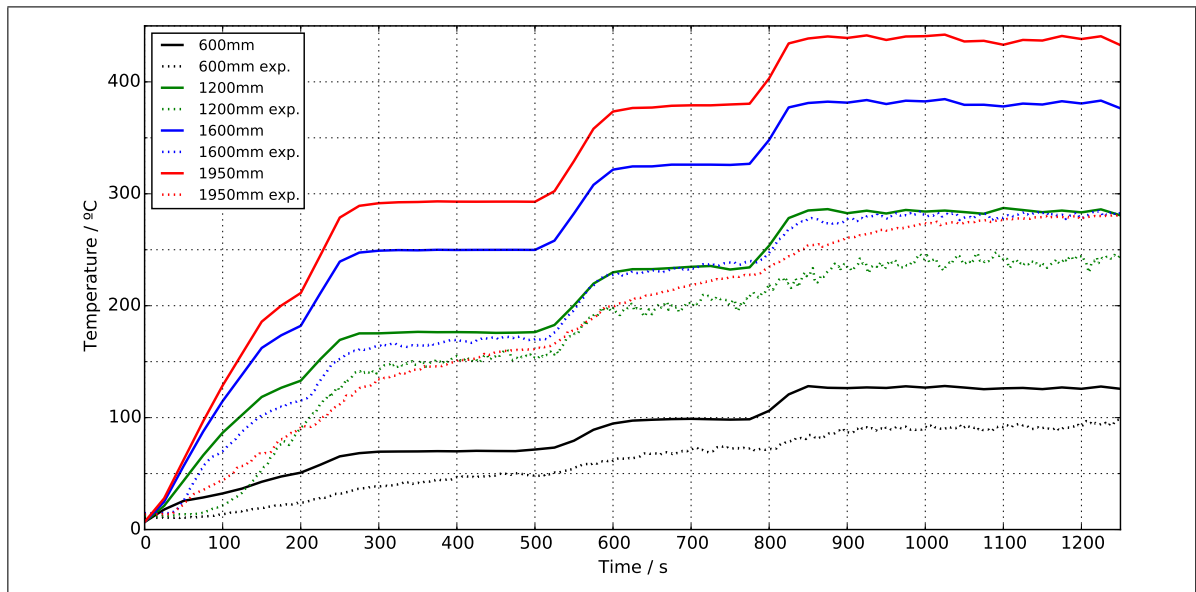


Figure 4.18. Transient thermocouple temperatures at four heights along the central tree D15. Begins at the opening of propane supply to the burners, and ends at the ending of the 99% HRR steady state period.

5. CHAPTER 5: DISCUSSION

In Chapter 1: Introduction, it is stated that the goal was to assess FireFOAM's capabilities of simulating real-scale fires, by evaluating if result quality is of appropriate precision for engineering fire hazard analysis purposes. As the line of work aims towards Dynamic Mesh Refinement using OpenFOAM's flexible capabilities, it was also part of the scope to test the solver's behavior in coarse meshes that involve real-time simulations. This chapter will therefore discuss FireFOAM's strengths and constraints regarding real-scale fire modeling quality, also considering associated runtime. Finally, the chapter is concluded with a discussion on simulation agreement with experimental data, considering the previously discussed strengths and constraints.

5.1. FireFOAM Strengths

5.1.1. Real-Time Computing Capabilities

Results show that FireFOAM can run coarse meshes such that computing is done in real time, and even faster. Within the tested cases, M30 and M40 could manage to run faster than real time, where M30 —consisting of 29k rectangular prism-shaped elements in a conform mesh— simulated 750 s of fire in 676 s of clock time. As results showed, even though both the M30 and M40 case were not mesh-converged in any case and their predictions did not show good agreement, flow patterns in streamline maps and temperature distribution in contours showed physical consistency. These cases captured both buoyancy effects of air entrainment from the lower opening and hot gas outflow from the upper opening. Temperature maps showed temperature stratification, with increasing temperature towards the roof where the hot gas layer was located. In the far-field —tested considering the central tree—, both M30 and M40 showed results that diverge no more than $\pm 50^{\circ}\text{C}$ from the finest M7.5 case. A discrepancy of this order can be simply because the temperature is averaged over a bigger cell, and not because the coarse mesh fails reproducing the physical phenomena. If

this is the case, one can conclude that no precision is lost when modeling the far field with coarse meshes, at least regarding thermodynamic properties (such as temperatures).

FireFOAM's parallel processing capabilities also show promise for reducing computing time. For the baseline M20 case with 144 solid angles in the FVDOM model, computing time was reduced from 10 000 s to 3 600 s —saving 1hr 46min— when sub-dividing the domain in four parts —two z-wise and two in y-wise— and assigning each part to a processor (4 processors in total). This was done through the 'decomposeParDict' script in the '/system' directory. Further increase to more parallel processors did not show runtime reductions, given that message passing protocols made the process slower when adding additional processors.

The amount of solid angles implemented in the FVDOM radiation model also showed mayor differences in runtime, therefore another good opportunity was detected when reducing runtime is key. Considering that for ETFT Test I the M20 case had a converged G -field with 16 solid angles, considerable time reduction is seen when passing from the most FVDOM demanding case (R400, 9 000 s of runtime) to the converged one (R16, 3130 s). Here, a time saving of 1h 40 min was accomplished considering same-quality results. Therefore, it vital to know the required number of solid angles for each situation in order to get fast and converged results.

5.1.2. Convergence Capabilities

FireFOAM presents both solid angle and mesh -convergence. As mentioned, solid angle-convergence for the M20 was achieved for 16 solid angles, considering the convergence of the G -field at the back wall as a sufficient criteria. Mesh convergence for internal flow fields and far-field temperatures was achieved for the M10 case. The presence of these convergence types proves the solver is internally consistent, given that past certain resolution the results do not considerably vary. Plus, simulated data tended to converge towards experimental measurements, therefore major errors are probably systematic —mainly solver input, such as gas properties, compartment thermal boundary layer properties, initial and boundary

conditions, among others—, and/or due to solving relaxations, admissible error parameters, and calculation error (interpolation, differentiation and integration). A major systematic discrepancy was caused by the uneven experimental propane distribution among burners, which caused experimental under-estimations of approximately 40°C in the smoke layer between Burner pairs 6-2, compared to an even propane distribution scenario.

The tested cases did not show convergence in two cases: near-flame region temperatures and compartment opening inflow/outflow velocities. In both situations, refinement from M10 to M7.5 showed considerable improvement in the results' agreement with experimental data, therefore finer meshes are needed in these sectors. This is reasonable, as in near-flame zones fire dynamics is dominated by heat release processes and buoyant acceleration, and near the openings velocity fields are controlled by turbulence near geometric discontinuities —e.g. near the hanger tip—. These phenomena are known to require higher mesh resolution, as the LES-filter has to be able to capture small eddies that control turbulent mixing and hence reaction rate in the near-flame zone, and be able to capture small eddies adjacent to the hanger tip as to correctly model outflow dynamics. This is why mesh refinement in flame regions and in eddy-inducing geometric discontinuities is a common practice in Combustion CFD. This fact justifies the tremendous advantage that Dynamic Mesh Refinement could bring in FireFOAM, where near-flame and geometric discontinuity local refinement promises to reduce computing time and deliver good quality results.

5.2. FireFOAM Limitations

5.2.1. Irregular-Tetrahedron Meshes

First observed, was the fact that the solver could not get a converged full-time result of the of the T20 case —irregular mesh of 20cm side tetrahedron elements—. This type of mesh construction is known to cause convergence problems in the Finite Volumes discretization concept, mainly because the method constantly evaluates fluxes through faces that are not parallel to the cell normal. This results in gradient corrections that may lead to larger numerical

errors. Approximation of gradients in each element and time-step, makes the solver prone to mayor computing time and divergence. This could be a problem if dynamic meshing is implemented with tetrahedral-refinement, as the time gained by the local refinement can be lost in solving complex tetrahedron geometries. A key consideration for dynamic local refinement is therefore to implement an algorithm that locally refines with volumes whose cell normals are parallel to the local flux.

5.2.2. Steady-State Boundary Conditions

As noted in Section 3.3.5. Transient Temperature Profiles: Central Tree, temperature measurements of the 1.95m-high thermocouple in the central tree never achieved steady state throughout the whole extension of Test I. This indicates that the roof boundary layer has a characteristic gas-phase heating time higher than the time extension of any steady state HRR periods. Therefore, a transient heat dissipation boundary condition is vital. As noted, all cases were run using the steady state 1D heat transfer boundary condition 'external-WallHeatFluxTemperature', which by construction cannot model the transient roof boundary layer observed though out the entire extension of ETFT Test I. This implies an even larger error, given that heat dissipation through the compartment walls represents between 15 and 20% of the total energy flux leaving the compartment (Maluk et al., 2017). An implementation and testing of the 1D transient diffusion equation for each patch would be key for addressing this limitation. This would in turn increase computing time, as the solid-phase wall has to be as well meshed and solved. Gas phase-solid phase interaction is a mayor research area in Fire Science (Ren et al., 2016).

Using a heat transfer steady state boundary condition in the transient boundary layer can also explain why energy conservation behaves so poorly (see Fig. 4.7). The problem extends to the M10 case, where in steady state HRR periods about 10-20% of the heat release rate is not accounted for. Simulated heat dissipation through walls represented less than 1% of the total heat release rate, far less than the experimental 15-20% estimated by Maluk et al. (2017). This previous experimental estimation probably fills the unaccounted 10-20%

gap observed in the M10 energy balance exercise, most probably caused by the deficient modeling of the boundary layer.

5.2.3. Sensitivity to Domain Construction

Results were found to be considerably sensitive to the secondary domain construction, which was constructed with the purpose of realistically capturing air entrainment into the compartment. It is important to note that air entrainment modeling is vital, as the air flux is not prescribed and therefore the solver has to establish it. Differences in entrainment air flux cause considerable changes in the solver solution, as air contains oxidant—which controls the spatial distribution of prescribed heat release rate—and inert nitrogen—which controls cooling of the compartment interior—.

As Figure 5.1 illustrates, three different secondary domain constructions were tested: one with air entrainment at compartment floor level (case A), another including the room floor with closed boundaries (case B), and another the same as the previous but with an open surface for entrainment from below the compartment (case C).

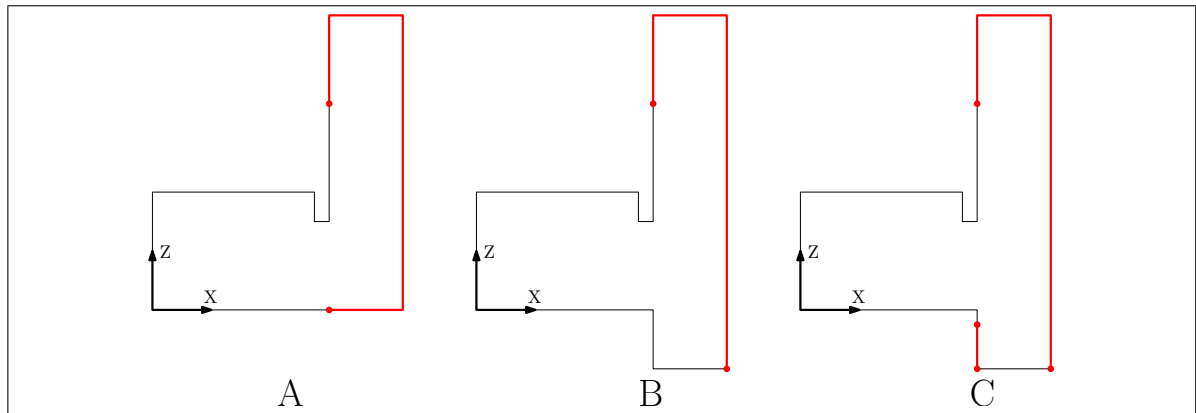


Figure 5.1. Compartment X-Z plane cut. Thin black lines represent wall boundaries, and thick red lines entrainment zones. These constructions are defined through the boundary condition set. Chosen cuts, for simplicity, do not include propane burners. Images not scaled to real dimensions.

As explained in Section 3.3. Numerical Setup, case C was chosen for this simulations as air entrainment from below the compartment showed to be considerable. Among these cases (A, B and C) displacements of the neutral plane were observed at the compartment opening plane, with magnitudes of 0-10cm, and changes in peak inflow/outflow velocities with magnitudes of 0-0.2 m/s. Case C showed considerable air entrainment from below the compartment, and as the experimental setup contains this opening (refer to Fig. 3.2) it was considered that it best represented the experimental conditions. This opening contained pillar-like structures along the y-span (refer again to Fig. 3.2) which were not included for lack of information about them.

As seen in Section 4.3.4 Opening Plane Temperature and Velocity Profiles, both simulated inlet and outlet mean velocities were underestimated. Inflow underestimations were in the order of 0.1 m/s (aprox. 10% of the mean experimental inflow velocity) and for outflow in the order of 0.6 m/s (aprox. 27% of the mean experimental outflow velocity). These errors can be attributed to the following fact: as air entrainment is considerably sensible to the computational domain and boundary conditions, the computational setup for ETFT Test I has to be represented more accurately.

5.3. Simulated Data Agreement with Experimental Measurements

Simulated results, in general, showed reasonable agreement with experimental data as they managed to capture magnitude orders, and in some cases even precisely reproduce some of the measured fields. Well reproduced phenomena were:

- Flow patterns at the compartment interior.
- Neutral plane height and pronounced temperature gradients at openings.
- Cold-air/hot-smoke interface height at the compartment interior.
- Temperatures in the cold air zone and in most of the hot-gas zone (below the roof thermal boundary layer).

Poorly reproduced phenomena were:

- Incident radiation in the hot-gas zone, over the 1.4m-high thermocouple bead.
- Temperatures at the top half of the smoke layer, specially inside the roof boundary layer.
- Mean inflow and outflow velocities at the compartment interior.
- Transient behavior of near-roof temperatures.

G -field over-estimations of around 100% for the M20 case are expected to be associated with coarse meshing, given the estimation improvement seen when refining to the M10 case. Simulated thermocouple temperatures at the top-half of the smoke layer were over-estimated mostly because FVDOM over-estimated the G -field in this zone. This is expected to originate from the under-estimation of compartment inflow and outflow velocities. As inflow is under-estimated, less inert nitrogen enters the compartment to cool down the hot smoke layer. In the other hand, as outflow is under-estimated, hot gases do not leave the compartment at the experimental rate, therefore huge accumulation of hot gases cause the ceiling radiation to increase. This is consistent, given the fact that finer meshes (M10 and M7.5) predicted higher outflow velocities, hence less intensive roof-level irradiation, and therefore simulated thermocouple temperatures closer to the experimental measurements. Given the above, part of the error can be eliminated by mesh refinement, as finer meshes capture better flow dynamics at the opening plane. Inaccuracies in the computational domain with respect to the real experimental conditions, can account for the systematical component of this error. Further systematical over-estimation of hot-gas layer temperatures (even in the M7.5 and M10 cases) is attributed to the poor reproduction of heat dissipation through walls, modeled by the stationary 'externalWallHeatFluxTemperature' boundary condition, which was established in an actually transient boundary layer. This layer remained transient during the whole extension of ETFT Test I, as measured by the 1.95m-high thermocouple at the central tree. Another systematic error in the estimation of smoke layer temperatures was the uneven propane distribution among burners, where the propane mass flow was weighted toward Burner 1a. This lowered the expected heat release rate among Burner pairs 6-2, and hence

measured hot-gas layer temperatures systematically reduced compared to an even-propane-distribution case (as simulated in FireFOAM). This experimental fall-back was estimated to be 40°C.

It is important to remember that when simulating real-scale fires like ETFT Test I, mesh resolution is limited as large domains intrinsically demand more cells. For example, an M5 case was planned but could not be run as the host computer could not handle the almost 4 million cells that composed this mesh. Up-to-date validation exercises typically consist of smaller domains, therefore smaller cells are viable. Vilfayeau et al. (2016) computed a 2m x 2m x 0.85m domain with local refinement of 2-12mm cells, getting nearly 1-2 million cells. Better results would have been obtained in the ETFT Test I simulations if more mesh resolution could have been afforded, specially regarding roof temperatures and velocity fields at openings.

6. CONCLUSIONS

The applied methodology allowed a detailed evaluation of FireFOAM's performance in simulating real-scale fires, accounting also for runtimes with different solver conditions. Conditions mainly consisted in different meshes —both in sizes and geometry of elements—, different FVDOM resolutions —controlled through the number of solid angles—, and different numbers of parallel processors for computing results. Both theory-based and experimentally-based techniques were implemented. Theory-based validation included mass balances, energy balances, and streamline maps, as to test the solver's internal consistency. Experimentally-based validation consisted in a comparison of the solver's results with the ETFT Test I (J. Hidalgo et al., 2017) measurements, testing variables such as temperature, incident radiation and velocity fields. It was found that a 10cm rectangular-prism mesh with 144 solid angles in FVDOM is sufficient to accurately capture main phenomena in full-scale compartment fires.

6.1. Relevant findings

Results and discussions suggest that the Finite Volumes method does not deliver conservative results, at least for the 20 cm and 10 cm cubic meshes. For the M10 case, mass and energy balances showed discrepancies of +7% and 0 to -20% regarding conservation, respectively. M20 experienced time-steps with discontinual solutions, therefore it was not considered appropriate for the mass-energy balance analysis. The simulated energy distribution showed poor results, because apart from reaching -20% of unaccounted energy flow, simulations captured almost no heat dissipation through walls (<1% of the Heat Release Rate), compared to the experimental value of 20% of the Heat Release Rate (Maluk et al., 2017). Reasonably good agreement of simulated data with experimental data, is attributed to the fact that the mayor energy dissipation fraction, corresponding enthalpy of hot gases leaving the compartment, is precisely simulated. Simulations indicated that in average 75% of the Heat Release Rate is dissipated as enthalpy of hot gases leaving the compartment, which is in good agreement with the experimental fraction of aprox. 70% reported by Maluk

et al. (2017). Also, supporting the previous conclusion, is the fact that the simulated 8% of radiation leaving through the openings is in good agreement with the predicted experimental 10% (Maluk et al., 2017). Both these similarities explain well reproduced phenomena such as flow patterns, neutral plane and cold-air/hot-gas interface heights, and temperatures in the cold-air zone and in the first half of the hot-gas zone. Temperature overestimations at the top-half of the hot-gas zone—from heights 1.4m to 2m—were attributed to inaccurate simulation of flow dynamics at the compartment opening, heat dissipation through the roof, and experimental uneven distribution of propane among burners—given that an even distribution was simulated—. Inaccurate simulation of opening plane flow dynamics—due to computational domain inaccuracies—yielded underestimations of the compartment ventilation rate, hence causing overestimations in the upper smoke-layer temperatures. Underestimations in the ventilation rate means less inert nitrogen entering the compartment and less smoke evacuation, resulting in less smoke-layer cooling and elevated irradiation near the roof.

Results suggested that the above errors are reduced, up to a certain point, as the mesh is refined. Main systematic errors—i.e. errors not reduced by mesh refinement—in the simulated upper-smoke layer are mainly attributed to both the implementation of a steady state heat dissipation boundary condition—given that there was an experimentally-observed transient boundary layer even in steady state heat release rate periods—, and inaccuracies when reproducing experimental conditions through the computational domain—where it was proven that results are considerably sensible to its construction—. The implementation of a steady state heat dissipation model can also explain poor energy conservation during transient periods, specially significant in the M20 case. This can also explain the discontinual mass and energy fluxes obtained in transient time steps for the M20 case. The above findings suggest that in real-scale fires, heat dissipation through walls is relevant—20% of the heat release rate according to Maluk et al. (2017)—and therefore a one-dimensional steady-state heat transfer boundary condition is not accurate enough. Another systematic error estimated as a 40°C fall-back in measured hot-layer temperatures, was the experimental propane flow weighting from Burner pairs 6-2 towards pair 1.

Runtime studies and results on the far-field (central tree), near-flame zone (tree B2) and openings suggest that implementation of dynamic meshing in FireFOAM could bring important benefits. Both M30 and M40 meshes —made from conform 30cm and 40cm cubic elements, respectively— run faster than real-time (see Figure 4.1). In the other hand, refinement in the far-field from M40 to M7.5 does not show simulation convergence towards experimental data—in average 25°C differences considering temperatures in the 100-400°C range—, while in the near-flame and openings zones mesh refining implies mesh-convergence towards experimental data with considerable improvements—up to 150°C in the smoke layer—. Considering the above, FireFOAM can reasonably reproduce the far-field with real-time running meshes, and at the same time is benefited by refinement in the near-flame zone and in the vicinity of geometric discontinuities such as the hanger tip. As this methodology is the principle of dynamic meshing, results support the idea of implementing it on FireFOAM in scenarios where the fire front moves.

6.2. Methodology limitations

The employed methodology can be improved by increasing its capacity of explaining FireFOAM limitations found within this work. For example, the implementation of the transient 1D thermal diffusion equation as an internal wall boundary condition could have verified/discarded some systematic error in both the energy balance and smoke layer modeling. To eliminate the systematic error due to experimental uneven propane distribution, an improved methodology could have incorporated this imbalance in the simulation's fuel inlet boundary conditions. Regarding the mass/energy balance studies, part of the discrepancy was attributed to both poor solver sensitivity and incorporation of models into the conservative form of the Navier-Stokes system, but no tangible reason was found. Including into the methodology a prior revision of solving relaxations, sensitivity parameters, and effects of sub-grid scale modeling on energy conservation could have been essential to address the non-conservative results. On the other hand, the use of finer meshes could have been key for verifying mesh convergence in the flame zone and near the openings. Literature has shown

that typical mesh sizes in the near-flame region for fire validation exercises are in the 1mm-15mm range, as indicated by Ren et al. (2016), Vilfayeau et al. (2016), and Y. Wang et al. (2011), among others. Therefore, implementing smaller mesh sizes in the near-flame zone could have given a tangible value of the convergence resolution for real-scale fires, which is valuable input data for Dynamic Mesh Refinement algorithms. It is important to note that this improvement would require access to high performance clusters for computing, as the 17.8m x 4.9m x 2m compartment is a computationally intensive domain for fine meshes. Finally, a mesh-specific study involving the necessary solid angles for G -field convergence could provide valuable insight for future work. Knowing the minimum number of solid angles required for each mesh size promises to deliver reliable results with considerable saved runtime, as it was found that an increasing number of solid angles brought considerable increases in runtime. A precise estimation of the G -field is important, as it plays a mayor role in the thermocouple correction model.

6.3. Future work

Shortening execution time (also referred as runtime, or computing time) in fire simulations is of essence in the fire hazard analysis field. In this work, FireFOAM has proven to have exceptional capabilities for reducing runtime, consisting in both parallel processing and mesh refinement. Given this, proposed future work centers in implementing Dynamic Mesh Refinement in FireFOAM when simulating fire scenarios with flame spread/extinction. Valuable insight was obtained regarding FireFOAM's runtime versus result-quality trade-off. The solver's main strengths and constrains were identified, providing guidelines for future investigators in this line of work. Recommendations for future work include:

- A revision of the effects of sub-grid scale models (e.g. one-equation eddy viscosity model) on energy conservation, as in these exercises the Finite Volumes method was not found to be conservative, with 10-20% errors.
- Along with the previous point, perform a revision of the solver's sensitivity parameters for solving conservation equations.

- Implementation of a transient heat transfer boundary condition, rather than the steady state one used herein.
- Perform mesh refinement with elements whose cell-normals are parallel to the local flow, as it was found that for ETFT Test I irregular tetrahedrons increased runtime and induced divergence when implemented in the whole domain. Given the rectangular-prism geometry of the ETFT compartment, the ideal elements for this case were found to be rectangular prisms with cell normals parallel to the Cartesian planes.
- Refine using mesh resolutions on the 1-15mm range, as it is common practice in literature. Supporting the above statement, it was seen how the M7.5 case (75 mm) was not mesh-convergent for near-flame zones and in the vicinity of geometric discontinuities.
- Following the previous recommendation, perform mesh refinement in the near-flame zone and near geometric discontinuities, and mesh coarsening in the far-field and flame-extinction zones.

Finally, as meshing times are negligible compared to computing times, Dynamic Mesh Refinement again proves to be an excellent opportunity for reducing computing times while obtaining good result quality. As OpenFOAM counts with a solver capable of computing irregular-meshed domains and subdividing these domains for parallel processing, future work in this line of investigation has to take advantage of these two capabilities. Most commercial solvers do not provide these levels of flexibility, which were herein demonstrated to provide considerable time saving.

REFERENCES

- Almeida, Y. P., Lage, P. L., & Silva, L. F. L. (2015). Large eddy simulation of a turbulent diffusion flame including thermal radiation heat transfer. *Applied Thermal Engineering*, 81, 412–425.
- Bauwens, C., & Dorofeev, S. (2014). Cfd modeling and consequence analysis of an accidental hydrogen release in a large scale facility. *International Journal of Hydrogen Energy*, 39(35), 20447–20454.
- Geuzaine, C., & Remacle, J.-F. (2009). Gmsh: A 3-D finite element mesh generator with built-in pre- and post-processing facilities. *International Journal for Numerical Methods in Engineering*, 79, 1309 – 1331.
- Hidalgo, J., Cowlard, A., Abecassis-Empis, C., Maluk, C., Majdalani, A., Kahrmann, S., . . . Torero, J. (2017). An experimental study of full-scale open floor plan enclosure fires. *Fire Safety Journal*, 89, 22–40.
- Hidalgo, J. P., Maluk, C., Cowlard, A., Abecassis-Empis, C., Krajcovic, M., & Torero, J. L. (2017). A thin skin calorimeter (tsc) for quantifying irradiation during large-scale fire testing. *International Journal of Thermal Sciences*, 112, 383–394.
- Magnussen, B. F., & Hjertager, B. H. (1977). On mathematical modeling of turbulent combustion with special emphasis on soot formation and combustion. In *Symposium (international) on combustion* (Vol. 16, pp. 719–729).
- Maluk, C., Linnan, B., Wong, A., Hidalgo, J. P., Torero, J. L., Abecassis-Empis, C., & Cowlard, A. (2017). Energy distribution analysis in full-scale open floor plan enclosure fires. *Fire Safety Journal*.
- Maragkos, G., Rauwoens, P., & Merci, B. (2012). Application of fds and firefoam in large

eddy simulations of a turbulent buoyant helium plume. *Combustion Science and Technology*, 184(7-8), 1108–1120.

McCaffrey, B. (1979). Purely buoyant diffusion flames: Some experimental results. final report. *Chemical and Physical Processes in Combustion. The National Institute of Standards and Technology (NIST), Miami Beach*, 49.

Mishra, S. C., & Roy, H. K. (2007). Solving transient conduction and radiation heat transfer problems using the lattice boltzmann method and the finite volume method. *Journal of Computational Physics*, 223(1), 89–107.

OpenCFD. (2007, 11 April). OpenFOAM - The Open Source CFD Toolbox - User's Guide (1.4 ed.) [Computer software manual]. United Kingdom.

Ren, N., de Vries, J., Chaos, M., & Yi, W. (2015). Firefoam modeling of standard class 2 commodity rack storage fires. *Conference Paper*.

Ren, N., de Vries, J., Zhou, X., Chaos, M., Meredith, K. V., & Wang, Y. (2017). Large-scale fire suppression modeling of corrugated cardboard boxes on wood pallets in rack-storage configurations. *Fire Safety Journal*, 91, 695–704.

Ren, N., Wang, Y., Vilfayeau, S., & Trouvé, A. (2016). Large eddy simulation of turbulent vertical wall fires supplied with gaseous fuel through porous burners. *Combustion and Flame*, 169, 194–208.

Runefors, M., Anderson, J., Wahlqvist, J., Huang, C., & Husted, B. (2016). A comparison of radiative transfer models in firefoam and fds. In *14th international conference and exhibition on fire science and engineering (interflam 2016), windsor, uk 4-6 july 2016*.

Torero, J. L., Majdalani, A. H., Abecassis-Empis, C., & Cowlard, A. (2014). Revisiting the compartment fire. *Fire Safety Science*, 11, 28–45.

Trouvé, A., & Wang, Y. (2010). Large eddy simulation of compartment fires. *International*

Journal of Computational Fluid Dynamics, 24(10), 449–466.

Vilfayeau, S. (2015). *Large eddy simulation of fire extinction phenomena* (Unpublished doctoral dissertation). University of Maryland.

Vilfayeau, S., White, J., Sunderland, P., Marshall, A., & Trouvé, A. (2016). Large eddy simulation of flame extinction in a turbulent line fire exposed to air-nitrogen co-flow. *Fire Safety Journal*, 86, 16–31.

Wang, C., Wen, J. X., & Chen, Z. (2014). Simulation of large-scale lng pool fires using firefoam. *Combustion Science and Technology*, 186(10-11), 1632–1649.

Wang, Y., Chatterjee, P., & de Ris, J. L. (2011). Large eddy simulation of fire plumes. *Proceedings of the Combustion Institute*, 33(2), 2473–2480.

Welch, S., Jowsey, A., Deeny, S., Morgan, R., & Torero, J. L. (2007). Bre large compartment fire tests characterising post-flashover fires for model validation. *Fire Safety Journal*, 42(8), 548–567.

APPENDIX

A. ELABORATING AN OPENFOAM CASE USING GMSH

This Annex has the purpose of providing documentation about the whole process of elaborating the FireFOAM tested cases. The following documentation contains the steps from elaborating the mesh in Gmsh (Geuzaine & Remacle, 2009), to post-processing OpenFOAM results in ParaView. Ubuntu was used as the preferred OS, as it supports Gmsh, OpenFOAM and ParaView. Posterior post-processing, i.e. plotting relevant data from the '.csv'¹ files after they are extracted from ParaView, is done through Python. Specific Python data managing is out of this Annex scope, as the analyst can use a software of its preference (e.g. Mathematica, MatLab, among others).

A.1. Case Setup

Before constructing the mesh, it is recommended to set the case directory as to save the mesh. An easy way to start familiarizing with the software is to grab an OpenFOAM Tutorial case, delete the contents in the 'constant/polMesh' directory, and start from section A.2. Creating the Mesh. OpenFOAM case directories consist of tree folders: '0', 'system' and 'constant'. The '0' directory contains a set of scripts, each one corresponding to the boundary and initial conditions of each physical quantity. The 'system' directory contains the solver parameters, as temporal scheme in the 'controlDict' script, parallel processing decomposition structure in the 'decomposeParDict' script, and Finite Volumes solving parameters and relaxations in the 'fvSchemes' and 'fvSolutions' scripts. The 'createPatchDict' and 'topoSetDict' are used when creating the mesh. Finally, the 'constant' directory contains physical parameters, such as the FVDOM number of solid angles, chemical reactions, thermodynamic properties, and LES constants among others. For more information on the meaning of each script, the reader is directed to the OpenFOAM User Guide (OpenCFD, 2007) and/or the Lid-driven cavity flow tutorial².

¹A '.csv' corresponds to a Comma Separated Value file, which contains rows of information, with each row with data values separated by commas. Excel considers these commas as a cell space, therefore a '.csv' file is equivalent to an Excel table.

²<https://cfd.direct/openfoam/user-guide/cavity/>

A.2. Creating the Mesh

Both the geometry and the mesh are constructed in Gmsh, which can be downloaded for free from the developer Geuzaine and Remacle (2009) website³, which also contains a Reference Manual. In the Ubuntu version of the software, the Gmsh executable is contained inside the '/bin' directory inside the installation folder. When the 'gmsh' executable is opened, a graphic interface will appear, and here lies the option of opening the '.geo' file containing the geometry. These files contain the domain information necessary for the software to reproduce it. Domains in Gmsh are constructed with the Point as a basic element. Points are then connected together through a Line, and multiple Lines are defined in an array named the Line Loop, which is later to be defined as a Plane Surface. Line Loops can be defined with Lines in a clock-wise or counter-clock-wise orientation, which determines the Plane's normal direction. Plane Surfaces are finally extruded into Volumes. Figures A.1, A.2 and A.3 show the '.geo' script of the simulated domain's XZ plane, later to be extruded. Figure A.4 shows the geometry generated by the shown script. The script can be written line-by-line, or using the Gmsh graphic interphase. For this work, it was written line-by-line as it provides more flexibility despite of being a slower method. When writing line-by-line, one has to make sure that the first line of the '.geo' file contains the '//+' characters (see Figure A.1).

```
//+
gridsize=0.20;
wRoof=gridsize; //gridsizeOr0.1
lx=4.9-wRoof;
ly=17.8;
lz=2;
lQuem=0.5;
hQuem=0.35;
hRoof=0.5;
hStage=1;
hFloor=0.2; //gridsizeOr0.2
hFacade=2;
zDom=4;
xDom=2;
deltaNode=0.25;
```

Figure A.1. Characteristic domain and mesh dimensions defined in Gmsh.

³<http://gmsh.info/>

```

//C1
Point(1) = {0,0,lz-hRoof,gridsize};
Point(2) = {0,0,lz,gridsize};
Point(3) = {1.4-lQuem/2,0,lz,gridsize};
Point(4) = {1.4-lQuem/2,0,lz-hRoof,gridsize};
Line(5) = {1,2};
Line(6) = {2,3};
Line(7) = {3,4};
Line(8) = {4,1};
Line Loop(9) = {5,6,7,8};
Plane Surface(10) = {9};
Transfinite Line{5,7}=hRoof/gridsize+1+deltaNode;
Transfinite Line{6,8}=(1.4-lQuem/2)/gridsize+1+deltaNode;
Transfinite Surface{10};
Recombine Surface{10};

//C2
Point(11) = {1.4+lQuem/2,0,lz,gridsize};
Point(12) = {1.4+lQuem/2,0,lz-hRoof,gridsize};
Line(13) = {3,11};
Line(14) = {11,12};
Line(15) = {12,4};
Line Loop(16) = {-7,13,14,15};
Plane Surface(17) = {16};
Transfinite Line{14}=hRoof/gridsize+1+deltaNode;
Transfinite Line{13,15}=lQuem/gridsize+1+deltaNode;
Transfinite Surface{17};
Recombine Surface{17};

//C3
Point(18) = {1.4+2-lQuem/2,0,lz,gridsize};
Point(19) = {1.4+2-lQuem/2,0,lz-hRoof,gridsize};
Line(20) = {11,18};
Line(21) = {18,19};
Line(22) = {19,12};
Line Loop(23) = {-14,20,21,22};
Plane Surface(24) = {23};
Transfinite Line{21}=hRoof/gridsize+1+deltaNode;
Transfinite Line{20,22}=(2-lQuem/2)/gridsize+1+deltaNode;
Transfinite Surface{24};
Recombine Surface{24};

//C4
Point(25) = {1.4+2+lQuem/2,0,lz,gridsize};
Point(26) = {1.4+2+lQuem/2,0,lz-hRoof,gridsize};
Line(27) = {18,25};
Line(28) = {25,26};
Line(29) = {26,19};
Line Loop(30) = {-21,27,28,29};
Plane Surface(31) = {30};
Transfinite Line{28}=hRoof/gridsize+1+deltaNode;
Transfinite Line{27,29}=lQuem/gridsize+1+deltaNode;
Transfinite Surface{31};
Recombine Surface{31};

//C5
Point(32) = {lx,0,lz,gridsize};
Point(33) = {lx,0,lz-hRoof,gridsize};
Line(34) = {25,32};
Line(35) = {32,33};
Line(36) = {33,26};
Line Loop(37) = {-28,34,35,36};
Plane Surface(38) = {37};
Transfinite Line{35}=hRoof/gridsize+1+deltaNode;
Transfinite Line{34,36}=(lx-1.4-2-lQuem/2)/gridsize+1+deltaNode;
Transfinite Surface{38};
Recombine Surface{38};

//C6
Point(39) = {0,0,hQuem,gridsize};
Point(40) = {1.4-lQuem/2,0,hQuem,gridsize};
Line(41) = {39,1};
Line(42) = {4,40};
Line(43) = {40,39};
Line Loop(44) = {41,-8,42,43};
Plane Surface(45) = {44};
Transfinite Line{41,42}=(1.5-hQuem)/gridsize+1+deltaNode;
Transfinite Line{43}=(1.4-lQuem/2)/gridsize+1+deltaNode;
Transfinite Surface{45};
Recombine Surface{45};

//C7
Point(46) = {1.4+lQuem/2,0,hQuem,gridsize};
Line(47) = {12,46};
Line(48) = {46,40};
Line Loop(49) = {-42,-15,47,48};
Plane Surface(50) = {49};
Transfinite Line{47}=(1.5-hQuem)/gridsize+1+deltaNode;
Transfinite Line{48}=lQuem/gridsize+1+deltaNode;
Transfinite Surface{50};
Recombine Surface{50};

//C8
Point(51) = {1.4+2-lQuem/2,0,hQuem,gridsize};
Line(52) = {19,51};
Line(53) = {51,46};
Line Loop(54) = {-47,-22,52,53};
Plane Surface(55) = {54};
Transfinite Line{52}=(1.5-hQuem)/gridsize+1+deltaNode;
Transfinite Line{53}=(2-lQuem/2)/gridsize+1+deltaNode;
Transfinite Surface{55};
Recombine Surface{55};

//C9
Point(56) = {1.4+2+lQuem/2,0,hQuem,gridsize};
Line(57) = {26,56};
Line(58) = {56,51};
Line Loop(59) = {-52,-29,57,58};
Plane Surface(60) = {59};
Transfinite Line{57}=(1.5-hQuem)/gridsize+1+deltaNode;
Transfinite Line{58}=lQuem/gridsize+1+deltaNode;
Transfinite Surface{60};
Recombine Surface{60};

//C10
Point(61) = {lx,0,hQuem,gridsize};
Line(62) = {33,61};
Line(63) = {61,56};
Line Loop(64) = {-57,-36,62,63};
Plane Surface(65) = {64};
Transfinite Line{62}=(1.5-hQuem)/gridsize+1+deltaNode;
Transfinite Line{63}=(lx-1.4-2-lQuem/2)/gridsize+1+deltaNode;
Transfinite Surface{65};
Recombine Surface{65};

//C11
Point(62) = {0,0,0,gridsize};
Point(63) = {1.4-lQuem/2,0,0,gridsize};
Line(64) = {62,39};
Line(65) = {40,63};
Line(66) = {63,62};
Line Loop(67) = {64,-43,65,66};
Plane Surface(68) = {67};
Transfinite Line{64,65}=hQuem/gridsize+1+deltaNode;
Transfinite Line{66}=(1.4-lQuem/2)/gridsize+1+deltaNode;
Transfinite Surface{68};
Recombine Surface{68};

//C12
Point(69) = {1.4+lQuem/2,0,0,gridsize};
Line(70) = {46,69};
Line(71) = {69,63};
Line Loop(72) = {-65,-48,70,71};
Plane Surface(73) = {72};
Transfinite Line{70}=hQuem/gridsize+1+deltaNode;
Transfinite Line{71}=lQuem/gridsize+1+deltaNode;
Transfinite Surface{73};
Recombine Surface{73};

```

Figure A.2. Geometry and mesh definition. Part I.

```

//C13
Point(74) = {1.4+2-lQuem/2,0,0,gridsize};
Line(75) = {51,74};
Line(76) = {74,69};
Line Loop(77) = {-70,-53,75,76};
Plane Surface(78) = {77};
Transfinite Line(75)=hQuem/gridsize+1+deltaNode;
Transfinite Line(76)=(2-lQuem/2*2)/gridsize+1+deltaNode;
Transfinite Surface(78);
Recombine Surface(78);

//C14
Point(79) = {1.4+2+lQuem/2,0,0,gridsize};
Line(80) = {56,79};
Line(81) = {79,74};
Line Loop(82) = {-75,-58,80,81};
Plane Surface(83) = {82};
Transfinite Line(80)=hQuem/gridsize+1+deltaNode;
Transfinite Line(81)=lQuem/gridsize+1+deltaNode;
Transfinite Surface(83);
Recombine Surface(83);

//C15
Point(84) = {lx,0,0,gridsize};
Line(85) = {61,84};
Line(86) = {84,79};
Line Loop(87) = {-80,-63,85,86};
Plane Surface(88) = {87};
Transfinite Line(85)=hQuem/gridsize+1+deltaNode;
Transfinite Line(86)=(lx-1.4-2-lQuem/2)/gridsize+1+deltaNode;
Transfinite Surface(88);
Recombine Surface(88);

//C16
Point(89) = {lx+wRoof,0,lz-hRoof,gridsize};
Point(90) = {lx+wRoof,0,hQuem,gridsize};
Line(91) = {33,89};
Line(92) = {89,90};
Line(93) = {90,61};
Line Loop(94) = {-62,91,92,93};
Plane Surface(95) = {94};
Transfinite Line(92)=(1.5-hQuem)/gridsize+1+deltaNode;
Transfinite Line(91,93)=wRoof/gridsize+1+deltaNode;
Transfinite Surface(95);
Recombine Surface(95);

//C17
Point(96) = {lx+wRoof,0,0,gridsize};
Line(97) = {90,96};
Line(98) = {96,84};
Line Loop(99) = {-85,-93,97,98};
Plane Surface(100) = {99};
Transfinite Line(97)=hQuem/gridsize+1+deltaNode;
Transfinite Line(98)=wRoof/gridsize+1+deltaNode;
Transfinite Surface(100);
Recombine Surface(100);

//C18
Point(101) = {lx+wRoof+xDom,0,0,gridsize};
Point(102) = {lx+wRoof+xDom,0,hQuem,gridsize};
Line(103) = {90,102};
Line(104) = {102,101};
Line(105) = {101,96};
Line Loop(106) = {-97,103,104,105};
Plane Surface(107) = {106};
Transfinite Line(104)=hQuem/gridsize+1+deltaNode;
Transfinite Line(103,105)=xDom/gridsize+1+deltaNode;
Transfinite Surface(107);
Recombine Surface(107);

//C19
Point(108) = {lx+wRoof+xDom,0,lz-hRoof,gridsize};
Line(109) = {89,108};
Line(110) = {108,102};
Line Loop(111) = {-92,109,110,-103};
Plane Surface(112) = {111};
Transfinite Line(110)=(1.5-hQuem)/gridsize+1+deltaNode;
Transfinite Line(109)=xDom/gridsize+1+deltaNode;
Transfinite Surface(112);
Recombine Surface(112);

//C20
Point(113) = {lx+wRoof,0,lz,gridsize};
Point(114) = {lx+wRoof+xDom,0,lz,gridsize};
Line(115) = {89,113};
Line(116) = {113,114};
Line(117) = {114,108};
Line Loop(118) = {115,116,117,-109};
Plane Surface(119) = {118};
Transfinite Line(115,117)=hRoof/gridsize+1+deltaNode;
Transfinite Line(116)=xDom/gridsize+1+deltaNode;
Transfinite Surface(119);
Recombine Surface(119);

//C21
Point(120) = {lx+wRoof,0,lz+hFacade,gridsize};
Point(121) = {lx+wRoof+xDom,0,lz+hFacade,gridsize};
Line(122) = {113,120};
Line(123) = {120,121};
Line(124) = {121,114};
Line Loop(125) = {122,123,124,-116};
Plane Surface(126) = {125};
Transfinite Line(122,124)=hFacade/gridsize+1+deltaNode;
Transfinite Line(123)=xDom/gridsize+1+deltaNode;
Transfinite Surface(126);
Recombine Surface(126);

//C22
Point(127) = {lx+wRoof,0,-hFloor,gridsize};
Point(128) = {lx+wRoof+xDom,0,-hFloor,gridsize};
Line(129) = {96,127};
Line(130) = {127,128};
Line(131) = {128,101};
Line Loop(132) = {-129,-130,-131,-105};
Plane Surface(133) = {132};
Transfinite Line(129,131)=hFloor/gridsize+1+deltaNode;
Transfinite Line(130) = xDom/gridsize+1+deltaNode;
Transfinite Surface(133);
Recombine Surface(133);

//C23
Point(134) = {lx+wRoof,0,-hStage,gridsize};
Point(135) = {lx+wRoof+xDom,0,-hStage,gridsize};
Line(136) = {127,134};
Line(137) = {134,135};
Line(138) = {135,128};
Line Loop(139) = {-136,130,-138,-137};
Plane Surface(140) = {139};
Transfinite Line(136,138)=(hStage-hFloor)/gridsize+1+deltaNode;
Transfinite Line(137)=xDom/gridsize+1+deltaNode;
Transfinite Surface(140);
Recombine Surface(140);

//C24
Point(141) = {lx+wRoof,0,lz+zDom,gridsize};
Point(142) = {lx+wRoof+xDom,0,lz+zDom,gridsize};
Line(143) = {120,141};
Line(144) = {141,142};
Line(145) = {142,121};
Line Loop(146) = {143,144,145,-123};
Plane Surface(147) = {146};
Transfinite Line(143,145)=(zDom-hFacade)/gridsize+1+deltaNode;
Transfinite Line(144)=xDom/gridsize+1+deltaNode;
Transfinite Surface(147);
Recombine Surface(147);

```

Figure A.3. Geometry and mesh definition. Part II.

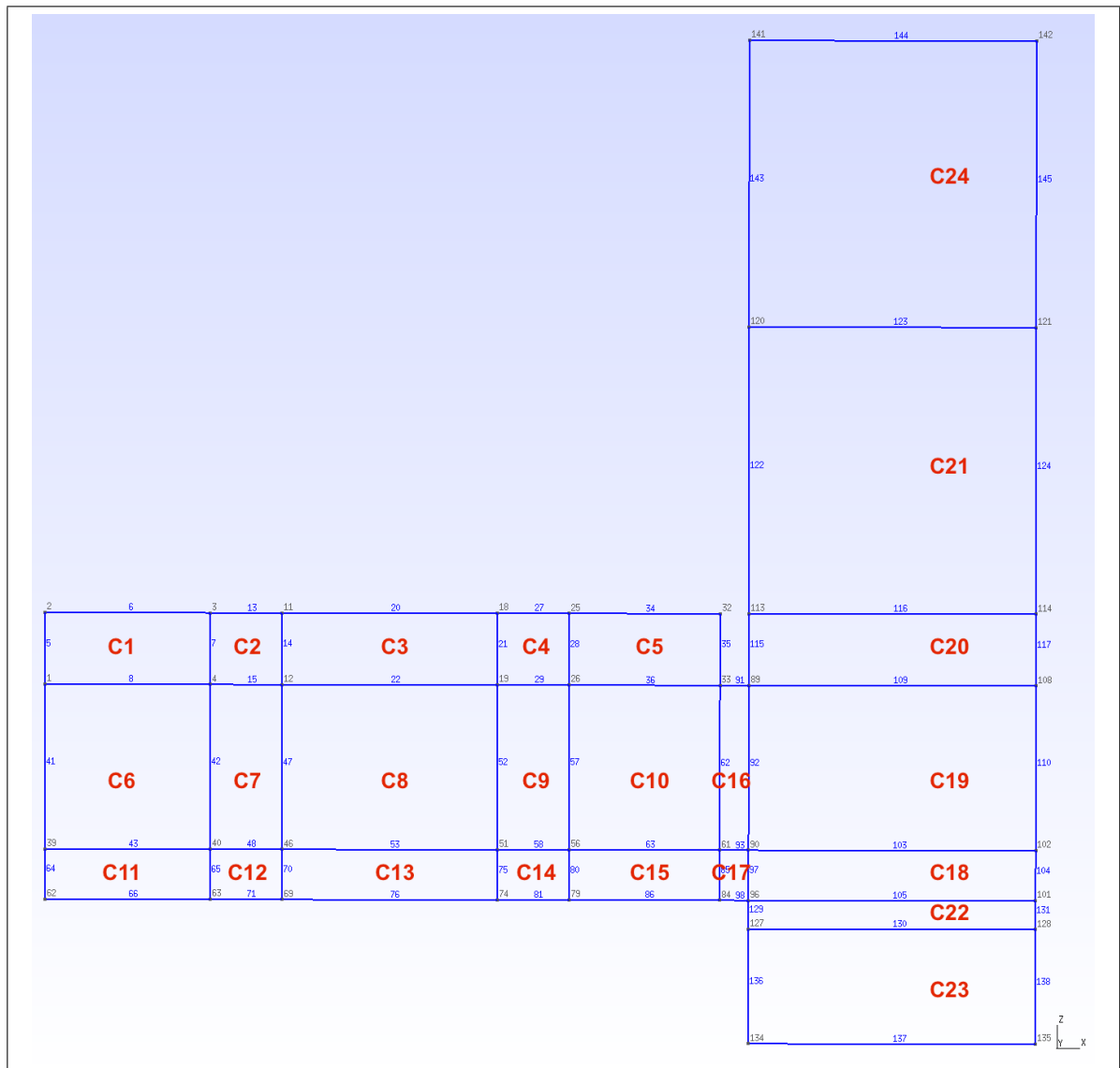


Figure A.4. X-Z view of the Domain, with Lines and Points numerated using functions located in the Tools-Options-Geometry Menu. Lines and Points can also be viewed in the graphic interphase by locating the cursor on top. C12 and C14 dimensions are those of the burners.

Each 'C' face corresponds to a Plane Surface. Line labels can be activated in the Tools-Options-Geometry Menu, facilitating the definition of Line Loops. The geometry was constructed this way—with various plane surfaces—as to generate a conform mesh constructed

by rectangular prisms, given that Gmsh by default meshes with irregular tetrahedrons. Tetrahedrons are not preferred when solving with the Finite Volumes Method, as they increase computing time, decrease precision and induce numerical instabilities. Note that all Plane Surfaces 'C' share the same sides —i.e. there is no Plane Surface with a side of different dimension compared to its neighbour—: this is done for the Extrusion to correctly work, as if neighbor sides are of different dimensions, the Extrusion algorithm will create two separate surfaces from each side. This creates conflicts when later defining boundary conditions.

Said the above, note from the script (Figures A.2 and A.3) that after the Plane Surface is defined, a Transfinite Line algorithm is applied over each line contained in the Plane Surface. This is done to control the number of mesh nodes in each line: the algorithm picks the lines inside the '{}' brackets, and meshes them with a number of nodes equal to the truncated value of the right-hand-side after the equivalence '='. The rule for the number of nodes in each line, shown in the script, is explained in Chapter 3: Methodology. Progressions for local mesh refinement can be used in this 1D algorithm, but as they are outside the scope of this Annex, the reader is redirected to the Gmsh Reference Manual in case interested.

After applying the Transfinite Line 1D meshing algorithm over each Line contained in the Plane Surface, the Transfinite Surface and Recombine 2D algorithms have to be applied on the Plane Surface. This combination generates a structured rectangular mesh over each surface, as the algorithm connects the nodes generated along the Transfinite Lines. Once these meshing rules are established in the '.geo file', executing the 'Mesh 2D' function — located in the left side hierarchic menu— generates the desired mesh (later to be extruded), as indicated in Figure A.5.

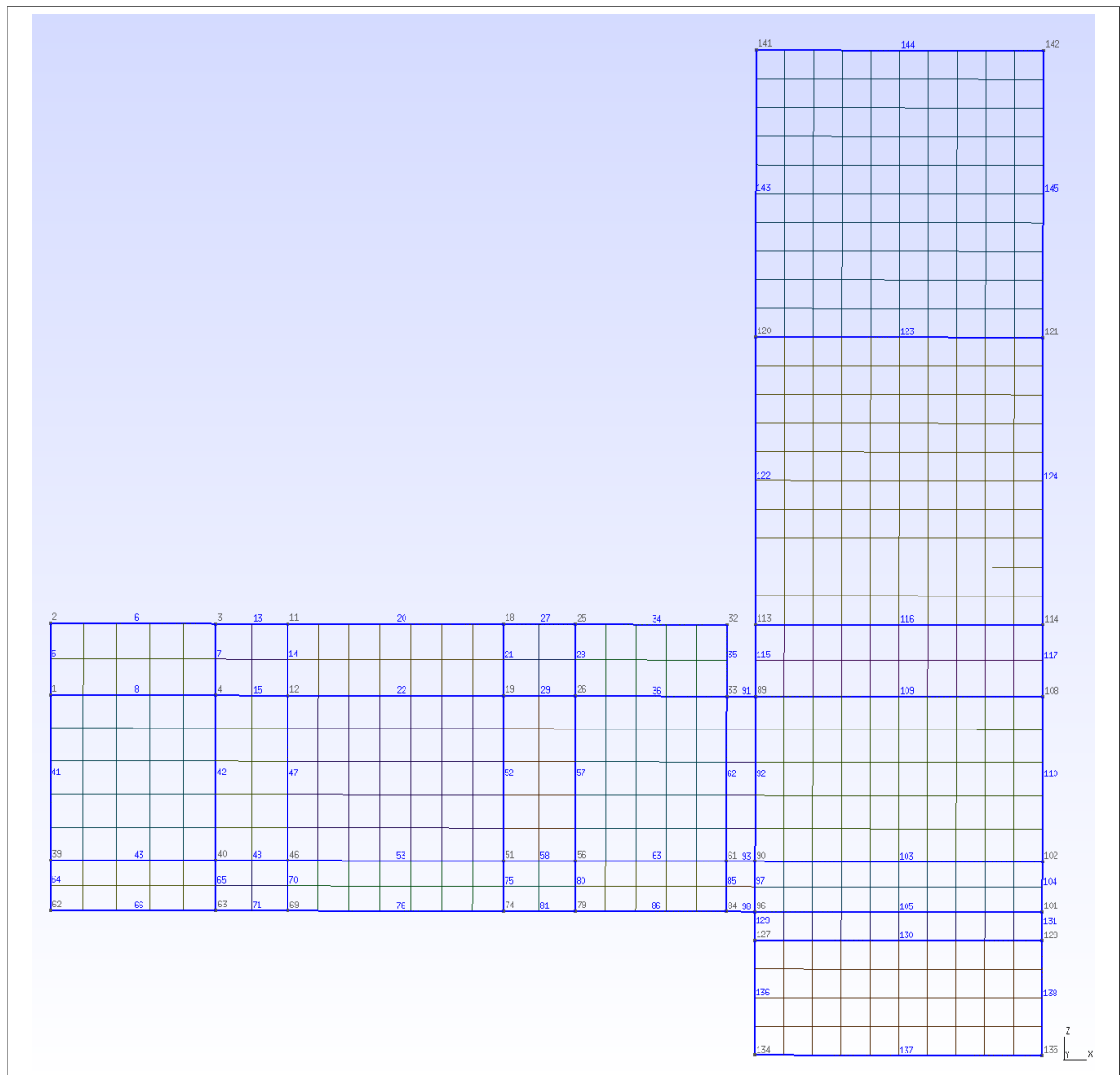


Figure A.5. X-Z plane mesh, constructed by regular quadrangles. Mean grid size: 20cm.

The Extrusion has to be made in a sequence, as it is desired that the interior of the burners are not meshed. Each X-Z layer of Plane Surfaces are Extruded in the Y-direction a number of cell-layers equal to the truncated value of the Extrusion length divided in the desired grid size. The Plane Surface labels —for performing the extrusion— can be set as visible in the Tools-Options-Geometry Menu. The extrusions are done in such way that, when the extruded

layer contains burner side faces, the extrusion is not done on those particular faces, leaving the burner interior unmeshed. Figures A.6, A.7 and A.8 show the extrusion script and the complete extruded domain. Note that after each burner extrusion, the burner inferior Line has to be defined as to redefine the whole burner side face as a Plane Surface. This Plane Surface is lost when the burner side face was not included as an extruded face. Over this redefined Plane Surface, the respective Transfinite Line, Transfinite Surface and Recombine Surface algorithms have to be applied.

```

//Ext1
newEntities1[] =
Extrude{0,(1.4-lQuem/2),0}
{
    Surface{140,133,68,73,78,83,88,100,107,45,51}
    Layers{(1.4-lQuem/2)/gridsize};
    Recombine;
};

//Ext1Quem
newEntities1q[] =
Extrude{0,lQuem,0}
{
    Surface{169,191,213,257,301,323,345,367,389}
    Layers{lQuem/gridsize};
    Recombine;
};

//L6a
Line(2000) = {337,336};
Line Loop(2001) = {-723,-856,-743,2000};
Plane Surface(2002) = {2001};
Transfinite Line(2000)=lQuem/gridsize+1+deltaNode;
Transfinite Surface(2002);
Recombine Surface{2002};

//L6b
Line(2003) = {347,346};
Line Loop(2004) = {-745,-900,-765,2003};
Plane Surface(2005) = {2004};
Transfinite Line(2003)=lQuem/gridsize+1+deltaNode;
Transfinite Surface(2005);
Recombine Surface{2005};

//Ext2
newEntities2[] =
Extrude{0,3-lQuem,0}
{
    Surface
{697,719,741,2002,763,2005,785,807,829,851,873,895,}
    Layers{(3-lQuem)/gridsize};
    Recombine;
};

//Ext2Quem
newEntities2q[] =
Extrude{0,lQuem,0}
{
    Surface
{2027,2049,2071,2115,2159,2181,2203,2225,2247,2269,}
    Layers{lQuem/gridsize};
    Recombine;
};

//L5a
Line(4000) = {653,652};
Line Loop(4001) = {-2581,-2714,-2601,4000};
Plane Surface(4002) = {4001};
Transfinite Line(4000)=lQuem/gridsize+1+deltaNode;
Transfinite Surface(4002);
Recombine Surface{4002};

//L5b
Line(4003) = {663,662};
Line Loop(4004) = {-2603,-2758,-2623,4003};
Plane Surface(4005) = {4004};
Transfinite Line(4003)=lQuem/gridsize+1+deltaNode;
Transfinite Surface(4005);
Recombine Surface{4005};

//Ext3
newEntities3[] =
Extrude{0,3-lQuem,0}
{
    Surface
{2555,2577,2599,4002,2621,4005,2643,2665,2687,2709,}
    Layers{(3-lQuem)/gridsize};
    Recombine;
};

//Ext3Quem
newEntities3q[] =
Extrude{0,lQuem,0}
{
    Surface
{4027,4049,4071,4115,4159,4181,4203,4225,4247,4269,}
    Layers{lQuem/gridsize};
    Recombine;
};

//L4a
Line(6000) = {969,968};
Line Loop(6001) = {-4581,-4714,-4601,6000};
Plane Surface(6002) = {6001};
Transfinite Line(6000)=lQuem/gridsize+1+deltaNode;
Transfinite Surface(6002);
Recombine Surface{6002};

//L4b
Line(6003) = {979,978};
Line Loop(6004) = {-4603,-4758,-4623,6003};
Plane Surface(6005) = {6004};
Transfinite Line(6003)=lQuem/gridsize+1+deltaNode;
Transfinite Surface(6005);
Recombine Surface{6005};

//Ext4
newEntities4[] =
Extrude{0,3-lQuem,0}
{
    Surface
{4555,4577,4599,6002,4621,6005,4643,4665,4687,4709,}
    Layers{(3-lQuem)/gridsize};
    Recombine;
};

//Ext4Quem
newEntities4q[] =
Extrude{0,lQuem,0}
{
    Surface
{6027,6049,6071,6115,6159,6181,6203,6225,6247,6269,}
    Layers{lQuem/gridsize};
    Recombine;
};

//L3a
Line(8000) = {1285,1284};
Line Loop(8001) = {-6581,-6714,-6601,8000};
Plane Surface(8002) = {8001};
Transfinite Line(8000)=lQuem/gridsize+1+deltaNode;
Transfinite Surface(8002);
Recombine Surface{8002};

//L3b
Line(8003) = {1295,1294};
Line Loop(8004) = {-6603,-6758,-6623,8003};
Plane Surface(8005) = {8004};
Transfinite Line(8003)=lQuem/gridsize+1+deltaNode;
Transfinite Surface(8005);
Recombine Surface{8005};

```

Figure A.6. Definition of extrusions on the '.geo' file. Part I.

```

//Ext5
newEntities5[] =
Extrude{0,3-lQuem,0}
{
    Surface
    {6555,6577,6599,8002,6621,8005,6643,6665,6687,6709,6
    Layers{(3-lQuem)/gridsize};
    Recombine;
};

//Ext5Quem
newEntities5q[] =
Extrude{0,lQuem,0}
{
    Surface
    {8027,8049,8071,8115,8159,8181,8203,8225,8247,8269,8
    Layers{lQuem/gridsize};
    Recombine;
};

//L2a
Line(10000) = {1601,1600};
Line Loop(10001) = {-8581,-8714,-8601,10000};
Plane Surface(10002) = {10001};
Transfinite Line{10000}=lQuem/gridsize+1+deltaNode;
Transfinite Surface{10002};
Recombine Surface{10002};

//L2b
Line(10003) = {1611,1610};
Line Loop(10004) = {-8603,-8758,-8623,10003};
Plane Surface(10005) = {10004};
Transfinite Line{10003}=lQuem/gridsize+1+deltaNode;
Transfinite Surface{10005};
Recombine Surface{10005};

//Ext6
newEntities6[] =
Extrude{0,3-lQuem,0}
{
    Surface
    {8555,8577,8599,10002,8621,10005,8643,8665,8687,8709
    Layers{(3-lQuem)/gridsize};
    Recombine;
};

//Ext6Quem
newEntities6q[] =
Extrude{0,lQuem,0}
{
    Surface
    {10027,10049,10071,10115,10159,10181,10203,10225,102
    Layers{lQuem/gridsize};
    Recombine;
};

//L1a
Line(12000) = {1917,1916};
Line Loop(12001) = {-10581,-10714,-10601,12000};
Plane Surface(12002) = {12001};
Transfinite Line{12000}=lQuem/gridsize+1+deltaNode;
Transfinite Surface{12002};
Recombine Surface{12002};

//L1b
Line(12003) = {1927,1926};
Line Loop(12004) = {-10603,-10758,-10623,12003};
Plane Surface(12005) = {12004};
Transfinite Line{12003}=lQuem/gridsize+1+deltaNode;
Transfinite Surface{12005};
Recombine Surface{12005};

//Ext7
newEntities7[] =
Extrude{0,1.4-lQuem/2,0}
{
    Surface
    {10555,10577,10599,12002,10621,12005,10643,10665,106
    Layers{(1.4-lQuem/2)/gridsize};
    Recombine;
};

```

Figure A.7. Definition of extrusions on the '.geo' file. Part II.

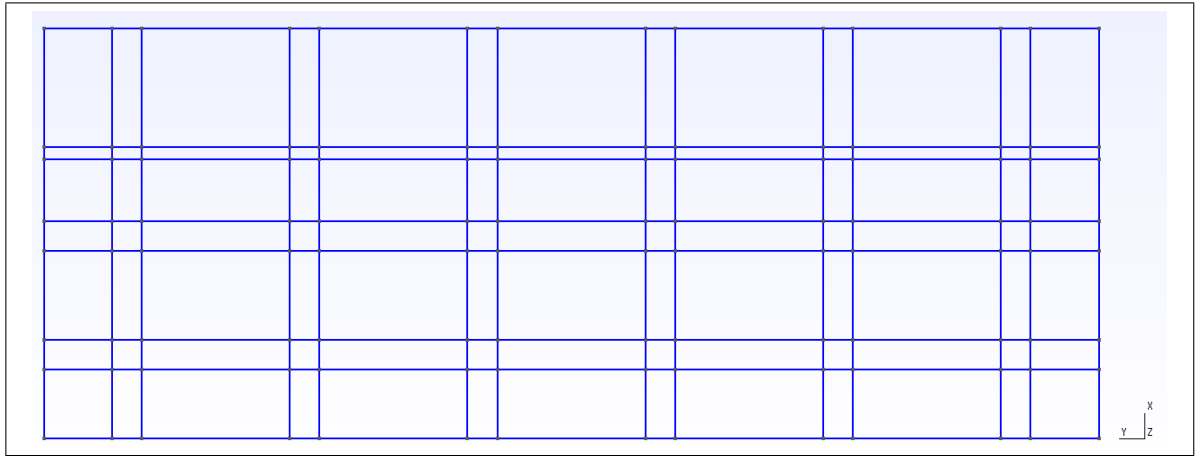


Figure A.8. Extrusions of the X-Z layer through the Y direction. Surface labels removed for simplicity. Label visibility can be activated in the Tools-Options-Geometry Menu, as to facilitate the extrusion task.

Once the whole volume is defined, the domain's peripheral surfaces have to be established as the respective boundaries for the boundary conditions. This is done through the Physical Surface function, where inside the brackets '{}' goes the name of the surface and after the equivalence '=' goes an array of the respective Plane Surfaces. These can be viewed through the Tools-Options-Geometry Menu, for labeling purposes. The domain's volume has to be defined as well, now using the Physical Volume function. Figure A.9 shows the final part of the script, where surface and volume labeling is done. For checking the labeling, as to be sure that it was correctly performed, the Tools-Visibility Menu —different from the Tools-Options-Geometry Menu— can be used.

```

//Definiciones
Physical Surface("rearAndSideWalls") =
{68,73,78,83,88,100,45,50,55,60,65,95,10,17,24,31,38,200,
Physical Surface("roof") =
{512,996,2370,2854,4370,4854,6370,6854,8370,8854,10370,10
Physical Surface("floor") =
{212,740,2070,2598,4070,4598,6070,6598,8070,8598,10070,10
Physical Surface("hanger") =
{604,1088,2462,2946,4462,4946,6462,6946,8462,8946,10462,1
Physical Surface("burnerFixedWalls") =
{736,2594,4594,6594,8594,10594,235,2002,2093,4002,4093,60
Physical Surface("burner") = {872,2730,4730,6730,8730,107
Physical Surface("sideEnt") =
{156,684,2014,2542,4014,4542,6014,6542,8014,8542,10014,10
Physical Surface("roomFloor") =
{178,706,2036,2564,4036,4564,6036,6564,8036,8564,10036,10
Physical Surface("facade") = {640,1124,2498,2982,4498,498
Physical Surface("outlet") = {666,1150,2524,3008,4524,500
Physical Volume("Vol") =
{1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22

//END OF GEO FILE|

```

Figure A.9. Defining the domain volume and surfaces. Surface labels are later used for boundary conditions.

Finally, once all of the above is performed, the 'Mesh 3D' algorithm —located in the left-hand-side hierarchic Menu— has to be run in order to create the mesh. Gmsh reports in the 'log' (located downwards in the graphic interphase) the meshing process, errors, and final observations (number of elements and of vertices, among others). The mesh is saved as a '.msh' file by going to the File Menu and clicking the Save Mesh function (see Figure A.10). The mesh is stored in the same folder and with the same name as the '.geo' file.

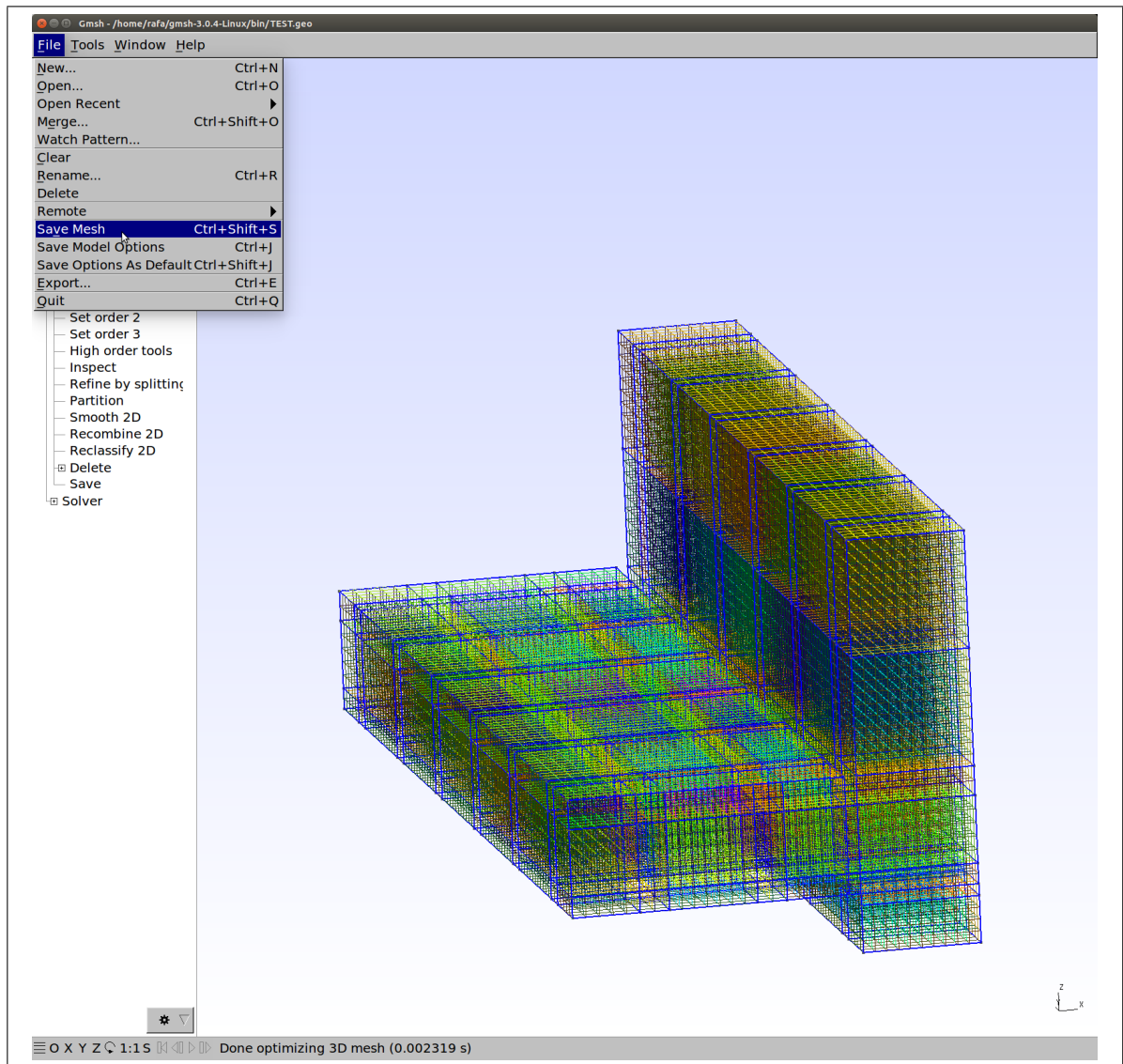


Figure A.10. Saving the '.msh' file

A.3. Exporting the Mesh to OpenFOAM

Gmsh exports the mesh to the OpenFOAM directory '/constant/polyMesh' from the generated '.msh' file, therefore it is fundamental that the whole case is set, and the '/constant/polyMesh' directory is empty. In case of using an OpenFOAM tutorial as a test case, delete

the 'blockMeshDict' file from this directory. From the Terminal and inside the '.msh' file directory, type the command:

gmshToFoam meshName.msh -case directoryLocationAndName

The 20cm cubic mesh used in this work is called 'EtftGrid20.msh' and the OpenFOAM directory '20cm144sa', therefore the necessary code line is shown in Figure A.11.



```
rafa@rafa: ~/gmsh-3.0.4-Linux/bin
rafa@rafa:~$ cd gmsh-3.0.4-Linux/
bin/ share/
rafa@rafa:~$ cd gmsh-3.0.4-Linux/bin/
rafa@rafa:~/gmsh-3.0.4-Linux/bin$ gmshToFoam EtftGrid20.msh -case 3rdRUN/20cm144sa
```

Figure A.11. Exporting the mesh into the OpenFOAM case directory.

This command generates files inside the previously empty '/constant/polyMesh' directory, as shown in Figure A.12.

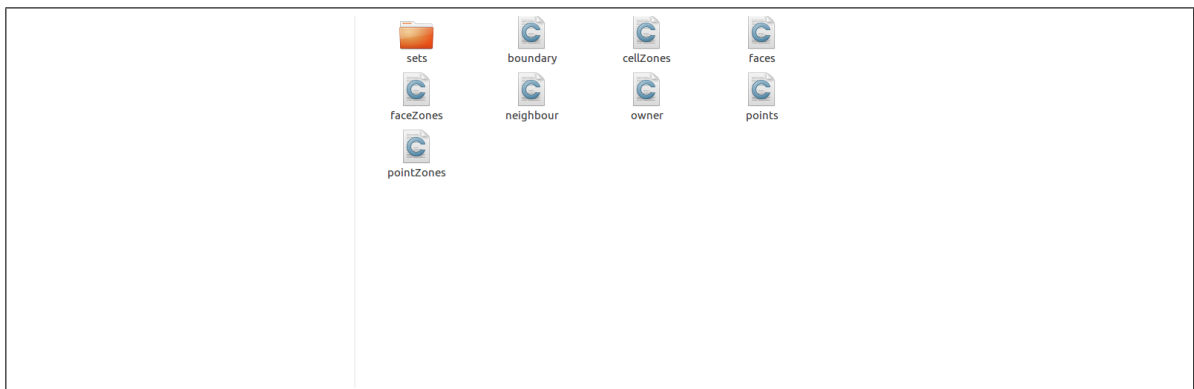


Figure A.12. '/constant/polyMesh' directory, after exporting the mesh.

Herein, only the 'boundary' Unix Executable script needs to be modified (see Figure A.13). After opening it with a text editor, each wall boundary has to be defined as 'wall', inlets/outlets/side as 'patch', and in case a 2D domain is being solved, the front and back faces of the 1-cell extruded domain have to be tagged as 'empty'. Note that the boundary names correspond to the ones assigned with the Physical Surface command in the '.geo' file. 'nFaces' indicates the number of mesh patches in the surface. Any additional surface named 'defaultPatches' corresponds to peripheral surfaces not defined as Physical Surfaces, or to internal surfaces adjacent other internal surfaces with different dimensions (e.g. in case that the 'C' Plane Surfaces neighbor sides are not of the same dimensions). If this case, the errors have to be corrected in such way that no 'defaultFaces' unknown to the solver are present.


```

/*-----* C++ -*-----*/
=====
V   F i e l d
   O p e r a t i o n
   A n d
   M a n i p u l a t i o n
=====

OpenFOAM: The Open Source CFD Toolbox
Version: 4.1
Web: www.OpenFOAM.org

FoamFile
{
    version      2.0;
    format        binary;
    class        polyBoundaryMesh;
    location      "constant/polyMesh";
    object        boundary;
}
// *****

10
(
    rearAndSideWalls
    {
        type            wall;
        physicalType    wall;
        nFaces          1130;
        startFace       125360;
    }
    sideEnt
    {
        type            patch;
        physicalType    patch;
        nFaces          4616;
        startFace       126490;
    }
    roomFloor
    {
        type            wall;
        physicalType    wall;
        nFaces          982;
        startFace       131106;
    }
    floor
    {
        type            wall;
        physicalType    wall;
        nFaces          1756;
        startFace       132008;
    }
    burnerFixedWalls
    {
        type            wall;
        physicalType    wall;
        nFaces          192;
        startFace       133764;
    }
    hanger
    {
        type            wall;
        physicalType    wall;
        nFaces          410;
        startFace       133956;
    }
    roof
    {
        type            wall;
        physicalType    wall;
        nFaces          1722;
        startFace       134366;
    }
    facade
    {
        type            wall;
        physicalType    wall;
        nFaces          820;
        startFace       136088;
    }
    outlet
    {
        type            patch;
        physicalType    patch;
        nFaces          820;
        startFace       136908;
    }
    burner|
    {
        type            patch;
        physicalType    patch;
        nFaces          48;
        startFace       137728;
    }
}
// *****

```

Figure A.13. Edited 'boundary' script for the 20cm mesh.

The final step before running the case, is to elaborate the 'Allrun' file inside the case directory. This script provides the function execution sequence for the case. As the ETFT cases correspond to a modified version of the tutorial 'smallPoolFire3D', the 'Allrun' file is similar except for the fact that the 'blockMeshDict' file is not executed. This file is not executed because the mesh was already built in Gmsh, and therefore the 'snappyHexMesh' was not used. The 'Allrun' file for the cases herein is shown in Figure A.14.

```
#!/bin/sh
# Source tutorial run functions
. $WM_PROJECT_DIR/bin/tools/RunFunctions

# Set application name
application='getApplication'

cp 0/ph_rgh.orig 0/ph_rgh

runApplication decomposePar -force

# Run
runParallel $application

#-----
```

Figure A.14. 'Allrun' script for the ETFT Test I cases.

Note that in this script, the hydro-static pressure field is generated for time-step zero from the file 'ph_rgh.orig' in the '/0' directory, and the 'decomposePar' utility for decomposing the domain for parallel processing is executed. To finally execute the case, go to the case directory from the Terminal, and execute the command './Allrun'. From this moment, the OpenFOAM case will start running, and the computing output will be written in the 'log.FireFOAM' script.

A.4. Viewing and Post-Processing Results

As the solver computes the solution, folders containing each time-step will appear in the case directory. Inside each folder, binary scripts of the solved fields are contained. ParaView interprets this binary solution, showing human-readable results. To read these, go to the case directory through the Terminal, and once inside, type:

paraFoam &

and a ParaView window with the case will open. If the case was run using the 'decomposePar' utility, time-step folders of each decomposed domain part are located within each respective processor folder. Hence, if four processors were used, the 'processor0', 'processor1', 'processor2' and 'processor3' folders should appear, each one containing the time-steps. To reconstruct the fields into the case directory, go through the Terminal to the case directory —where the 'processorX' folders are located— and type:

reconstructPar

This command will reconstruct the fields from each decomposed processor into the directory. Now one can delete the 'processorX' folders —as to save disk space— and execute the command:

paraFoam &

which as stated, will open a ParaView window with the simulation results. To view the fields, select the 'eye icon' and then the desired fields on the left-downside bar as shown in Figure A.15. Note that some fields, such as density and combustion product concentration among others, are not present as they are not initialized from time-step '0'. To recover these fields, it is recommended that previously to opening ParaView, the '0' folder is removed from the case directory. This way, the '.csv' files can be obtained including these not initialized fields.

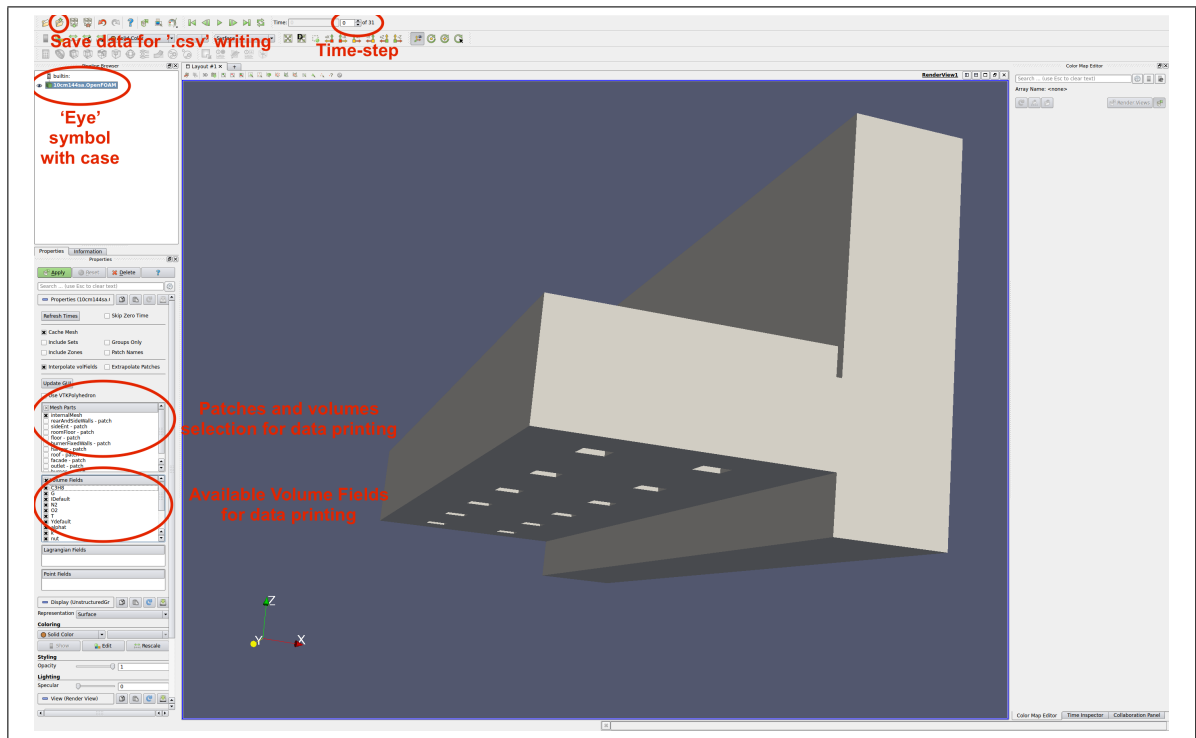


Figure A.15. Case selection, patch/volume selection, and field extraction in ParaView.

To select the physical location of the extracted data, the user has to make use of the Filters tab. Filters can also generate additional useful data, as cell dimensions for calculating mass/energy fluxes. Some available Filters are:

- **Extract Volume** — '.csv' file with data points/cells in the selected volume. Used for changes in compartment internal energy and HRR calculations (cell data).
- **Slice** — '.csv' file with data points/cells along a selected plane. Used for contours (data points) and energy flux calculations (cell data).
- **Plot Along Line** — '.csv' file with data points/cells along a selected line. Used for thermocouple tree temperature profiles and probe velocity profiles.
- **Mesh Quality** — This filter creates an additional field containing relevant cell information. It is used in the energy balance for calculating fluxes through surfaces

and/or global properties in the compartment, as it delivers cell surface area (for the Slice Filter) and cell volume (for the Extract Volume Filter).

After selecting and applying the Filter, click the Save Data option on the File menu, and a window will appear for the user to select the desired '.csv' writing options, as shown in Figure A.16.

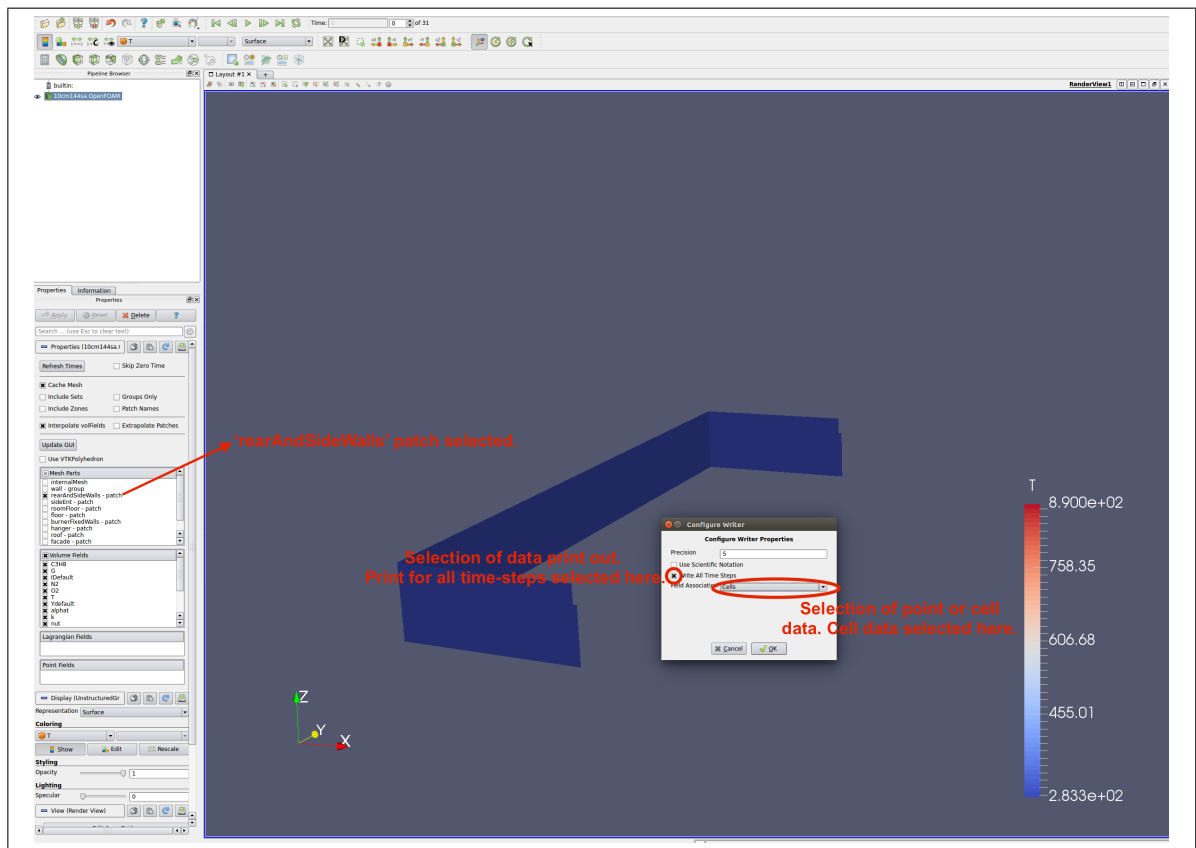


Figure A.16. '.csv' file being extracted for the 'rearAndSideWalls' patch. Here no slice was used, as this patch was directly chosen. To perform a Slice, the 'internalField' part has to be selected. Data is being printed as a list of cells contained in the patch, with its respective fields including its surface area (created with the Mesh Quality Filter).

The '.csv' file will contain the fields selected at the beginning. Point data prints in the '.csv' data for a set of points with its marked coordinates. Cell data gives a list of the cells

contained in the domain extracted with the Filter (3D volume, slice, or line). The 'write for all time-steps' option gives a '.csv' file for each time-step, ideal for plotting time-series and time-averaged values. Time-averaged values are necessary specially for fine meshes ($<20\text{cm}$), as the solver captures higher local rates of change given the smaller LES-filter.