

POLITECNICO DI MILANO

Facoltà di Ingegneria dei Processi Industriali

Corso di Laurea in Ingegneria Chimica



Sviluppo di modelli di combustione turbolenta per fiamme diffusive

RELATORE

Prof. Tiziano Faravelli

CORRELATORE

Ing. Alberto Cuoci

TESI DI LAUREA DI

Emanuele Francesco Leoni

Matr. N. 725188

Anno Accademico 2009/2010

Ai miei nonni

Ringraziamenti

Desidero ringraziare innanzitutto il professor Faravelli, relatore della tesi, per l'apporto fornitomi durante tutto il lavoro e per i consigli sempre puntuali e precisi. Ringrazio l'ing. Cuoci, la sua esperienza e le sue capacità informatiche sono state fondamentali durante questa tesi. Ringrazio anche l'ing. Lucchini del dipartimento di Energia: la sua disponibilità e la competenza nell'ambito del software *OpenFOAM* hanno contribuito a risolvere molti problemi incontrati durante il lavoro. Ringrazio tutti i professori, ricercatori e dottorandi del terzo piano che mi hanno accolto tra loro durante quest'anno, e in particolare: il professor Ranzi, per la sua grande cordialità, l'ing. Frassoldati per i suoi consigli, gli ingegneri Grana, Maffei e Sommariva, per la loro infinita pazienza e sopportazione e l'ing. Manenti per le indicazioni su Bergamo Alta, ma non solo. Ringrazio anche l'ing. Totaro, l'ing. Barker-Hemings, l'ing. Brambilla e l'ing. Gamba per la compagnia. Ringrazio i professori Pierucci, Manca, Dente, Bozzano, Pellegrini. Desidero inoltre ricordare il professor Biardi, scomparso da poco, e che ho avuto il piacere di conoscere più da vicino essendo stato rappresentante degli studenti nel Consiglio della Facoltà da lui presieduta. Un ringraziamento speciale va alla mia famiglia, che in questi anni di università mi è sempre stata vicina e mi ha supportato economicamente e non solo. Non sono stato sempre uno studente modello, ma spero di averle dato almeno qualche minima soddisfazione. Ringrazio i miei parenti: i miei nonni che ormai non ci sono più ma sono sicuro hanno vigilato dall'alto su di me, gli zii e le zie che mi hanno sempre sostenuto e anche tutti gli altri. Ringrazio i miei amici, è impossibile citarli tutti, ma sono sicuro che senza la compagnia che sono grato di avere avuto al mio fianco durante questi 6 anni di università sarebbe stato tutto meno bello e sicuramente più difficile. Infine voglio ringraziare Eloisa, che è entrata prepotentemente nella mia vita solo da poco tempo, ma l'ha già cambiata radicalmente. Questi ultimi mesi di tesi sono stati difficili, ma sicuramente molto più belli con lei al mio fianco.

Abstract

La simulazione numerica di fiamme turbolente non premiscelate (diffusive) è molto importante e richiesta in ambito non solo accademico, ma anche industriale grazie alle molteplici applicazioni di queste fiamme in diversi campi. Qualsiasi simulazione numerica in ambiente turbolento è per sua natura molto complessa e richiede l'utilizzo di modelli per risolvere le equazioni del trasporto. L'introduzione della combustione all'interno di una simulazione di questo tipo contribuisce ad elevare notevolmente il grado di complessità del sistema risultante. Infatti il termine che tiene conto della velocità di reazione aumenta la non linearità dell'equazione e, anche in questo caso, si devono utilizzare specifici modelli di combustione che semplifichino il problema portando comunque a risultati accettabili. In questo lavoro di tesi è stato utilizzato per la simulazione di queste fiamme un programma *open source*, *OpenFOAM*. Questo programma contiene, nella versione base, un modello di turbolenza, denominato *k-epsilon*, tuttavia è privo di alcuni modelli di combustione. Lo scopo del lavoro è stato perciò quello di implementare, all'interno del codice, due modelli di questo tipo, l'*Eddy Dissipation (ED)* e lo *Steady Laminar Flamelets (SLF)*, e verificarne il corretto funzionamento. Quest'ultima fase è stata effettuata simulando alcune fiamme già ampiamente modellate e di cui la letteratura offre un'elevata quantità di dati sperimentali. La convalida dei risultati è stata effettuata sia attraverso il confronto coi dati sperimentali stessi, sia attraverso il confronto con il software *Ansys Fluent*, codice commerciale ampiamente utilizzato in simulazioni di questo tipo e quindi utile come punto di riferimento. I risultati ottenuti sono stati abbastanza soddisfacenti: il modello *ED* conferma i suoi limiti intrinseci nella stima dei dati sperimentali, ma in questo caso il confronto con *Fluent* ha permesso di verificare che il modello fosse implementato nel modo corretto. Per quanto riguarda il modello *SLF*, più complesso, si sono ottenuti risultati in linea con i dati sperimentali per un elevato numero di casi. Probabilmente l'accordo andrebbe a migliorare ancora tarando il modello di turbolenza per ogni singola fiamma, dato che nel nostro caso il modello è stato utilizzato con i parametri standard. Analizzando i risultati si può affer-

mare che i modelli sono stati implementati all'interno del programma in maniera ottimale e pertanto l'obiettivo del lavoro è stato raggiunto.

Indice

Ringraziamenti	ii
Abstract	iii
Elenco delle figure	xi
Introduzione	1
1 La turbolenza	3
1.1 Introduzione	3
1.2 Equazioni di trasporto	4
1.2.1 Conservazione della massa	5
1.2.2 Conservazione della quantità di moto	5
1.2.3 Conservazione delle specie	6
1.2.4 Conservazione dell'energia	7
1.3 Leggi costitutive	8
1.3.1 Legge di Newton	8
1.3.2 Legge di Fick	9
1.3.3 Legge di Fourier	10
1.3.4 Equazione di stato	10
1.4 Equazioni di Navier-Stokes	10

1.5	Caratteristiche della turbolenza	13
1.6	Modellazione della turbolenza	15
1.6.1	Mediazione secondo Reynolds	16
1.6.2	Mediazione secondo Favre	17
2	<i>CFD</i> e modelli di turbolenza	19
2.1	Introduzione	19
2.2	Discretizzazione e creazione della <i>mesh</i>	20
2.2.1	Metodo ai volumi finiti	21
2.2.2	Condizioni al contorno	24
2.3	Risoluzione delle equazioni algebriche	24
2.3.1	Metodo <i>SIMPLE</i>	25
2.4	Modelli di turbolenza	26
2.5	Approccio <i>RANS</i>	29
2.5.1	Equazioni mediate	29
2.5.2	Flussi turbolenti di specie ed entalpia	30
2.5.3	Sforzi di Reynolds	30
2.5.4	Interazione cinetica-turbolenza	33
2.5.5	Modelli di irraggiamento	34
2.6	Codici utilizzati	34
3	Interazione cinetica-turbolenza	36
3.1	Introduzione	36
3.2	Modelli <i>RRA</i>	40
3.2.1	Modello <i>Eddy Break Up (EBU)</i>	40
3.2.2	Modello <i>Eddy Dissipation (ED)</i>	41
3.2.3	Modello <i>Finite Rate Chemistry (FR)</i>	42
3.2.4	Modello <i>EDFR</i>	43

3.2.5	Modello <i>Eddy Dissipation Concept (EDC)</i>	43
3.2.6	Modello <i>Transported PDF</i>	46
3.3	Modelli <i>PVA</i>	47
3.3.1	<i>Mixture fraction</i>	47
3.3.2	Equazioni di trasporto	49
3.3.3	Modelli <i>presumed PDF</i>	50
3.3.4	Modello miscelato-bruciato (<i>mixed is burnt</i>)	51
3.3.5	Modello di equilibrio	52
3.3.6	Modello <i>Steady Laminar Flamelets</i>	52
3.3.7	Modello <i>Laminar Flamelets</i> non stazionarie	55
4	Il software <i>OpenFOAM</i>	56
4.1	Introduzione	56
4.2	<i>Open source</i>	56
4.3	Il <i>C++</i> e la programmazione ad oggetti	58
4.3.1	Incapsulamento	59
4.3.2	Mascheramento	59
4.3.3	Polimorfismo	60
4.3.4	Ereditarietà	60
4.4	Il codice <i>OpenFOAM</i>	60
4.4.1	La struttura di <i>OpenFOAM</i>	62
4.4.1.1	I solutori	62
4.4.1.2	Le librerie	64
4.4.1.3	I casi	65
4.4.1.4	Le <i>utilities</i>	67
4.4.2	La programmazione <i>C++</i> in <i>OpenFOAM</i>	67
4.4.3	La creazione della <i>mesh</i> in <i>OpenFOAM</i>	70

5	Implementazione di modelli di combustione	74
5.1	Introduzione	74
5.2	Modello <i>Eddy Dissipation</i>	75
5.2.1	L'implementazione	75
5.2.2	Il set del caso	78
5.3	Modello <i>Steady Laminar Flamelets</i>	79
5.3.1	La creazione della <i>look-up table</i>	80
5.3.2	Il solutore <i>rhoSimpleFlamelets</i>	81
5.3.3	Le librerie <i>OpenSMOKE</i> e <i>flameletsThermo</i>	83
5.3.4	Il set del caso	84
5.3.5	Schema cinetico	85
6	Risultati	87
6.1	Introduzione	87
6.2	Le fiamme studiate	88
6.2.1	Fiamma Sandia H_2/He	88
6.2.2	Fiamma Sandia $CH_4/H_2/N_2$	89
6.2.3	Fiamma Sandia $CO/H_2/N_2$ A	90
6.2.4	Fiamma Sandia <i>Piloted CH_4/Air</i>	91
6.2.5	Fiamma Sandia <i>Bluff-Body</i>	92
6.3	Convalida modello <i>Eddy Dissipation</i>	94
6.3.1	Fiamma Sandia $H_2/He0$	96
6.3.2	Fiamma Sandia $CH_4/H_2/N_2$ A	99
6.4	Convalida modello <i>Steady Laminar Flamelets</i>	101
6.4.1	Fiamma Sandia $H_2/He0$	102
6.4.2	Fiamma Sandia $H_2/He2$	106
6.4.3	Fiamma Sandia $H_2/He4$	108

6.4.4	Fiamma Sandia $CH_4/H_2/N_2$ A	111
6.4.5	Fiamma Sandia $CH_4/H_2/N_2$ B	114
6.4.6	Fiamma Sandia $CO/H_2/N_2$ A	117
6.4.7	Fiamma Sandia <i>Piloted</i> CH_4/Air D	119
6.4.8	Fiamma Sandia <i>Bluff-Body</i> HM1	121
Conclusioni		125
A Costruzione della libreria di <i>SLF</i> non adiabatiche		128
A.1	Equazioni delle <i>flamelets</i> stazionarie	128
A.1.1	Estensione alla combustione non adiabatica	129
A.1.1.1	Metodo transitorio	130
A.1.1.2	Discretizzazione alle differenze finite	131
A.1.2	Costruzione della libreria <i>flamelet</i> non adiabatica	131
A.1.2.1	β -PDF	132
A.1.2.2	<i>Clipped-Gaussian PDF</i>	133
A.1.2.3	Distribuzione <i>log-normal</i>	134
A.1.2.4	Integrazione numerica	134
A.1.2.5	Integrazione rispetto a f	134
A.1.2.6	Integrazione rispetto a χ_{st}	135
B Risultati aggiuntivi		137
B.1	Modello <i>ED</i>	137
B.1.1	Fiamma Sandia H_2/He 0	137
B.1.2	Fiamma Sandia $CH_4/H_2/N_2$ A	140
B.2	Modello <i>SLA</i>	144
B.2.1	Fiamma Sandia H_2/He 0	144
B.2.2	Fiamma Sandia H_2/He 2	148

INDICE

B.2.3	Fiamma Sandia H_2/He_4	152
B.2.4	Fiamma Sandia $CH_4/H_2/N_2$ A	154
B.2.5	Fiamma Sandia $CH_4/H_2/N_2$ B	158
B.2.6	Fiamma Sandia $CO/H_2/N_2$ A	162
B.2.7	Fiamma Sandia <i>Piloted</i> CH_4/Air D	165
B.2.8	Fiamma Sandia <i>Bluff-Body</i> HM1	169
Elenco dei simboli		173
Bibliografia		178

Elenco delle figure

1.1	Differenza tra moto laminare e moto turbolento	3
1.2	Rappresentazione della macroscale L	14
1.3	Oscillazione di una componente della velocità intorno al valore medio	15
2.1	Evoluzione della temperatura nel tempo simulata con tecnica <i>DNS</i> , <i>RANS</i> e <i>LES</i> in un bruciatore turbolento	28
3.1	Descrizione dei tempi di residenza	38
3.2	Schema dei modelli di combustione	40
3.3	Modello per il trasferimento di energia turbolenta dalle strutture più grandi alle più piccole	44
3.4	Modello di Tennekes per le regioni <i>fine structures</i>	44
3.5	Schema di una <i>fine structure</i> approssimata a un reattore perfettamente miscelato	46
3.6	Geometria di una fiamma a controdiffusione	53
4.1	Struttura di <i>OpenFOAM</i> [36]	61
4.2	Contenuto della cartella principale del programma	63
4.3	Contenuto della cartella del solutore <i>speciesSimpleFoam</i>	63
4.4	Contenuto della cartella <i>src</i>	65
4.5	Contenuto della cartella di un caso	66
4.6	diagramma <i>UML</i> per la classe <i>GeometricField</i>	70

ELENCO DELLE FIGURE

4.7	<i>mesh</i> di esempio visualizzata attraverso il software <i>paraview</i>	73
5.1	Schema del modello <i>flamelets</i> in <i>OpenFOAM</i>	80
5.2	Struttura di una libreria di <i>flamelets</i> non adiabatica	81
6.1	<i>mesh</i> utilizzata per la fiamma <i>Sandia H₂/He</i>	89
6.2	<i>mesh</i> utilizzata per la fiamma <i>Sandia CH₄/H₂/N₂</i>	90
6.3	Fotografia della fiamma <i>Sandia CH₄/H₂/N₂ A</i> [47]	90
6.4	<i>mesh</i> utilizzata per la fiamma <i>Sandia CO/H₂/N₂ A</i>	91
6.5	Fotografia della fiamma <i>Sandia CO/H₂/N₂ A</i> [47]	91
6.6	<i>mesh</i> utilizzata per la fiamma <i>Sandia Piloted CH₄/Air</i>	92
6.7	Fotografia della fiamma <i>Sandia Piloted CH₄/Air</i> [47]	92
6.8	Schema di un bruciatore <i>bluff-body</i>	93
6.9	<i>mesh</i> utilizzata per la fiamma <i>Sandia Bluff-Body</i>	93
6.10	Fotografia della fiamma <i>Sandia Bluff-Body</i> [47]	93
6.11	Profili di velocità lungo l'asse di simmetria per la fiamma <i>H₂/He0</i> a freddo (a sinistra) e con reazione chimica (a destra)	94
6.12	ingrandimento della mappa della velocità assiale nel cannello di ingresso del combustibile per la fiamma <i>H₂/He0</i>	95
6.13	mappe di temperatura della fiamma <i>H₂/He0</i> modellata con <i>OpenFOAM</i> e <i>Fluent</i>	96
6.14	Fiamma <i>H₂/He0</i> . Profili assiali di temperatura, velocità e composizione. Confronto tra modelli (linea azzurra <i>Fluent</i> , linea rossa <i>OpenFOAM</i>) e dati sperimentali (punti in verde) [2]	97
6.15	Fiamma <i>H₂/He0</i> . Profili radiali di temperatura e frazione massiva di <i>H₂O</i> calcolati a 253 mm dallo sbocco. Confronto tra modelli (linea azzurra <i>Fluent</i> , linea rossa <i>OpenFOAM</i>) e dati sperimentali (punti in verde) [2] . . .	98
6.16	Fiamma <i>H₂/He0</i> . Profili radiali di temperatura e frazione massiva di <i>H₂O</i> calcolati a 675 mm dallo sbocco. Confronto tra modelli (linea azzurra <i>Fluent</i> , linea rossa <i>OpenFOAM</i>) e dati sperimentali (punti in verde) [2] . . .	98

ELENCO DELLE FIGURE

6.17	mappe di temperatura della fiamma $CH_4/H_2/N_2$ A modellata con <i>OpenFOAM</i> e <i>Fluent</i>	99
6.18	Fiamma $CH_4/H_2/N_2$ A. Profili assiali di temperatura, velocità e composizione. Confronto tra modelli (linea azzurra <i>Fluent</i> , linea rossa <i>OpenFOAM</i>) e dati sperimentali (punti in verde) [47]	100
6.19	Fiamma $CH_4/H_2/N_2$ A. Profili radiali di temperatura e frazione massiva di CO_2 calcolati a 160 mm dallo sbocco. Confronto tra modelli (linea azzurra <i>Fluent</i> , linea rossa <i>OpenFOAM</i>) e dati sperimentali (punti in verde) [47]	101
6.20	Fiamma $CH_4/H_2/N_2$ A. Profili radiali di temperatura e frazione massiva di CO_2 calcolati a 640 mm dallo sbocco. Confronto tra modelli (linea azzurra <i>Fluent</i> , linea rossa <i>OpenFOAM</i>) e dati sperimentali (punti in verde) [47]	101
6.21	Mappe di temperatura per la fiamma $H_2/He0$. A sinistra è visualizzata la simulazione adiabatica, a destra quella non adiabatica	103
6.22	Fiamma $H_2/He0$. Profili assiali di temperatura, composizione, <i>mixture fraction</i> e varianza. Confronto tra modelli (linea azzurra <i>Fluent</i> adiabatico, linea rossa <i>OpenFOAM</i> adiabatico, linea blu <i>OpenFOAM</i> non adiabatico) e dati sperimentali (punti in verde) [2]	104
6.23	Fiamma $H_2/He0$. Profili radiali delle frazioni massive di OH calcolati a 422 mm (a sinistra) e 675 mm (a destra) dallo sbocco. Confronto tra modelli (linea azzurra <i>Fluent</i> adiabatico, linea rossa <i>OpenFOAM</i> adiabatico, linea blu <i>OpenFOAM</i> non adiabatico) e dati sperimentali (punti in verde) [2]	105
6.24	Fiamma $H_2/He0$. Profili radiali di temperatura e <i>mixture fraction</i> calcolati a 675 mm dallo sbocco. Confronto tra modelli (linea azzurra <i>Fluent</i> adiabatico, linea rossa <i>OpenFOAM</i> adiabatico, linea blu <i>OpenFOAM</i> non adiabatico) e dati sperimentali (punti in verde) [2]	105
6.25	Mappe di temperatura per la fiamma $H_2/He2$. A sinistra è visualizzata la simulazione adiabatica, a destra quella non adiabatica	106
6.26	Fiamma $H_2/He2$. Profili assiali di temperatura, composizione, <i>mixture fraction</i> e varianza. Confronto tra modelli (linea azzurra <i>Fluent</i> adiabatico, linea rossa <i>OpenFOAM</i> adiabatico, linea blu <i>OpenFOAM</i> non adiabatico) e dati sperimentali (punti in verde) [2]	107

ELENCO DELLE FIGURE

6.27 Fiamma H_2/He_2 . Profili radiali di temperatura calcolati a 352 mm (a sinistra) e 562.5 mm (a destra) dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [2] 108

6.28 Fiamma H_2/He_2 . Profili radiali di *mixture fraction* calcolati a 352 mm (a sinistra) e 562.5 mm (a destra) dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [2] 108

6.29 Mappe di temperatura per la fiamma H_2/He_4 . A sinistra è visualizzata la simulazione adiabatica, a destra quella non adiabatica 109

6.30 Fiamma H_2/He_4 . Profili assiali di temperatura, composizione, *mixture fraction* e varianza. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [2] 110

6.31 Fiamma H_2/He_4 . Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 281 mm. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [2] 111

6.32 Mappe di temperatura della fiamma $CH_4/H_2/N_2$ A. A sinistra è visualizzata la simulazione adiabatica, a destra quella non adiabatica 112

6.33 Fiamma $CH_4/H_2/N_2$ A. Profili assiali di temperatura, composizione, *mixture fraction* e varianza. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [47] 113

6.34 Fiamma $CH_4/H_2/N_2$ A. Profili radiali delle frazioni massive di reagenti e prodotti calcolati a 160 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [47] . . . 114

6.35 Mappe di temperatura della fiamma $CH_4/H_2/N_2$ B. A sinistra è visualizzata la simulazione adiabatica, a destra quella non adiabatica 115

ELENCO DELLE FIGURE

6.36	Fiamma $CH_4/H_2/N_2$ B. Profili assiali di temperatura, composizione, <i>mixture fraction</i> e varianza. Confronto tra modelli (linea azzurra <i>Fluent</i> adiabatico, linea rossa <i>OpenFOAM</i> adiabatico, linea blu <i>OpenFOAM</i> non adiabatico) e dati sperimentali (punti in verde) [47]	116
6.37	Mappe di temperatura della fiamma $CO/H_2/N_2$ A. A sinistra è visualizzata la simulazione adiabatica, a destra quella non adiabatica	117
6.38	Fiamma $CO/H_2/N_2$ A. Profili assiali di temperatura, composizione, <i>mixture fraction</i> e varianza. Confronto tra modelli (linea azzurra <i>Fluent</i> adiabatico, linea rossa <i>OpenFOAM</i> adiabatico, linea blu <i>OpenFOAM</i> non adiabatico) e dati sperimentali (punti in verde) [4]	118
6.39	Mappe di temperatura della fiamma <i>Piloted</i> CH_4/Air D. A sinistra è visualizzata la simulazione adiabatica, a destra quella non adiabatica	119
6.40	Fiamma <i>Piloted</i> CH_4/Air D. Profili assiali di temperatura, composizione, <i>mixture fraction</i> e varianza. Confronto tra modelli (linea azzurra <i>Fluent</i> adiabatico, linea rossa <i>OpenFOAM</i> adiabatico, linea blu <i>OpenFOAM</i> non adiabatico) e dati sperimentali (punti in verde) [3]	120
6.41	Fiamma <i>Piloted</i> CH_4/Air D. Profili radiali di temperatura e <i>mixture fraction</i> calcolati a 216 mm dallo sbocco. Confronto tra modelli (linea azzurra <i>Fluent</i> adiabatico, linea rossa <i>OpenFOAM</i> adiabatico, linea blu <i>OpenFOAM</i> non adiabatico) e dati sperimentali (punti in verde) [3]	121
6.42	Mappe di temperatura della fiamma <i>Bluff-Body</i> HM1. A sinistra è visualizzata la simulazione adiabatica, a destra quella non adiabatica	121
6.43	Ingrandimento del campo di temperatura nella zona del <i>bluff-body</i> in 4 step iterativi diversi	122
6.44	Fiamma <i>Bluff-Body</i> HM1. Profili assiali di temperatura, composizione, <i>mixture fraction</i> e varianza. Confronto tra modelli (linea azzurra <i>Fluent</i> adiabatico, linea rossa <i>OpenFOAM</i> adiabatico, linea blu <i>OpenFOAM</i> non adiabatico) e dati sperimentali (punti in verde) [1]	123
6.45	Fiamma <i>Bluff-Body</i> HM1. Profili radiali di temperatura e frazione massiva di <i>OH</i> calcolati a 13 mm dallo sbocco. Confronto tra modelli (linea azzurra <i>Fluent</i> adiabatico, linea rossa <i>OpenFOAM</i> adiabatico, linea blu <i>OpenFOAM</i> non adiabatico) e dati sperimentali (punti in verde) [1]	124

6.46	Fiamma <i>Bluff-Body HM1</i> . Profili radiali di temperatura e frazione massiva di <i>OH</i> calcolati a 120 mm dallo sbocco. Confronto tra modelli (linea azzurra <i>Fluent</i> adiabatico, linea rossa <i>OpenFOAM</i> adiabatico, linea blu <i>OpenFOAM</i> non adiabatico) e dati sperimentali (punti in verde) [1]	124
A.1	Struttura di una libreria di <i>flamelets</i> non adiabatica	130
B.1	Fiamma <i>H₂/He0</i> . Profili radiali di temperatura, velocità e composizione calcolati a 84 mm dallo sbocco. Confronto tra modelli (linea azzurra <i>Fluent</i> , linea rossa <i>OpenFOAM</i>) e dati sperimentali (punti in verde) [2]	137
B.2	Fiamma <i>H₂/He0</i> . Profili radiali di temperatura, velocità e composizione calcolati a 253 mm dallo sbocco. Confronto tra modelli (linea azzurra <i>Fluent</i> , linea rossa <i>OpenFOAM</i>) e dati sperimentali (punti in verde) [2]	138
B.3	Fiamma <i>H₂/He0</i> . Profili radiali di temperatura, velocità e composizione calcolati a 422 mm dallo sbocco. Confronto tra modelli (linea azzurra <i>Fluent</i> , linea rossa <i>OpenFOAM</i>) e dati sperimentali (punti in verde) [2]	138
B.4	Fiamma <i>H₂/He0</i> . Profili radiali di temperatura, velocità e composizione calcolati a 675 mm dallo sbocco. Confronto tra modelli (linea azzurra <i>Fluent</i> , linea rossa <i>OpenFOAM</i>) e dati sperimentali (punti in verde) [2]	139
B.5	Fiamma <i>CH₄/H₂/N₂ A</i> . Profili radiali di temperatura, velocità e composizione calcolati a 40 mm dallo sbocco. Confronto tra modelli (linea azzurra <i>Fluent</i> , linea rossa <i>OpenFOAM</i>) e dati sperimentali (punti in verde) [47]	140
B.6	Fiamma <i>CH₄/H₂/N₂ A</i> . Profili radiali di temperatura, velocità e composizione calcolati a 160 mm dallo sbocco. Confronto tra modelli (linea azzurra <i>Fluent</i> , linea rossa <i>OpenFOAM</i>) e dati sperimentali (punti in verde) [47]	141
B.7	Fiamma <i>CH₄/H₂/N₂ A</i> . Profili radiali di temperatura, velocità e composizione calcolati a 320 mm dallo sbocco. Confronto tra modelli (linea azzurra <i>Fluent</i> , linea rossa <i>OpenFOAM</i>) e dati sperimentali (punti in verde) [47]	142
B.8	Fiamma <i>CH₄/H₂/N₂ A</i> . Profili radiali di temperatura, velocità e composizione calcolati a 640 mm dallo sbocco. Confronto tra modelli (linea azzurra <i>Fluent</i> , linea rossa <i>OpenFOAM</i>) e dati sperimentali (punti in verde) [47]	143

B.9 Fiamma $H_2/He0$. Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 84 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [2] . . . 144

B.10 Fiamma $H_2/He0$. Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 253 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [2] . . . 145

B.11 Fiamma $H_2/He0$. Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 422 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [2] . . . 146

B.12 Fiamma $H_2/He0$. Profili radiali di Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 675 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [2] 147

B.13 Fiamma $H_2/He2$. Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 70 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [2] . . . 148

B.14 Fiamma $H_2/He2$. Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 211 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [2] . . . 149

B.15 Fiamma $H_2/He2$. Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 352 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [2] . . . 150

B.16 Fiamma $H_2/He2$. Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 562.5 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [2] . . . 151

B.17 Fiamma H_2/He_4 . Profili radiali di temperatura, composizione, <i>mixture fraction</i> e varianza calcolati a 187.5 mm dallo sbocco. Confronto tra modelli (linea azzurra <i>Fluent</i> adiabatico, linea rossa <i>OpenFOAM</i> adiabatico, linea blu <i>OpenFOAM</i> non adiabatico) e dati sperimentali (punti in verde) [2] . . .	152
B.18 Fiamma H_2/He_4 . Profili radiali di temperatura, composizione, <i>mixture fraction</i> e varianza calcolati a 375 mm dallo sbocco. Confronto tra modelli (linea azzurra <i>Fluent</i> adiabatico, linea rossa <i>OpenFOAM</i> adiabatico, linea blu <i>OpenFOAM</i> non adiabatico) e dati sperimentali (punti in verde) [2] . . .	153
B.19 Fiamma $CH_4/H_2/N_2$ A. Profili radiali di temperatura, composizione, <i>mixture fraction</i> e varianza calcolati a 40 mm dallo sbocco. Confronto tra modelli (linea azzurra <i>Fluent</i> adiabatico, linea rossa <i>OpenFOAM</i> adiabatico, linea blu <i>OpenFOAM</i> non adiabatico) e dati sperimentali (punti in verde) [47]	154
B.20 Fiamma $CH_4/H_2/N_2$ A. Profili radiali di temperatura, composizione, <i>mixture fraction</i> e varianza calcolati a 160 mm dallo sbocco. Confronto tra modelli (linea azzurra <i>Fluent</i> adiabatico, linea rossa <i>OpenFOAM</i> adiabatico, linea blu <i>OpenFOAM</i> non adiabatico) e dati sperimentali (punti in verde) [47]	155
B.21 Fiamma $CH_4/H_2/N_2$ A. Profili radiali di temperatura, composizione, <i>mixture fraction</i> e varianza calcolati a 320 mm dallo sbocco. Confronto tra modelli (linea azzurra <i>Fluent</i> adiabatico, linea rossa <i>OpenFOAM</i> adiabatico, linea blu <i>OpenFOAM</i> non adiabatico) e dati sperimentali (punti in verde) [47]	156
B.22 Fiamma $CH_4/H_2/N_2$ A. Profili radiali di temperatura, composizione, <i>mixture fraction</i> e varianza calcolati a 640 mm dallo sbocco. Confronto tra modelli (linea azzurra <i>Fluent</i> adiabatico, linea rossa <i>OpenFOAM</i> adiabatico, linea blu <i>OpenFOAM</i> non adiabatico) e dati sperimentali (punti in verde) [47]	157
B.23 Fiamma $CH_4/H_2/N_2$ B. Profili radiali di temperatura, composizione, <i>mixture fraction</i> e varianza calcolati a 40 mm dallo sbocco. Confronto tra modelli (linea azzurra <i>Fluent</i> adiabatico, linea rossa <i>OpenFOAM</i> adiabatico, linea blu <i>OpenFOAM</i> non adiabatico) e dati sperimentali (punti in verde) [47]	158

B.24 Fiamma $CH_4/H_2/N_2$ B. Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 160 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [47] 159

B.25 Fiamma $CH_4/H_2/N_2$ B. Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 320 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [47] 160

B.26 Fiamma $CH_4/H_2/N_2$ B. Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 640 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [47] 161

B.27 Fiamma $CO/H_2/N_2$ A. Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 92 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [4] . . 162

B.28 Fiamma $CO/H_2/N_2$ A. Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 183 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [4] 163

B.29 Fiamma $CO/H_2/N_2$ A. Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 275 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [4] 164

B.30 Fiamma <i>Piloted CH₄/Air D</i> . Profili radiali di temperatura, composizione, <i>mixture fraction</i> e varianza calcolati a 14 mm dallo sbocco. Confronto tra modelli (linea azzurra <i>Fluent</i> adiabatico, linea rossa <i>OpenFOAM</i> adiabatico, linea blu <i>OpenFOAM</i> non adiabatico) e dati sperimentali (punti in verde)[3]	165
B.31 Fiamma <i>Piloted CH₄/Air D</i> . Profili radiali di temperatura, composizione, <i>mixture fraction</i> e varianza calcolati a 108 mm dallo sbocco. Confronto tra modelli (linea azzurra <i>Fluent</i> adiabatico, linea rossa <i>OpenFOAM</i> adiabatico, linea blu <i>OpenFOAM</i> non adiabatico) e dati sperimentali (punti in verde) [3]	166
B.32 Fiamma <i>Piloted CH₄/Air D</i> . Profili radiali di temperatura, composizione, <i>mixture fraction</i> e varianza calcolati a 216 mm dallo sbocco. Confronto tra modelli (linea azzurra <i>Fluent</i> adiabatico, linea rossa <i>OpenFOAM</i> adiabatico, linea blu <i>OpenFOAM</i> non adiabatico) e dati sperimentali (punti in verde) [3]	167
B.33 Fiamma <i>Piloted CH₄/Air D</i> . Profili radiali di temperatura, composizione, <i>mixture fraction</i> e varianza calcolati a 432 mm dallo sbocco. Confronto tra modelli (linea azzurra <i>Fluent</i> adiabatico, linea rossa <i>OpenFOAM</i> adiabatico, linea blu <i>OpenFOAM</i> non adiabatico) e dati sperimentali (punti in verde) [3]	168
B.34 Fiamma <i>Bluff-Body HM1</i> . Profili radiali di temperatura, composizione, <i>mixture fraction</i> e varianza calcolati a 13 mm dallo sbocco. Confronto tra modelli (linea azzurra <i>Fluent</i> adiabatico, linea rossa <i>OpenFOAM</i> adiabatico, linea blu <i>OpenFOAM</i> non adiabatico) e dati sperimentali (punti in verde) [1]	169
B.35 Fiamma <i>Bluff-Body HM1</i> . Profili radiali di temperatura, composizione, <i>mixture fraction</i> e varianza calcolati a 45 mm dallo sbocco. Confronto tra modelli (linea azzurra <i>Fluent</i> adiabatico, linea rossa <i>OpenFOAM</i> adiabatico, linea blu <i>OpenFOAM</i> non adiabatico) e dati sperimentali (punti in verde) [1]	170

ELENCO DELLE FIGURE

B.36 Fiamma <i>Bluff-Body HM1</i> . Profili radiali di temperatura, composizione, <i>mixture fraction</i> e varianza calcolati a 90 mm dallo sbocco. Confronto tra modelli (linea azzurra <i>Fluent</i> adiabatico, linea rossa <i>OpenFOAM</i> adiabatico, linea blu <i>OpenFOAM</i> non adiabatico) e dati sperimentali (punti in verde) [1]	171
B.37 Fiamma <i>Bluff-Body HM1</i> . Profili radiali di temperatura, composizione, <i>mixture fraction</i> e varianza calcolati a 120 mm dallo sbocco. Confronto tra modelli (linea azzurra <i>Fluent</i> adiabatico, linea rossa <i>OpenFOAM</i> adiabatico, linea blu <i>OpenFOAM</i> non adiabatico) e dati sperimentali (punti in verde) [1]	172

Introduzione

Le fiamme turbolente non premiscelate (o diffuse) sono molto utilizzate in ambito industriale per la loro grande efficienza e per questioni di sicurezza, in quanto la premiscelazione dei reagenti causerebbe rischi maggiori di incendi e esplosioni. Tipiche fiamme di questo tipo vengono usate nelle caldaie e nei bruciatori industriali. D'altra parte una combustione di questa natura causa la formazione di grandi quantità di inquinanti rispetto a quella in cui i reagenti sono miscelati prima dell'ignizione, e in fase di progettazione di nuovi bruciatori si deve evidentemente tener conto di leggi da rispettare sempre più restrittive. Per questo è grande la richiesta anche da parte del mondo industriale di software computazionali in grado di descrivere adeguatamente queste fiamme e allo stesso tempo di simulare in maniera realistica la formazione di specie inquinanti. Rispetto ad un'analisi sperimentale, le simulazioni di fluidodinamica computazionale (*CFD*) offrono numerosi vantaggi: innanzitutto il costo è decisamente inferiore sia in termini di risorse che di tempo utilizzato, e si possono cambiare le condizioni di una simulazione in maniera solitamente molto rapida. Inoltre i fenomeni sono studiati nelle loro dimensioni reali evitando così problemi di scala. Possono essere simulate situazioni anche pericolose che, riprodotte in laboratorio, richiederebbero le dovute e costose precauzioni e infine si possono valutare alcune variabili fisiche difficilmente misurabili con precisione. D'altra parte bisogna dire che, nell'ambito dei calcoli di *CFD*, vengono utilizzati dei modelli semplificati in modo da trovare un compromesso tra costi computazionali e accuratezza dei risultati; per questo è necessario convalidare i modelli e i programmi utilizzati caso per caso ed è per tale motivo che le simulazioni numeriche ancora non possono sostituire completamente l'analisi sperimentale. In questo contesto si inserisce il presente lavoro di tesi, svolto all'interno del dipartimento di Ingegneria Chimica "Giulio Natta" del Politecnico di Milano. L'obiettivo del lavoro è stato quello di verificare il funzionamento di alcuni modelli fluidodinamici già ampiamente utilizzati nello studio di fiamme diffuse utilizzando un software *opensource*, *OpenFOAM*, che solo negli ultimi tempi sta assumendo una certa importanza in ambito

accademico. Si è scelto di utilizzare questo software in quanto è gratuito e liberamente modificabile, cioè possono essere implementati al suo interno i modelli più disparati. Questa duttilità lo rende uno strumento molto potente e in grado di competere con altri software molto più diffusi, ma anche molto più costosi, come ad esempio *Ansys Fluent*. La combustione in ambiente turbolento è uno dei fenomeni ancora oggi più difficili da modellare. Nei primi due capitoli di questa tesi sono descritti i fenomeni di turbolenza e le modalità con cui i software di fluidodinamica computazionale cercano di risolverli. Nel terzo capitolo è trattato il problema dell'interazione cinetica-turbolenza, in cui ai fenomeni appena discussi viene aggiunta la combustione aumentando notevolmente la complessità del sistema risultante. Per riuscire a risolvere questo problema sono stati sviluppati nel tempo diversi tipi di modelli a complessità crescente, tutt'ora utilizzati nei più importanti software di fluidodinamica. Questi però non sono presenti nella versione base di *OpenFOAM* perciò si è resa necessaria la loro implementazione. Nel quarto capitolo vengono spiegate le caratteristiche e il funzionamento del software utilizzato mentre nel quinto è descritta nei dettagli l'implementazione di alcuni di questi modelli, che è stata realizzata utilizzando il linguaggio *C++* e la programmazione a oggetti, che offre notevoli vantaggi. Successivamente i modelli inseriti sono stati collaudati attraverso la simulazione di numerose fiamme diffusive, in modo da verificarne il funzionamento all'interno di *OpenFOAM*. Il sesto capitolo è quindi dedicato alla presentazione dei risultati, convalidati attraverso due tipi di confronti: il primo con i risultati ottenuti usando gli stessi modelli all'interno del software *Fluent* che, dato il suo largo utilizzo, può ritenersi particolarmente affidabile e il secondo con i dati sperimentali disponibili in letteratura per le fiamme modellate.

Capitolo 1

La turbolenza

1.1 Introduzione

I regimi di moto con cui un fluido in movimento si può presentare sono due: laminare e turbolento. Un flusso si dice laminare quando il moto del fluido avviene in maniera regolare, tramite lo scorrimento di strati infinitesimi gli uni sugli altri senza alcun tipo di rimescolamento, neppure su scala infinitesima. Si parla invece di flusso turbolento quando le forze viscosive non sono sufficienti a contrastare le forze di inerzia e il moto avviene di conseguenza in maniera caotica, senza seguire traiettorie ordinate. Possiamo apprezzare la differenza tra questi due tipi di moto in figura 1.1. I problemi studiati in questo lavoro di tesi riguardano situazioni in cui i flussi reattivi si presentano in condizioni di moto turbolento. In questo capitolo si descriverà sinteticamente la turbolenza e le equazioni che la caratterizzano, tenendo in considerazione la presenza di reazioni chimiche. Dopo una panoramica sulle equazioni del trasporto che stanno alla base della trattazione teorica di qualsiasi flusso, si entrerà nel merito della turbolenza e si introdurranno i concetti che

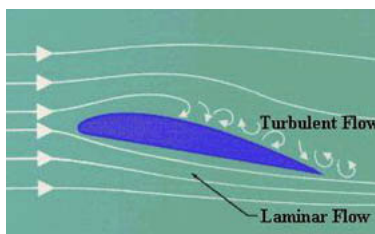


Figura 1.1: Differenza tra moto laminare e moto turbolento

stanno alla base dei modelli utilizzati per risolverla, illustrati successivamente nel secondo capitolo.

1.2 Equazioni di trasporto

Per modellare qualsiasi tipo di flusso, quindi anche un flusso turbolento reattivo, è necessario conoscere le equazioni di conservazione [9]. Come punto di partenza definiamo l'equazione di conservazione di una generica proprietà estensiva in una ben determinata regione dello spazio, che prende il nome di volume di controllo. Normalmente essa viene scritta come una relazione tra la velocità di cambiamento della proprietà stessa e le determinazioni esterne ed interne al volume di controllo. Prendendo come riferimento una generica quantità ϕ nel volume V che ha superficie S , si scrive l'equazione di conservazione nella forma generale (indipendente dal sistema di riferimento):

$$\int_V \frac{\partial}{\partial t}(\rho\phi) dV + \int_S \rho\phi\vec{u} \cdot \underline{n} dS = \int_S \vec{Q}_S(\phi) \cdot \underline{n} dS + \int_V Q_V(\phi) dV \quad (1.1)$$

dove \underline{n} è il versore ortogonale ad ogni punto della superficie S , considerato positivo uscente, \vec{u} è la velocità del fluido e ρ è la sua densità. Il primo termine a sinistra dell'uguale è la velocità di cambiamento della quantità ϕ nel volume, mentre il secondo è il flusso convettivo di ϕ attraverso la superficie S che, come si può vedere, è legato alla velocità convettiva del fluido. A destra dell'uguale il primo termine è il flusso diffusivo di ϕ attraverso la superficie S e il secondo termine è la produzione di ϕ nel volume V . Si può a questo punto applicare il teorema della divergenza di Gauss in modo da trasformare gli integrali di superficie in integrali di volume. Se poi si considera un volume di controllo infinitesimo, si ottiene la formulazione differenziale indipendente dal tipo di sistema di riferimento:

$$\frac{\partial}{\partial t}(\rho\phi) + \nabla(\rho\vec{u}\phi) = \nabla\vec{Q}_S(\phi) + Q_V(\phi) \quad (1.2)$$

In alternativa, introducendo la derivata lagrangiana (o sostanziale)

$$\frac{D(\phi)}{Dt} = \frac{\partial\phi}{\partial t} + \vec{u} \cdot \nabla\phi \quad (1.3)$$

la formulazione risulta:

$$\rho \frac{D(\phi)}{Dt} = \nabla\vec{Q}_S(\phi) + Q_V(\phi) \quad (1.4)$$

Nei paragrafi successivi verranno presentate le diverse equazioni di conservazione della massa, della quantità di moto, delle specie e dell'energia, trascurando i passaggi matematici delle dimostrazioni [9].

1.2.1 Conservazione della massa

La conservazione della massa è chiamata anche equazione di continuità e si ottiene in forma integrale a partire dalla equazione precedentemente descritta [9]. Il semplice significato di tale equazione è che la massa si conserva:

$$\int_V \frac{\partial \rho}{\partial t} dV + \int_S \rho \vec{u} \cdot \underline{n} dS = 0 \quad (1.5)$$

dove ρ è la densità del fluido e \vec{u} è la sua velocità. La formula è più breve della precedente perchè nei problemi di ingegneria chimica la velocità con la quale la massa cambia all'interno del volume di controllo è imputabile solo ai flussi di materia attraverso la superficie S . Applicando come nel paragrafo precedente il teorema della divergenza, il termine di convezione viene trasformato in un integrale di volume e la forma differenziale, indipendente dal sistema di riferimento, si scrive come

$$\frac{\partial \rho}{\partial t} + \nabla(\rho \vec{u}) = 0 \quad (1.6)$$

1.2.2 Conservazione della quantità di moto

L'equazione della quantità di moto [9] considera le forze agenti sul volume di controllo: forze superficiali (pressione, tensione superficiale, sforzi tangenziali) e forze di campo (gravità, forza di Coriolis, forza elettromagnetica). La forma integrale è:

$$\int_V \frac{\partial}{\partial t}(\rho \vec{u}) dV + \int_S \rho \vec{u} \vec{u} \cdot \underline{n} dS = \int_S \underline{\underline{\sigma}} \cdot \underline{n} dS + \int_V \rho \vec{f} dV \quad (1.7)$$

dove con $\underline{\underline{\sigma}}$ si intende il tensore degli sforzi, mentre il termine sorgente a destra ha la seguente forma

$$\rho \vec{f} = \rho \sum_{k=1}^{N_C} \vec{f}_k \omega_k \quad (1.8)$$

in cui \vec{f} è la forza di volume agente sulla k -esima specie e N_C è il numero totale di specie. Se consideriamo soltanto la forza di gravità, allora \vec{f} coincide con il vettore accelerazione di gravità \vec{g} . Il significato di questa equazione è che la velocità con cui cambia la quantità di moto all'interno del volume di controllo è uguale alla somma delle forze di volume e di superficie agenti sul fluido. Applicando di nuovo il teorema della divergenza, otteniamo la forma differenziale:

$$\frac{\partial}{\partial t}(\rho\vec{u}) + \nabla(\rho\vec{u}\vec{u}) = \rho\vec{f} + \nabla\vec{\underline{\underline{\sigma}}} \quad (1.9)$$

o in forma più compatta

$$\rho \frac{D\vec{u}}{Dt} = \rho\vec{f} + \nabla\vec{\underline{\underline{\sigma}}} \quad (1.10)$$

Questa è un'equazione vettoriale che condensa un numero di equazioni scalari dipendenti dalle dimensioni geometriche del problema, una per ogni direzione. In un riferimento di coordinate cartesiane, nella generica direzione i -esima si ha:

$$\rho \frac{Du_i}{Dt} = \frac{\partial}{\partial t}(\rho u_i) + \nabla(\rho u_i \vec{u}) = \rho f_i + \nabla \vec{\sigma}_i \quad (1.11)$$

dove $\vec{\sigma}_i$ è il vettore colonna i -esima del tensore degli sforzi.

1.2.3 Conservazione delle specie

Questa equazione di conservazione viene introdotta solo nel caso in cui il fluido sia composto da diverse specie e sta ad indicare la conservazione della massa della specie k -esima [9]. Le due forme integrale e differenziale sono le seguenti:

$$\int_V \frac{\partial}{\partial t}(\rho\omega_k) dV + \int_S \rho\vec{u}\omega_k \cdot \underline{n} dS = - \int_S \vec{j}_k \cdot \underline{n} dS + \int_V \dot{\Omega}_k dV \quad k = 1, \dots, N_C \quad (1.12)$$

$$\rho \frac{D\omega_k}{Dt} = \frac{\partial}{\partial t}(\rho\omega_k) + \nabla(\rho\vec{u}\omega) = -\nabla\vec{j}_k + \dot{\Omega}_k \quad k = 1, \dots, N_C \quad (1.13)$$

dove N_C è il numero delle specie, \vec{j}_k è il flusso diffusivo molecolare, $\dot{\Omega}_k$ è la velocità di formazione massiva della specie k per unità di volume e ω_k è la frazione massiva della specie k .

Bisogna a questo punto fare presente due definizioni logiche:

$$\sum_{k=1}^{N_C} \vec{j}_k = \vec{0} \quad (1.14)$$

$$\sum_{k=1}^{N_C} \dot{\Omega}_k = 0 \quad (1.15)$$

Esse permettono di ottenere, sommando tutte le N_C equazioni della conservazione delle specie, l'equazione di continuità. Solitamente si esprime il flusso diffusivo molecolare facendo riferimento ad una velocità di diffusione \vec{V}_k :

$$\vec{j}_k = \rho \omega_k \vec{V}_k \quad (1.16)$$

1.2.4 Conservazione dell'energia

L'equazione di conservazione dell'energia può essere scritta in forme differenti [9]. Il punto di partenza per la derivazione di diverse varianti è la conservazione dell'energia totale specifica per unità di massa \hat{E} . Tale energia è data dalla somma dell'energia cinetica specifica \hat{E}_k (associata al moto convettivo della materia) e della energia interna specifica \hat{U} (associata allo stato molecolare della materia e quindi, trascurando l'energia potenziale, funzione di temperatura, pressione e composizione del sistema in quanto variabile di stato):

$$\hat{E} = \hat{E}_k + \hat{U}(p, T, \omega_k) = \frac{1}{2} \vec{u} \cdot \vec{u} + \hat{U}(p, T, \omega_k) \quad (1.17)$$

L'equazione di conservazione nella forma integrale è quindi:

$$\int_V \frac{\partial}{\partial t} (\rho \hat{E}) dV + \int_S \rho \hat{E} \vec{u} \cdot \underline{n} dS = - \int_S \vec{q} \cdot \underline{n} dS + \int_S \underline{\underline{\sigma}} \cdot \vec{u} \cdot \underline{n} dS + \int_V \dot{Q} dV + \int_V \rho \sum_{k=1}^{N_C} \omega_k f_k \cdot (\vec{u} + \vec{V}_k) dV \quad (1.18)$$

in cui \dot{Q} è il termine sorgente dell'energia per il volume (cioè l'energia dovuta a flusso di radiazione o altra forma di scintilla), mentre \vec{q} è il flusso termico. Il quarto termine a destra dell'uguale è la potenza prodotta dalle forze di volume sulla specie k . La forma

vettoriale è quindi:

$$\rho \frac{D\hat{E}}{Dt} = \frac{\partial}{\partial t}(\rho\hat{E}) + \nabla(\rho\vec{u}\hat{E}) = -\nabla\vec{q} + \nabla(\underline{\underline{\sigma}} \cdot \vec{u}) + \dot{Q} + \rho \sum_{k=1}^{N_C} \omega_k f_k^{\vec{}} \cdot (\vec{u} + \vec{V}_k) \quad (1.19)$$

Quest'ultima equazione rispetta il primo principio della termodinamica secondo il quale la variazione dell'energia nell'unità di tempo di una particella di fluido è uguale alla somma del lavoro fatto sulla particella e della velocità di apporto di calore alla particella.

1.3 Leggi costitutive

Le equazioni di conservazione viste nel paragrafo precedente costituiscono sostanzialmente un sistema di equazioni differenziali che può essere applicato a qualsiasi fluido. Il numero di equazioni però è decisamente inferiore a quello delle incognite e ciò rende il sistema indeterminato. Infatti le incognite sono la densità ρ , il vettore della velocità del fluido \vec{u} , la pressione p , l'energia \hat{E} , le N_C frazioni massive ω_k e in aggiunta il tensore degli sforzi $\underline{\underline{\sigma}}$, il flusso diffusivo molecolare \vec{j}_k e il flusso di energia \vec{q} . Con l'espressione "leggi costitutive" si intendono alcune relazioni che ci permettono di conoscere le ultime tre variabili incognite [9]. Esse dipendono dalle proprietà del fluido e per rendere possibile la chiusura del sistema (cioè la sua risoluzione) devono essere espresse in funzione delle incognite principali. Inoltre è necessaria un'equazione di stato che permetta di determinare le variabili termodinamiche della miscela di gas. Nei paragrafi seguenti sono mostrate le leggi costitutive per i fluidi Newtoniani e l'equazione dei gas perfetti.

1.3.1 Legge di Newton

Il tensore degli sforzi $\underline{\underline{\sigma}}$ per i fluidi Newtoniani sta ad indicare la velocità molecolare di trasporto della quantità di moto e si può scrivere utilizzando la forma generalizzata della legge di Newton della viscosità:

$$\underline{\underline{\sigma}} = 2\mu\underline{\underline{D}} - \left(p + \frac{2}{3}\mu\nabla\vec{u}\underline{\underline{I}} \right) = \left(2\mu\underline{\underline{D}} - \frac{2}{3}\mu\nabla\vec{u}\underline{\underline{I}} \right) - p\underline{\underline{I}} = \underline{\underline{\tau}} - p\underline{\underline{I}} \quad (1.20)$$

dove μ è la viscosità dinamica, $\underline{\underline{I}}$ è la matrice identità, p è la pressione statica e $\underline{\underline{D}}$ è il tensore delle deformazioni:

$$\underline{\underline{D}} = \frac{1}{2}[\nabla \cdot \vec{u} + (\nabla \cdot \vec{u})^T] \quad (1.21)$$

Il tensore indicato con il simbolo $\underline{\underline{\tau}}$ è la parte viscosa del tensore degli sforzi $\underline{\underline{\sigma}}$. Si può ottenere che:

$$\underline{\underline{\tau}} = 2\mu\underline{\underline{D}} - \frac{2}{3}\mu\nabla\vec{u}\underline{\underline{I}} \quad (1.22)$$

che poi in coordinate cartesiane risulta:

$$\sigma_{ij} = \tau_{ij} - p\delta_{ij} = \left(2\mu D_{ij} - \frac{2}{3}\mu\frac{\partial u_j}{\partial x_j}\delta_{ij}\right) - p\delta_{ij} \quad (1.23)$$

Invece il tensore delle deformazioni risulta:

$$D_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) \quad (1.24)$$

dove δ_{ij} è il delta di Krönecker che assume valore unitario se $i = j$ e valore nullo in caso contrario.

1.3.2 Legge di Fick

Il flusso diffusivo o la velocità diffusiva si possono ottenere risolvendo il seguente sistema [42]

$$\nabla X_k = \sum_{p=1}^{N_C} \frac{X_k X_p}{\Gamma_{kp}} (V_p - V_k) + (\omega_k - X_k) \frac{\nabla p}{p} + \frac{\rho}{p} \sum_{p=1}^{N_C} \omega_k \omega_p (f_k - f_p) \quad k = 1, \dots, N_C \quad (1.25)$$

dove Γ_{kp} è il coefficiente binario di diffusione massivo della specie k nella specie p e X_k è la frazione molare della specie k . Questo sistema è lineare e di dimensioni N_C^2 da risolvere in ogni direzione, per ogni punto e in ogni istante. Di solito per ragioni di costo computazionale si semplifica introducendo la legge di Fick. Tale legge permette di scrivere la velocità diffusiva o il flusso molecolare diffusivo come:

$$\omega_k \vec{V}_k = -\Gamma_{k,mix} \nabla \omega_k \quad (1.26)$$

che utilizzando la 1.16 si traduce in:

$$\vec{j}_k = -\rho \cdot \Gamma_{k,mix} \nabla \omega_k \quad (1.27)$$

dove $\Gamma_{k,mix}$ è il coefficiente di diffusione della specie k nella miscela.

1.3.3 Legge di Fourier

Il flusso di energia si valuta, come indicato nell'equazione 1.28, come somma di un primo contributo che rappresenta la legge di Fourier e un secondo termine associato alla diffusione delle specie con valore di entalpia differente (solo per miscele di specie diverse):

$$\vec{q} = -\lambda_t \nabla T + \rho \sum_{k=1}^{N_C} \hat{H}_k \omega_k \vec{V}_k \quad (1.28)$$

dove λ_t è la conducibilità termica e \hat{H}_k è l'entalpia specifica della specie k .

1.3.4 Equazione di stato

Attraverso l'equazione di stato si può esprimere una variabile di stato come funzione di altre due variabili del sistema come, per esempio

$$p = p(\rho, T) \quad (1.29)$$

Nelle applicazioni di questo lavoro di tesi le condizioni sono tali da accettare l'ipotesi che il comportamento della miscela di gas sia approssimabile al comportamento di un gas perfetto, pertanto l'equazione di stato si riduce a:

$$p = \frac{\rho}{M_w} RT \quad (1.30)$$

1.4 Equazioni di Navier-Stokes

Unendo le equazioni di conservazione e le equazioni costitutive analizzate nei paragrafi precedenti si crea un sistema di equazioni differenziali alle derivate parziali per fluidi Newtoniani:

Conservazione della massa

$$\frac{\partial \rho}{\partial t} + \nabla(\rho \vec{u}) = 0 \quad (1.31)$$

Conservazione della quantità di moto

$$\rho \frac{D\vec{u}}{Dt} = \rho \vec{f} - \left(\nabla p + \frac{2}{3} \mu \nabla \vec{u} \right) + \nabla \left[\mu (\nabla \cdot \vec{u} + (\nabla \cdot \vec{u})^T) \right] \quad (1.32)$$

Conservazione delle specie

$$\rho \frac{D\omega_k}{Dt} = \nabla (\rho \cdot \Gamma_{k,mix} \nabla \omega_k) + \dot{\Omega}_k \quad (1.33)$$

Conservazione dell'energia

$$\rho \frac{D\hat{E}}{Dt} = -\nabla(\rho \vec{u}) + \nabla(\underline{\tau} \cdot \vec{u}) + \nabla \left(\lambda_t \nabla T - \rho \sum_{k=1}^{N_C} \hat{H}_k \omega_k \vec{V}_k \right) + \dot{Q} + \rho \sum_{k=1}^{N_C} \omega_k \hat{f}_k \cdot (\vec{u} + \vec{V}_k) \quad (1.34)$$

Sull'ultima equazione scritta si possono effettuare alcuni passaggi ulteriori: se definiamo l'entalpia totale specifica \hat{H}_{tot} e l'entalpia specifica \hat{H} come:

$$\hat{H}_{tot} = \hat{E} + \frac{p}{\rho} \quad (1.35)$$

$$\hat{H} = \hat{U} + \frac{p}{\rho} \quad (1.36)$$

e introducendole all'interno della conservazione dell'energia, si ottengono le seguenti equazioni di conservazione dell'entalpia totale e specifica:

$$\rho \frac{D\hat{H}_{tot}}{Dt} = \frac{\partial p}{\partial t} - \nabla \vec{q} + \nabla(\underline{\tau} \cdot \vec{u}) + \dot{Q} + \rho \sum_{k=1}^{N_C} \omega_k \hat{f}_k \cdot (\vec{u} + \vec{V}_k) \quad (1.37)$$

$$\rho \frac{D\hat{H}}{Dt} = \frac{Dp}{Dt} - \nabla \vec{q} + \underline{\tau} : \nabla \vec{u} + \dot{Q} + \rho \sum_{k=1}^{N_C} \omega_k \hat{f}_k \cdot \vec{V} \quad (1.38)$$

si può inoltre indicare con \dot{Q}_{diss} il termine sorgente dovuto al calore viscoso che è sempre una quantità positiva:

$$\dot{Q}_{diss} = \underline{\tau} : \nabla \vec{u} > 0 \quad (1.39)$$

Le equazioni appena presentate costituiscono un problema, essendo di difficile implementazione nei codici *CFD*. Infatti viene usata l'entalpia sensibile \hat{H}_s , che contiene il termine $\sum_{k=1}^{N_C} \Delta \hat{H}_{f,k}^0 \cdot \omega_k$ in aggiunta all'entalpia specifica \hat{H} :

$$\hat{H}_s = \hat{H} - \sum_{k=1}^{N_C} \Delta \hat{H}_{f,k}^0 \cdot \omega_k \quad (1.40)$$

Utilizzando la conservazione delle specie, l'equazione dell'entalpia sensibile è ottenibile da quella dell'entalpia specifica:

$$\rho \frac{D\hat{H}_s}{Dt} = \dot{Q}_R + \frac{Dp}{Dt} + \nabla(\lambda_t \nabla T) - \nabla \left(\rho \sum_{k=1}^{N_C} \Delta \hat{H}_{s,k} \cdot \omega_k \vec{V}_k \right) + \dot{Q}_{diss} + \dot{Q} + \rho \sum_{k=1}^{N_C} \omega_k \vec{f}_k \cdot \vec{V}_k \quad (1.41)$$

dove il termine \dot{Q}_R è il calore rilasciato a causa delle reazioni chimiche:

$$\dot{Q}_R = - \sum_{K=1}^{N_C} \Delta \hat{H}_{f,k}^0 \dot{\Omega}_k \quad (1.42)$$

Solitamente si deriva un'equazione di conservazione riferita alla temperatura, se le condizioni sono lontane da quelle soniche

$$\rho \frac{D\hat{H}_s}{Dt} = \sum_{k=1}^{N_C} \hat{H}_{s,k} \rho \frac{D\omega_k}{Dt} + \rho \hat{C}_P \frac{DT}{Dt} \quad (1.43)$$

Introducendo questa equazione nella 1.41 si ottiene:

$$\rho \hat{C}_P \frac{DT}{Dt} = \dot{Q}_R^* + \frac{Dp}{Dt} + \nabla(\lambda_t \nabla T) - \left(\rho \sum_{k=1}^{N_C} C_{P,k} \omega_k \vec{V}_k \right) \cdot \nabla T + \dot{Q}_{diss} + \dot{Q} + \rho \sum_{k=1}^{N_C} \omega_k \vec{f}_k \cdot \vec{V}_k \quad (1.44)$$

con \dot{Q}_R^* da intendersi come il calore rilasciato dalla combustione:

$$\dot{Q}_R^* = - \sum_{K=1}^{N_C} \Delta \hat{H}_k \dot{\Omega}_k = - \sum_{k=1}^{N_C} \hat{H}_{s,k} \dot{\Omega}_k - \sum_{K=1}^{N_C} \Delta \hat{H}_{f,k}^0 \dot{\Omega}_k = - \sum_{k=1}^{N_C} \hat{H}_{s,k} \dot{\Omega}_k + \dot{Q}_R \quad (1.45)$$

1.5 Caratteristiche della turbolenza

I flussi turbolenti sono caratterizzati da alcune proprietà [18], elencate di seguito:

- sono fortemente non stazionari: riportando la velocità in funzione del tempo otterremo un comportamento caotico e *random*
- sono moti intrinsecamente tridimensionali: la velocità istantanea è sempre una funzione delle tre coordinate dello spazio
- il grado di vorticità è elevatissimo: lo *stretching* (allungamento) dei vortici è uno dei meccanismi principali che causano l'aumento dell'intensità della turbolenza.
- la turbolenza incrementa in misura significativa il mescolamento delle specie chimiche e più in generale di tutti i processi diffusivi
- si possono individuare all'interno della turbolenza delle strutture coerenti
- i moti turbolenti possono essere caratterizzati da fluttuazioni su un range di tempi e lunghezze caratteristiche estremamente ampi e questa è una delle cause di maggiore difficoltà nella simulazione numerica di tali moti.

Per risolvere problemi di natura turbolenta bisogna ovviamente partire dalle equazioni di Navier-Stokes viste nel paragrafo precedente. E' importante notare come i metodi per risolvere queste equazioni cambino completamente a seconda del regime di moto in cui ci si trova.

Esiste un parametro che caratterizza il tipo di flusso analizzato e discrimina tra flusso laminare e turbolento [18]. Questo parametro è il numero di Reynolds, che esprime il rapporto tra le forze inerziali e quelle viscosi.

$$Re = \frac{ul}{\nu} \tag{1.46}$$

dove u è la velocità caratteristica del fluido, l la lunghezza caratteristica del problema e ν la viscosità cinematica del fluido, cioè il rapporto tra la viscosità dinamica e la sua densità. Per fluidi in quiete cioè con velocità nulla, ovviamente anche $Re = 0$. Se invece il numero di Reynolds assume valori molto piccoli, significa che gli sforzi viscosi dominano nei confronti delle forze inerziali e il regime di moto è laminare. Viceversa, un valore molto alto del numero implica che gli sforzi viscosi siano quasi trascurabili rispetto alle

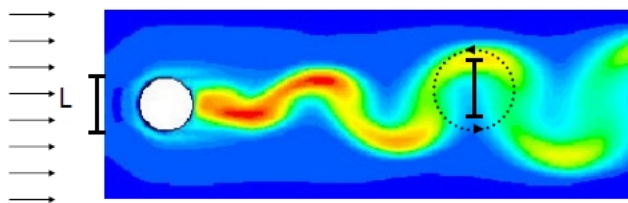


Figura 1.2: Rappresentazione della macroscale L

forze inerziali e il moto è turbolento. Ovviamente per ogni problema in esame e diversa geometria esistono dei valori limite che caratterizzano il tipo di moto. Esiste anche una fase definita di transizione che sta ad indicare quella zona intermedia tra i due valori limite in cui non si può stabilire la tipologia di flusso.

Il moto turbolento è caratterizzato dall'interazione di vortici di dimensione variabile attraverso i quali fluisce l'energia che viene dissipata in calore da parte della viscosità. Il processo di trasferimento di energia dai vortici più grandi verso scale sempre più piccole che ne causano la dissipazione è l'oggetto della *Teoria di Kolmogorov* [18]. Il primo elemento da considerare è la scala della turbolenza λ , che indica le dimensioni dei vortici. Ad ogni scala è associata un'ampiezza della fluttuazione di velocità, v_λ , che corrisponde all'incirca alla velocità del vortice di scala λ relativa al moto medio. Si può definire anche un numero di Reynolds turbolento, equivalente a $Re_\lambda = \frac{\lambda v_\lambda}{\nu}$. In ogni elemento di fluido sono presenti vortici di diverse dimensioni, che vanno dalla macroscale L alla microscale di Kolmogorov l_d . La macroscale è la misura delle massime dimensioni dei vortici turbolenti che si sviluppano durante il moto, oltre i quali le fluttuazioni turbolente non sono più correlate tra loro. Essa è sostanzialmente dell'ordine di grandezza dell'apparato che genera la turbolenza, perciò dipende in gran parte dalle dimensioni del sistema. Questi grandi vortici possiedono bassa frequenza e grande lunghezza d'onda (vedi figura 1.2). Come detto l'energia cinetica turbolenta si trasferisce dai vortici più grandi a quelli più piccoli. Questo succede senza dissipazione di energia fino a quando il numero di Reynolds turbolento si mantiene elevato, in quanto le forze inerziali rendono trascurabili gli sforzi viscosi, responsabili della dissipazione. Quando la scala raggiunge valori tali per cui $Re_\lambda \simeq 1$, la dissipazione prevale sul trasferimento e quindi tutta l'energia trasferita fino a quel momento viene dissipata. La microscale di Kolmogorov è la scala al di sotto della quale non è più possibile trasferire energia ma solo dissiparla. Pertanto non esistono vortici di scala più piccola della microscale.

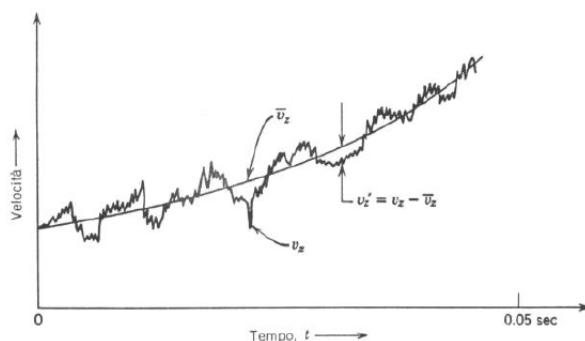


Figura 1.3: Oscillazione di una componente della velocità intorno al valore medio

1.6 Modellazione della turbolenza

Ritornando alle equazioni di Navier-Stokes, vediamo come queste risentono del cambiamento di regime di moto. Tutte queste equazioni contengono dei termini non lineari, che sono espressione degli sforzi inerziali. Ad esempio nel caso della conservazione della quantità di moto (equazione 1.32) il termine non lineare è dato dalla derivata sostanziale $\frac{D\vec{u}}{Dt}$. A questo punto è chiaro come questi termini diventino importanti quando le forze inerziali dominano sugli sforzi viscosi, ovvero in condizioni in cui il numero di Reynolds è molto alto e il regime è quindi in condizioni turbolente. Proprio per questo motivo nel caso in cui il numero di Reynolds sia basso e il moto quindi laminare, le equazioni possono essere risolte in maniera abbastanza agevole nelle variabili istantanee (\vec{u} nel caso in esame), anche in maniera numerica. Il campo di moto può quindi essere definito.

Diverso è il caso in cui il regime sia turbolento. Come già detto esso è fortemente non stazionario. Tutte le variabili istantanee si comportano in maniera caotica e non determinabile puntualmente, pertanto non è possibile lavorare con queste per ottenere considerazioni deterministiche sul campo di moto. In pratica non è possibile definire il valore locale istantaneo delle variabili e non si possono integrare le equazioni del moto nelle variabili stesse. Il problema però può essere affrontato facendo ricorso a considerazioni di tipo statistico [18]. Il moto turbolento possiede infatti delle proprietà *medie temporali* molto significative: infatti pur non essendo assicurata in esso nessuna riproducibilità locale istantanea, è invece assicurata una riproducibilità media temporale, senza la quale non si potrebbe fare alcuna predizione relativa a questo tipo di moto. Questo aspetto è messo in luce in figura 1.3. Tale media sarà fatta su intervalli di tempo abbastanza grandi rispetto al periodo delle fluttuazioni turbolente, in maniera tale da assicurare la riproducibilità dei valori mediati locali. Le grandezze fluttuanti possono essere rappresentate da una serie di

contributi funzione del tempo. Ad uno di questi contributi si associa un periodo caratteristico più grande degli altri (*periodo fondamentale*), al quale sono collegabili i periodi di tutti gli altri contributi. E' come se esistesse una frequenza fondamentale alla quale è associabile un tempo caratteristico. Mediando su tale tempo le variabili si ottengono delle grandezze che, date in ingresso condizioni al contorno stazionarie, si comportano mediamente in maniera stazionaria. Per poter calcolare i valori istantanei in ogni punto, ai valori mediati si devono sovrapporre dei contributi fluttuanti, che variano localmente in funzione del tempo. E' da notare che, dal momento che la turbolenza dissipa grosse quantità di energia coinvolte nel movimento, il suo innesco è legato strettamente all'esistenza di forze viscoso contrapposte a forze inerziali. Le forze viscoso sono legate ai gradienti di velocità (che presentano il loro massimo in prossimità di pareti solide) e perciò si può tendenzialmente affermare che sono proprio gli ostacoli solidi nel fluido a generare la turbolenza, che poi si sviluppa anche a distanza dal solido stesso. Riassumendo, le equazioni di bilancio non hanno in base a quanto detto alcuna soluzione "deterministica" in regime turbolento. Da esse, tuttavia, si possono ricavare almeno delle informazioni riguardanti i valori medi locali (di velocità, pressione, composizione e temperatura). Quello che viene fatto perciò è mediare le equazioni stesse. Di seguito vengono introdotti i due metodi di mediazione più comuni, la mediazione di Reynolds e quella di Favre. Nel capitolo successivo verrà chiarito come le mediazioni possano essere utilizzate dai modelli matematici che rendono possibile una definizione accettabile del campo di moto turbolento.

1.6.1 Mediazione secondo Reynolds

La quantità media $\bar{\phi}$ può essere definita in molti modi. Consideriamo il seguente:

$$\bar{\phi}(x_i, t) = \frac{1}{N} \sum_{k=1}^N \phi^k(x_i, t) \quad (1.47)$$

dove N è il numero di campioni. Il valore istantaneo della variabile ϕ è pari alla somma del valore medio e della sua fluttuazione ϕ' :

$$\phi(x_i, t) = \bar{\phi}(x_i, t) + \phi'(x_i, t) \quad (1.48)$$

Il valore medio dell'espressione precedente $\bar{\phi}$ si definisce come media temporale di ϕ con intervallo di mediazione Δt sufficientemente maggiore della scala temporale dei moti turbolenti:

$$\bar{\phi}(x_i, t) = \frac{1}{\Delta t} \int_t^{t+\Delta t} \phi(x_i, t) dt \quad (1.49)$$

La media delle fluttuazioni è nulla per definizione stessa della fluttuazione:

$$\overline{\phi'}(x_i, t) = \frac{1}{\Delta t} \int_t^{t+\Delta t} \phi'(x_i, t) dt = 0 \quad (1.50)$$

Considerando invece la media del prodotto delle fluttuazioni di due variabili ϕ' e ψ' , essa è diversa da zero:

$$\overline{\phi'\psi'}(x_i, t) = \frac{1}{\Delta t} \int_t^{t+\Delta t} \phi'(x_i, t) \cdot \psi'(x_i, t) dt \neq 0 \quad (1.51)$$

La mediazione secondo Reynolds ad esempio dell'equazione di continuità, risulta:

$$\frac{\partial \bar{\rho}}{\partial t} + \nabla (\bar{\rho}\bar{u} + \overline{\rho'u'}) = 0 \quad (1.52)$$

1.6.2 Mediazione secondo Favre

L'equazione di continuità mediata secondo Reynolds contiene la nuova quantità $\overline{\rho'u'}$ incognita che necessita di essere modellata. La mediazione della densità secondo Reynolds produce, a seconda dell'equazione da mediare, molti altri termini incogniti $\overline{\rho'\phi'}$ costituiti dalla fluttuazione di densità e dalla fluttuazione della generica quantità ϕ , in quanto nei flussi reattivi la densità è soggetta a variazioni. La mediazione secondo Favre, essendo una mediazione pesata sulla massa, risolve questo problema. Allo stesso modo di prima si può scrivere:

$$\phi(x_i, t) = \tilde{\phi}(x_i, t) + \phi''(x_i, t) \quad (1.53)$$

e la quantità $\tilde{\phi}$ è definita come:

$$\tilde{\phi}(x_i, t) = \frac{\overline{\rho\phi}}{\bar{\rho}} \quad (1.54)$$

La fluttuazione ϕ'' gode delle seguenti importanti proprietà:

$$\tilde{\phi}''(x_i, t) = \frac{\overline{\rho(\phi - \tilde{\phi})}}{\bar{\rho}} = 0 \quad (1.55)$$

$$\overline{\rho\phi''}(x_i, t) = 0 \tag{1.56}$$

$$\overline{\phi''}(x_i, t) = -\frac{\overline{\rho'\phi'}}{\bar{\rho}} \neq 0 \tag{1.57}$$

Capitolo 2

CFD e modelli di turbolenza

2.1 Introduzione

Per risolvere problemi di tipo turbolento si utilizza un approccio numerico. La *fluidodinamica computazionale (CFD, Computational Fluid Dynamics)* è la tecnica che permette lo studio dei problemi di fluidodinamica attraverso l'uso di un calcolatore. Essa viene utilizzata moltissimo sia in ambito industriale sia di ricerca, in tutti i problemi che coinvolgono dei fluidi. Come già accennato nel capitolo precedente, i problemi studiati in questo lavoro di tesi, ossia problemi di combustione in regime turbolento, sarebbero impossibili da risolvere senza l'ausilio della *CFD*. Essa si basa su modelli, cioè rappresentazioni semplificate della realtà fisica. Ovviamente questi modelli non garantiscono una assoluta validità dei risultati, anche perchè le equazioni differenziali, che contengono già semplificazioni, devono per poter essere risolte essere prima trasformate in algebriche attraverso una discretizzazione, che comporta ulteriori approssimazioni. Il metodo di discretizzazione che verrà qui presentato e di cui fanno uso i programmi utilizzati per il nostro lavoro viene detto *metodo dei volumi finiti*. Infine nella risoluzione delle equazioni si utilizzano degli algoritmi risolutivi basati su metodi iterativi che non portano ovviamente a una soluzione esatta del problema. Le equazioni sono risolte in maniera *segregata* e per calcoli stazionari ci si avvale di un algoritmo particolare definito *SIMPLE*.

Nonostante l'impossibilità di fornire una soluzione precisa l'uso della *CFD* comporta però notevoli vantaggi, poichè è possibile descrivere eventi complessi analizzando gli eventi elementari che li caratterizzano. Si possono studiare moltissimi diversi casi con costi e tempi molto inferiori rispetto alla pratica sperimentale, che talvolta può anche essere difficoltosa

o addirittura impraticabile; inoltre è possibile analizzare la sensitività di tali fenomeni a variazioni di condizioni operative o di geometria in maniera molto veloce.

In questo capitolo analizzeremo nel dettaglio gli elementi che compongono un codice di calcolo *CFD*, che sono stati accennati brevemente in questa introduzione. Si noti che nei prossimi paragrafi si è cercato di essere sintetici e di spiegare i metodi e le tecniche in maniera discorsiva, senza entrare nei dettagli matematici. Per un approfondimento si veda [48]. In seguito ampio spazio è riservato alla descrizione dei modelli utilizzati per descrivere la turbolenza. Infine si introdurranno brevemente i due codici di calcolo utilizzati in questo lavoro di tesi, *Fluent* e *OpenFOAM*.

2.2 Discretizzazione e creazione della *mesh*

Abbiamo già accennato che la *CFD* cerca di risolvere i problemi di fluidodinamica trattando un fluido continuo in un modo discreto. Il metodo più semplice per raggiungere questo obiettivo è discretizzare il dominio dello spazio in un certo numero di celle che vanno a formare una griglia (*mesh*) di calcolo e quindi applicare degli algoritmi su ogni cella per risolvere le equazioni del moto in maniera iterativa. La *mesh* può essere regolare o irregolare, le celle possono avere svariate forme a seconda del problema da studiare e possono essere graduate, ovvero essere più fitte in alcune regioni e più lasche in altre. Questo dipende dall'accuratezza richiesta rispetto ai tempi di calcolo. Normalmente un'elevata risoluzione spaziale è necessaria in regioni dove i gradienti e l'intensità di miscelazione sono elevati. E' chiaro che solo una buona ottimizzazione del dominio di calcolo può portare a ottenere soluzioni accettabili in tempi ragionevoli. I metodi più utilizzati per discretizzare la *mesh* sono tre:

- Metodo ai volumi finiti (*FVM*, *Finite Volume Method*)
- Metodo alle differenze finite (*Finite Difference Method*)
- Metodo agli elementi finiti (*FEM*, *Finite Element Method*)

Essi ovviamente differiscono nei metodi utilizzati per rendere algebriche le equazioni differenziali. Possono funzionare ovviamente sia in regime stazionario, quindi considerando nulla la derivata rispetto al tempo, sia in regime transitorio. Tra i tre il più utilizzato è il metodo ai volumi finiti, su cui si basano anche i codici utilizzati in questo lavoro. Per

questo motivo il prossimo paragrafo viene dedicato interamente alla spiegazione di questa tecnica. Per quanto riguarda il metodo alle differenze finite, non è molto utilizzato e si basa sull'approssimazione delle derivate o tramite sviluppi in serie di Taylor o tramite *fitting* attraverso una funzione polinomiale. Ovviamente in questo caso l'approssimazione è introdotta dal fatto che in entrambi i casi si ha un errore dovuto al troncamento. Le equazioni in questo caso sono risolte sui nodi della griglia.

Il metodo agli elementi finiti invece è molto utilizzato e si presta in particolare quando il dominio di calcolo ha forma complessa, è variabile, oppure quando l'accuratezza richiesta alla soluzione non è omogenea sul dominio, o infine quando la soluzione non è regolare. Il metodo si basa sulla suddivisione del continuo in elementi di forma definita e dimensioni contenute; ogni singolo elemento finito viene considerato un campo di integrazione numerica di caratteristiche omogenee. Su ciascun elemento caratterizzato da questa forma elementare, la soluzione del problema è espressa dalla combinazione lineare di funzioni dette funzioni di base o funzioni di forma. L'esempio tipico è quello che fa riferimento a funzioni polinomiali, cioè la soluzione complessiva del problema viene approssimata con una funzione polinomiale a pezzi. Il numero di coefficienti che identifica la soluzione su ogni elemento è dunque legato al grado del polinomio scelto. Questo, a sua volta, governa l'accuratezza della soluzione numerica trovata [49]

2.2.1 Metodo ai volumi finiti

Il metodo dei volumi finiti [48] prevede che l'equazione differenziale da risolvere sia posta in termini integrali, come nell'equazione 1.1, che viene qui riportata:

$$\int_V \frac{\partial}{\partial t}(\rho\phi) dV + \int_S \rho\phi\vec{u} \cdot \underline{n} dS = \int_S \vec{Q}_S(\phi) \cdot \underline{n} dS + \int_V Q_V(\phi) dV \quad (2.1)$$

L'idea su cui si basa il metodo dei volumi finiti è semplice: il dominio di calcolo viene suddiviso in un numero finito di piccoli volumi di controllo costruendo una griglia di calcolo appropriata. In questo metodo, a differenza di quello delle differenze finite, le equazioni vengono risolte non sui nodi della griglia ma sui volumi di controllo. Si va cioè a determinare il valore della variabile incognita ϕ al centro di ciascun volume di controllo. L'equazione scritta sopra può essere applicata su tutti i volumi di controllo ma anche su tutto il dominio di calcolo. Infatti se andassimo a sommare le equazioni per i singoli volumi di controllo, otterremmo l'equazione di conservazione globale. Quindi la proprietà di conservazione globale viene garantita utilizzando questo metodo e ciò rappresenta uno

dei principali vantaggi che lo rendono così importante nel campo della *CFD*. Per ottenere un'equazione algebrica per ciascun volume di controllo, gli integrali di superficie e di volume devono essere opportunamente discretizzati attraverso delle formule di quadratura.

- Integrali di superficie: l'integrale di superficie si può scrivere sempre come somma degli integrali di superficie su tutte le facce che costituiscono la superficie totale di controllo del singolo volume:

$$\int_S f dS = \sum_k \int_{S_k} f dS \quad (2.2)$$

Prendendo in considerazione una singola faccia della cella, per calcolarne l'integrale esattamente, bisognerebbe conoscere la funzione integranda f , informazione che ovviamente non è disponibile, perchè si hanno a disposizione solo le informazioni relative ai centroidi dei volumi di controllo. Perciò è necessario ricostruire le informazioni attraverso l'uso delle sole variabili definite nei centroidi. Per ottenere questo obiettivo è necessaria una doppia approssimazione: innanzitutto l'integrale è approssimato utilizzando generalmente le informazioni di un solo punto della faccia e in secondo luogo i valori scelti sulla faccia sono a loro volta disponibili solo in forma approssimata, poichè sono ottenuti a partire dalle uniche informazioni disponibili, ovvero quelle dei valori calcolati nel centro delle celle della griglia. L'approssimazione più semplice per la descrizione dell'integrale di superficie è l'uso della *regola del punto di mezzo*: l'integrale viene approssimato in pratica come il prodotto della funzione integranda al centro della faccia per l'area della faccia considerata. Il valore di f al centro della faccia non è però disponibile, ma può essere ottenuto solo attraverso una opportuna interpolazione. Altre approssimazioni di questo integrale utilizzano la *regola del trapezio* e la *regola di Simpson*

- Integrali di volume: anche in questo caso l'approccio più semplice consiste nell'andare a sostituire l'integrale di volume con il prodotto del valor medio della funzione integranda per il volume di controllo. Il problema ovviamente sta nell'individuare il valor medio. La cosa più logica da fare è prenderlo esattamente pari al valore della funzione nel centroide del volume di controllo. Questo valore è già disponibile, in quanto tutte le variabili principali e quelle derivate sono note nei centroidi della griglia di calcolo senza la necessità di ricorrere ad alcuna interpolazione. Si possono ottenere approssimazioni di ordine superiore utilizzando il valore della funzione in

più punti; queste informazioni possono essere ottenute solo andando ad interpolare i valori nodali, utilizzando opportune formule di interpolazione. Sono possibili in questo senso numerose soluzioni. Le principali, da cui poi se ne possono sviluppare altre più complesse, sono:

Interpolazione Upwind (UDS): è una soluzione molto semplice che consiste nel costruire un' interpolazione lineare di tipo *backward* o *forward* a seconda della direzione del vettore della velocità. In sostanza il valore della funzione nel centro della faccia considerata è pari al valore in uno dei due centroidi le cui celle hanno in comune la faccia in questione. Questo a seconda appunto della direzione della velocità sulla faccia stessa. Questa soluzione è l'unica che soddisfa il criterio di *boundedness* (cioè che non comporta la possibilità di soluzioni oscillanti) in maniera incondizionata. Tuttavia è un'approssimazione poco accurata, che introduce una diffusività numerica non trascurabile.

Interpolazione lineare centrata (CDS): consiste in una interpolazione lineare che prende in considerazione i due centroidi già visti in precedenza e interpola per ottenere il valore nella faccia richiesta. E' uno schema accurato al secondo ordine e l'errore di troncamento in questo caso è proporzionale al quadrato del passo della griglia, sia con griglia uniforme che variabile. Come ogni schema di ordine due o superiore, anche questo tipo di approssimazione può portare a soluzioni oscillanti. Comunque si tratta dello schema del secondo ordine più utilizzato e semplice da implementare.

Interpolazione quadratica Upwind (QUICK): il passaggio successivo è ovviamente l'approssimazione del profilo della variabile in questione attraverso una parabola piuttosto che una retta. Si devono pertanto utilizzare punti ulteriori della griglia. Questo schema è ovviamente più complesso del metodo *CDS*; esso è comunque abbastanza popolare e porta a errori di troncamento del terzo ordine sia su griglie a passo uniforme che a passo variabile. Se però questo schema viene utilizzato insieme alla *regola del punto di mezzo* per l'approssimazione degli integrali di superficie, l'approssimazione globale risultante è tuttavia del secondo ordine. Pertanto, nonostante il metodo *QUICK* sia più accurato del metodo *CDS*, entrambi alla fine mostrano una convergenza pratica del secondo ordine e le differenze risultano spesso essere molto piccole.

Schemi di ordine superiore: interpolazioni di ordine superiore hanno senso essenzialmente solo quando gli integrali sono risolti utilizzando approssimazioni di ordine elevato. Ad esempio se si utilizza la *regola di Simpson* per gli integrali di superficie è necessario interpolare con polinomi che siano almeno di terzo ordine, il che porta

ad un'errore di interpolazione del quarto ordine.

2.2.2 Condizioni al contorno

Ogni cella di calcolo porta alla scrittura di una equazione algebrica dedicata alla variabile ϕ . Gli integrali di volume si calcolano allo stesso modo anche per le celle che appartengono ai contorni della griglia. Viceversa, i flussi attraverso le facce dei volumi di controllo che coincidono con i contorni richiedono dei trattamenti speciali. Questi flussi sul contorno devono essere noti oppure essere espressi in funzione delle sole variabili interne al dominio di calcolo. Dal momento che non portano equazioni aggiuntive, non devono allo stesso tempo introdurre nuove incognite. Poichè inoltre non esistono punti o nodi esterni ai contorni, le approssimazioni dovranno utilizzare interpolazioni *one side* oppure opportune estrapolazioni.

2.3 Risoluzione delle equazioni algebriche

Dopo aver discretizzato le equazioni differenziali esse devono essere risolte numericamente. Qui bisogna innanzitutto fare una prima distinzione. Nel caso in cui il regime sia transitorio si deve risolvere anche il termine di derivata in funzione del tempo, che rende il problema molto più complicato. Infatti sarà necessaria, oltre alla discretizzazione spaziale, anche una sorta di discretizzazione temporale, che permetta di arrivare a una soluzione del sistema di equazioni differenziali. Molto più semplice è il caso in cui il regime sia stazionario. Infatti il termine derivativo nel tempo si annulla e il problema si risolve in un sistema di equazioni algebriche, normalmente non lineari. Esistono numerosi metodi per risolvere sistemi di *ODE*, ma dato che i problemi studiati in questo lavoro di tesi sono tutti stazionari, ci concentreremo solo sul metodo utilizzato per risolvere il sistema algebrico. In questo caso vi sono due approcci possibili: il primo consiste nel risolvere tutte le equazioni contemporaneamente, cioè in maniera accoppiata. In sostanza si va a risolvere il sistema nella sua interezza. Il metodo utilizzato invece dai programmi più diffusi di fluidodinamica computazionale compresi quelli utilizzati in questo lavoro utilizzano una risoluzione di tipo segregato. In pratica si utilizza un falso transitorio nel quale ad ogni step iterativo si risolve un'equazione alla volta, tenendo una variabile come incognita e utilizzando come valori delle altre incognite i valori dello step iterativo precedente. Procedendo in questa maniera si raggiunge un punto in cui tutti i valori delle variabili incognite rimangono stazionari

e il sistema si può considerare risolto. L'analisi della convergenza può essere effettuata valutando i residui. Questi sono una misura della velocità di cambiamento delle correzioni sulle variabili calcolate ad ogni step, dove per correzioni si intendono i cambiamenti delle variabili stesse da uno step iterativo a quello precedente. I residui si calcolano dividendo la correzione per lo step temporale. Resta ancora un problema da risolvere: la risoluzione dell'equazione di continuità e di quella dei momenti effettuata con questo approccio porta a grosse instabilità, perciò è stato necessario sviluppare un ulteriore algoritmo in grado di gestire l'accoppiamento pressione-velocità.

2.3.1 Metodo *SIMPLE*

Prima di introdurre l'algoritmo *SIMPLE* è necessario distinguere tra due diversi tipi di simulazione numerica: se il flusso viene considerato incomprimibile significa che lungo il dominio di calcolo non si possono avere grosse variazioni di pressione, o salti di pressione. Questo introduce sostanzialmente una semplificazione notevole. Un flusso si può considerare incomprimibile quando le velocità in gioco sono molto basse rispetto alla velocità del suono, tipicamente il numero di Mach deve essere inferiore a 0.3. Quando il numero di Mach supera questo valore si parla invece di flusso comprimibile e la semplificazione non può più essere attuata.

Nel caso in cui il flusso sia incomprimibile si possono verificare ancora due casi. Il primo è quello in cui il fluido stesso sia incomprimibile (ad esempio un liquido), la densità rimane perciò costante e l'equazione di continuità 1.6 assume una forma di questo tipo:

$$\nabla \vec{u} = 0 \tag{2.3}$$

In questa equazione non compare esplicitamente la pressione e sebbene la velocità sia funzione della pressione, la sua risoluzione diretta porta a instabilità. Per questo si adotta una tecnica: partendo da un valore di primo tentativo sulla velocità si inserisce all'interno dell'equazione dei momenti e si calcola il valore di pressione corrispondente. A questo punto si calcola attraverso la 2.3 il valore di velocità che soddisfa l'equazione per quel valore di pressione e si ripete il procedimento fino a convergenza. In questo consiste sostanzialmente il metodo *SIMPLE*. Nei problemi studiati però i gas utilizzati non sono da considerarsi fluidi incomprimibili, e pertanto l'equazione di continuità non si traduce direttamente nella 2.3. Per poter utilizzare questo metodo ugualmente si ricorre a un artificio. Essendo il flusso incomprimibile le differenze di pressione nel dominio saranno

minime, pertanto è possibile utilizzare due diversi tipi di pressione. Da una parte la pressione fluidodinamica che è quella che si ottiene risolvendo le equazioni e che è variabile lungo la griglia. Dall'altra parte la pressione termodinamica, che è costante e solitamente pari a 1 atmosfera, che viene usata nel calcolo della densità tramite la legge dei gas perfetti secondo

$$\rho = \frac{p \cdot M_w}{R \cdot T} \quad (2.4)$$

dove M_w è il peso molecolare e R la costante dei gas perfetti. Quindi utilizzando sempre la stessa pressione, la densità non dipende più da questa e perciò è possibile applicare lo stesso metodo di prima per calcolare in maniera accoppiata velocità e pressione.

2.4 Modelli di turbolenza

Andando a risolvere numericamente i problemi di turbolenza si incontrano notevoli difficoltà dovute alle ragioni già citate. I fenomeni turbolenti creano grossi problemi, tanto che all'aumentare del numero di Reynolds e al complicarsi delle geometrie aumentano in maniera rilevante i costi computazionali. Una delle cause sta nel fatto che, all'aumentare del numero di Reynolds, si allargano notevolmente gli intervalli di scale in gioco, sia spaziali che temporali ed è pertanto necessario utilizzare griglie di calcolo molto raffinate e passi temporali alquanto ristretti. La grande estensione delle scale in gioco deve essere trattata accuratamente dato che vortici anche di dimensioni molto diverse interagiscono tra di loro. Più precisamente sappiamo dalla teoria di Kolmogorov [18], che il rapporto tra la scala dissipativa l_d e la scala integrale L è dell'ordine

$$\frac{l_d}{L} \simeq Re^{-\frac{3}{4}} \quad (2.5)$$

In una simulazione numerica corretta, che deve coprire tutto il range di lunghezze, sarebbe necessaria una griglia con $N^3 = Re^{\frac{9}{4}}$ punti. Inoltre è stato dimostrato da Blazek che il numero di passi temporali è proporzionale a Re^3 . E' quindi evidente che per numeri di Reynolds alti come quelli che caratterizzano il moto turbolento, non è possibile realizzare praticamente un numero così elevato di passi temporali. Per risolvere numericamente un flusso turbolento è stato allora necessario sviluppare delle tecniche per semplificare i problemi permettendo di diminuire il numero di punti della griglia diminuendo nello stesso

tempo il costo computazionale. Si possono distinguere tre livelli di approccio, che verranno qui presentati brevemente.

- Il primo livello è basato sulla *DNS* (Direct Numerical Simulation) [42], dove le equazioni di Navier-Stokes vengono risolte senza alcun modello per i flussi turbolenti (ovviamente dopo aver discretizzato il dominio). Tutte le scale turbolente sono perciò determinate esplicitamente e i loro effetti sulla combustione sono catturati dalla simulazione. Ovviamente per poter determinare tutte le scale di turbolenza è necessaria una griglia molto fitta e il costo computazionale di una simulazione di questo tipo è altissimo. E' per questo che la tecnica *DNS* sta prendendo piede solo negli ultimi anni, grazie alla possibilità di utilizzare calcolatori sempre più performanti. Nonostante abbia rivoluzionato l'analisi della combustione turbolenta, il suo utilizzo è comunque ancora limitato a studi accademici in flussi molto semplici, ad esempio la combustione turbolenta in un piccolo spazio cubico.
- Nel secondo livello di approccio, chiamato *LES* (*Large Eddy Simulation*) [42], non tutte le scale turbolente vengono calcolate esplicitamente, bensì solo le più grandi. Le scale più piccole viceversa sono modellate utilizzando regole di chiusura dette *subgrid*. Le equazioni di bilancio sono ottenute filtrando le equazioni istantanee. Una simulazione *LES* di flussi reagenti determina la posizione istantanea di un fronte di fiamma a larga scala, ma un modello *subgrid* è richiesto per tenere in considerazione gli effetti delle piccole scale di turbolenza sulla combustione.
- Il terzo livello, *RANS* (*Reynolds Averaged Navier Stokes*) [42] è stato storicamente il primo approccio possibile in quanto la simulazione istantanea del flusso in condizioni turbolente era impossibile. Questa tecnica è stata perciò sviluppata in modo da risolvere i problemi per i valori medi delle variabili, secondo l'approccio statistico che consiste nel mediare nel tempo le diverse equazioni di cui si è parlato nel paragrafo 1.6. A causa della non linearità delle equazioni di Navier-Stokes, nelle equazioni mediate compaiono dei termini addizionali, noti come sforzi di Reynolds, che rendono il numero di incognite superiore al numero di equazioni, rendendo il problema indeterminato. E' possibile ricavare equazioni esatte per tali sforzi, ma queste contengono a loro volta altre correlazioni incognite di ordine superiore e andando avanti se ne generano sempre di più. Da notare è che all'aumentare del loro ordine non è detto che questi termini diventino trascurabili. In questo consiste il cosiddetto *problema di chiusura della turbolenza*. Per chiudere il sistema è comunque necessario troncare

la gerarchia delle equazioni a un certo ordine, anche se questo comporta una perdita di informazioni. Si ricorre pertanto a modelli di turbolenza che permettono di esprimere le correlazioni incognite in funzione delle grandezze mediate, in maniera esplicita o tramite la risoluzione di equazioni di trasporto.

In figura 2.1 vengono messe in luce molto chiaramente le differenze tra i tre modelli appena presentati. Si nota chiaramente come l'approccio *DNS* segua tutte le fluttuazioni della turbolenza. L'approccio *RANS* al contrario tiene conto solo del valore medio delle fluttuazioni, mentre il *LES* si pone tra i due precedenti andando a seguire solo le macrofluttuazioni e non quelle più piccole. Come già accennato, in termini di costi computazionali l'approccio *DNS* è il più preciso quindi anche quello che richiede maggiori risorse ed è perciò limitato a numeri di Reynolds molto bassi e geometrie semplici, richiedendo un numero molto elevato di celle della griglia. Il metodo *LES* lavora con griglie più lasche e può essere usato anche con numeri di Reynolds relativamente alti, ma richiede dei modelli *subgrid*, dai quali dipendono la qualità del calcolo e l'accuratezza dei risultati. La tecnica *RANS* infine è molto utilizzata nelle pratiche ingegneristiche, poichè è molto meno onerosa, ma la sua validità è strettamente legata ai modelli di chiusura delle equazioni turbolente e al modello di combustione utilizzato. Nel paragrafo successivo di questo capitolo analizzeremo in dettaglio l'approccio *RANS*, in quanto sta alla base del funzionamento di entrambi i codici utilizzati in questo lavoro.

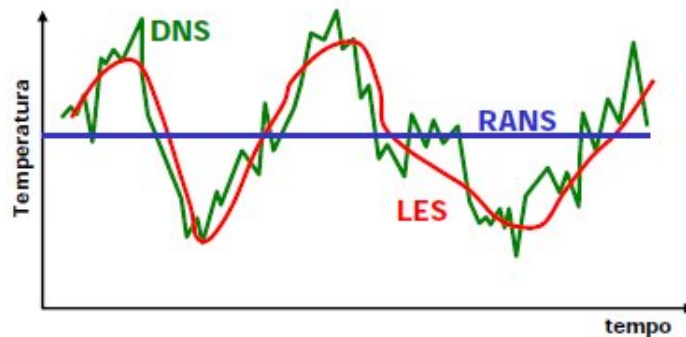


Figura 2.1: Evoluzione della temperatura nel tempo simulata con tecnica *DNS*, *RANS* e *LES* in un bruciatore turbolento

2.5 Approccio *RANS*

2.5.1 Equazioni mediate

Di seguito vengono mostrate le espressioni delle equazioni di conservazione nella forma mediata secondo Favre (paragrafo 1.6.2). Per maggiori dettagli sulla derivazione di queste espressioni si consultino [16, 51, 10, 20]

Conservazione della massa

$$\frac{\partial \bar{\rho}}{\partial t} + \nabla \cdot (\bar{\rho} \tilde{\mathbf{u}}) = 0 \quad (2.6)$$

Conservazione della quantità di moto

$$\frac{\partial}{\partial t} (\bar{\rho} \tilde{\mathbf{u}}) + \nabla \cdot (\bar{\rho} \tilde{\mathbf{u}} \tilde{\mathbf{u}}) = -\nabla \bar{p} + \nabla \cdot \bar{\underline{\underline{\tau}}} + \nabla \cdot (\bar{\rho} \widetilde{\mathbf{u}'' \mathbf{u}''}) + \bar{\rho} \vec{f} \quad (2.7)$$

Conservazione delle specie

$$\frac{\partial}{\partial t} (\bar{\rho} \tilde{\omega}_k) + \nabla \cdot (\bar{\rho} \tilde{\mathbf{u}} \tilde{\omega}_k) = \nabla \cdot (\overline{\rho \cdot \Gamma_{k,mix} \nabla \omega_k}) + \bar{\Omega}_k - \nabla \cdot (\bar{\rho} \widetilde{\omega_k'' \mathbf{u}''}) \quad (2.8)$$

Conservazione dell'energia

$$\frac{\partial}{\partial t} (\bar{\rho} \tilde{H}_s) + \nabla \cdot (\bar{\rho} \tilde{\mathbf{u}} \tilde{H}_s) = \bar{Q}_R + \frac{D\bar{p}}{Dt} + \nabla \cdot (\bar{\lambda} \nabla T) + \bar{Q}_{diss} + \bar{Q} - \nabla \cdot \left(\rho \sum_{k=1}^{N_C} \hat{H}_{s,k} \omega_k \vec{V}_k \right) - \nabla \cdot (\bar{\rho} \widetilde{\mathbf{u}'' \hat{H}_s''}) \quad (2.9)$$

dove $\frac{D\bar{p}}{Dt} = \frac{\partial \bar{p}}{\partial t} + \bar{\mathbf{u}} \nabla \bar{p} = \frac{\partial \bar{p}}{\partial t} + \tilde{\mathbf{u}} \nabla \bar{p} + \bar{\mathbf{u}}'' \nabla \bar{p}$ e nell'ultima, per semplicità, si è trascurato il termine dovuto a forze esterne di volume. Si arriva quindi ad una forma delle equazioni molto simile a quella derivante dalla mediazione con Reynolds, con i vantaggi derivanti però dalla media di Favre. I termini aggiuntivi incogniti che richiedono chiusura sono:

- il tensore degli sforzi di Reynolds $\nabla \cdot (\bar{\rho} \widetilde{\mathbf{u}'' \mathbf{u}''})$
- il flusso turbolento delle specie $\nabla \cdot (\bar{\rho} \widetilde{\mathbf{u}'' \omega''})$
- il flusso turbolento di entalpia $\nabla \cdot (\bar{\rho} \widetilde{\mathbf{u}'' H_s''})$

Il procedimento attraverso il quale si forniscono le equazioni aggiuntive per la determinazione di questi termini è definito *chiusura della turbolenza o modellazione della turbolenza*. Inoltre la mediazione di Favre introduce anche il termine $\overline{\dot{\Omega}_k}$ che rappresenta la velocità di formazione della specie k : la modellazione di questo termine è trattata nel capitolo seguente che riguarda l'interazione tra cinetica e turbolenza e costituisce uno dei problemi maggiori nella simulazione numerica di flussi reattivi per via della sua non linearità. Ora verranno analizzati invece i modelli che è possibile utilizzare per chiudere le equazioni rispetto alle tre categorie di variabili aggiuntive incognite che richiedono chiusura.

2.5.2 Flussi turbolenti di specie ed entalpia

Solitamente un flusso può essere visto come direttamente proporzionale ad un gradiente. In particolare i flussi dovuti alla turbolenza di specie ed entalpia si possono vedere come:

$$\overline{\rho \tilde{u}'' \tilde{\omega}''} = -\frac{\mu_t}{Sc_{t,k}} \nabla \tilde{\omega}_k \quad k = 1, \dots, N_c \quad (2.10)$$

$$\nabla (\overline{\rho \tilde{u}'' H_s''}) = -\frac{\mu_t}{\sigma_t} \nabla \tilde{H}_s \quad (2.11)$$

dove μ_t è la viscosità dinamica turbolenta che sarà definita in seguito, $Sc_{t,k}$ è il numero di Schmidt turbolento per la specie k , normalmente pari a 0.7 [21] e σ_t è una costante assunta solitamente pari a 0.85 [21]. La validità di tali assunzioni non è assoluta, ma legata al fatto che non ci siano flussi turbolenti caratterizzati da trasporto contro gradiente.

2.5.3 Sforzi di Reynolds

Alla base della valutazione degli sforzi di Reynolds sta l'ipotesi di Bousinesq secondo la quale anche per i flussi turbolenti ci deve essere un legame con i gradienti che li provocano. Il flusso di quantità di moto turbolento, in analogia al trasporto molecolare della quantità di moto, si può esprimere in funzione del gradiente di velocità, mediante l'introduzione della viscosità turbolenta μ_t . Gli sforzi di Reynolds hanno la stessa forma del tensore degli sforzi (vedi paragrafo 1.3.1):

$$\overline{\rho \tilde{u}'' \tilde{u}''} = \overline{\rho \tilde{u}'' \tilde{u}''} = \mu_t \underline{\underline{\tilde{\tau}}} + \frac{2}{3} \overline{\rho} k \quad (2.12)$$

L'espressione dell'energia cinetica turbolenta k è data da:

$$k = \frac{1}{2} \widetilde{u_i'' u_i''} \quad (2.13)$$

Grazie all'ipotesi di Bousinesq, il problema della chiusura delle equazioni si traduce nella stima della viscosità turbolenta. Questa proprietà può essere valutata in diversi modi: utilizzando equazioni algebriche che non richiedono equazioni di bilancio aggiuntive, oppure con chiusura ad una equazione e con chiusura a due equazioni. L'approccio più semplice è quello sviluppato da Prandtl:

$$\mu_t = \bar{\rho} l_{mix}^2 \left| \underline{\underline{\tilde{S}}} \right| \quad (2.14)$$

dove l_{mix} è la lunghezza di miscelazione (una sorta di libero cammino medio) e $\underline{\underline{\tilde{S}}}$ è il tensore degli sforzi medio:

$$\underline{\underline{\tilde{S}}} = \frac{1}{2} \left(\frac{\partial \tilde{u}_i}{\partial x_j} - \frac{\partial \tilde{u}_j}{\partial x_i} \right) \quad (2.15)$$

Si usano poi formule empiriche per valutare l_{mix} che sono fortemente dipendenti dalla geometria del flusso e non possono perciò essere generalizzate.

Per quanto riguarda i modelli di chiusura a una equazione va citato quello di Prandtl-Kolmogorov [44] che lega la viscosità turbolenta all'energia cinetica turbolenta:

$$\mu_t = \bar{\rho} C_\mu l_{mix} \sqrt{k} \quad (2.16)$$

dove C_μ è la costante del modello pari a 0.09 e l_{mix} è una lunghezza caratteristica che dipende dalla geometria del flusso.

Gli approcci migliori rimangono comunque quelli a due equazioni, che permettono di stimare la viscosità in maniera migliore e tengono conto sia dell'energia cinetica turbolenta k che della sua velocità di dissipazione ε . Nel modello $k - \varepsilon$, dovuto a Jones e Launder [25], la viscosità turbolenta è ricavabile dalla seguente espressione:

$$\mu_t = \bar{\rho} C_\mu \frac{k^2}{\varepsilon} \quad (2.17)$$

Si usano poi due equazioni di trasporto delle quantità k ed ε . Il punto di partenza per ottenere un'equazione di trasporto dell'energia cinetica è l'equazione di Navier-Stokes, mediata secondo Favre e opportunamente rimaneggiata:

$$\frac{\partial}{\partial t} (\bar{\rho} k) + \nabla \cdot (\bar{\rho} \tilde{u} k) = \nabla \cdot \left[\left(\mu + \frac{\mu_t}{\sigma_k} \right) \nabla k \right] + P_k - \bar{\rho} \varepsilon \quad (2.18)$$

L'equazione per la velocità di dissipazione dell'energia cinetica turbolenta è derivata empiricamente:

$$\frac{\partial}{\partial t} (\bar{\rho}\varepsilon) + \nabla \cdot (\bar{\rho}\tilde{u}\varepsilon) = \nabla \cdot \left[\left(\mu + \frac{\mu_t}{\sigma_\varepsilon} \right) \nabla \varepsilon \right] + \frac{\varepsilon}{k} (C_{1\varepsilon}P_k - C_{2\varepsilon}\bar{\rho}\varepsilon) \quad (2.19)$$

in cui il termine P_k è il termine sorgente dato dalla seguente espressione

$$P_k = \bar{\rho}\widetilde{u''u''}\nabla\tilde{u} \quad (2.20)$$

A seconda dei diversi valori dati alle costanti si hanno diversi tipi di modello. Nel modello $k - \varepsilon$ *standard* proposto da Laufer e Spalding [27] le costanti assumono i seguenti valori: $C_\mu = 0.09$, $\sigma_k = 1.0$, $\sigma_\varepsilon = 1.3$, $C_{1\varepsilon} = 1.44$, $C_{2\varepsilon} = 1.92$. Questo modello è affidabile in situazioni semplici, ma fornisce previsioni poco soddisfacenti in condizioni più complesse, ad esempio la presenza di componente tangenziale nei getti o di effetti di galleggiamento. Spesso perciò i valori di queste costanti vengono modificati per avere una migliore aderenza alle situazioni reali. Ad esempio il valore di $C_{1\varepsilon}$ è portato spesso per fiamme a getto a 1.6 (Morse 1977). Dopo il modello $k - \varepsilon$ *standard*, molte modifiche sono state proposte per descrivere situazioni particolari di moto di fluido non descritte correttamente dal modello originario. Di seguito verranno esposti brevemente i più importanti.

Il modello $k - \varepsilon$ *standard RNG* (*ReNormalization Group*) deriva dall'utilizzo di una teoria statistica. Sono presenti rispetto al modello standard dei termini differenti nelle equazioni di trasporto di k ed ε e i valori delle costanti sono diversi. Il termine aggiuntivo nell'equazione della velocità di dissipazione permette di migliorare l'accuratezza per fluidi con deformazioni rapide. Si può descrivere anche l'effetto dello *swirl* (ovvero la componente rotazionale) sulla turbolenza e i numeri di Prandtl turbolenti vengono stimati in modo analitico. Si può inoltre applicare anche a flussi con bassi numeri di Reynolds.

Un secondo modello derivante da quello standard è il $k - \varepsilon$ *realizable* che utilizza una differente equazione per ε e una diversa formulazione per la viscosità turbolenta. Il termine *realizable* significa che il modello soddisfa pienamente le costanti matematiche sugli sforzi di Reynolds, cosa non possibile con i modelli visti in precedenza. Tale modello risulta adeguato quando i getti hanno ricircolazioni, rotazioni o anche separazioni.

Un altro modello è il $k - \omega$, sviluppato da Wilcox, che contiene modifiche per la descrizione di flussi con bassi numeri di Reynolds e fluidi comprimibili. Con la grandezza ω si intende la velocità di dissipazione dell'energia turbolenta specifica. Questo modello descrive in maniera agevole diversi tipi di getti, da piani a radiali. Esiste anche una variante di questo

modello chiamata $k - \omega$ SST (*Shear-Stress Transport*) che contiene delle modifiche per migliorare le predizioni vicino alle pareti e la definizione di viscosità turbolenta rispetto al gradiente di velocità.

Un modello che realizza una chiusura del secondo ordine senza però la necessità dell'approssimazione di Boussinesq è il modello *RSM* (*Reynolds Stress Model*). Se le forze di volume hanno molta importanza e se il campo di moto è complesso, l'ipotesi di Boussinesq non permette un'accurata valutazione degli sforzi di Reynolds, sebbene k venga stimata con la massima accuratezza possibile. Alla base di questo modello sta l'idea di risolvere delle equazioni di trasporto dedicate a ciascuna singola componente del tensore degli sforzi. L'equazione fondamentale è:

$$\frac{\partial}{\partial t} \left(\overline{\rho u_i'' u_j''} \right) + \frac{\partial}{\partial x_k} \left(\overline{\rho u_k u_i'' u_j''} \right) = P_{ij} + D_{ij} - E_{ij} + \Pi_{ij} + \Omega_{ij} \quad (2.21)$$

Da essa discendono sei equazioni alle derivate parziali, una per il trasporto di ogni componente indipendente del tensore (essendo il tensore simmetrico). Questo modello permette di descrivere due fenomeni fisici riguardanti l'energia cinetica turbolenta che non sono ancora stati trattati con i modelli fin qui descritti: la correlazione pressione/*strain* e la rotazione, rispettivamente contenute nei termini Π_{ij} e Ω_{ij} . I termini P_{ij} , produzione, D_{ij} , diffusione e E_{ij} , dissipazione, sono presenti nell'energia cinetica turbolenta. I termini Π_{ij} , P_{ij} e D_{ij} sono da modellare, mentre gli altri non lo richiedono. Questo modello permette di descrivere rapidi cambiamenti nello *strain rate*, *swirl* e rotazione e contiene una maggior rigorosità intrinseca. I risultati sono comunque influenzati dai modelli di chiusura per i termini che lo richiedono. Il carattere principale di questo modello è la sua validità generale senza necessità di aggiustamenti di costanti da farsi caso per caso.

2.5.4 Interazione cinetica-turbolenza

Il termine $\overline{\Omega_k}$ presente nell'equazione delle specie introduce un nuovo problema. Infatti il termine reattivo essendo legato all'equazione di Arrhenius per la stima della costante cinetica di velocità, possiede una intrinseca non linearità e la procedura di mediazione su questo termine non è semplice. Si ricorre quindi a modelli semplificati per descrivere correttamente l'interazione cinetica-turbolenza. Esistono diversi tipi di approccio e verranno dettagliatamente discussi nel terzo capitolo.

2.5.5 Modelli di irraggiamento

Il termine sorgente \overline{Q}_R contenuto nell'equazione 2.9 va modellato utilizzando uno dei diversi modelli di irraggiamento disponibili, tra cui si possono citare il modello *P-1*, il *Roseland* e il *DO (Discrete Ordinates)* [21]. In questo lavoro però l'unico modello su cui val la pena soffermarsi è un modello molto semplificato che può però essere utilizzato per fiamme non confinate come quelle studiate e che per questo motivo viene anche proposto sul sito nel quale sono disponibili i dati sperimentali delle fiamme [47]. Inoltre è stato molto semplice implementarlo nel programma utilizzato, che al momento non offre la possibilità di usare modelli di irraggiamento sofisticati. In questo modello, basato sul RADCAL [23], il termine sorgente viene calcolato con questa formula:

$$Q_R = 4\sigma_{SB} \sum_i^{N_c} (p_i \cdot a_{p,i}) \cdot (T^4 - T_b^4) \quad N_c = \text{numero specie} \quad (2.22)$$

in cui σ_{SB} è la costante di Stefan Boltzmann pari a $5.669e-08 \frac{W}{m^2K^4}$, p_i è la pressione parziale della specie considerata, $a_{p,i}$ è il coefficiente di assorbimento medio di Planck della stessa specie, T è la temperatura raggiunta e T_b è solitamente pari alla temperatura ambiente, o comunque alla temperatura delle pareti. La semplicità di questo modello risiede anche nel fatto che tra le specie responsabili dell'irraggiamento vengono considerate solo l'acqua, il monossido di carbonio, l'anidride carbonica e il metano. Perciò solo queste specie vengono utilizzate nella formula appena descritta e i coefficienti di assorbimento vengono calcolati tramite delle correlazioni che dipendono dalla temperatura e disponibili su [47].

2.6 Codici utilizzati

In questo lavoro di tesi sono stati utilizzati due diversi codici di calcolo CFD, *Ansys Fluent* e *OpenFOAM (Field Operation And Manipulation)*. Entrambi sono codici *general-purpose*, ovvero destinati alla risoluzione di una vasta casistica di problemi su geometrie generiche. Il primo è un codice commerciale e proprietario, ovvero l'utente può utilizzarlo per risolvere i problemi ma non è concesso l'accesso diretto al codice, pertanto è possibile apportare modifiche molto limitate. Va comunque detto che il codice fornito, nonostante il costo molto elevato, è molto completo e permette di utilizzare un'ampia gamma di modelli e di metodi risolutivi. Il secondo codice invece è un codice *open source*, perciò gratuito e liberamente modificabile. Ovviamente il codice è molto meno completo, perciò alcuni

modelli utilizzati in questo lavoro di tesi sono stati direttamente implementati. Entrambi i programmi utilizzano il metodo dei volumi finiti per la discretizzazione della *mesh* e l'approccio di risoluzione della turbolenza *RANS*. Inoltre in entrambi è possibile utilizzare il modello di chiusura $k - \varepsilon$ visto in precedenza. L'unica differenza sostanziale riscontrata tra i due codici sta negli schemi numerici utilizzati per arrivare alla soluzione. Rimandando ad altri testi e alla *user guide* [21] per una descrizione più dettagliata del codice *Fluent*, in questa tesi abbiamo dedicato il quarto capitolo a una panoramica su *OpenFOAM*, sul suo funzionamento e anche sul software *open source* in generale.

Capitolo 3

Interazione cinetica-turbolenza

3.1 Introduzione

Come già accennato nel paragrafo 2.5.4 una delle difficoltà maggiori quando viene utilizzato un approccio *RANS* per risolvere la turbolenza sta nella modellazione del termine reattivo. I fenomeni termofluidodinamici di interesse ingegneristico, come quelli che si riscontrano nei bruciatori, sono in generale caratterizzati da elevata turbolenza. E' indubbia l'importanza di conoscere a fondo l'interazione tra i fenomeni fluidodinamici e quelli chimici: le reazioni di combustione, a causa dell'elevato rilascio di calore, modificano in maniera accentuata la densità e la viscosità dei gas, provocando forti accelerazioni che influenzano il campo di moto dei fluidi. I problemi in cui sono coinvolti correnti turbolente o processi di combustione sono già tra i più complessi per quanto riguarda la loro modellazione numerica e pertanto l'accoppiamento di questi due fenomeni (ovvero la combustione turbolenta) crea un problema ancora più complesso. Le interazioni tra cinetica e turbolenza intervengono per scale molto vaste di condizioni del flusso. La diversa intensità dell'interazione causa fenomeni differenti: se l'interazione è debole l'aspetto fisico è semplicemente quello di rendere più rugoso il fronte di fiamma; se al contrario l'interazione è forte possono avvenire grosse modifiche sia nel campo di moto turbolento che nelle reazioni chimiche. E' noto che l'effetto di un campo di moto turbolento sia quello di portare i reagenti ad un più efficace mescolamento, migliorando la combustione; tuttavia, se troppo intensa, la turbolenza può portare all'estinzione della fiamma. La maggior parte dei combustori creano condizioni tali per cui il campo di moto dei fluidi (nelle fiamme non premiscelate) sia turbolento. Quasi tutti i generatori di potenza operano in questo modo,

dato che la turbolenza favorisce la velocità del consumo di massa dei reagenti, portando la a valori molto più elevati rispetto alle fiamme laminari e questo ovviamente aumenta notevolmente la potenza prodotta dal generatore. Dal punto di vista matematico, riprendendo l'equazione di trasporto per la singola specie mediata secondo Reynolds, vediamo che contiene un termine sorgente che può essere scritto in questo modo:

$$\overline{\dot{\Omega}_k} = M_{w,k} \sum_{j=1}^{N_R} \left(\nu_{k,j}^f - \nu_{k,j}^b \right) \overline{r_j} = M_{w,k} \sum_{j=1}^{N_R} \left(\nu_{k,j}^f - \nu_{k,j}^b \right) \left[K_j^f \prod_{k=1}^{N_C} (C_k)^{\nu_{k,j}^f} - K_j^b \prod_{k=1}^{N_C} (C_k)^{\nu_{k,j}^b} \right] \quad (3.1)$$

in cui $M_{w,k}$ è il peso molecolare della specie k -esima, C_k è la concentrazione molare della specie k , $\overline{r_j}$ è la velocità di reazione netta valutata in condizioni medie (velocità di reazione diretta meno velocità di reazione inversa per una generica reazione reversibile) e $\nu_{k,j}$ è il coefficiente stechiometrico e, considerando una cinetica semplice, anche ordine di reazione per la specie k nella reazione j -esima. Gli apici f e b indicano la reazione diretta e quella inversa. La costante K presente nell'ultima formulazione può essere espressa attraverso l'equazione di Arrhenius:

$$K = A \cdot e^{\left(-\frac{E_a}{RT}\right)} \quad (3.2)$$

dove A è il fattore preesponenziale, E_a l'energia di attivazione, R la costante dei gas e T la temperatura. Come già accennato, il problema sta nella non linearità di questo termine esponenziale, che causa grossi problemi nella procedura di mediazione. Infatti valutare il termine sorgente chimico usando i valori di concentrazione e temperatura medi significa trascurare completamente gli effetti della turbolenza e commettere un grave errore. Una reazione chimica è un processo attraverso il quale avviene la trasformazione di specie chimiche e possiede un tempo caratteristico [13]. Quest'ultimo è indispensabile per poter valutare l'effetto della turbolenza sulle velocità di formazione di ogni specie chimica. I tempi della chimica possono essere definiti in modo rigoroso tramite gli autovalori della matrice Jacobiana delle velocità di formazione. In condizioni isoterme il generico elemento di tale matrice è:

$$J_{ij} = \frac{\partial \dot{\Omega}_k}{\partial C_j} \quad \text{con } i, j = 1, \dots, N_c \quad (3.3)$$

dove C_j è la concentrazione della specie j -esima e N_C il numero delle specie. Ogni elemento di questa matrice e quindi anche gli autovalori sono perciò dipendenti dalla

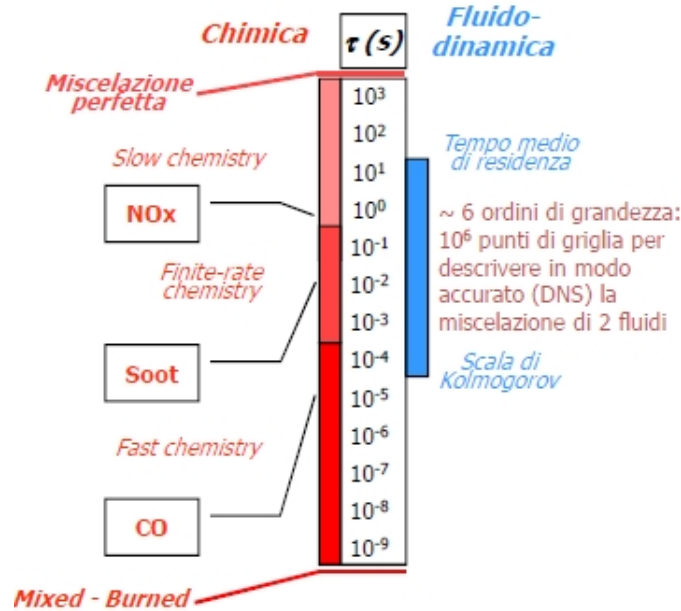


Figura 3.1: Descrizione dei tempi di residenza

concentrazione e dai parametri cinetici. Per una matrice $N_C \times N_C$, detti μ_i gli autovalori, si possono definire i tempi caratteristici della chimica come:

$$\tau_{c,j} = \frac{1}{|Re(\mu_j)|} \quad (3.4)$$

con j indice riferito alla j -esima specie chimica del sistema. Dato che l'autovalore dipende dalla composizione locale in ogni istante, anche il tempo caratteristico avrà la stessa dipendenza. Il tempo chimico può assumere valori elevati se l'autovalore è piccolo, ovvero se la chimica è lenta (*slow chemistry*), o al contrario assumere valori molto bassi se la chimica è veloce e l'autovalore di conseguenza grande (*fast chemistry*). Nei processi di combustione, i tempi caratteristici della chimica possono variare da valori molto bassi (10^{-9} secondi) a valori molto più elevati, dell'ordine di 10^2 secondi, coprendo quindi più di 10 ordini di grandezza. Tale scala è da confrontarsi con quella dei tempi caratteristici della turbolenza, i quali si possono pensare come degli indicatori della velocità di miscelazione. Questa scala copre un intervallo che va dal valore del tempo caratteristico associato ai vortici di Kolmogorov fino al tempo di residenza associato alla macroscale, passando attraverso diverse condizioni di micro-miscelazione turbolenta differenti. In figura 3.1 sono indicati i tempi caratteristici sia della chimica che della turbolenza. Dati questi intervalli si possono

individuare tre tipi di regimi differenti:

- Chimica veloce (*fast chemistry*) quando il tempo chimico è minore del tempo caratteristico di Kolmogorov
- Chimica lenta (*slow chemistry*) quando viceversa il tempo chimico è maggiore del tempo di miscelazione τ_{mix}
- *Finite-rate chemistry* quando il comportamento è intermedio ai due precedenti e i tempi caratteristici sono simili

Solitamente si discrimina tra le diverse situazioni utilizzando il rapporto tra il tempo di micro-miscelazione e il tempo chimico della specie *j-esima*. Tale rapporto è definito come il numero adimensionale di Damköhler della specie *j-esima*:

$$Da_j = \frac{\tau_{mix}}{\tau_{c,j}} \quad (3.5)$$

Quando il numero di Damköhler è molto maggiore di uno siamo perciò in una zona in cui le reazioni chimiche sono governate dal mixing turbolento. Viceversa quando il numero è molto minore di uno è la chimica a prevalere sulla fluidodinamica. Le grosse difficoltà numeriche sorgono per il fatto che, considerando un problema chimico complesso, esistono situazioni in cui si verificano entrambe le condizioni, o comunque il numero di Damköhler si trova in una posizione intermedia tra le due [15]. Un modello di interazione cinetica-turbolenza, nella valutazione del termine sorgente chimico deve necessariamente tenere in considerazione il rapporto tra i tempi della turbolenza e della chimica.

Per risolvere il problema si potrebbe per esempio pensare a uno sviluppo in serie di Taylor della velocità di reazione media. Questo comporterebbe alcune difficoltà: in primo luogo verrebbero introdotte nuove quantità incognite che necessiterebbero di ulteriori equazioni; inoltre troncando l'espansione si avrebbe un errore non trascurabile. Pertanto quando si utilizza un approccio *RANS* l'accoppiamento tra cinetica e turbolenza può essere descritto solo attraverso modelli appropriati. Come si vede nella figura 3.2 essi sono raggruppati in due grandi categorie, a seconda dell'approccio utilizzato. Nella prima (*Reaction Rate Approach, RRA*) viene modellato solo il termine sorgente delle equazioni di trasporto e si risolve un'equazione di trasporto per ogni singola specie. Nella seconda (*Primitive Variables Approach, PVA*) si risolvono le equazioni di trasporto solo per un numero ristretto di variabili fondamentali che permettono di descrivere in maniera univoca i campi di concentrazione e temperatura. Inoltre dalla figura si nota anche come questi modelli siano legati

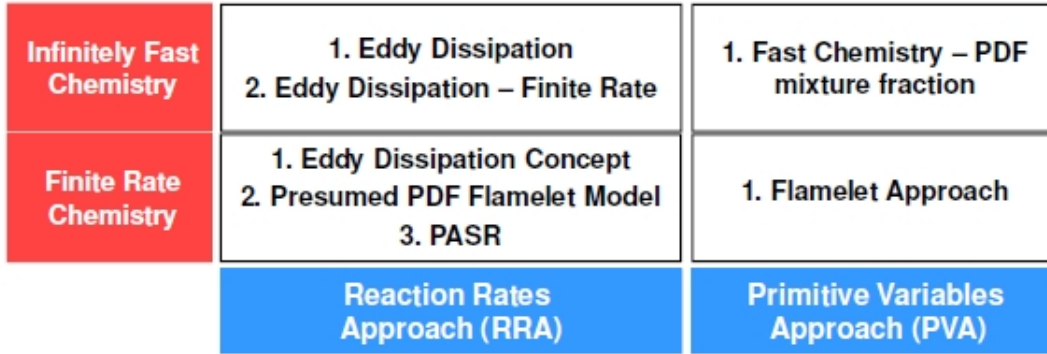


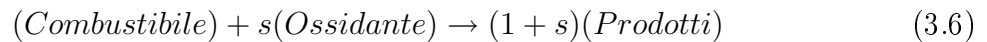
Figura 3.2: Schema dei modelli di combustione

alla velocità delle reazioni chimiche rispetto alla miscelazione turbolenta, di cui è stato appena parlato. I diversi modelli presenti nello schema verranno descritti nel dettaglio nel corso di questo capitolo.

3.2 Modelli *RRA*

3.2.1 Modello *Eddy Break Up (EBU)*

Il modello *Eddy Break Up* è stato sviluppato da Spalding nel 1971 [45] e si applica a fiamme turbolente premiscelate supponendo i numeri di Reynolds e di Damköhler elevati. L'ipotesi che sta alla base del modello è che la velocità del processo di combustione sia controllata dalla miscelazione turbolenta e non dalla chimica delle reazioni di ossidazione; la chimica è considerata perciò infinitamente veloce, secondo l'approccio noto come *mixed is burnt*. Quindi la velocità di reazione media risulta proporzionale al tempo di miscelazione $\frac{\varepsilon}{k}$. Se si considera la generica reazione:



si ha che la velocità di reazione del combustibile (indicato con pedice F) è data da:

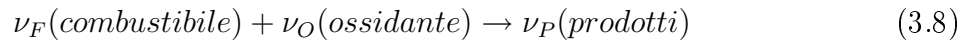
$$\overline{\dot{\Omega}}_F = \bar{\rho} C_{EBU} \frac{\varepsilon}{k} \left(\widetilde{\omega_F''^2} \right)^{0.5} \quad (3.7)$$

in cui C_{EBU} è una costante caratteristica del modello pari circa a uno, mentre $\widetilde{\omega_F''^2}$ indica la varianza della frazione massiva del combustibile. Questo modello viene molto utilizzato

nei codici commerciali per i suoi costi computazionali alquanto bassi in quanto la velocità mediata viene scritta come funzione di quantità medie già note senza dover scrivere equazioni di trasporto aggiuntive. Questo però comporta alcuni errori: il modello infatti, oltre a non considerare gli effetti della cinetica chimica, tende a sovrastimare la velocità di reazione soprattutto nelle regioni del fluido molto stressate in cui il rapporto $\frac{\varepsilon}{k}$ è elevato.

3.2.2 Modello *Eddy Dissipation (ED)*

Il modello *Eddy Dissipation* è l'estensione del modello *Eddy Break Up* al caso di fiamme non premiscelate [31]; in questi tipi di fiamme avviene, a causa della turbolenza, un lento mescolamento tra il combustibile e il comburente all'interno della zona di reazione in cui la combustione è molto veloce; si considera la velocità di reazione come controllata dal mixing turbolento e si dice che la combustione è *mixing limited* dato che il grado di mescolamento risulta limitante rispetto alla velocità di reazione. Pertanto le velocità cinetiche di reazione possono essere trascurate e il termine sorgente viene legato come nel modello precedente al tempo necessario per il mescolamento dei reagenti a livello molecolare. Considerando un'unica reazione:



La velocità di reazione media è presa come il valore minimo tra le due seguenti espressioni di velocità

$$\overline{\dot{\Omega}_{i,r}} = \nu_{i,r} M_{w,i} A \bar{\rho} \frac{\varepsilon}{k} \min \left(\frac{\tilde{\omega}_F}{\nu_{F,r} M_{w,F}}, \frac{\tilde{\omega}_O}{\nu_{O,r} M_{w,O}} \right) \quad (3.9)$$

$$\overline{\dot{\Omega}_{i,r}} = \nu_{i,r} M_{w,i} A B \bar{\rho} \frac{\varepsilon}{k} \frac{\sum_P \tilde{\omega}_P}{\sum_j \nu_{j,r} M_{w,j}} \quad (3.10)$$

dove $\tilde{\omega}$ è la frazione massiva mediata, rispettivamente del combustibile (F), dell'ossidante (O) e dei prodotti (P). ν e M_w rappresentano i coefficienti stechiometrici e il peso molecolare, in particolare il pedice i si riferisce a tutte le specie mentre il pedice j solo ai prodotti. A e B sono due costanti del modello stimate in maniera empirica a 4 e 0.5. Il pedice r indica la reazione che si sta considerando, in quanto il modello potrebbe essere utilizzato anche con più di una reazione. La prima espressione tiene conto di quale sia il fattore limitante tra il combustibile e l'ossidante rispetto al rapporto stechiometrico tra essi; la seconda considera come limitante la presenza o meno di prodotti caldi che inneschino il

processo di combustione.

Il mdello *ED*, per come è costruito, si applica con successo a flussi turbolenti nei quali la velocità di reazione chimica è alta rispetto ai processi di mixing turbolento; al contrario i processi in cui la cinetica chimica può limitare la velocità di reazione sono simulati in modo poco accurato. Inoltre il modello presuppone di simulare una combustione stabile, infatti non vengono considerati nè fenomeni di ignizione nè di estinzione della fiamma. Questo modello ha inoltre il problema di sovrastimare le temperature: non si tengono in considerazione infatti le reazioni di dissociazione e quindi le condizioni di equilibrio tra le specie. Se per esempio si considera la reazione



non si tiene conto della dissociazione dell'anidride carbonica che avviene alle alte temperature:



Questa reazione causa un abbassamento della temperatura e quindi quella ottenuta attraverso il modello -risulta sensibilmente maggiore di quella sperimentale. Inoltre il modello ottiene risultati accettabili solo se applicato ad un'unica reazione. Nel caso di una miscela di combustibili è necessario creare pseudospecie perdendo ulteriormente accuratezza nella soluzione.

3.2.3 Modello *Finite Rate Chemistry (FR)*

Il modello *Finite Rate Chemistry* si applica in casi in cui i tempi caratteristici della chimica sono maggiori di quelli della turbolenza (cioè quando $Da \ll 1$). Il termine reattivo mediato viene definito come:

$$\overline{\dot{\Omega}_r} = \tilde{K}_r \prod_{j=1}^{NC} [\widetilde{C}_{j,r}]^{\nu_{j,r}} = A\tilde{T}^n \cdot e^{\left(-\frac{E_a}{RT}\right)} \cdot \prod_{j=1}^{NC} [\widetilde{C}_{j,r}]^{\nu_{j,r}} \quad (3.13)$$

in cui i parametri sono gli stessi presentati nelle equazioni 3.1 e 3.2. Il modello *Finite Rate* da buoni risultati se applicato in situazioni dove il parametro limitante è la chimica e si può per questo motivo utilizzare sia in flussi laminari che turbolenti

3.2.4 Modello *EDFR*

L'*EDFR* non è altro che l'unione dei modelli *Eddy Dissipation* e *Finite Rate*. Sostanzialmente si considera come velocità di reazione mediata la minima tra quelle calcolate con il primo modello e con il secondo:

$$\overline{\dot{\Omega}}_k = \min \left(\overline{\dot{\Omega}}_k^{FR}, \overline{\dot{\Omega}}_k^{ED} \right) \quad (3.14)$$

In pratica le due velocità vengono calcolate separatamente. Infatti in alcuni step la velocità può essere controllata dalla cinetica chimica mentre in altri dalla miscelazione turbolenta. Questo approccio permette di tenere in conto varie tipologie di fenomeni, sia con bassi che con alti numeri di Damköhler. In particolare è adatto quando ci si trova nei due casi limite, in cui uno dei due tempi caratteristici prevale sull'altro.

L'utilizzo di questo modello risulta però problematico nel caso in cui vi siano meccanismi cinetici con tanti step di reazione; infatti il modello ripete il confronto tra *ED* e *FR* per ogni step e per ogni cella della griglia di calcolo. Alcune reazioni potrebbero subire la limitazione dovuta al *mixing* che le porta ad essere poste alla stessa velocità anche se alcune sono cineticamente più veloci di altre.

3.2.5 Modello *Eddy Dissipation Concept (EDC)*

Il modello *Eddy Dissipation Concept* è una derivazione dell'*Eddy Dissipation* ed è stato sviluppato da Magnussen nel 1976 [30, 22]. Il tentativo di questo modello è quello di incorporare gli effetti della cinetica chimica nelle fiamme turbolente. Esso si basa su una descrizione molto accurata dei vortici turbolenti e della loro dissipazione e sull'assunzione che le reazioni chimiche possano aver luogo solo quando la miscelazione dei reagenti avviene al livello della scala molecolare e a una temperatura sufficientemente alta. Come già visto un flusso turbolento consiste nell'insieme di vortici di differenti dimensioni. Secondo la teoria di Kolmogorov l'energia cinetica viene trasportata, con una dissipazione trascurabile, dai vortici più grandi a quelli più piccoli fino ad arrivare alla scala di Kolmogorov, dove la dissipazione ad opera delle forze viscosse trasforma tutta l'energia cinetica in calore. Questa teoria è valida solo se ci si trova in condizioni di omogeneità tridimensionale e isotropia locale. Si può però affermare che per numeri di Reynolds sufficientemente alti il campo di moto turbolento presenti ampie regioni in cui la dissipazione è trascurabile e dove l'energia turbolenta viene perciò in gran parte trasferita e altre regioni dove si ha miscelazione su

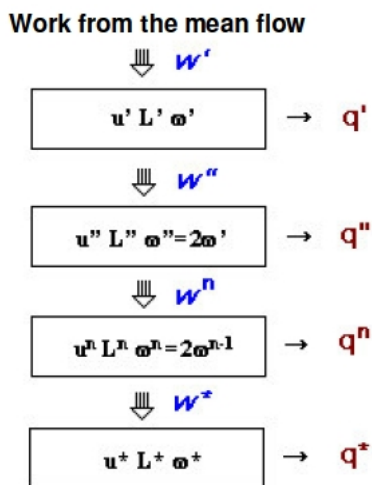


Figura 3.3: Modello per il trasferimento di energia turbolenta dalle strutture più grandi alle più piccole

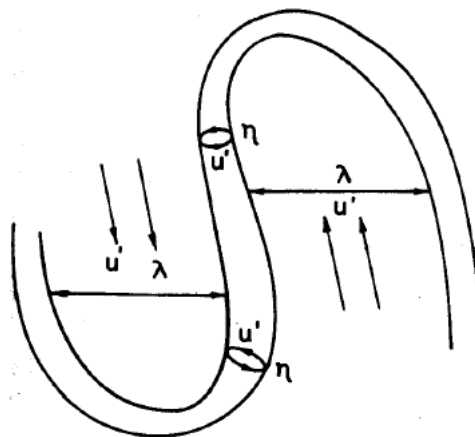


Figura 3.4: Modello di Tennekes per le regioni *fine structures*

scala molecolare, chiamate *fine structures*, all'interno delle quali avviene la dissipazione. Il processo a cascata attraverso il quale l'energia viene trasferita tra i vortici è visibile in figura 3.3, nella quale si nota come ogni livello della cascata sia caratterizzato da una velocità turbolenta u , una scala L e uno *strain rate* definito come $\omega = \frac{u}{L}$. Il trasferimento di energia dovuto agli sforzi è rappresentato dal lavoro W e q è il calore dissipato. Ovviamente q^* che rappresenta il calore dissipato alla microscala rappresenta la quasi totalità dell'energia inizialmente presente. Le *fine structures* sono contenute in regioni ben definite del fluido, visibili in figura 3.4 secondo il modello sviluppato da Tennekes [22]. Tali regioni possono essere viste come degli insiemi grandi e sottili di vortici ripiegati e aggrovigliati su se stessi e le cui dimensioni sono molto maggiori di quelle delle *fine structures*, ossia dei microvortici contenuti all'interno. Queste zone si vengono a formare negli spazi tra i vortici più grandi maggiormente sottoposti agli sforzi. Le *fine structures* sono fondamentali per lo sviluppo del modello *EDC*. Il punto di partenza infatti sta nell'assumere che tutte le reazioni chimiche che coinvolgono specie in fase gas avvengano all'interno di queste zone, dove i reagenti possono miscelarsi. La conseguenza è che nel fluido circostante è come se le reazioni fossero inibite, venendo a mancare in pratica il contatto a livello molecolare tra i reagenti. Questa è una grande semplificazione, ma la validità dell'assunzione è stata provata in molte applicazioni pratiche. L'approccio utilizzato in questo modello sta nel trattare le *fine structures* come se fossero dei reattori perfettamente miscelati con un

tempo di residenza che chiameremo τ^* e dove la frazione massiva della specie k -esima è ω_i^* . Definiamo con γ_λ la frazione volumetrica o massiva della *fine structure*, funzione delle proprietà turbolente. All'interno di queste microstrutture le reazioni avvengono per un tempo τ^* e su un volume effettivo pari a V^* secondo meccanismi anche complessi, pertanto è possibile utilizzare schemi cinetici anche molto dettagliati. Le equazioni per il calcolo di queste grandezze sono ricavabili attraverso un procedimento matematico che verrà tralasciato e per cui si rimanda a [30] e si traducono nelle seguenti:

$$\tau^* = 0.41 \left(\frac{\nu}{\varepsilon} \right)^{\frac{1}{2}} \quad (3.15)$$

$$\gamma_\lambda = 2.13 \left(\frac{\nu \varepsilon}{k^2} \right)^{\frac{1}{4}} \quad (3.16)$$

$$V^* = \gamma_\lambda V \quad (3.17)$$

dove ν è la viscosità laminare cinematica. Considerando lo scambio massivo esistente tra le *fine structures* e le zone circostanti, la velocità di trasferimento netta massiva media della specie k -esima tra le *fine structures* e il fluido circostante si può esprimere come

$$\overline{\dot{\Omega}}_k = \frac{\rho \gamma_\lambda^2}{\tau^* (1 - \gamma_\lambda^3)} (\omega_i^* - \omega_i^0) \quad (3.18)$$

in cui ρ è la densità e ω_i^* e ω_i^0 sono le frazioni massive della specie i nelle *fine structures* e nel fluido circostante. Tra gli aspetti principali vi è quello secondo il quale tutte le reazioni chimiche il cui tempo caratteristico per la specie limitante è maggiore del tempo di residenza τ^* vengono considerate come estinte. Solo una parte delle *fine structures* può reagire ed è quella che è riscaldata sufficientemente. Solitamente si usa un fattore correttivo χ per indicare tale frazione e quindi per tenere conto degli effetti della chimica *finite rate* quando l'assunzione della chimica veloce è utilizzata. L'effetto della cinetica viene considerato trattando le *fine structures* reattive come dei reattori miscelati messi in condizione di scambiare massa ed energia con il fluido circostante. Nel caso in cui il trasferimento di calore per effetto dell'irraggiamento sia trascurabile i reattori sono adiabatici e partendo dalle equazioni caratteristiche di questo tipo di reattori [22, 16], otteniamo dei reattori stazionari perfettamente miscelati (rappresentati in figura 3.5) con tempo di residenza τ^* e descritti dalla seguente equazione:

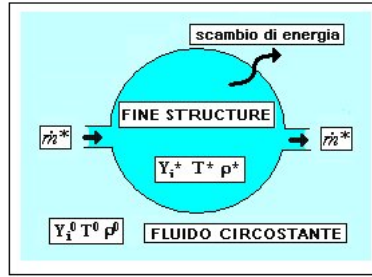


Figura 3.5: Schema di una *fine structure* approssimata a un reattore perfettamente miscelato

$$\frac{\omega_i^* - \omega_i^0}{\tau^*} = r_i \quad (3.19)$$

in cui r_i è il termine netto di produzione chimica calcolato attraverso le espressioni di Arrhenius, che rende l'equazione altamente non lineare e perciò richiede per la soluzione un algoritmo particolarmente robusto. La soluzione di questa equazione corrisponde alla ω_i^* da inserire nell'equazione 3.18, che è il termine sorgente da inserire nell'equazione di bilancio delle specie (2.8).

Nel caso in cui non si possa considerare il reattore come adiabatico, è necessario determinare la potenza netta per unità di volume \dot{Q}^* trasferita tra le *fine structures* e il fluido circostante:

$$\dot{Q}^* = \frac{\rho^*}{\tau^* (1 - \gamma_k^3)} \sum_{k=1}^{N_C} \left(\omega_i^* \hat{H}_k^* - \omega_i^0 \hat{H}_k^0 \right) \quad (3.20)$$

Il modello *EDC* può pertanto, approssimando le *fine structures* a dei reattori ideali, permettere l'implementazione di meccanismi cinetici dettagliati all'interno di flussi reattivi in condizioni turbolente e può essere utilizzato quando l'assunzione di chimica veloce non può essere fatta in alcun modo. Tuttavia sorge il problema che i meccanismi cinetici dettagliati hanno delle caratteristiche che rendono la loro integrazione numerica un problema *stiff* e quindi molto impegnativo sul piano delle risorse computazionali rispetto ad altri modelli di combustione.

3.2.6 Modello *Transported PDF*

Pur non entrando nel dettaglio, merita di essere menzionato un ultimo modello della categoria *RRA*. I metodi fin qui visti cercano di realizzare una chiusura riguardante il

termine di velocità di reazione media introdotto nelle equazioni di trasporto delle specie dopo aver mediato secondo Favre. Questi modelli possono talvolta fallire nella predizione del comportamento di alcune fiamme turbolente. L'alternativa è sostituire la mediazione secondo Favre con una equazione di trasporto della funzione densità di probabilità (*PDF*). Il tentativo è quello di calcolare, mediante un'equazione di trasporto, la forma della *PDF*. Il principale vantaggio è legato al fatto che tale modello è una sorta di descrizione statistica del sistema che permette perciò applicazioni sia ai casi premiscelati che no. Inoltre con questo metodo non è necessaria la chiusura sul termine sorgente che è come visto altamente non lineare. Gli svantaggi sono che modellare con questo metodo la miscelazione si rivela difficile e il costo computazionale è molto elevato.

3.3 Modelli *PVA*

Come già detto, in questo tipo di approccio, vengono risolte solo un numero ristretto di variabili fondamentali, definite primitive, dalle quali è possibile risalire a tutte le altre, in particolare temperatura e composizione. Per rendere più chiara la questione è necessario introdurre il concetto di *mixture fraction* [22, 13].

3.3.1 *Mixture fraction*

Nella combustione diffusiva il combustibile e l'ossidante entrano nella zona di reazione in due correnti distinte e chiameremo le portate massive m_1 e m_2 rispettivamente. Sommando tutte le reazioni è possibile scrivere un'unica reazione globale del tipo:



dove ovviamente F indica il combustibile, O l'ossidante e P i prodotti. r rappresenta il rapporto massivo $\frac{\text{aria}}{\text{combustibile}}$. Il rapporto di equivalenza quindi può essere visto come

$$\varphi = \frac{r_{st}}{r} \quad (3.22)$$

in cui r_{st} si riferisce alle condizioni stechiometriche. Definiamo la *mixture fraction* come la frazione massiva di combustibile nella miscela prima che avvenga la combustione:

$$f = \frac{m_1}{m_1 + m_2} = \frac{1}{1 + r} = \frac{\varphi}{\varphi + r_{st}} \quad (3.23)$$

La frazione massiva di *fuel* nella miscela incombusta, $\omega_{F,u}$ è proporzionale alla frazione massiva del *fuel* stesso nella corrente di combustibile $\omega_{F,1}$ (infatti entrambe le correnti possono contenere degli inerti):

$$\omega_{F,u} = \omega_{F,1} f \quad (3.24)$$

Allo stesso modo nel caso dell'ossidante:

$$\omega_{O_2,u} = \omega_{O_2,2}(1 - f) \quad (3.25)$$

in cui $\omega_{O_2,2}$ è la frazione massiva di ossidante all'interno della corrente di aria. Quando φ assume valore pari a 1 ci si trova in condizioni stechiometriche e la *mixture fraction* relativa è ovviamente:

$$f_{st} = \frac{1}{1 + r_{st}} \quad (3.26)$$

Nel caso in cui la *mixture fraction* assuma valori inferiori a quello stechiometrico la miscela viene definita magra, cioè ci si trova in carenza di combustibile. Se al contrario il valore è superiore a quello stechiometrico la miscela è ricca di combustibile. E' importante a questo punto introdurre la frazione massiva degli elementi, in quanto mentre quella delle specie non viene conservata durante la combustione, quella degli elementi si conserva. La frazione massiva del generico elemento j viene definita come:

$$Z_j = \sum_{i=1}^{N_C} \frac{a_{ij} W_j}{W_i} \cdot \omega_i \quad j = 1, 2, \dots, n_e \quad (3.27)$$

in cui N_c è il numero di specie chimiche ed n_e il numero di elementi contenuti all'interno del sistema, a_{ij} il numero di atomi dell'elemento j in una molecola di specie i , W_j il peso atomico dell'elemento e ω_i la frazione massiva della specie i . La *mixture fraction*, allora, partendo dalla 3.24, può essere vista come:

$$f = \frac{\omega_{F,u}}{\omega_{F,1}} = \frac{\sum_{j=1}^{n_e} Z_{j,F}}{\omega_{F,1}} \quad (3.28)$$

Dalla 3.24 e 3.25, inoltre:

$$\omega_{F,u} + \omega_{O_2,u} = \omega f + \omega(1 - f) \quad (3.29)$$

cioè:

$$\sum_{j=1}^{ne} Z_{j,F} + \sum_{j=1}^{ne} Z_{j,O_2} = \sum_{j=1}^{ne} Z_{j,1} f + \sum_{j=1}^{ne} Z_{j,2} (1 - f) \quad (3.30)$$

che accorpando i due termini a sinistra diventa:

$$\sum_{j=1}^{ne} Z_j = \sum_{j=1}^{ne} (Z_{j,1} f + Z_{j,2} (1 - f)) \quad (3.31)$$

Da quest'ultima espressione si ricava una diversa definizione di *mixture fraction*:

$$f = \frac{Z_j - Z_{j,2}}{Z_{j,1} - Z_{j,2}} \quad (3.32)$$

in cui Z_j è la frazione massiva dell'elemento j nella miscela, mentre $Z_{j,1}$ e $Z_{j,2}$ si riferiscono rispettivamente alla corrente in ingresso di *fuel* e ossidante. Se il coefficiente di diffusività è lo stesso per tutte le specie, allora l'equazione 3.32 è la stessa per tutti gli elementi [7].

3.3.2 Equazioni di trasporto

Si dimostra [22] che l'equazione di conservazione della *mixture fraction* ha la stessa formula di quella scritta per la frazione massiva elementare, è controllata dai termini di convezione e diffusione ma ha la particolarità di non contenere al suo interno il termine sorgente:

$$\frac{\partial}{\partial t}(\rho f) + \frac{\partial}{\partial x_i}(\rho u_i f) = \frac{\partial}{\partial x_i} \left(\rho \Gamma \frac{\partial f}{\partial x_i} \right) \quad (3.33)$$

Mediando questa equazione di trasporto secondo Favre si ottiene:

$$\bar{\rho} \frac{\partial \tilde{f}}{\partial t} + \bar{\rho} \tilde{u}_i \frac{\partial \tilde{f}}{\partial x_i} = \frac{\partial}{\partial x_i} \left(\frac{\mu_t}{Sc_t} \frac{\partial \tilde{f}}{\partial x_i} \right) \quad (3.34)$$

In un ambiente turbolento la *mixture fraction* fluttua caoticamente intorno al suo valore medio in ogni punto della griglia. Per caratterizzare il processo di miscelazione e le reazioni chimiche, servono informazioni aggiuntive riguardanti le statistiche relative alle fluttua-

zioni della *mixture fraction*. Per modellare tali fluttuazioni si definisce la sua varianza con la relativa equazione di trasporto mediata secondo Favre:

$$\bar{\rho} \frac{\partial \tilde{f}''^2}{\partial t} + \bar{\rho} \tilde{u}_i \frac{\partial \tilde{f}''^2}{\partial x_i} = \frac{\partial}{\partial x_i} \left(\frac{\mu_t}{Sc_t} \frac{\partial \tilde{f}''^2}{\partial x_i} \right) + 2 \frac{\mu_t}{Sc_t} \left(\frac{\partial \tilde{f}}{\partial x_i} \right)^2 - \bar{\rho} \tilde{\chi} \quad (3.35)$$

Tra i termini a destra dell'uguale il primo rappresenta il trasporto turbolento, il secondo è la produzione delle fluttuazioni dovute ai gradienti medi e l'ultimo il termine di dissipazione, definito come:

$$\bar{\rho} \tilde{\chi} = \overline{\rho D |\nabla f|^2} \quad (3.36)$$

dove $\tilde{\chi}$ è lo *scalar dissipation rate*, modellato come vedremo in seguito. La varianza \tilde{f}''^2 è un parametro fondamentale per modellare la chiusura dell'interazione cinetica-turbolenza. La *mixture fraction* determina la posizione della fiamma, mentre le fluttuazioni (e quindi la varianza) sono responsabili delle deformazioni del fronte di fiamma. La varianza è indice di disomogeneità nella miscelazione dei reagenti e perciò nelle fiamme non premiscelate è uno dei parametri di controllo fondamentali. Quando la cinetica è veloce rispetto alla velocità di miscelazione, il sistema reagisce in maniera rapida ai cambiamenti delle proprietà locali causate dalla miscelazione [50]. Le velocità di reazione più elevate si hanno nei punti in cui la *mixture fraction* assume valori vicini a quello stechiometrico.

3.3.3 Modelli *presumed PDF*

Quando la varianza indica piccole fluttuazioni di f intorno al valore stechiometrico siamo in condizioni di miscelazione veloce e pertanto intensa reattività e il tempo caratteristico delle reazioni è quello della scala di Kolmogorov. Quando al contrario le fluttuazioni sono significative il processo di miscelazione risulta essere più lento del tempo delle reazioni chimiche e pertanto si può applicare un approccio *fast chemistry*. In questo caso è utilizzabile l'approccio *presumed PDF*, attraverso il quale ad ogni iterazione i campi di temperatura e composizione della fiamma vengono valutati tramite la conoscenza dei valori di *mixture fraction*, varianza, *scalar dissipation rate* e, nel caso di fiamma non adiabatica, entalpia. Infatti questo modello si basa sull'ipotesi che l'interazione tra chimica e turbolenza sia tenuta in conto andando a valutare i campi di variazione di queste proprietà nella fiamma reale, attraverso una funzione densità di probabilità (*PDF*) $\tilde{P}(f; \underline{x}, t)$, che quantifica la

proprietà che la variabile f assuma un valore appartenente a un certo intervallo per valori assegnati spaziali (\underline{x}) e temporali (t). Per un generico scalare si può stimare:

$$\tilde{\phi} = \int_0^1 P(f) \cdot \phi(f) \cdot df \quad (3.37)$$

in cui $P(f)$ è la *PDF* della *mixture fraction* e $\phi(f)$ è la relazione che lega lo scalare alla *mixture fraction*. Se il sistema non è adiabatico, allora si utilizza una *PDF* che dipende sia dalla composizione che dall'entalpia, ma si semplifica assumendo che le fluttuazioni di entalpia non dipendano dal valore dell'entalpia stessa, ovvero che le perdite di entalpia non influenzino le fluttuazioni della stessa. Questo si traduce in:

$$\tilde{\phi} = \int_0^1 P(f) \cdot \phi(f, \tilde{H}) \cdot df \quad (3.38)$$

Le forme di *PDF* più utilizzate sono *double-delta PDF*, *clipped Gaussian PDF* e β -*PDF*. In questo modo una volta che le equazioni di trasporto della varianza e della *mixture fraction* sono risolte, la *PDF* è nota in ogni punto del dominio di calcolo. Il vantaggio di tale metodo deriva dall'assunzione, come già accennato, che lo stato termochimico del fluido è determinato univocamente trasportando questi due scalari, ovvero bastano due sole equazioni e una aggiuntiva dell'entalpia se il sistema non è adiabatico. Lo *scalar dissipation rate* infatti viene calcolato attraverso un'equazione algebrica. Le variabili sono recuperabili da una libreria di *flamelets (look-up table)*, dove sono disponibili la temperatura e la composizione corrispondente ad ogni valore di *mixture fraction* media, della sua varianza, dello *scalar dissipation rate* e dell'entalpia media. I valori medi degli scalari in una cella sono interpolati da quelli circostanti. Le interazioni cinetica-turbolenza sono perciò in questo caso pre-processate e i calcoli non sono da effettuarsi ad ogni iterazione. Si possono perciò facilmente anche trattare degli schemi cinetici dettagliati. Si possono utilizzare tre diversi tipi di approccio, citati in ordine di complessità crescente: l'approccio *mixed is burnt*, l'approccio d'equilibrio e l'approccio *Steady Laminar Flamelets*, in cui la fiamma viene vista come un insieme di fiamme laminari stazionarie.

3.3.4 Modello miscelato-bruciato (*mixed is burnt*)

E' l'approssimazione più semplice, in quanto le reazioni sono considerate irreversibili e infinitamente veloci. Non vi è perciò alcun bisogno di informazioni circa la cinetica delle

reazioni chimiche: la composizione si ricava dalla stechiometria della reazione di combustione in base al fatto che reagenti e prodotti non possono coesistere. Lo svantaggio ovviamente è l'accuratezza decisamente bassa conseguente al fatto di non poter utilizzare schemi cinetici dettagliati.

3.3.5 Modello di equilibrio

Questo tipo di approssimazione è più accurata della precedente: per ogni valore della *mixture fraction* media, della sua varianza e dell'entalpia, si minimizza l'energia libera di Gibbs della miscela di gas, in modo da valutare la composizione e la temperatura alle condizioni di equilibrio termodinamico. Anche in questo caso non sono necessari dati cinetici, ma solo le specie presenti nel sistema. Poichè questo approccio è ancora basato sull'assunzione di chimica veloce, è anch'esso impreciso e in particolare non è assolutamente in grado di descrivere la formazione di quelle specie che hanno una chimica molto lenta rispetto ai tempi caratteristici della turbolenza (ad esempio *soot* e ossidi di azoto)

3.3.6 Modello *Steady Laminar Flamelets*

Quando si descrive una fiamma non premiscelata è necessario a volte descrivere anche gli effetti di non equilibrio. Questo approccio (che chiameremo d'ora in poi *SLF*) modella la fiamma turbolenta come un insieme di fiamme laminari stazionarie [22]. In pratica si assume che il bilancio locale di queste fiamme sia simile a quello di una fiamma laminare stazionaria a controdiffusione. Questo tipo di fiamma può essere modellata semplicemente e descritta attraverso il profilo della variabile *mixture fraction* lungo la coordinata in direzione normale alla fiamma. Una fiamma a controdiffusione è rappresentata in figura 3.6. La zona di reazione è posta nel punto in cui i reagenti raggiungono la concentrazione stechiometrica [17]. I parametri utilizzati per identificare le diverse fiamme prototipo sono la *mixture fraction* stessa e lo *scalar dissipation rate* χ . Quest'ultimo è un'espressione della velocità di miscelazione e può essere interpretato come l'inverso del tempo caratteristico di diffusione. Pertanto quando χ è elevato si ha una diffusione molto veloce e χ è legato alla velocità di dissipazione delle fluttuazioni di f nel flusso turbolento. Dal punto di vista fisico se χ è elevato, il gradiente di *mixture fraction* e il profilo di temperatura sono ripidi e la zona di reazione è stretta. Per ulteriori incrementi di χ il gradiente diffusivo di calore supera il rilascio di calore nella fiamma e la *flamelet* si estingue. Pertanto utilizzando tale

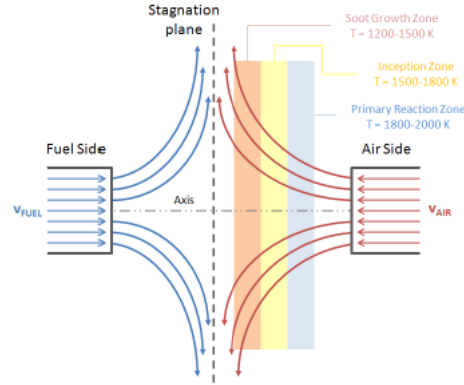


Figura 3.6: Geometria di una fiamma a controdiffusione

approcio è possibile tenere conto anche dei fenomeni di allontanamento dall'equilibrio o addirittura dei fenomeni di estinzione ad opera dello *strain* dovuto alla turbolenza. In quelle zone vicino alle condizioni stechiometriche di fiamma, individuate da un valore di *mixture fraction* f_{st} , la velocità di reazione è importante mentre, allontanandosi da queste zone, essa assume valori molto bassi. La zona reattiva viene considerata come un piccolo strato di gas con spessore al di sotto della scala di Kolmogorov. Il modello che deve essere ora scritto per descrivere una fiamma laminare diffusiva in una dimensione consta di equazioni per le specie e la temperatura, scritte però in funzione della *mixture fraction* [41]

$$\frac{1}{2}\rho\chi\frac{1}{Le_i}\frac{\partial^2\omega_i}{\partial f^2} + \dot{\Omega}_i - \frac{1}{2}\frac{\partial\omega_i}{\partial f}\rho\chi\frac{1}{Le_i^2}\frac{\partial Le_i}{\partial f} - \frac{1}{2}\left(1 - \frac{1}{Le_i}\right)\left(\frac{\partial\rho\chi}{\partial f} + \rho\chi\frac{\hat{C}_P}{\lambda}\frac{\partial}{\partial f}\left(\frac{\lambda_d}{\hat{C}_P}\right)\right) = 0 \quad (3.39)$$

$$\frac{1}{2}\rho\hat{C}_P\chi\frac{\partial^2 T}{\partial f^2} - \sum_{i=1}^{N_C} H_i\dot{\Omega}_i + \frac{\rho\chi}{2}\frac{\partial T}{\partial f}\left[\frac{\partial\hat{C}_P}{\partial f} + \sum_{i=1}^{N_S}\frac{1}{Le_i}\hat{C}_{P,i}\frac{\partial\omega_i}{\partial f}\right] = 0 \quad (3.40)$$

Il numero di Lewis è il rapporto tra la diffusività termica e materiale di ciascuna specie:

$$Le_i = \frac{\lambda_d}{\rho\hat{C}_P} \quad (3.41)$$

Lo *scalar dissipation rate*, indicato come visto in precedenza con la lettera χ è definito come:

$$\chi = \chi_{st} \cdot g(f) \quad (3.42)$$

dove il pedice st indica la quantità nella zona stechiometrica, mentre g è una funzione della *mixture fraction* derivata dal modello monodimensionale della fiamma laminare a controdiffusione:

$$g(f) = \exp\left(-2 \left[\operatorname{erf}^{-1}(2f - 1)\right]^2\right) \quad (3.43)$$

dove erf è la funzione degli errori. Lo *scalar dissipation rate* medio in condizioni stechiometriche viene modellato in questo modo:

$$\tilde{\chi}_{st} = C_\chi \tilde{f}''^2 \frac{\tilde{\epsilon}}{\tilde{k}} \quad (3.44)$$

in cui C_χ è una costante solitamente pari a 2 [39]. La *mixture fraction* e la sua varianza sono ottenute dalle rispettive equazioni di trasporto. La varianza dello *scalar dissipation rate* viene invece per semplicità posta uguale a zero. A questo punto è importante chiarire la procedura per valutare il valore medio di uno scalare (ad esempio temperatura e composizione). Anche qui si deve integrare una opportuna *PDF* dipendente oltre che dalla *mixture fraction* e dall'entalpia, anche dallo *scalar dissipation rate* in condizioni stechiometriche:

$$\tilde{\phi} = \int_0^1 \int_0^\infty \int_{-\infty}^\infty P(f) \cdot P(\chi_{st}) \cdot P(\hat{H}) \phi(f, \chi_{st}, \hat{H}) \cdot df \cdot d\chi_{st} \cdot d\hat{H} \quad (3.45)$$

Solitamente si sceglie la β -*PDF* per quanto riguarda la *mixture fraction* e delta di *Dirac* (ovvero nessuna distribuzione) per χ e l'entalpia. In realtà aumentando lo sforzo computazionale richiesto dal problema si possono utilizzare *PDF* più complesse per questi ultimi e ottenere risultati più accurati.

E' conveniente disporre i risultati riguardanti le *flamelets* in cosiddette *look-up tables*, di modo che essi siano ordinati e facilmente accessibili dall'esterno. Temperatura e composizione si ricavano quindi estraendo i dati dalle tabelle attraverso i valori calcolati di *scalar dissipation rate*, *mixture fraction* (e varianza) ed entalpia. In tal modo il calcolo della chimica viene disaccoppiato da quello della miscelazione turbolenta, in quanto questa è preprocessata e i risultati sono inseriti in questa tabella a cui si accede durante lo svolgimento del codice *CFD*. Questi ultimi passaggi sono ripresi e approfonditi in appendice A, dove si spiegherà nel dettaglio la modalità di costruzione di tali *look-up tables*.

3.3.7 Modello *Laminar Flamelets* non stazionarie

Nelle simulazioni in cui lo *strain* turbolento è molto basso, il modello appena visto tende a dare previsioni per le specie vicine alle condizioni di equilibrio. Questo porta a commettere stime sbagliate per tutte quelle specie che hanno un'evoluzione lenta, ovvero in cui la scala temporale del fenomeno chimico è molto più grande dell'inverso dello *scalar dissipation rate*. Il modello delle *Laminar Flamelets* non stazionarie può descrivere in maniera corretta anche queste situazioni. In questo caso infatti le equazioni vengono trasformate da stazionarie a non stazionarie, introducendo un termine di dipendenza dal tempo sia per la composizione che per la temperatura. La dipendenza dal tempo permette di descrivere fenomeni che avvengono su una scala temporale diversa da quella della diffusione. Il vantaggio derivante dall'utilizzo di tale modello è di poter simulare geometrie complesse anche con schemi cinetici dettagliati, con un vantaggio dal punto di vista del tempo di calcolo.

Capitolo 4

Il software *OpenFOAM*

4.1 Introduzione

Questo capitolo è dedicato al software utilizzato in questo lavoro di tesi. Come già accennato nel secondo capitolo, per le simulazioni di fiamme diffusive trattate in questo lavoro sono stati utilizzati i software *Ansys Fluent* e *OpenFOAM*. In realtà bisogna specificare che il primo dei due è stato utilizzato solo come software di supporto per confrontare la validità dei risultati ottenuti con il secondo. Infatti *OpenFOAM*, acronimo di *Open Field Operation And Manipulation*, è un software *open source*, che in sostanza significa libero (non proprietario) e liberamente modificabile. E' possibile apportare infatti qualsiasi aggiunta e modifica sul codice, scritto in linguaggio di programmazione *C++* e basato sulla programmazione ad oggetti. In questo capitolo, oltre a una panoramica generale sul concetto di *open source* e sulla programmazione a oggetti mediante *C++*, si cercherà di descrivere *OpenFOAM*, sia mostrandone il funzionamento e la struttura, sia specificando le peculiarità dal punto di vista della programmazione e dell'implementazione di nuove funzionalità.

4.2 *Open source*

In informatica il termine *open source* (che significa letteralmente “sorgente aperta”) indica un software rilasciato con una licenza per cui il codice sorgente è lasciato disponibile a eventuali sviluppatori [49]. Durante gli anni Ottanta e Novanta, con la crescita esponenziale legata al mondo dell'informatica, si è passati da un modo di agire “libero” alle restrizioni

del software proprietario. La nascita dell'*open source* ha cercato di essere un'inversione di tendenza, con un ritorno alla libera circolazione delle informazioni relative ai programmi. La vera rivoluzione quindi sta nel fatto che software di questo tipo, oltre a essere gratuiti e liberamente scaricabili e utilizzabili, sono pensati per poter essere modificati da chiunque. In questo modo diverse realtà anche distanti tra di loro possono interagire contribuendo ad ottenere alla fine un prodotto sempre migliore. Altro aspetto fondamentale è che un software *open source* può in ogni caso essere distribuito da chiunque (per fare un semplice esempio non è reato produrne copie su cd vergini e distribuirle anche a pagamento) sia nella versione originale sia nelle versioni modificate e migliorate.

La storia dell'*open source* inizia con la creazione della *Free Software Foundation* da parte di Richard Stallman, che partì dall'idea che il software nelle istituzioni accademiche era sempre stato libero. Nel 1984 Stallman, sentendosi oppresso dai limiti dei software proprietari, promosse un progetto denominato *GNU* per la creazione di un sistema operativo libero. Nello stesso tempo egli enunciò i principi fondamentali del *free software* su cui si basa il modello *open source*. Ancora oggi non è stato creato il *kernel* del sistema operativo *GNU*, ma è solo grazie a tale iniziativa che pochi anni dopo, grazie a Linus Torvald, studente di informatica presso l'università di Helsinki, vide la luce il *kernel linux*, sviluppo di *minix*, sistema operativo basato su *unix* che Torvald aveva creato per uso personale. *Linux* venne subito distribuito con una licenza libera e per capire lo sviluppo che ci può essere stato negli ultimi 20 anni basti pensare alle numerosissime diverse distribuzioni di *linux* che si possono utilizzare al giorno d'oggi, fornite nella maggior parte dei casi di interfacce grafiche create ad hoc, e che ormai possono essere praticamente utilizzate sui computer personali in alternativa ai sistemi operativi *Microsoft Windows* e *Mac OS* senza grossi problemi per l'utente medio. In particolare negli anni sono stati sviluppati software *open source* di supporto ai sistemi operativi *linux* proprio con il fine di renderli competitivi con i sistemi proprietari.

In sostanza i vantaggi nell'utilizzo di software *open source* sono molteplici: oltre al prezzo, che può essere considerato il fattore principale, perchè nella maggior parte dei casi questi software sono gratuiti, bisogna aggiungere che se un programma è ben ideato, questo potrà usufruire del lavoro e dell'assistenza di una grande quantità di programmatori da tutto il mondo che contribuiranno a perfezionarlo e migliorarlo. In tal modo l'autore avrà la possibilità di fornire ai propri clienti un programma assai migliore di quello che avrebbe creato da sè e di farsi pagare per questo e per la successiva assistenza. Inoltre mentre i software proprietari ad ogni rilascio devono seguire precise regole di *marketing*, che portano ogni volta a dover inserire nuove funzionalità o nuove caratteristiche per giustificare

l'acquisto di una nuova licenza da parte dell'utente, nei software *open source* chi definisce le caratteristiche e l'evoluzione del programma stesso sono gli utenti della comunità di sviluppo. L'utente di un software di questo tipo poi, se ne è in grado, può perfezionare il prodotto come meglio crede, disponendo di tutti i mezzi per farlo e non dovendo dipendere da nessun fornitore. Infine questi software sono generalmente multiplatforma, il che significa che se un software viene inizialmente programmato per essere utilizzato su *linux*, solitamente viene creata anche la versione per *Windows* e *macOS*, o comunque tendenzialmente non sarebbe difficile crearla. Il processo contrario, soprattutto di alcuni software a pagamento, è molto più difficile. D'altra parte si possono mettere in luce alcuni svantaggi: innanzitutto il supporto al cliente che può essere dato a un software proprietario non esiste per l'*open source* se non a pagamento e in alcuni casi può essere fondamentale. Inoltre spesso i software proprietari, nonostante il *gap* si stia molto assottigliando, sono più semplici da utilizzare, inoltre non tutti gli utenti sono disposti a spendere del tempo per imparare a usare certi tipi di software, quando pagando potrebbero utilizzare prodotti più immediati.

Dopo aver messo in luce vantaggi e svantaggi resta da dire che nella società vi è una netta divisione tra chi pensa che il software libero non sia assolutamente in grado di competere con quello proprietario, che per definizione è migliore e chi dall'altra parte, sposando in pieno la causa dell'*open source*, ne fa una ragione e uno stile di vita e critica in continuazione i produttori di software proprietari. Probabilmente, per quanto detto finora, nessuna delle due fazioni ha completamente ragione.

In questo lavoro di tesi si è deciso di sfruttare i vantaggi dell'*open source* utilizzando il software *OpenFOAM* che si inserisce proprio in questo contesto: funziona solo in ambienti *linux* e non è di semplice utilizzo come alcuni codici commerciali, ma d'altra parte è gratuito e soprattutto il codice è disponibile e lo si può manipolare a piacimento. Nel nostro caso si è cercato di valutarne le possibilità di utilizzo a livello accademico nello studio di fiamme turbolente non premiscelate.

4.3 Il *C++* e la programmazione ad oggetti

Il *C++* è un linguaggio di programmazione nato nel 1983 come miglioramento del linguaggio *C* e ormai è largamente diffuso. La principale caratteristica del *C++* è il fatto di permettere la programmazione orientata agli oggetti [12] rispetto a quella procedurale, utilizzata ad esempio dal suo predecessore. In un linguaggio di tipo procedurale il cuore

della programmazione sono le *subroutine* ovvero le funzioni utilizzate per implementare gli algoritmi risolutivi, che vengono intese come blocchi di codice distinti che interagiscono fra loro. In un linguaggio orientato agli oggetti invece si raggruppa in un'unica entità (detta classe) sia i dati che le procedure (cioè le funzioni) che operano su di essi, creando appunto un "oggetto" dotato di proprietà (dati) e metodi (procedure) che operano sui dati dell'oggetto stesso. Un programma viene perciò progettato e realizzato attraverso oggetti che interagiscono gli uni con gli altri. Quattro sono le principali caratteristiche che rendono la programmazione ad oggetti differente e migliore rispetto a quella procedurale.

4.3.1 Incapsulamento

In un linguaggio procedurale vi è una netta distinzione tra dati e funzioni. Le funzioni contengono le implementazioni di un algoritmo e servono ad elaborare i dati. Questo significa che chi utilizza una di queste funzioni deve conoscere molti particolari su come è stata programmata in modo da fornirle i dati in modo corretto, e spesso servono informazioni che in realtà non sarebbero necessarie. Inoltre potrebbero sorgere problemi quando si va a utilizzare un algoritmo in maniera iterativa, e si desidera risolvere più problemi contemporaneamente, in particolare nella gestione dei dati e nella memorizzazione. In una programmazione a oggetti i dati e le funzioni sono *incapsulati* in un'unica entità chiamata oggetto. È l'oggetto che gestisce contemporaneamente i dati e le funzioni che li elaborano. L'utente vede i dati solo attraverso il filtro dell'oggetto e non deve necessariamente conoscere le modalità attraverso cui vengono manipolati funzioni e dati. Inoltre ogni oggetto gestisce solo i suoi dati, che non interferiscono con quelli di un altro oggetto della stessa classe. Si possono avere perciò contemporaneamente attivi molti oggetti che risolvono problemi analoghi senza doversi preoccupare dei dati necessari ad ognuno di essi all'interno delle funzioni che utilizzano.

4.3.2 Mascheramento

In una programmazione procedurale in cui i dati e le funzioni sono entità distinte è l'utente il responsabile del corretto uso dei dati fra le diverse chiamate alle varie funzioni. Non si può evitare a priori un uso scorretto dei dati. Utilizzando gli oggetti è invece possibile *mascherare* quei dati che l'utente non deve manomettere e controllare l'accessibilità a quei dati. Questo è possibile nella dichiarazione in cui i vari elementi della classe possono

essere definiti privati e quindi non utilizzabili dall'esterno, protetti, cioè utilizzabili ma non modificabili, oppure pubblici, cioè utilizzabili dall'esterno.

4.3.3 Polimorfismo

In un linguaggio procedurale è obbligatorio utilizzare un identificatore differente per ogni funzione anche quando si ha a che fare con funzioni che concettualmente risolvono problemi simili. In un linguaggio orientato agli oggetti invece il polimorfismo permette di creare diverse funzioni con lo stesso nome, le quali verranno richiamate a seconda degli argomenti che di volta in volta gli vengono passati. In questo modo il codice risulta decisamente più pulito e concettualmente chiaro.

4.3.4 Ereditarietà

L'ereditarietà è forse la proprietà più importante e che porta i maggiori vantaggi alla programmazione ad oggetti. Essa consiste nella possibilità di derivare nuove classi da una classe preesistente in modo che gli oggetti delle nuove classi ereditino le caratteristiche degli oggetti della classe madre. Durante il processo di derivazione ovviamente si possono aggiungere nuove funzionalità o modificarle ampliando la classe preesistente. Questa proprietà fornisce ai linguaggi come il *C++* una potenzialità enorme nel riutilizzo e nell'estensione di programmi già esistenti. E' possibile oltretutto utilizzare delle classi per derivarne di nuove senza necessariamente conoscere il codice sorgente delle classi madre. Questa potenzialità del *C++* è dovuta soprattutto all'uso delle funzioni virtuali. Utilizzando tali funzioni è possibile creare una classe base comune a più classi che fornisce la stessa interfaccia per l'utente. La classe base si prende cura di selezionare la funzione appropriata appartenente a una sua classe derivata quando l'utente utilizza un oggetto di quella classe derivata.

4.4 Il codice *OpenFOAM*

OpenFOAM (*Open Field Operation And Manipulation*) è una libreria di calcolo *open source* sviluppata in *C++* basata sulla meccanica dei continui. E' stata creata per la prima volta nel 1993 ed è tutt'ora in continuo sviluppo (nel 2009 è stata rilasciata la versione 1.6).

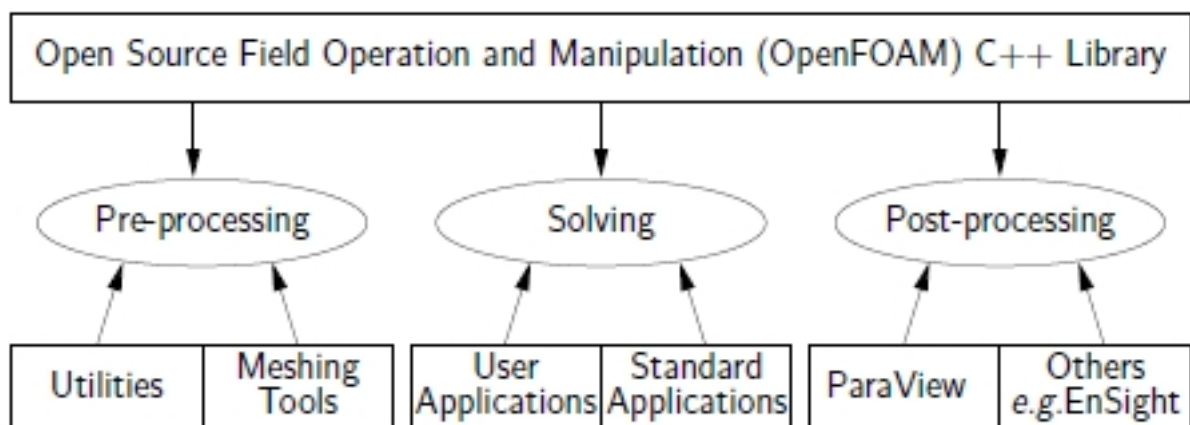


Figura 4.1: Struttura di *OpenFOAM* [36]

È prodotta per ambienti *linux* ed è distribuita dalla OpenCFD Ltd [37] con licenza GNU GPL (*General Public License*), i cui termini sono infatti elencati nel primo capitolo della *user guide* del programma [36]. Viene utilizzato per risolvere una grande varietà di problemi, ma è pensato soprattutto per potere eseguire simulazioni di fluidodinamica (*CFD*). *OpenFOAM* perciò fa suoi tutti i vantaggi dei programmi *open source* di cui si è parlato nel paragrafo 4.2, e della programmazione orientata agli oggetti disponibile attraverso il linguaggio *C++*. Grazie a questo linguaggio tra l'altro il codice assume alcune caratteristiche che rendono l'implementazione di nuove funzionalità più semplice e immediata. Queste caratteristiche verranno più precisamente illustrate in seguito. Nonostante sia già disponibile la versione 1.6, nel nostro lavoro è stata utilizzata la 1.5 *dev*, già utilizzata da tempo, quindi maggiormente testata e contenente alcune applicazioni non presenti nella versione base.

Come la maggior parte dei simulatori di *CFD*, oltre al vero e proprio programma di simulazione, *OpenFOAM* si avvale di alcune procedure di *pre-processing* e *post-processing* dei dati, le prime necessarie alla preparazione della simulazione e le seconde utili alla visualizzazione dei risultati della stessa. Lo schema generale è visibile in figura 4.1. Come si vede la simulazione vera e propria viene effettuata attraverso delle applicazioni, definite *solutori*, cioè degli eseguibili che possono essere quelli standard già presenti nella versione distribuita del programma, oppure possono essere creati ad hoc dall'utente che possieda le competenze necessarie in materia di programmazione e di *CFD*. Il *pre-processing* è affidato a delle *utilities*, eseguibili che svolgono altri tipi di funzioni, a partire dalla generazione della *mesh* fino ad esempio all'inserimento di condizioni non costanti sui diversi contorni.

Il *post-processing* è invece per lo più affidato a un programma grafico, *Paraview*, utile a visualizzare i risultati, e comunque ad altre *utilities* adeguate allo scopo. E' necessario aggiungere all'elenco degli elementi caratterizzanti questo software anche le *librerie* che, sebbene non direttamente utilizzabili dall'utente, contengono le dichiarazioni e il codice delle classi utilizzate e sono pertanto fondamentali per il funzionamento del codice stesso. Infine restano da citare i *case*, ossia le simulazioni vere e proprie, in cui sono presenti i files contenenti le condizioni iniziali e le specifiche della simulazione e i files contenenti i risultati.

4.4.1 La struttura di *OpenFOAM*

Una volta che *OpenFOAM* è stato compilato all'interno del sistema (nel nostro caso si è utilizzato un computer con processore dual core T5500, 2 gb di ram e con installato il sistema operativo *Ubuntu 9.10*), avremo una cartella denominata "*OpenFOAM*" al cui interno sono presenti tre sottocartelle: la prima, *OpenFOAM-1.5-dev*, contiene il programma vero e proprio, la seconda, *nomeutente-1.5-dev*, è adibita ai solutori e alle simulazioni impostate dall'utente. La terza e ultima cartella, *Third Party*, include tutti i software esterni di cui *OpenFOAM* si avvale nel suo funzionamento, ad esempio il già citato *paraview*, ma anche, per esempio, il compilatore di *C++*.

Il contenuto della cartella principale del programma è visibile in figura 4.2. Val la pena focalizzarsi su alcune sue sottocartelle. La cartella *applications* contiene tutti i codici degli eseguibili già presenti nel programma, sia i solutori che le *utilities*. Nella cartella *doc* è contenuta ovviamente la documentazione del programma, cioè la *user guide* [36] e la *programmers guide* [35]. La cartella *src* contiene tutte le librerie di cui si è parlato precedentemente e verrà discussa in seguito. La cartella *tutorials* contiene dei casi base preimpostati che possono aiutare l'utente nel set delle proprie simulazioni o possono venire utilizzati nell'apprendimento del programma. Infine la cartella *wmake* contiene per lo più files necessari alla compilazione. Analizziamo ora nel dettaglio gli elementi che compongono il programma.

4.4.1.1 I solutori

Come già detto il solutore è un eseguibile che permette di risolvere un determinato problema, quindi di svolgere una certa simulazione. Tutti i solutori hanno un *background* simile, ma si differenziano ovviamente a seconda dei problemi che devono risolvere. E'

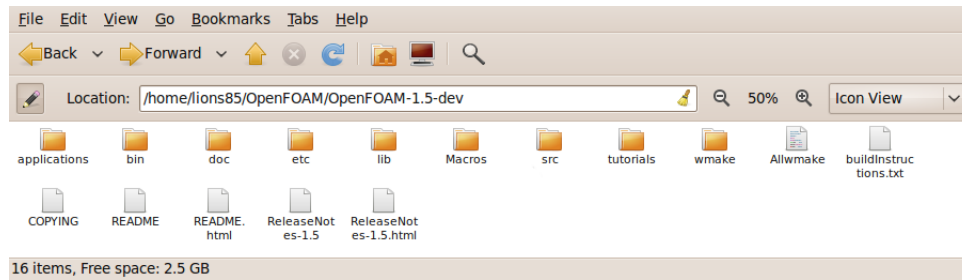


Figura 4.2: Contenuto della cartella principale del programma

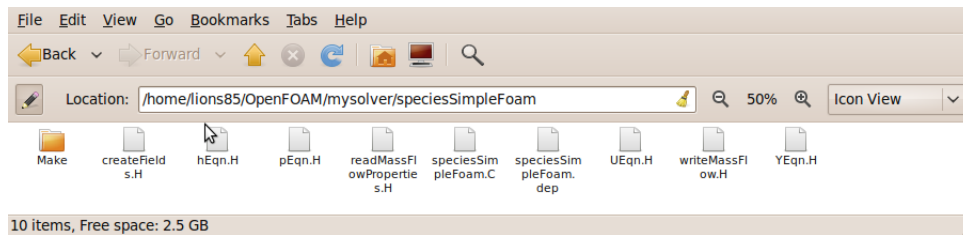


Figura 4.3: Contenuto della cartella del solutore *speciesSimpleFoam*

utile mostrare un solutore standard per analizzarne la struttura. Per farlo si utilizzerà il solutore stazionario dal quale si è partiti per implementare i modelli di combustione, chiamato *speciesSimpleFoam*, che non è presente nei solutori rilasciati nella versione base del programma, ma ci è stato fornito dal dipartimento di Energia del Politecnico di Milano. Questo solutore permette di simulare la diffusione di diverse specie in ambiente turbolento ed è una derivazione del solutore stazionario *simpleFoam* presente nella versione base, che però in aggiunta risolve le equazioni di trasporto delle specie. La cartella contenente i files del risolutore è visibile in figura 4.3. Nella sottocartella *Make* sono contenuti i files creati con la compilazione e anche due altri files: il primo, *files*, serve per indicare il codice da compilare e a indicare la cartella in cui si vuole compaia il file eseguibile *.bin*. Il secondo, *options*, contiene riferimenti a tutte le librerie da includere nel solutore stesso. Nella cartella principale invece è presente il file contenente il *main* (nel nostro caso *speciesSimpleFoam.C*), al cui interno sono richiamati tutti gli altri files della cartella: *createFields.H* contiene il codice attraverso cui vengono creati tutti i campi utilizzati nel programma (ad esempio velocità, pressione ecc.ecc.) e la creazione degli altri oggetti utilizzati nel programma stesso. Ad esempio questa è l’inizializzazione del campo di velocità:

```
volVectorField U
(
    IOobject
    ( "U",
      runtime.timeName(),
      mesh,
      IOobject::MUST_READ,
      IOobject::AUTO_WRITE
    ),
    mesh
);
```

mentre questa è la creazione dell'oggetto relativo alla turbolenza:

```
autoPtr<compressible::RASModel> turbulence
(
    compressible::RASModel::New
    (
        rho,
        U,
        phi,
        thermo()
    )
);
```

Gli altri files, in cui è presente la parola *eqn*, sono anch'essi richiamati nel *main* e contengono il codice per la risoluzione delle equazioni del trasporto, rispettivamente della velocità, dell'entalpia, della pressione e delle specie. Infine i files *readMassFlowProperties.H* e *writeMassFlow.H* sono files aggiuntivi che permettono una verifica delle portate di fluido in ingresso e in uscita.

I solutori devono essere compilati dalla *shell* di *Ubuntu* attraverso il comando *wmake*, che in caso di errori nel codice restituisce un *output* che consente di riconoscere dove si trova l'errore e correggerlo.

4.4.1.2 Le librerie

La cartella *src* nominata in precedenza contiene un numero molto grande di sottocartelle, come si vede in figura 4.4. Le librerie infatti sono suddivise in grandi categorie a seconda

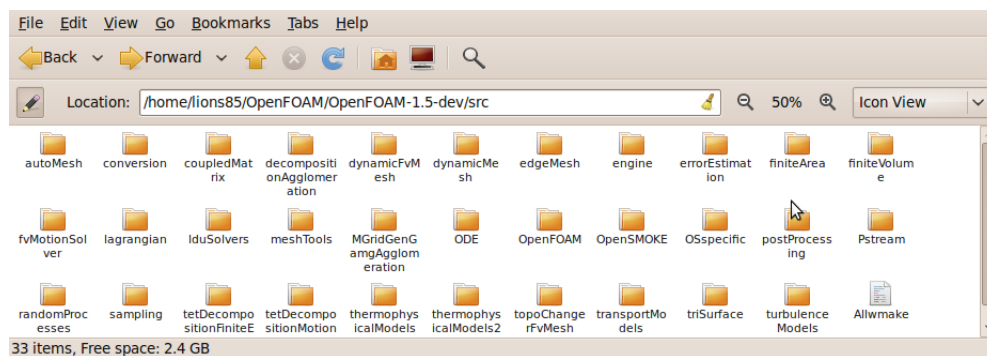


Figura 4.4: Contenuto della cartella *src*

della loro funzione. Ad esempio nella cartella *finiteVolume* sono presenti tutte le librerie attraverso cui vengono risolte le equazioni con la discretizzazione dei volumi finiti, in quella *ODE* sono contenute quelle per la risoluzione di equazioni differenziali ordinarie e così via. Particolare attenzione va posta alla cartella *turbulenceModels*, molto importante in quanto contiene anche l'implementazione del modello di turbolenza *k-epsilon*. Infine la cartella *thermophysicalModels* contiene l'implementazione della termodinamica, fondamentale per la soluzione dei nostri problemi. Ovviamente tutte queste librerie sono programmate utilizzando l'approccio di programmazione ad oggetti, quindi ognuna contiene sostanzialmente delle dichiarazioni di classi, spesso incastonate una nell'altra, secondo il concetto di ereditarietà e comunque collegate tra di loro come approfondiremo in seguito. Per la compilazione di tutte le librerie di una certa cartella si può utilizzare da terminale il comando *Allwmake*, mentre se si vuole compilare solo una singola libreria il comando è *wmake libso*. La compilazione avviene nello stesso modo dei solutori.

4.4.1.3 I casi

La cartella di un caso si presenta all'inizio della simulazione come in figura 4.5. Nella cartella *0* sono presenti i files in cui impostare le condizioni al contorno. Di tali files si parlerà nel prossimo capitolo quando sarà presentata l'implementazione dei modelli di combustione. Nella cartella *system* sono presenti i seguenti files:

- *controlDict*: in questo dizionario si possono impostare i parametri della simulazione, tra cui lo *step* iterativo a cui cominciare la simulazione, l'intervallo di iterazioni dopo le quali scrivere i risultati e la precisione di scrittura.
- *fvSchemes*: in questo file vanno inseriti gli schemi numerici visti nel paragrafo 2.2.1.

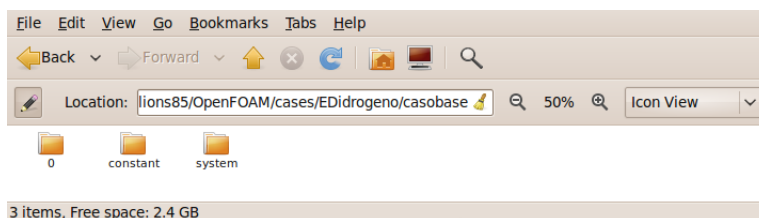


Figura 4.5: Contenuto della cartella di un caso

- *fvSolution*: in questo file vanno inseriti i fattori di sottorilassamento e le tolleranze per le grandezze di cui si va a risolvere l'equazione del trasporto. I fattori di sottorilassamento sono utilizzati dal risolutore per aiutare a raggiungere la convergenza. Più piccoli sono questi fattori più aumentano i tempi di calcolo ma allo stesso tempo le probabilità di raggiungere la convergenza. Le tolleranze invece rappresentano dei parametri che indicano il livello a cui i residui sono abbastanza piccoli da ritenere la soluzione sufficientemente accurata [37, 21].

All'interno della cartella *constant* sono presenti dei files in cui è possibile assegnare dati o parametri necessari per la simulazione. Ovviamente questi files saranno diversi a seconda del solutore usato. Nel nostro caso i files comuni a tutti i risolutori sono stati:

- *RASProperties*: in questo file vanno settati i parametri della turbolenza e il modello utilizzato.
- *environmentalProperties*: qui vengono inserite le forze agenti dall'esterno sul sistema, e.g. la forza gravitazionale.
- *massFlowProperties*: qui vanno inseriti i nomi dei contorni di ingresso e uscita per il calcolo della portata massiva in entrata e in uscita.
- *polyMesh*: in questa cartella sono presenti i files relativi alla costruzione della *mesh* nonché il file *blockMeshDict*, dizionario attraverso cui è possibile facilmente creare la propria *mesh*.

Altri files contenuti in questa cartella variano a seconda dei risolutori utilizzati nel nostro lavoro, e ne discuteremo nel prossimo capitolo.

Una volta che il caso è stato simulato, ad ogni *step* in cui si è deciso di far scrivere i risultati, viene creata una nuova cartella simile a quella iniziale ma che ha come nome lo step iterativo a cui si riferisce e che contiene appunto i valori di tutte le grandezze che potranno essere poi visualizzati nella fase di *post processing*.

4.4.1.4 Le *utilities*

Come già detto le *utilities* sono degli eseguibili che vengono utilizzati a supporto del programma in particolar modo nella gestione del *pre* e del *post processing*. Verranno qui elencate soltanto le *utilities* che sono state usate nel corso del nostro lavoro:

- *blockMesh*: è utilizzata per la creazione della *mesh* a partire da un file di input. Questo argomento sarà trattato nel paragrafo 4.4.3
- *checkMesh*: controlla la *mesh* presente all'interno del caso e verifica la presenza di eventuali anomalie.
- *mapFields*: attraverso questa *utility* è possibile rimappare tutti i campi della *mesh* di una nuova simulazione partendo da una già esistente. Molto utile per esempio nel caso in cui sia necessario aumentare i punti della griglia e non si voglia ricominciare la simulazione da capo ma da una soluzione già trovata precedentemente.
- *funkySetFields/GroovyBC*: permettono di inserire condizioni non costanti sui contorni. Queste applicazioni sono state utilizzate in maniera molto limitata in quanto è stato più comodo inserire le condizioni variabili direttamente nel codice del solutore.
- *paraFoam/foamToVTK*: queste applicazioni permettono di visualizzare i risultati all'interno del software *paraview*. La prima apre direttamente i risultati nel programma, ma nel nostro caso per un difetto di compilazione dello stesso *paraview* non è stato possibile utilizzarla. E' stata perciò utilizzata la seconda, che converte le cartelle dei risultati in un formato leggibile direttamente da questo software.
- *sample*: questa applicazione estrae i risultati della simulazione attraverso un dizionario (*sampleDict*) che può essere aggiunto nella cartella *system*. I risultati possono essere esportati in files di formati diversi, e sono disponibili numerose opzioni da inserire nel dizionario. Nel nostro caso è stato utilizzato per visualizzare i risultati lungo l'asse di simmetria delle fiamme e in direzione radiale lungo alcune traverse di riferimento.

4.4.2 La programmazione *C++* in OpenFOAM

Il codice *OpenFOAM* possiede alcune peculiarità che rendono la programmazione più semplice. La prima di queste caratteristiche si trova nella classe `geometricField <Type>` [35]

attraverso la quale è possibile creare oggetti in cui i campi delle variabili sono associati alla *mesh*. Un oggetto di questa classe contiene i valori della grandezza presa in considerazione in ogni cella della *mesh* e sui contorni. Sono presenti inoltre le unità di misura e i valori della grandezza stessa all'iterazione precedente che devono essere disponibili se si usano fattori di sottorilassamento. A seconda del modo in cui si definiscono i campi all'interno della *mesh* si distingue tra tre tipi di `geometricField <Type>`:

`volField<Type>`: il campo è definito nei centri delle celle

`surfaceField<Type>`: il campo è definito sulle facce delle celle

`pointField<Type>`: il campo è definito sui vertici delle celle

Il vantaggio nell'utilizzare questa classe sta nel fatto che il programmatore non ha bisogno di pensare in termini di celle o facce della *mesh*, ma in termini di oggetti (e.g. velocità, densità, ecc.), definiti come campi contenenti valori e di cui non sono importanti le dimensioni e la grandezza perchè tutte queste caratteristiche sono conservate all'interno dell'oggetto. Per fare un esempio basti riguardare l'inizializzazione del campo di velocità (U) nel paragrafo 4.4.1.1: il campo è creato come un'oggetto di tipo `volVectorField` e al momento dell'inizializzazione assume già la dimensione della *mesh*.

Una conseguenza dell'uso di questo tipo di classe si ritrova nella discretizzazione delle equazioni. Abbiamo già visto che il metodo dei volumi finiti permette di convertire le *PDE* in un sistema di equazioni algebriche espresso in forma matriciale in questa maniera:

$$[A][x] = [b] \tag{4.1}$$

I vettori $[x]$ e $[b]$ per quanto visto sopra altro non sono che oggetti di tipo `volField<Type>`. La matrice $[A]$ invece è una lista di coefficienti e non può essere descritta da un oggetto di questo tipo. Perciò esiste una classe creata ad hoc: `fvMatrix<Type>`. Essa tra l'altro supporta molte delle operazioni standard tra matrici, come l'addizione, la sottrazione e la moltiplicazione. Ogni termine all'interno della *PDE* è rappresentato in OpenFOAM usando le classi `finiteVolumeMethod` e `finiteVolumeCalculus`, abbreviate in `fvm` e `fvc`. Esse contengono funzioni statiche che rappresentano gli operatori differenziali e la loro differenza è che le funzioni della prima classe restituiscono un oggetto di tipo `fvMatrix<Type>` mentre quelle della seconda restituiscono un `geometricField <Type>`. L'idea che sta alla base dell'uso di questi operatori è pensare alle equazioni differenziali come una somma di singoli operatori differenziali che possono essere discretizzati separatamente con differenti schemi di discretizzazione. Sono definiti ad esempio questi operatori:

- `fvm::ddt` = $\frac{\partial}{\partial t}$
- `fvm::div` = $\sum_i \frac{\partial}{\partial x_i}$
- `fvm::laplacian` = $\sum_i \frac{\partial^2}{\partial x_i^2}$

e anche molti altri per cui si rimanda alla guida per programmatori [35]. Risolvere un'equazione differenziale alle derivate parziali è quindi, tramite questi operatori, solo questione di combinare in modi differenti lo stesso set di operatori differenziali. Si può fare un esempio considerando l'equazione della conservazione della quantità di moto:

$$\frac{\partial \rho \vec{u}}{\partial t} + \nabla \cdot (\rho \vec{u} \vec{u}) - \nabla \cdot (\mu \nabla \vec{u}) = -\nabla p \quad (4.2)$$

Essa può essere molto semplicemente implementata in *OpenFOAM* in questa maniera:

```
solve
(
    fvm::ddt(rho, U)
  + fvm::div(phi, U)
  - fvm::laplacian(mu, U)
  ==
  -fvc::grad(p)
)
```

Per concludere questa carrellata sulle particolarità di *OpenFOAM* è opportuno mostrare un esempio di come il programma faccia largo uso della programmazione orientata agli oggetti e in particolare di come le classi siano legate e interagiscano tra loro. Per far questo a titolo di esempio mostriamo la classe `geometricField`. E' utile osservare la sua rappresentazione grafica *UML* (*Unified Modeling Language*), che può essere trovata sulla *C++ source guide* nel sito del programma per tutte le tipologie di classi e che per la classe in esame è visualizzata in figura 4.6. Il significato è il seguente: all'interno dei blocchi vi sono le diverse classi, e la classe da cui le altre sono generate ha lo sfondo grigio. Le frecce continue hanno il significato di eredità tra due classi, in particolare pubblica se il colore della freccia è blu, protetta se è verde e privata se è rossa. Una freccia tratteggiata viola significa che la classe è contenuta o utilizzata da un'altra classe. La freccia è etichettata con le variabili attraverso cui la classe a cui si punta è accessibile. Infine la freccia tratteggiata gialla

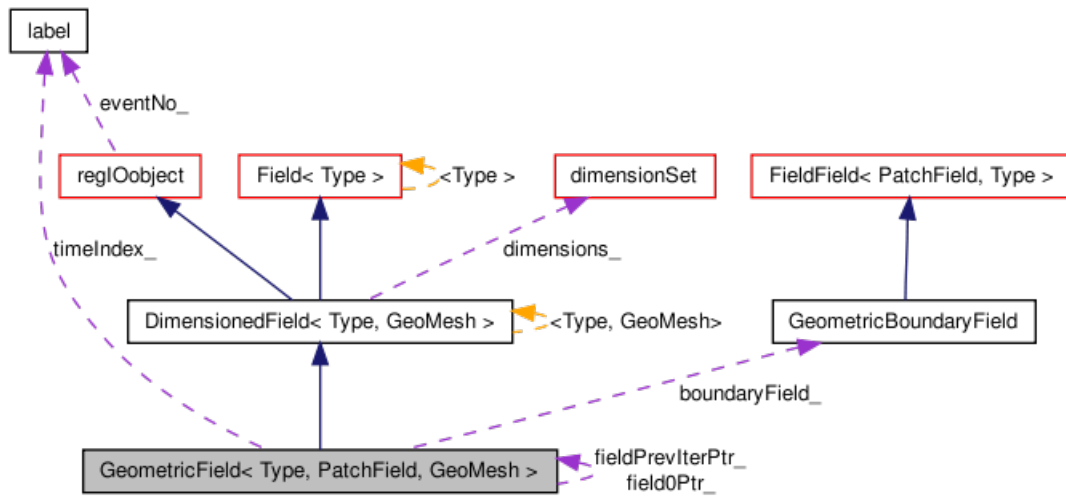


Figura 4.6: diagramma *UML* per la classe *GeometricField*

indica una relazione tra un'istanza *template* e la classe *template* da cui è stata istanziata. La freccia è etichettata con i parametri dell'istanza. Nel nostro esempio questo grafico ha il seguente significato: la classe *GeometricField*, come già detto, può rappresentare qualsiasi campo, e consiste di un *internalField* e un *boundaryField*. Il primo, rappresentato nella figura dalla classe *Field<Type>*, conserva la lista dei valori per ogni punto della *mesh*, mentre il secondo, rappresentato appunto da *GeometricBoundaryField*, conserva i valori dei contorni suddividendoli in una *patchField* per ogni contorno. Nella classe *dimensionSet* sono conservati i riferimenti alle unità di misura.

4.4.3 La creazione della *mesh* in *OpenFOAM*

La creazione della *mesh* in *OpenFOAM* può avvenire in molti modi: si possono ad esempio utilizzare diversi programmi oppure delle *utilities* che trasformino la *mesh* di *Fluent* in una utilizzabile in *OpenFOAM*. Il modo più semplice, che è stato usato nel nostro lavoro, è quello di utilizzare l'*utility* *blockMesh* che sfrutta il file *blockMeshDict*. Questo dizionario sostanzialmente è formato da 3 blocchi principali:

- **vertices**: in questo blocco vanno messe le coordinate dei vertici della *mesh*, come in questo esempio, in cui si vuole rappresentare un cubo di lato unitario (di fianco a ogni vertice, commentato, è stato posto il numero progressivo del vertice, che sarà utilizzato nella parte successiva del file):


```
vertices
(
  (0 0 0) //0
  (0 1 0) //1
  (1 1 0) //2
  (1 0 0) //3
  (0 0 1) //4
  (0 1 1) //5
  (1 1 1) //6
  (1 0 1) //7
);
```

- **blocks:** qui vanno posizionati in sequenza i diversi blocchi che costituiscono la *mesh*, scrivendo all'interno delle parentesi i numeri dei vertici che rispettivamente compongono un determinato blocco. Nella seconda parentesi vanno messi il numero di celle che si vuole avere in ognuna delle tre direzioni, e nell'ultima parentesi è possibile graduare la *mesh*. Continuando con l'esempio del cubo abbiamo un solo blocco, 10 celle in ogni direzione e il *grading* è lasciato al valore unitario:

```
blocks (
hex (0 1 2 3 4 5 6 7)          (10 10 10)    simpleGrading (1 1 1)
);
```

- **patches:** in quest'ultima sezione si devono inserire i vertici delle superfici di contorno della *mesh* e devono essere definiti i nomi e le funzioni di ciascun contorno. Sono possibili diverse opzioni:

patch: è una condizione generica, si usa solitamente per gli *inlet* e gli *outlet*

wall: è utilizzata per le pareti

simmetryPlane: si usa per i piani di simmetria

wedge: si usa per risolvere casi assialsimmetrici in due dimensioni. La *mesh* è uno spicchio del cilindro e la parte superiore e inferiore di questo spicchio devono avere questa condizione.

empty: si usa in casi come quello assialsimmetrico in cui si deve indicare un contorno senza spessore (come l'asse di simmetria del cilindro)

cyclic: si usa per piani ciclici

processor: talvolta c'è bisogno di dividere la *mesh* in più parti per svolgere i calcoli

su diversi processori. I contorni che congiungono tali parti devono avere questa condizione.

Nel nostro esempio abbiamo 6 superfici e per semplicità le considereremo tutte con un'unica condizione `wall` che chiameremo `parete`:

```
patches
(
    wall parete
    (
        (0 1 2 3)
        (4 5 6 7)
        (0 3 7 4)
        (1 2 6 5)
        (0 1 5 4)
        (2 3 7 6)
    )
);
```

Questa *mesh* di esempio può essere visualizzata in tre dimensioni tramite il software `paraview` ed è mostrata in figura 4.7. Nel nostro lavoro abbiamo avuto a che fare con simulazioni di tipo assialsimmetrico, in cui sostanzialmente la *mesh* è una sezione bidimensionale del cilindro corrispondente al bruciatore. La simulazione viene poi risolta in tre dimensioni attraverso un comando nel solutore. Se in *Fluent* la *mesh* corrisponde a una vera e propria griglia in due dimensioni, in *OpenFOAM* per poter effettuare una simulazione assialsimmetrica la *mesh* deve corrispondere a uno spicchio dell'intero cilindro, con l'angolo di 5° [36]. L'angolo è talmente piccolo che nei prossimi capitoli, quando verranno visualizzate le *mesh* utilizzate per simulare le nostre fiamme, non si potranno sostanzialmente distinguere da *mesh* bidimensionali.

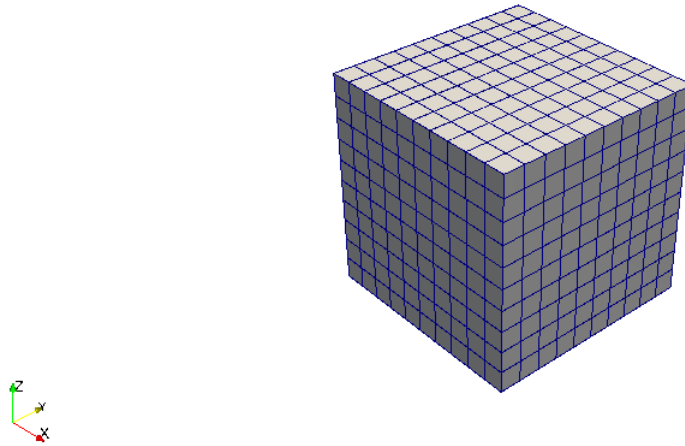


Figura 4.7: *mesh* di esempio visualizzata attraverso il software *paraview*

Capitolo 5

Implementazione di modelli di combustione

5.1 Introduzione

In questo capitolo verrà mostrata l'implementazione del codice necessaria per poter utilizzare in *OpenFOAM* i modelli di combustione visti nel terzo capitolo. L'unico modello di combustione già presente nella versione base del programma è il *Pasr*, modello su cui in questo lavoro non ci si è soffermati. Tale modello è tra l'altro disponibile nel programma base solo usando il solutore *reactingFoam* che è un solutore non stazionario e quindi si allontana dai nostri obiettivi. L'idea di base sarebbe stata quella di implementare tutti i principali modelli (*ED*, *EDFR*, *EDC*, *SLF*) per verificarne il funzionamento comparandone i risultati con quelli di *Fluent*, in cui i modelli sono già presenti, nonché con dei dati sperimentali. L'utilizzo di *OpenFOAM* ha richiesto parecchio tempo nel risolvere svariati tipi di problemi, per questo è stato possibile implementare solo due modelli, l'*ED* e l'*SLF*, che hanno comunque garantito la possibilità di collaudare in maniera adeguata il software. La prima parte del capitolo è dedicata perciò all'implementazione del modello *ED*, che si è rivelata abbastanza semplice dal punto di vista della programmazione, e viene mostrato come il caso base visto nel capitolo precedente si arricchisca di alcuni files con cui impostare il modello. La seconda parte vede l'implementazione del modello *SLF*, la sua descrizione e le differenze nei settaggi dei casi rispetto al modello precedente.

5.2 Modello *Eddy Dissipation*

5.2.1 L'implementazione

Abbiamo visto nel paragrafo 4.4.1.1 che tra i files del solutore *speciesSimpleFoam* ne sono presenti alcuni il cui codice risolve le diverse equazioni del trasporto. Nel modello *ED* l'equazione che si deve prendere in considerazione è quella di conservazione delle specie. Per comodità viene riportata qui sotto in una forma rielaborata rispetto a quella vista nel secondo capitolo (2.8):

$$\nabla \cdot (\rho \vec{u} \omega_k) = \nabla \cdot \left[\left(\rho \Gamma_{k,mix} + \frac{\mu_t}{Sc_{t,k}} \right) \nabla \omega_k \right] + \dot{\Omega}_k \quad (5.1)$$

In questa equazione la derivata in funzione del tempo viene tralasciata in quanto siamo sotto l'ipotesi di stato stazionario. Inoltre gli altri termini sono scritti in forma esplicita. Questa equazione viene risolta nel file *Yeqn.H* e il codice è riportato qui di seguito:

```
fvScalarMatrix YEqn
(
    mvConvection->fvmDiv(phi, Yi)
    - fvm::Sp(fvc::div(phi), Yi)
    - fvm::laplacian(turbulence->muEff(), Yi)
)
);
```

Le prime due righe sono gli operatori che risolvono la divergenza (**phi** è il prodotto di densità e velocità) mentre la terza riga risolve il laplaciano a destra dell'uguale. *OpenFOAM* utilizza il termine `muEff()`, che rappresenta la somma del contributo diffusivo e di quello turbolento contenuti all'interno delle parentesi tonde nell'equazione 5.1. Nello svolgimento delle simulazioni, cercando di capire il motivo di risultati non particolarmente positivi, si è scoperto che nel calcolo di questo termine viene utilizzato un numero di *Schmidt* pari a 1, quando invece il software *Fluent* usa 0.7, considerato anche in letteratura un valore migliore. Pertanto la terza riga è stata sostituita con questa

```
- fvm::laplacian(turbulence->mut()/0.70 + rho*D, Yi)
```

in cui D è la diffusività che deve essere precedentemente definita, Y_i la frazione massiva delle specie e gli altri termini sono facilmente intuibili. Come si può vedere in questa equazione manca il termine sorgente di reazione, il che significa che usando questo risolutore non è possibile far avvenire reazioni chimiche. Pertanto l'obiettivo è stato quello di aggiungere proprio questo termine. Esso viene modellato come visto nel paragrafo 3.2.2 e rielaborandone l'espressione possiamo ottenere che la generica velocità di reazione corrisponde al valore minimo tra queste due espressioni:

$$\bar{\dot{\Omega}}_r = \frac{1}{\nu_{F,r} M_{w,F}} A \bar{\rho} \frac{\varepsilon}{k} \min \left(\widetilde{\omega}_F, \frac{\widetilde{\omega}_O}{r_r} \right) \quad (5.2)$$

$$\bar{\dot{\Omega}}_r = \frac{1}{\nu_{F,r} M_{w,F}} A B \bar{\rho} \frac{\varepsilon}{k} \frac{\sum_P \widetilde{\omega}_P}{1 + r_r} \quad (5.3)$$

in cui $r_r = \frac{\nu_{O,r} M_{w,O}}{\nu_{F,r} M_{w,F}}$ e il pedice indica a quale reazione si riferisce.

Successivamente per calcolare la velocità di reazione di ogni specie è sufficiente applicare la seguente formula:

$$\bar{\dot{\Omega}}_{i,r} = \nu_{i,r} M_{w,i} \bar{\dot{\Omega}}_r \quad (5.4)$$

Per implementare tale termine sorgente si è deciso di creare una nuova classe adibita allo scopo, chiamata *EDmodel*, all'interno del solutore. E' interessante analizzare la porzione di codice in cui la velocità di reazione viene calcolata:

```
for (label i=0; i<NR; i++)
{
    sumYProducts[i] = 0.;
    for (label j=0; j<nub[i].size(); j++)
        sumYProducts[i] += Y[indexb[i][j]];
    rReactants[i] = A*rho*epsilon/k
        * min(Y[indexf[i][0]], Y[indexf[i][1]]/r[i] );
    rProducts[i] = A*B*rho*epsilon/k * sumYProducts[i]/(1.+r[i]);
    rr[i] = min(rReactant.difficils[i], rProducts[i])
        / (nuf[i][0]*MW[indexf[i][0]]); // [kmol/m3/s]
    for (int j=0; j<nuf[i].size(); j++)
```

```
    rrspecie[indexf[i][j]] -= rr[i]*MW[indexf[i][j]]*nuf[i][j];  
    for (int j=0; j<nub[i].size(); j++)  
        rrspecie[indexb[i][j]] += rr[i]*MW[indexb[i][j]]*nub[i][j];  
}
```

Il primo ciclo `for` sta ad indicare che le operazioni devono essere effettuate per tutte le reazioni (fino a un massimo di 3 reazioni, andando oltre il modello *ED* non da più risultati accettabili). Successivamente vengono calcolate le velocità secondo la formula vista in precedenza. `nuf` e `nub` sono rispettivamente i coefficienti stechiometrici dei reagenti e dei prodotti, mentre `indexf` e `indexb` sono gli indici delle specie all'interno del vettore che contiene tutte le specie, anche qui rispettivamente per i reagenti e per i prodotti. La funzione che calcola questa velocità è richiamata all'interno del file *Yeqn.H* tramite il seguente comando

```
ED.calcRate(Y, rho, k, epsilon, mesh);
```

e a questa funzione vengono passati i valori delle frazioni massive, di densità, di *k* e *epsilon* e la *mesh*. A questo punto si utilizza un'altra funzione della classe, chiamata `passRate`, che semplicemente restituisce il valore dell'elemento *i*-esimo del vettore `rrspecie` (dove *i* indica quindi la specie di cui si sta risolvendo l'equazione). Pertanto il codice presentato in precedenza si trasforma come segue:

```
fvScalarMatrix YEqn  
(  
    ( mvConvection->fvmDiv(phi, Yi)  
      - fvm::Sp(fvc::div(phi), Yi)  
      - fvm::laplacian(turbulence->mut())/0.70 + rho*D, Yi)  
      - ED.passRate(i)  
    )  
);
```

in cui a questo punto è contenuto il termine sorgente che tiene conto della reazione chimica. Resta da notare infine che per far avvenire l'ignizione con il modello *ED* è necessario imporre come condizione al contorno in ingresso o nell'interno della mesh almeno una piccolissima frazione di uno dei prodotti di reazione, in modo che la velocità calcolata rispetto ai prodotti non risulti sempre nulla e quindi sempre la minima tra le due, impedendo di fatto l'ignizione della fiamma.

5.2.2 Il set del caso

Una volta inserito nel codice il termine sorgente della reazione chimica, assume grande importanza il file contenuto nella cartella *constant* del caso e chiamato *thermophysical-Properties*: in questo file si devono innanzitutto dichiarare le specie inerti e inserire il percorso dei due files in formato *chemkin* attraverso i quali il programma legge la cinetica e la termodinamica. Nel file della cinetica vanno inseriti tutti gli elementi, tutte le specie e tutte le reazioni chimiche con i dati cinetici di ognuna (nonostante per il modello *ED* questi dati siano inutili). Ad esempio nel caso di una reazione di combustione di idrogeno, il file risulterebbe:

```
ELEMENTS
H O N
END
SPECIE
H2 O2 H2O N2
END
REACTIONS H2 + 0.5O2 => H2O          5.00E+8  0.0  15780.0
END
```

Nel file della termodinamica vanno inseriti i dati termodinamici delle specie presenti nel sistema appunto secondo lo standard del formato *chemkin*. Nel file *chemistryProperties*, anch'esso contenuto nella cartella *constant*, è possibile attivare o disattivare la chimica in modo da procedere se necessario con una simulazione a freddo.

Molto importante è a questo punto vedere l'impostazione delle condizioni al contorno: utilizzando questo modello, all'interno della cartella *0* del caso sono presenti diversi files, attraverso i quali si vanno a impostare le condizioni al contorno di temperatura, velocità, pressione, *k*, *epsilon* e di tutte le frazioni massive delle specie presenti. In ognuno dei files bisogna innanzitutto definire le unità di misura della grandezza considerata e definire il valore della grandezza nell'*internalfield*, cioè nell'interno della *mesh* all'inizio della simulazione, che di solito è uniforme e si definisce in questa maniera, ad esempio nel caso della velocità:

```
internalField  uniform (0 0 0);
```


A questo punto è necessario per ogni contorno stabilito in fase di creazione della *mesh* definire il valore della grandezza considerata. Prendendo sempre ad esempio la velocità e ponendo caso di considerare il contorno di ingresso del combustibile avremo:

```
boundaryField
{
    inletfuel
    {
        type fixedValue;
        value    uniform (0 0 296);
    }
}
```

Ovviamente andranno a questa aggiunte le condizioni per tutti gli altri contorni. I tipi di condizione che si possono inserire sono diversi, quelli utilizzati nelle nostre simulazioni sono stati:

- **fixedValue**: utilizzato per l'*inlet* e per le condizioni di parete ferma, implica l'imposizione di un valore fissato della grandezza.
- **inletOutlet**: questa condizione viene utilizzata in contorni di *outflow* da cui il fluido può uscire, ma può essere anche in grado di rientrare
- **zeroGradient**: è una condizione che sta a indicare gradiente nullo, ovvero il flusso della grandezza attraverso il contorno è nullo.

Altre possibili condizioni sono `fixedGradient`, `mixed`, `directionMixed`, `calculated`.

5.3 Modello *Steady Laminar Flamelets*

L'implementazione del modello *SLF* in *OpenFOAM* viene riassunta in maniera molto chiara in figura 5.2. Innanzitutto ci si appoggia a un codice esterno, chiamato *OpenSMOKE*, per la creazione delle libreria di *flamelets* (*look-up table*). Una volta creata la *look-up table* essa viene utilizzata nel nuovo solutore stazionario della fluidodinamica chiamato *rhoSimpleFlamelets*, che risolve le equazioni delle variabili primitive e si avvale inoltre di

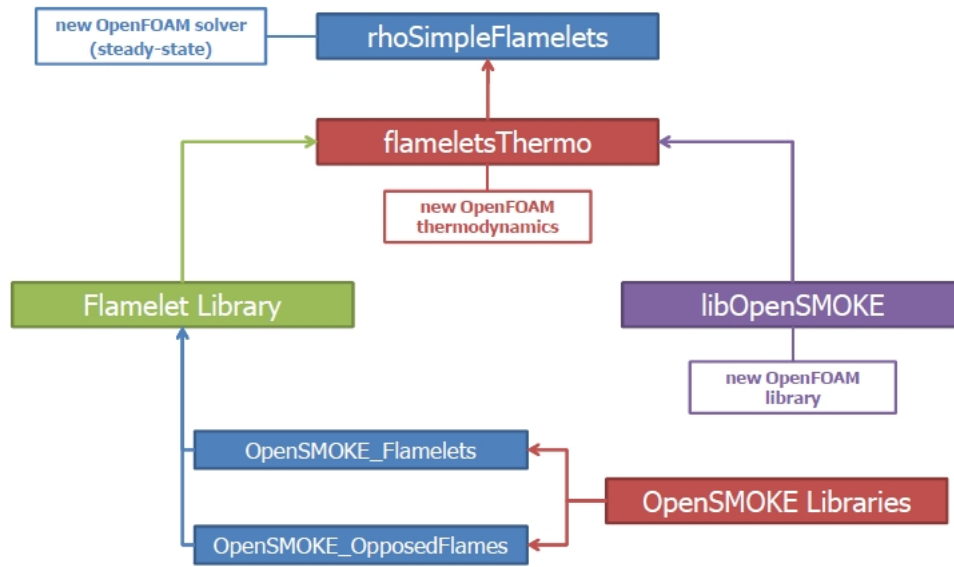


Figura 5.1: Schema del modello *flamelets* in *OpenFOAM*

due librerie costruite ad hoc: la libreria *flameletsThermo*, che gestisce la termodinamica al posto della libreria di default e la libreria *libOpenSMOKE* che gestisce il trasferimento di informazioni tra il solutore e la *look-up table*. Nel corso del paragrafo si cercherà di approfondire tutti questi aspetti.

5.3.1 La creazione della *look-up table*

Abbiamo visto come il modello *SLF* si basa sulla consultazione di una *look-up table* generata prima dello svolgimento della simulazione fluidodinamica, attraverso un software, *OpenSMOKE*, sviluppato dal dipartimento di Ingegneria Chimica del Politecnico di Milano. Questo software è stato utilizzato ma non è parte dello studio di questo lavoro di tesi. I modelli matematici attraverso cui si arriva alla generazione della libreria di *flamelets* sono comunque riportati in appendice A. Ciò che è importante far notare è il funzionamento della *look-up table*: per ogni valore di *scalar dissipation rate*, partendo da quello stechiometrico a quello in cui la fiamma si estingue, viene creata una tabella a doppia entrata con righe e colonne date dalla *mixture fraction* e dalla sua varianza. In questo modo i valori delle grandezze contenuti nella tabella possono essere consultati semplicemente a partire dai valori delle due variabili primitive *mixture fraction* (che richiede anche la conoscenza della varianza) e *scalar dissipation rate*, che sono calcolati nella simulazione *CFD*. Ovviamente le tabelle sono create utilizzando valori discreti delle variabili primitive, perciò è

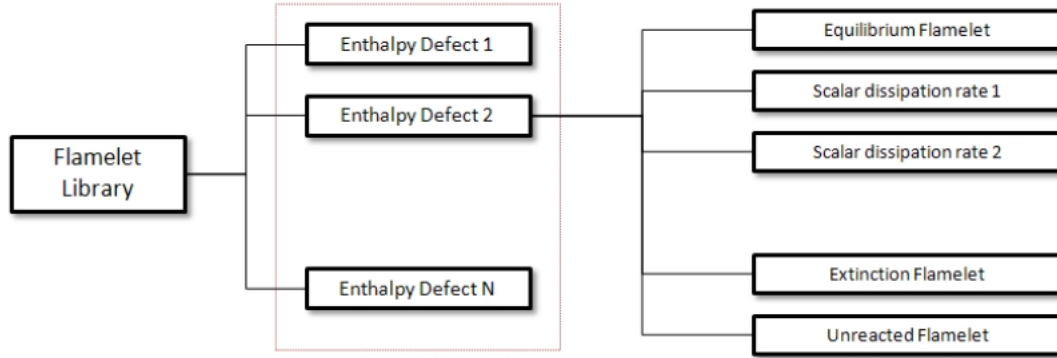


Figura 5.2: Struttura di una libreria di *flamelets* non adiabatica

chiaro che le grandezze da estrapolare saranno calcolate attraverso delle interpolazioni. Nel caso in cui la simulazione venga condotta in maniera non adiabatica, cioè tenendo conto dell'irraggiamento e quindi del rilascio di calore, la *look-up table* che si viene a creare assume una complessità maggiore. Viene infatti introdotto il difetto di entalpia (*enthalpy defect*), ossia la differenza tra l'entalpia attuale e l'entalpia che si avrebbe in una fiamma adiabatica. Per diversi valori di difetto di entalpia si crea un insieme di tabelle ognuna a diverso *scalar dissipation rate*. In pratica viene aggiunta una variabile primitiva attraverso cui il solutore va a consultare le tabelle. Lo schema generale della *look-up table* non adiabatica è mostrato in figura 5.2

5.3.2 Il solutore *rhoSimpleFlamelets*

Una volta creata la libreria attraverso il software *OpenSMOKE*, sulla simulazione numerica del codice *CFD* non peserà più anche la soluzione della chimica del sistema, ma solo quella della fluidodinamica, attraverso la risoluzione delle variabili primitive. Si devono risolvere perciò le due equazioni differenziali della *mixture fraction* e della sua varianza (3.32 e 3.35) e in aggiunta l'equazione per il calcolo del difetto di entalpia:

$$\bar{\rho} \frac{\partial \tilde{H}}{\partial t} + \bar{\rho} \tilde{u}_i \frac{\partial \tilde{H}}{\partial x_i} = \frac{\partial}{\partial x_i} \left(\Gamma_t \frac{\partial \tilde{H}}{\partial x_i} \right) \quad (5.5)$$

Per lo *scalar dissipation rate* non si trasporta alcuna equazione, ma lo si calcola algebricamente. Per questo il solutore, rispetto a quello standard visto nel capitolo precedente, non possiede più il file *Yeqn.H* che conteneva il codice per la risoluzione delle equazioni delle

specie, ma al suo posto il file *csiEqn.H* in cui vengono risolte le equazioni di *csi* (*mixture fraction*) e *csiv2* (varianza) in questa maniera:

```
fvScalarMatrix csiEqn
(
    (
        fvm::div(phi, csi)
        - fvm::Sp(fvc::div(phi), csi)
    )
);

fvScalarMatrix csiv2Eqn
(
    (
        fvm::div(phi, csiv2)
        - fvm::Sp(fvc::div(phi), csiv2)
        - fvm::laplacian(turbulence->mut()/sigmat, csiv2)
    )
);
```

Per ottenere una soluzione di primo tentativo più velocemente e con meno costo computazionale si può utilizzare al posto dell'equazione differenziale per il calcolo della varianza un'equazione algebrica, attivabile come vedremo in seguito nelle impostazioni del caso e che è qui riportata:

```
csiv2Cells = Cg*turbulence->mut()*magSqr(fvc::grad(csi))*turbulence->k() /
    (Cd*rho*(turbulence->epsilon()+epsilon_small));
```

k_small ed *epsilon_small* sono dei parametri che servono per evitare un errore di *floating point exception* nel momento in cui i denominatori dovessero risultare troppo vicini allo zero, ma devono essere impostati molto piccoli in modo da non modificare i risultati della simulazione.

Infine nello stesso file è calcolato come nella 3.44 anche lo *scalar dissipation rate* (*chi_st*):

```
chi_st=Cx*turbulence->epsilon()/(turbulence->k()+k_small)*csiv2;
```

Il file *hEqn.H* è invece adibito alla soluzione dell'equazione dell'entalpia, che nel caso la simulazione venga impostata in modalità non adiabatica, è:

```
Qrad = 4. * sigmaSB * as * (pow(thermo->T(),4) - pow(Tenv,4));

fvScalarMatrix hEqn
(
    fvm::div(phi, H)
    - fvm::Sp(fvc::div(phi), H)
    - fvm::laplacian(turbulence->mut()/sigmat, H)
    + Qrad
);
```

in cui il termine sorgente è calcolato attraverso il già citato modello di irraggiamento (cfr. paragrafo 2.5.5). Tramite il valore dell'entalpia viene calcolato il difetto di entalpia che si utilizza come parametro per accedere ai dati nella *look-up table*.

5.3.3 Le librerie *OpenSMOKE* e *flameletsThermo*

Come già accennato il solutore *rhoSimpleFlamelets* si avvale di due librerie:

Libreria *OpenSMOKE* La libreria *OpenSMOKE* è stata creata durante questo lavoro di tesi e contiene le classi che permettono al solutore di interagire con la *look-up table*. Infatti le tabelle di *flamelets* vengono raccolte in dei files binari, che devono essere interpretati. Ad ogni iterazione il solutore è in grado attraverso questa libreria di accedere alle tabelle, recuperare i dati di temperatura, densità e composizioni corrispondenti ai valori di *mixture fraction*, varianza, *scalar dissipation rate* e difetto di entalpia calcolati attraverso il codice *CFD*. A questo punto si procede iterativamente con questo metodo fino al raggiungimento della convergenza.

Libreria *flameletsThermo* Questa libreria è stata inserita all'interno del pacchetto che gestisce la termodinamica del sistema e sostanzialmente va a impostare un nuovo metodo con cui calcolare le grandezze termodinamiche: mentre con la libreria standard la temperatura viene ad ogni iterazione calcolata a partire dall'entalpia e anche la densità

viene calcolata attraverso una funzione appartenente alla classe *thermo*, utilizzando questa nuova libreria i valori di temperatura e densità, cioè quelli necessari alla risoluzione delle equazioni differenziali nello step successivo, non vengono calcolati ma semplicemente posti uguali a quelli estratti dalla *look-up table*.

5.3.4 Il set del caso

Il settaggio di un caso da risolvere attraverso il modello *SLF* è simile a quello del caso *ED* ma con alcune differenze sostanziali: innanzitutto all'interno della cartella *0* in cui impostare le condizioni al contorno non si trovano più le condizioni in ingresso per le specie (che vengono inserite in sede di creazione della *look-up table*) ma solo le condizioni in ingresso della *mixture fraction* e della sua varianza. La prima viene posta sempre uguale a uno nel contorno di ingresso del *fuel* e zero nel contorno di ingresso dell'aria, la seconda uguale a zero in entrambi i contorni. Passando alla cartella *constant* due files vanno presi in considerazione: il primo, *thermophysicalModels*, contiene una stringa in cui va modificata la libreria termodinamica da usare inserendo questa:

```
thermoType      pdfThermo<pureMixture<constTransport
                <pdfGasThermo<hConstThermo<pdfGasMixture>>>>>;
```

il secondo file è *flameletsProperties* ed è molto importante, perciò se ne riportano le parti salienti:

```
//Flamelet Library
libraryPath "constant/PDF-Library";
adiabaticMode off;
varianceEquation on;

5; // number of iterations to update the look-up table massFractionsUpdate
50; // number of iterations to update mass fractions species
"H2 OH H2O O2 "; // list of species to plot in paraview

//Scalar Dissipation Rate PDF
pdf "dirac";
sigma 1.31;
```

```
points 30;

// Equation constants
sigmat 0.85;
Cg 2.86;
Cd 2.00;
Cx 2.00;

k_small [0 2 -2 0 0 0 0] 1.e-3;
epsilon_small [0 2 -3 0 0 0 0] 1.e-3;
```

Nel primo blocco si può impostare la cartella in cui caricare la *look-up table*. E' possibile inoltre decidere se la simulazione deve essere svolta in modo adiabatico o non adiabatico e se si vuole risolvere l'equazione differenziale della varianza o solo il calcolo algebrico. Infine si può decidere ogni quante iterazioni la *look-up table* viene consultata, ogni quante iterazioni le frazioni massive vengono sovrascritte e quali sono le specie di cui si vuole visualizzare la frazione massiva. Nel secondo blocco si può decidere il tipo di *PDF* per lo *scalar dissipation rate* tra *Dirac* e *log normal* e nel secondo caso vanno impostati i parametri di questa distribuzione. Nel terzo blocco si possono impostare alcune costanti per la soluzione delle equazioni e i valori di `k_small` e `epsilon_small` di cui si è parlato in precedenza.

5.3.5 Schema cinetico

Utilizzando il modello *flamelets* abbiamo visto che la soluzione della cinetica del sistema viene disaccoppiata da quella della fluidodinamica. E' chiaro a questo punto che è possibile, a differenza del modello *ED*, utilizzare schemi cinetici anche molto complessi e dettagliati. Lo schema cinetico infatti viene utilizzato solo nella fase di costruzione delle librerie e una volta impostato il caso in *OpenFOAM* non serve più, ma i suoi benefici sono apprezzabili attraverso la maggiore accuratezza della *look-up table* creata. Nel nostro lavoro di tesi non è stato necessario utilizzare schemi molto dettagliati perchè il nostro obiettivo è stato soltanto quello di verificare il funzionamento del modello e perciò i combustibili utilizzati allo scopo sono stati molto semplici. Sono stati perciò usati degli schemi cinetici abbastanza contenuti: per le fiamme metano-idrogeno è stato usato uno schema *DRM19*, contenente 19 specie e 84 reazioni [19]. Per le fiamme a idrogeno e elio lo schema

utilizzato è stato il *PolimiH2_0810*, contenente 11 specie e 24 reazioni [43]. Per la fiamme *syngas* si è usato lo schema *PolimiH2/CO_0810*, contenente 14 specie e 27 reazioni [43].

Capitolo 6

Risultati

6.1 Introduzione

In questo capitolo verranno mostrati e analizzati i risultati ottenuti tramite le simulazioni con *OpenFOAM* su diverse tipologie di fiamme. La prima parte del capitolo è dedicata alla descrizione di tali fiamme, di cui si conoscono i dati sperimentali e che sono già state modellate in numerosi precedenti lavori. Per convalidare i risultati ottenuti con questo programma è stato necessario un confronto con i risultati ottenuti con *Fluent* e con i dati sperimentali. Nella prima fase, ovvero la convalida del modello *ED*, si sono svolte simulazioni adiabatiche, cioè senza considerare lo scambio termico dovuto all'irraggiamento e i confronti sono stati effettuati solo per due tipi di fiamma. Nella fase più corposa del lavoro, la convalida del modello *SLF*, è stato possibile svolgere anche simulazioni non adiabatiche che, tenendo conto del rilascio di calore per irraggiamento, portano a risultati più aderenti alla realtà dei dati sperimentali.

Bisogna aggiungere che l'obiettivo di questo lavoro è stato quello di convalidare i modelli implementati in *OpenFOAM* e perciò non si è cercato durante le simulazioni di ottenere risultati con un livello di precisione molto elevato, andando ad esempio a modificare i parametri o addirittura cambiando modello di turbolenza. Le simulazioni sono state svolte col modello *k-epsilon* standard (con coefficienti $C_1 = 1.44$ e $C_2 = 1.92$), in modo da ottenere un'adesione comunque soddisfacente ai dati sperimentali, cercando d'altra parte di convalidare i modelli sul maggior numero di casi possibili.

6.2 Le fiamme studiate

Le fiamme simulate in questo lavoro di tesi sono fiamme già modellate in precedenza e presentate in occasione dell' "International workshop on measurement and computation of turbulent nonpremixed flames". Questo workshop si svolge ogni due anni dal 1996 e ormai sul sito [47] sono presenti le caratteristiche e i dati sperimentali di numerose fiamme diffusive. Nel nostro lavoro sono state analizzate tre fiamme semplici, la prima alimentata a idrogeno e elio, la seconda a metano e idrogeno e la terza a *syngas*, una fiamma con bruciatore *bluff-body* e una fiamma con bruciatore pilota. Nel seguito del paragrafo le fiamme saranno descritte una per una.

6.2.1 Fiamma Sandia H_2/He

Questa fiamma semplice [2] consiste in un bruciatore cilindrico di diametro interno 3.75 mm e esterno 4.84 mm all'interno del quale viene iniettato il combustibile, che fuoriesce e che al contatto con l'aria esterna innesca la reazione. La temperatura di ingresso del *fuel* è di 295 K mentre quella dell'aria è di 294 K. In questa fiamma tre diverse situazioni sono state analizzate e riportate in tabella

nome	$H_2 : He$ (vol)	V_{IN} fuel
He0	100:0	296 m/s
He2	80:20	294 m/s
He4	60:40	256 m/s

Per tutte e tre le situazioni la velocità del *coflow* (aria) è stata supposta pari a 1 m/s. La *mesh* utilizzata in queste simulazioni, consiste in 20310 celle, il cannello di ingresso è lungo 70 mm, mentre la lunghezza dallo sbocco del *fuel* alla fine della griglia è di 862.5 mm. La larghezza della griglia è stata posta uguale a 150 mm. In figura 6.1 è stata riportata la *mesh* così come è visualizzata nel software *paraview* e anche un ingrandimento della parte in cui il *fuel* e l'aria entrano in contatto. Ovviamente si può notare come essa sia stata costruita in modo da avere una griglia più fitta nelle zone in cui avviene la reazione chimica e meno fitta nelle parti più lontane dal cuore della fiamma.

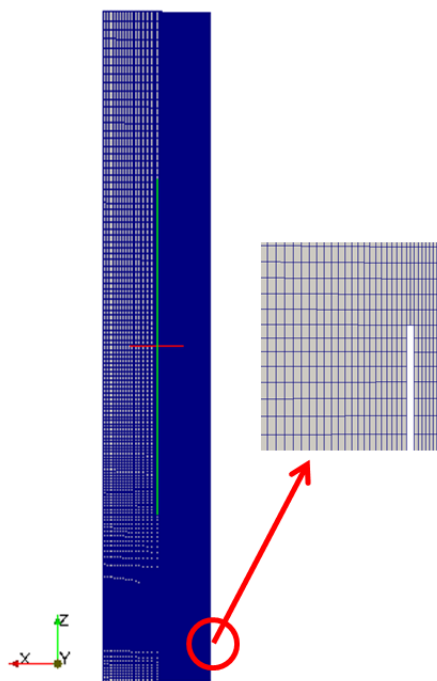


Figura 6.1: *mesh* utilizzata per la fiamma *Sandia H₂/He*

6.2.2 Fiamma Sandia $CH_4/H_2/N_2$

In questa seconda fiamma semplice [47] il bruciatore ha un diametro interno maggiore del precedente, pari a 8 mm, mentre quello esterno è 9.6 mm. La temperatura in ingresso è pari a 295 K per tutti i contorni. La composizione volumetrica del *fuel* in ingresso è la seguente: 22.1% CH_4 , 33.2% H_2 , 44.7% N_2 . Il *coflow* è ancora una volta semplicemente rappresentato da aria contenente lo 0.8% molare di acqua. Due fiamme sono state analizzate in questo caso. Nella prima (A) il *fuel* viene iniettato con una velocità di 42.2 m/s, nella seconda (B) la velocità è pari a 63.2 m/s. In entrambi i casi il *coflow* ha velocità pari a 0.3 m/s.

La *mesh* utilizzata per questo caso è molto simile alla precedente (fig. 6.2): le celle sono 21880, la lunghezza del cannello di ingresso è stata posta uguale a 120 mm. La lunghezza della parte in cui si sviluppa la fiamma è pari a 1040 mm, mentre la larghezza della griglia è pari a 200 mm. In figura 6.3 è mostrata una foto della fiamma reale.

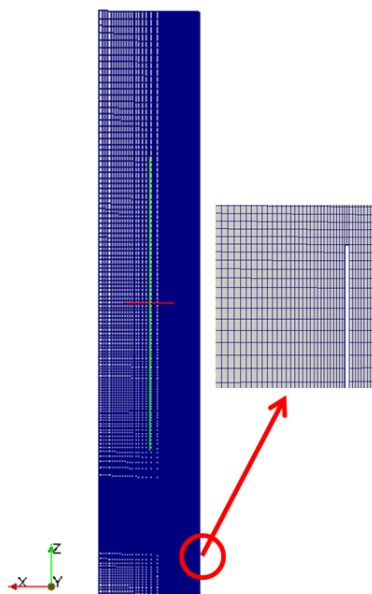


Figura 6.2: *mesh* utilizzata per la fiamma Sandia $CH_4/H_2/N_2$



Figura 6.3: Fotografia della fiamma Sandia $CH_4/H_2/N_2$ A [47]

6.2.3 Fiamma Sandia $CO/H_2/N_2$ A

Nell'ultima fiamma semplice che è stata modellata [4], il bruciatore ha un diametro interno di 4.58 mm e quello esterno è di 6.34 mm. La temperatura in ingresso è pari a 295 K per tutti i contorni. La composizione volumetrica del *fuel* in ingresso è: 40% CO , 30% H_2 , 30% N_2 . Il *coflow* è aria umida contenente l'1.2% molare di acqua. La *mesh* è ancora molto simile alle precedenti ed è visualizzata in figura 6.4: contiene 41447 celle e la lunghezza del cannello di ingresso è 91 mm. La lunghezza della parte successiva allo sbocco è pari a 458 mm e la larghezza è 150 mm. In figura 6.5 è visualizzata una fotografia della fiamma.

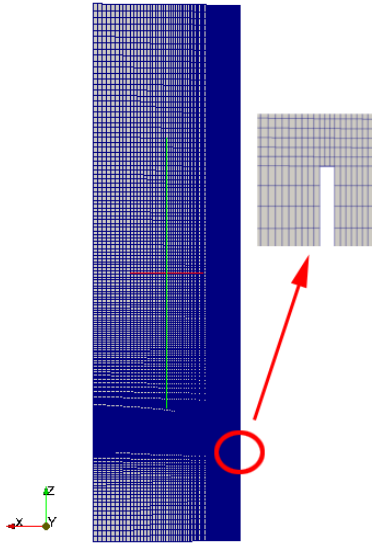


Figura 6.4: *mesh* utilizzata per la fiamma Sandia $CO/H_2/N_2$ A



Figura 6.5: Fotografia della fiamma Sandia $CO/H_2/N_2$ A [47]

6.2.4 Fiamma Sandia *Piloted* CH_4/Air

I bruciatori con pilota hanno una particolarità rispetto a quelli semplici. Infatti vi è un primo cannello che funziona da iniettore per il *fuel*, in questo caso una miscela formata dal 25% in volume di metano e dal 75% di aria e un secondo cannello in cui vengono iniettati dei fumi caldi di combustione, costituiti da una miscela di C_2H_2 , H_2 , aria, CO_2 e N_2 . Questo accorgimento viene utilizzato per consentire alla fiamma di rimanere ancorata al cannello. Infatti, essendo il metano particolarmente refrattario, potrebbero verificarsi accensioni in zone successive allo sbocco e questo provocherebbe la formazione di una fiamma non ancorata. Oltre all'uso di bruciatori pilota, per ovviare a questo problema, è possibile inserire in alimentazione una piccola quantità di idrogeno, meno refrattario, come fatto nelle fiamme precedenti, oppure utilizzare un altro accorgimento, che vedremo nel prossimo paragrafo. Il *coflow* è, come al solito, costituito da aria. Il *fuel* entra a una temperatura di 294 K, l'aria a 291 K, mentre la miscela di fumi ha una temperatura di 1880 K. Il bruciatore ha un diametro interno di 7.2 mm e uno esterno di 7.7 mm. Il pilota ha diametro interno 18.2 mm e esterno 18.9 mm. La conformazione studiata nel nostro lavoro (D) [3] vede una velocità del *fuel* pari a 49.6 m/s e una velocità dei fumi nel pilota pari a 11.4 m/s. Il *coflow* ha velocità 0.9 m/s. Nella figura 6.6 è visualizzabile la *mesh*

e nell'ingrandimento si può vedere come a differenza dei casi precedenti i canali di *inlet* siano due. In figura 6.7 è ancora presente una foto della fiamma in questione.

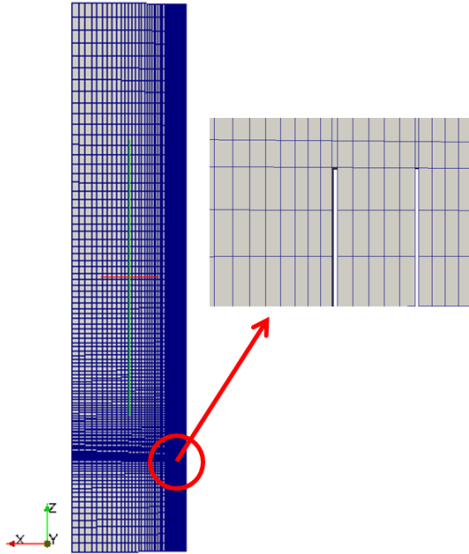


Figura 6.6: *mesh* utilizzata per la fiamma *Sandia Piloted CH₄/Air*



Figura 6.7: Fotografia della fiamma *Sandia Piloted CH₄/Air* [47]

6.2.5 Fiamma Sandia *Bluff-Body*

Anche la fiamma di un bruciatore *bluff-body* ha una particolarità rispetto a quelle semplici. In questo caso infatti, per favorire l'ancoraggio della fiamma, tra il cannello di imbocco del combustibile e la zona dove è presente l'aria si pone un cilindro di ceramica di larghezza definita chiamato appunto *bluff-body*, che serve a creare una zona di ricircolazione tale per cui siano presenti dei gas caldi che permettano alla fiamma di restare ancorata. Lo schema di un bruciatore *bluff-body* è visibile in figura 6.8. Nel nostro caso [1] il bruciatore ha un cannello centrale dal diametro di 3.6 mm, mentre il *bluff-body* ha un diametro di 50 mm. La composizione del *fuel* in ingresso è 50% in volume di metano e 50% di idrogeno. La temperatura di ingresso sia per il combustibile che per l'aria è 295 K. Le velocità in ingresso per la fiamma studiata (*HM1*) sono 118 m/s per il combustibile e 40 m/s per l'aria. Nelle figure 6.9 e 6.10 sono presentate la *mesh* utilizzata e la fotografia della fiamma. Si vede chiaramente come il *bluff-body* venga semplicemente modellato come uno spazio vuoto tra l'ingresso del combustibile e l'ingresso dell'aria. La presenza di questo

6. RISULTATI

accorgimento ha causato diversi problemi nello svolgimento della simulazione, che saranno esaminati durante l'analisi dei risultati.

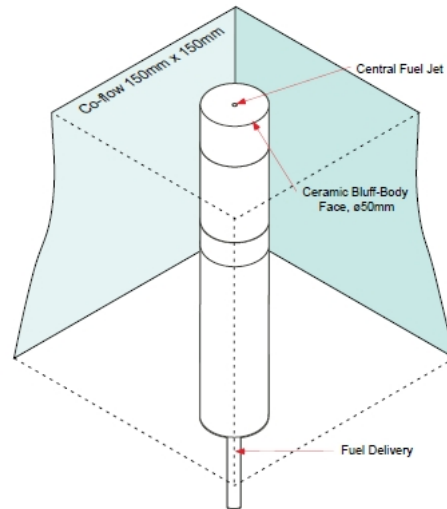


Figura 6.8: Schema di un bruciatore *bluff-body*

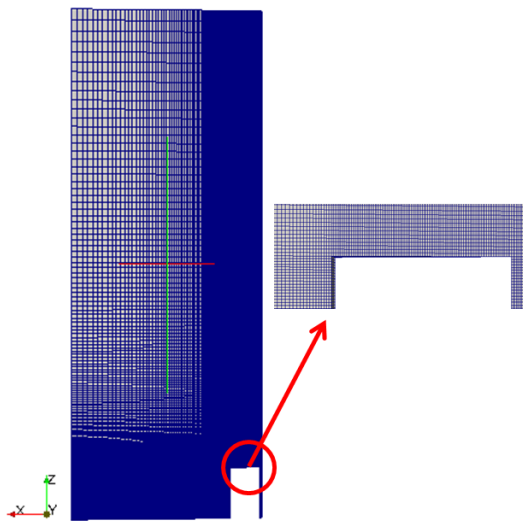


Figura 6.9: *mesh* utilizzata per la fiamma Sandia Bluff-Body

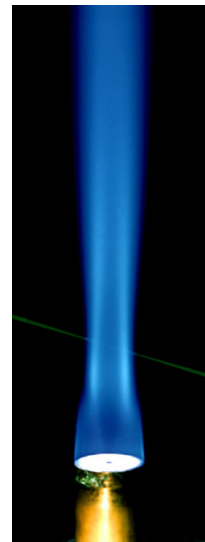


Figura 6.10: Fotografia della fiamma Sandia Bluff-Body [47]

6.3 Convalida modello *Eddy Dissipation*

Il modello *ED* è stato convalidato in questo lavoro di tesi confrontando i risultati ottenuti attraverso il software *OpenFOAM* con quelli ottenuti dal software *Fluent* per due delle fiamme semplici descritte in precedenza: la fiamma *Sandia H₂/He0*, cioè senza contenuto di elio in alimentazione e la fiamma *Sandia CH₄/H₂/N₂ A*. Il motivo per cui sono state modellate solo queste due fiamme è molto semplice: attraverso il modello *ED*, in accordo con quanto visto nel terzo capitolo, non si ottengono generalmente soluzioni precise che rispecchino i dati sperimentali, a causa della sua grande semplicità. Va detto inoltre che nel nostro lavoro, nonostante alcuni tentativi, non è stato possibile implementare in *OpenFOAM* lo stesso modello di irraggiamento di *Fluent*, per alcune difficoltà nel calcolo del coefficiente di assorbimento. Pertanto i confronti con i dati sperimentali risultano solo indicativi e in questa prima fase è sicuramente più utile verificare la corrispondenza tra i modelli dei due differenti software. Sono state analizzate solo due fiamme proprio per l'imprecisione del modello, in quanto si è ritenuto opportuno dare più spazio alla convalida del modello *SLF*, con cui si possono ottenere risultati più aderenti alla realtà. Durante la convalida dei modelli, prima di simulare la combustione vera e propria, sono state effettuate, per verificare il buon funzionamento del solutore di partenza, alcune simulazioni a freddo in cui le specie diffondono all'interno della *mesh* senza formazione di alcuna fiamma. I risultati ottenuti da tali simulazioni sono molto buoni, in quanto le curve calcolate con i due programmi coincidono quasi perfettamente (in fig. 6.11 possiamo vedere a sinistra il campo di velocità lungo l'asse per la fiamma *He0* a freddo).

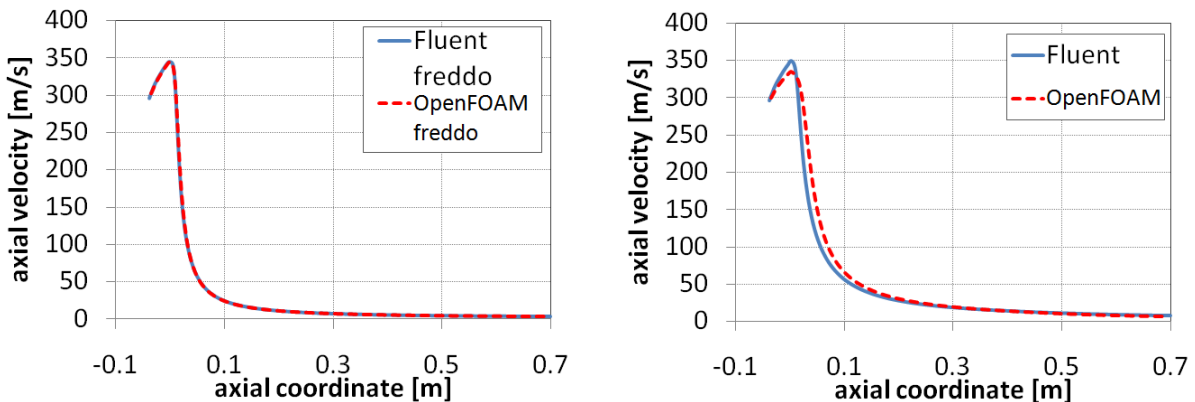


Figura 6.11: Profili di velocità lungo l'asse di simmetria per la fiamma *H₂/He0* a freddo (a sinistra) e con reazione chimica (a destra)

Introducendo la reazione chimica, cioè il termine reattivo nell'equazione delle specie, si ottiene un risultato comunque soddisfacente, ma che non vede le due curve sovrapporsi (fig. 6.11 a destra). La piccola differenza sulla velocità è indice come vedremo di una differenza anche più marcata su tutte le altre grandezze calcolate e il motivo di questa differenza va ricercato probabilmente nei differenti schemi numerici con i quali vengono risolte le equazioni differenziali. Infatti i due *software* utilizzano schemi differenti e non è stato possibile farli coincidere. Inoltre il termine sorgente di reazione aumenta la complessità dell'equazione di trasporto delle specie chimiche e pertanto le differenze fra gli schemi numerici o comunque le modalità di risoluzione delle equazioni differenziali vengono messe in risalto da questa complessità. Dalle figure appena viste si nota un'altra cosa significativa: osservando la parte del grafico in cui la coordinata assiale ha un valore negativo, cioè la zona del cannello di ingresso, la velocità aumenta lungo l'asse di simmetria invece di diminuire, raggiungendo un massimo in corrispondenza dello sbocco e abbassandosi solo successivamente. Questo comportamento si spiega in questo modo: entrando a velocità costante il flusso del combustibile viene rallentato in vicinanza delle pareti e ciò provoca un aumento della velocità nei punti in cui il fluido è più distante dalle pareti (e quindi per esempio in corrispondenza dell'asse).

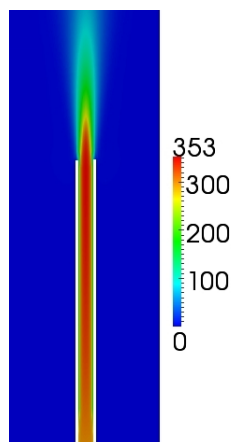


Figura 6.12: ingrandimento della mappa della velocità assiale nel cannello di ingresso del combustibile per la fiamma $H_2/He0$

A conferma di quanto detto vale la pena osservare l'ingrandimento del cannello di ingresso del combustibile in cui viene visualizzata la mappa di velocità tramite il software *paraview* (fig. 6.12). Si nota che il colore rosso acceso, a cui corrisponde il picco di velocità di 353 m/s, a fronte dei 296 in ingresso, viene raggiunto nel centro del cannello, mentre sulle

pareti la velocità è più bassa, intorno ai 200 m/s.
 Passiamo ora all'analisi dei risultati veri e propri.

6.3.1 Fiamma Sandia H_2/HeO

In figura 6.13 sono visualizzate le mappe di temperatura calcolate con OpenFOAM e Fluent. Esse sono molto importanti per avere un'idea indicativa della convergenza della simulazione e un'idea della forma della fiamma.

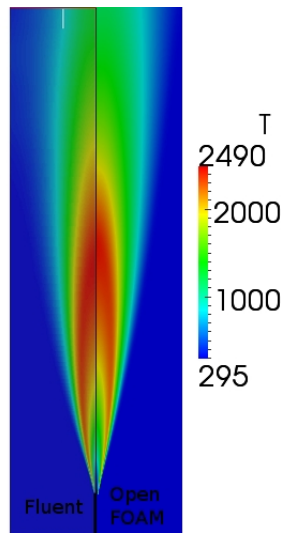


Figura 6.13: mappe di temperatura della fiamma H_2/HeO modellata con *OpenFOAM* e *Fluent*

Già analizzando queste due mappe si può vedere come i risultati siano sostanzialmente molto simili, ma anche come la fiamma modellata con *Fluent* risulti un poco più allungata. Di seguito (fig. 6.14) vengono riportati i profili assiali di velocità, temperatura e frazioni massive di idrogeno e acqua, visualizzati per entrambi i programmi e messi anche a confronto con i dati sperimentali ove disponibili. Da questi grafici si vede chiaramente come i profili assiali siano accettabili, in quanto le curve dei modelli risultano quasi coincidenti. Il motivo per cui probabilmente l'accordo non è perfetto è già stato messo in luce nel paragrafo precedente. La curva di temperatura di *OpenFOAM* risulta possedere un picco di temperatura inferiore e questo picco è spostato a destra lungo l'asse. Nella parte finale però è la fiamma di *Fluent* che raffreddandosi più lentamente risulta spostata verso destra, confermando quanto visto indicativamente nella mappa. La posizione del picco rispetto a quello dei dati sperimentali sembra comunque essere un punto a favore di

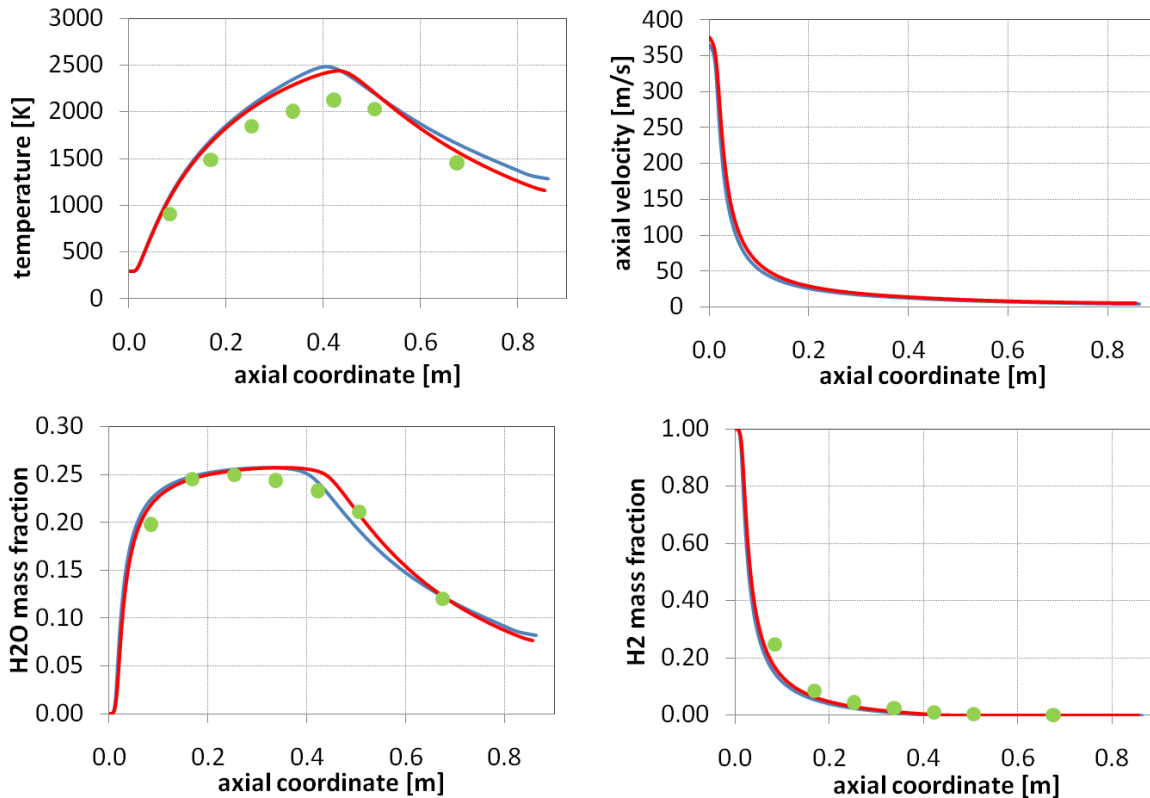


Figura 6.14: Fiamma $H_2/He0$. Profili assiali di temperatura, velocità e composizione. Confronto tra modelli (linea azzurra *Fluent*, linea rossa *OpenFOAM*) e dati sperimentali (punti in verde) [2]

OpenFOAM. Vale la pena anche ricordare il motivo per cui i dati sperimentali non sono in accordo perfetto coi modelli, in particolare per quanto riguarda la temperatura. Il modello *ED* infatti, per sua natura, tende a sovrastimare il picco di temperatura e il fatto di non considerare il rilascio di calore dovuto all'irraggiamento nelle simulazioni adiabatiche amplifica la differenza, tanto che il picco risulta essere più alto di oltre 300 gradi. Oltre ai profili sull'asse di simmetria, per ogni fiamma modellata sono stati visualizzati anche i profili radiali calcolati lungo alcune traverse. Tutti questi profili sono stati inseriti in appendice B e nel testo ne verranno inseriti alcuni particolarmente significativi o che ben rispecchiano il funzionamento del modello. Ad esempio in questo caso sono riportati in figura 6.15 e 6.16 i profili radiali di temperatura e frazione massiva di H_2O lungo due traverse, indicativi anche di tutti gli altri. Come si può notare, la bontà dell'accordo coi dati sperimentali diminuisce rispetto ai profili assiali, mentre l'accordo tra i due programmi resta sostanzialmente buono.

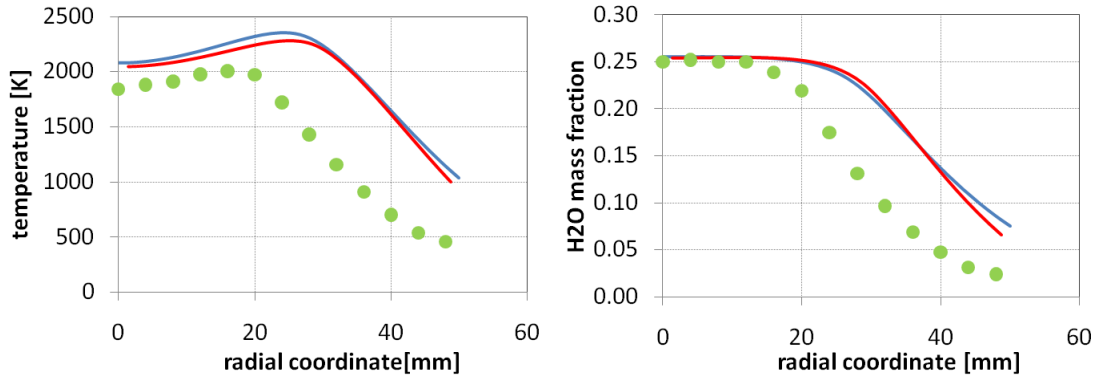


Figura 6.15: Fiamma $H_2/He0$. Profili radiali di temperatura e frazione massiva di H_2O calcolati a 253 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent*, linea rossa *OpenFOAM*) e dati sperimentali (punti in verde) [2]

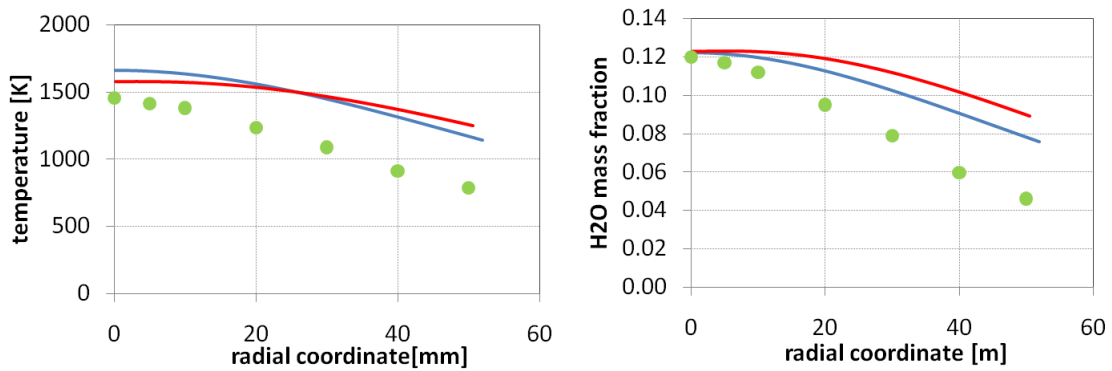


Figura 6.16: Fiamma $H_2/He0$. Profili radiali di temperatura e frazione massiva di H_2O calcolati a 675 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent*, linea rossa *OpenFOAM*) e dati sperimentali (punti in verde) [2]

6.3.2 Fiamma Sandia $CH_4/H_2/N_2$ A

Anche di questa fiamma sono disponibili i campi di temperatura calcolati con i due diversi software (fig. 6.17).

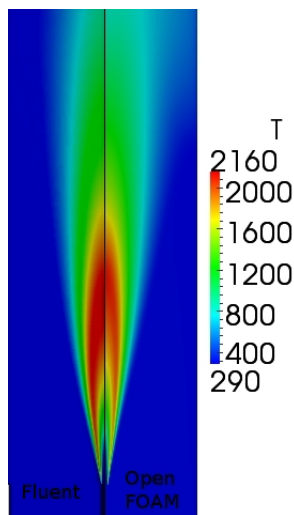


Figura 6.17: mappe di temperatura della fiamma $CH_4/H_2/N_2$ A modellata con *OpenFOAM* e *Fluent*

In questo caso la fiamma di *Fluent* sembra essere meno allungata rispetto alla precedente, ma si notano differenze abbastanza marcate nella parte finale del dominio di calcolo. Sono disponibili i profili della grandezze valutate precedentemente con l'aggiunta ovviamente di metano e anidride carbonica. In figura 6.18 si possono osservare le curve calcolate lungo l'asse di simmetria. Tali profili rispecchiano quanto già detto, cioè un buon riscontro tra i risultati calcolati con *OpenFOAM* e quelli calcolati con *Fluent*, anche se il picco di temperatura risulta sempre maggiore nella fiamma modellata con il secondo programma. Anche in questo caso è utile inserire alcuni profili radiali per mostrare l'accordo tra i modelli (fig. 6.19 e 6.20)

6. RISULTATI

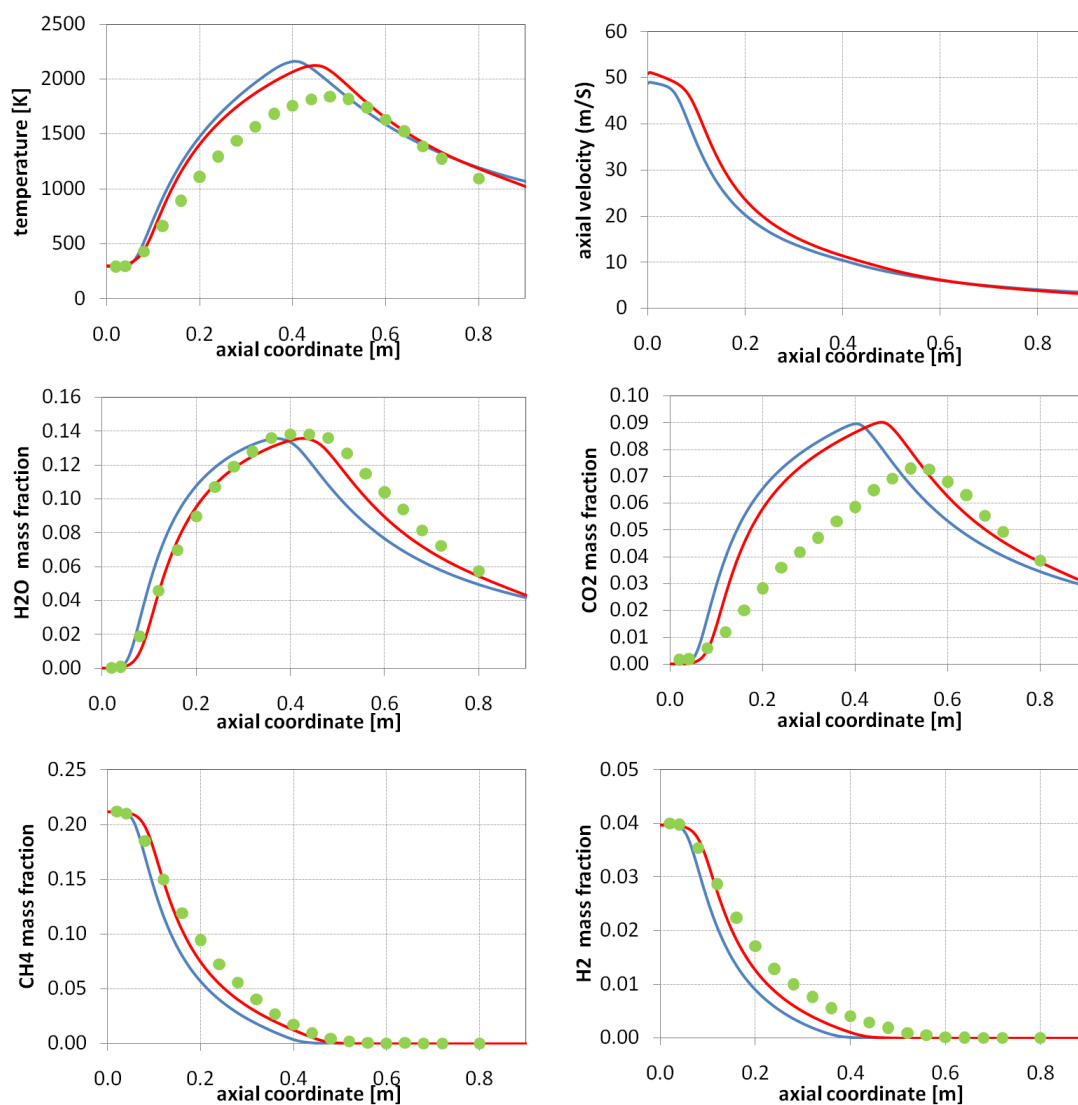


Figura 6.18: Fiamma $CH_4/H_2/N_2$ A. Profili assiali di temperatura, velocità e composizione. Confronto tra modelli (linea azzurra *Fluent*, linea rossa *OpenFOAM*) e dati sperimentali (punti in verde) [47]

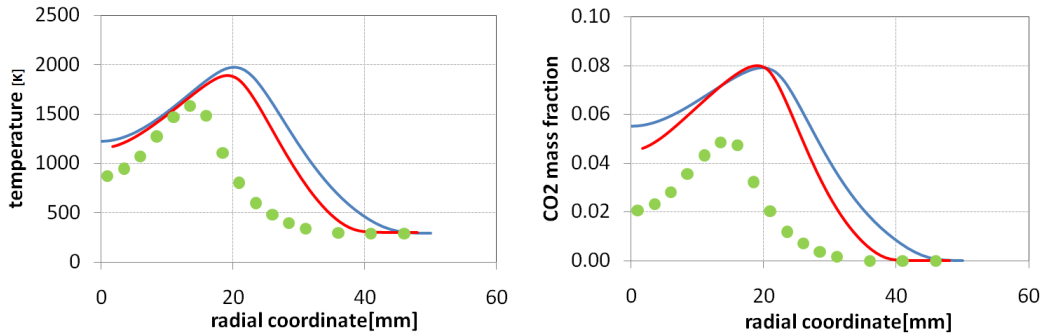


Figura 6.19: Fiamma $CH_4/H_2/N_2$ A. Profili radiali di temperatura e frazione massiva di CO_2 calcolati a 160 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent*, linea rossa *OpenFOAM*) e dati sperimentali (punti in verde) [47]

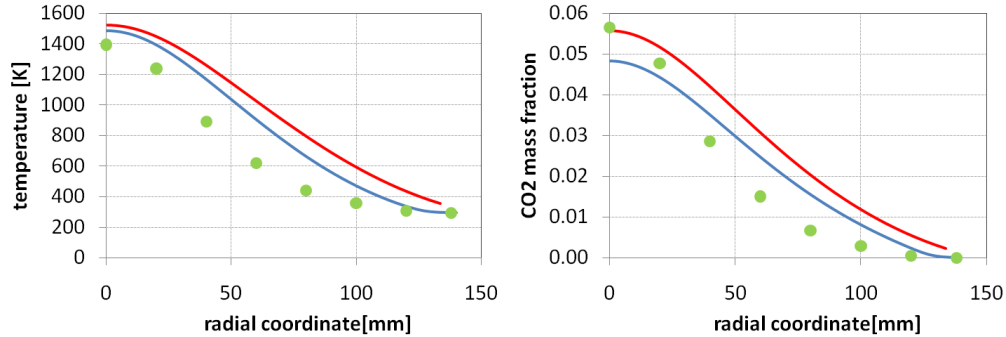


Figura 6.20: Fiamma $CH_4/H_2/N_2$ A. Profili radiali di temperatura e frazione massiva di CO_2 calcolati a 640 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent*, linea rossa *OpenFOAM*) e dati sperimentali (punti in verde) [47]

6.4 Convalida modello *Steady Laminar Flamelets*

Il modello *SLF*, potendosi avvalere di schemi cinetici dettagliati tramite la creazione delle librerie di *flamelets* esterna al codice *CFD*, è stato convalidato attraverso il confronto con *Fluent* ed in particolare attraverso i dati sperimentali disponibili per tutte le fiamme descritte in precedenza. Utilizzando *OpenFOAM* è stato possibile effettuare due tipi di simulazione: adiabatica e non adiabatica. La differenza sta nel fatto che, attivando la simulazione non adiabatica, il difetto di entalpia (cfr. 5.3.1) viene utilizzato come ulteriore parametro per la generazione e la consultazione delle *look-up tables* e il termine sorgente

dell'equazione che tiene conto dell'irraggiamento viene modellato in maniera molto semplice come indicato nel paragrafo 2.5.5. Il software *Fluent* utilizza un approccio diverso per la combustione non adiabatica, non del tutto rigoroso: invece di calcolare il difetto di entalpia e consultare le diverse librerie, utilizza sempre e solo la libreria relativa alla combustione adiabatica e durante lo svolgimento del codice *CFD* corregge la temperatura utilizzando dei modelli semiempirici disponibili nel programma [21]. Il problema di questo approccio, come verrà approfondito in appendice A, sta nel fatto che le frazioni delle specie non vengono modificate ad ogni iterazione per via del contributo radiativo, ma solo la temperatura. Si è ritenuto perciò opportuno, visto l'obiettivo di convalidare i risultati ottenuti con *OpenFOAM*, procedere nello stesso tempo a un confronto con *Fluent* sulla base della simulazione adiabatica e al confronto con i dati sperimentali del modello non adiabatico. In questo modo si potranno apprezzare anche le differenze tra i due modelli in *OpenFOAM* per capire dove l'irraggiamento assume un ruolo significativo. Come per il modello *ED*, per tutte le fiamme studiate verranno di seguito presentati i profili assiali di diverse grandezze, indicativi della bontà dei risultati ottenuti e alcuni profili radiali utili per commenti aggiuntivi. Tutti gli altri profili radiali calcolati sono disponibili in appendice B. Nei grafici bisogna porre attenzione al fatto che in alcuni casi le curve della simulazione adiabatica e non adiabatica potranno risultare sovrapposte e delle due si vedrà solo la più scura. Da notare infine che oltre alle grandezze già viste nel caso della convalida del modello *ED*, nei grafici per questo modello verranno visualizzate anche la *mixture fraction* e la sua varianza, che stanno alla base del funzionamento del modello stesso (cfr. 3.3.6). Per tutte le fiamme studiate verrà riportata anche la mappa di temperatura delle simulazioni adiabatica e non adiabatica di *OpenFOAM* visualizzate attraverso il software *paraview*.

6.4.1 Fiamma Sandia H_2/HeO

Dall'analisi dei primi risultati (fig 6.22) si può notare innanzitutto come rispetto al modello *ED* l'*SLF* rappresenti molto più fedelmente i valori dei dati sperimentali. Infatti tutti o quasi gli andamenti delle grandezze vengono rispecchiate nei modelli, in particolare nel modello non adiabatico di *OpenFOAM*, che riesce a modificare in maniera apprezzabile il campo di temperatura, come ci si aspettava, portando a un risultato decisamente migliore rispetto alla simulazione adiabatica.

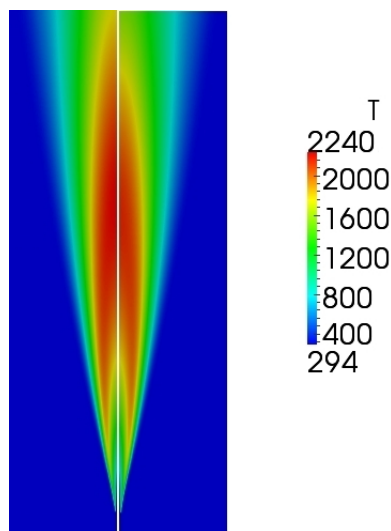


Figura 6.21: Mappe di temperatura per la fiamma $H_2/He0$. A sinistra è visualizzata la simulazione adiabatica, a destra quella non adiabatica

Unica stranezza da notare è nel grafico della frazione massiva di OH , in cui i dati sperimentali sembrano venire rappresentati in maniera migliore dalla curva adiabatica. Questo potrebbe essere dovuto a diversi fattori, tra cui ad esempio il dettaglio dello schema cinetico o la semplicità del modello di irraggiamento. Comunque, visto l'ordine di grandezza della frazione massiva del radicale OH , l'errore del modello non adiabatico non è macroscopico e alla fine è limitato ai due punti sperimentali che rappresentano il picco di concentrazione di OH . Questo viene messo in luce dai profili radiali di OH visibili in figura 6.23. In sostanza, i dati del primo grafico vengono presi su una traversa in prossimità del picco e si nota chiaramente come ancora la curva adiabatica sia migliore. Diverso è il caso della traversa successiva, in cui i ruoli si invertono ed è il modello non adiabatico a comportarsi meglio.

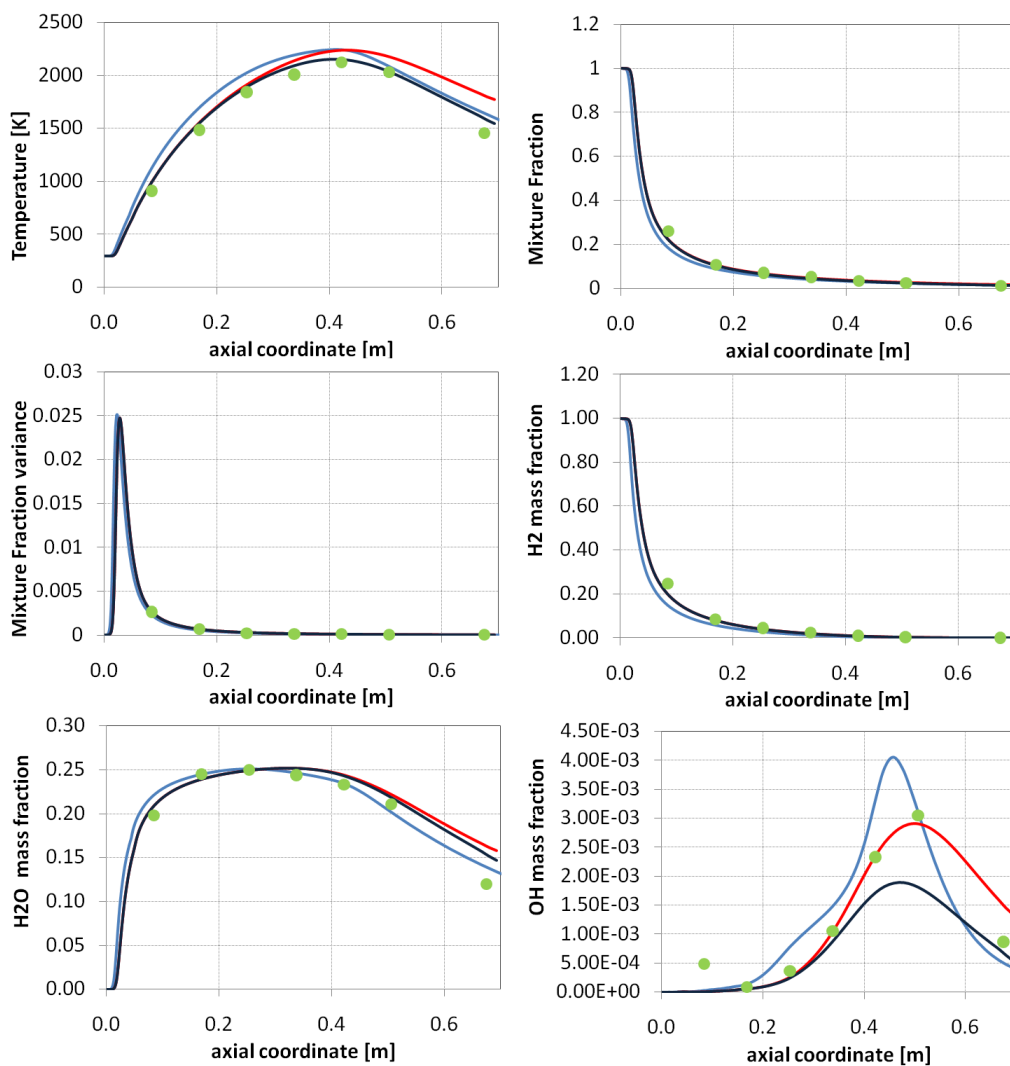


Figura 6.22: Fiamma $H_2/He0$. Profili assiali di temperatura, composizione, *mixture fraction* e varianza. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [2]

Un altro aspetto da notare è che nei profili radiali presi a distanza notevole dallo sbocco i risultati non sono più molto brillanti, come messo in luce dalla figura 6.24 in cui sono visualizzati a titolo indicativo i risultati per la temperatura e la *mixture fraction*.

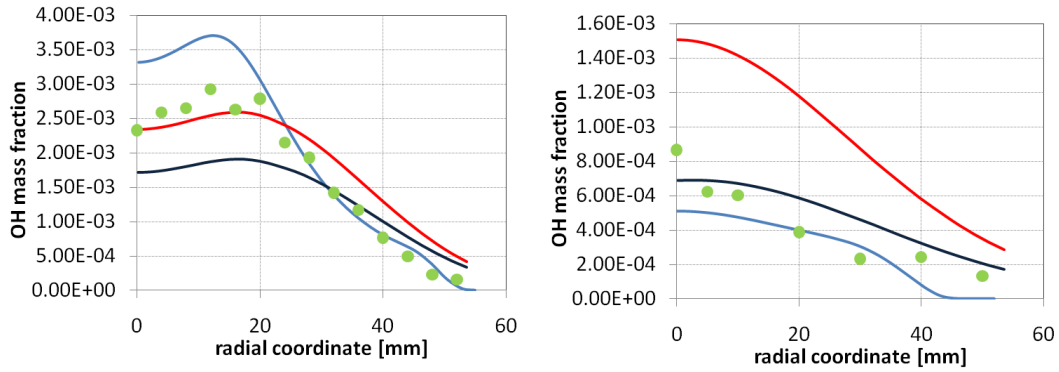


Figura 6.23: Fiamma $H_2/He0$. Profili radiali delle frazioni massive di OH calcolati a 422 mm (a sinistra) e 675 mm (a destra) dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [2]

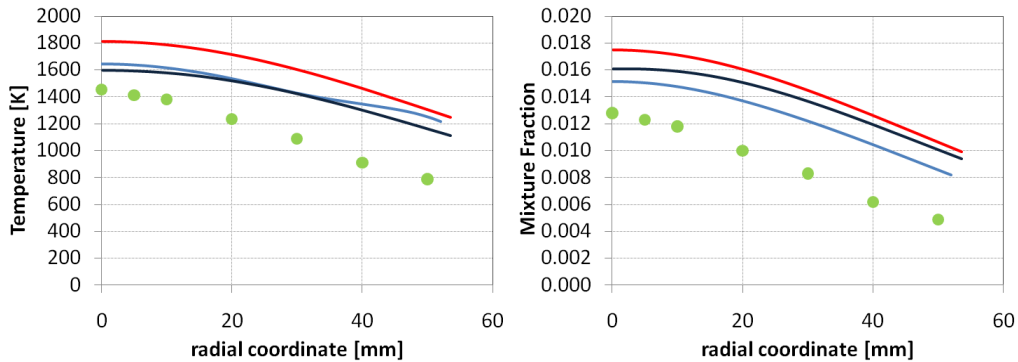


Figura 6.24: Fiamma $H_2/He0$. Profili radiali di temperatura e *mixture fraction* calcolati a 675 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [2]

Questo tipo di comportamento, come vedremo anche in seguito, è stato riscontrato frequentemente in questo lavoro di tesi. Sostanzialmente già nei grafici assiali si notano delle differenze coi dati sperimentali quando ci si avvicina alla fine del dominio di calcolo. La motivazione si trova proprio nel fatto che, a differenza di una fiamma reale, dove lo spazio è inconfinato, in una simulazione bisogna necessariamente dare alla *mesh* una dimensione ben definita e inserire una condizione al contorno in uscita che evidentemente rappresenta solo un'approssimazione della realtà fisica. A conferma di quanto detto è da notare che il problema si presenta allo stesso modo anche nelle curve modellate con il software *Fluent*. È importante valutare più in generale il confronto con lo stesso *Fluent*: le curve adiabatiche calcolate dai due programmi sono sostanzialmente simili in molti casi. Rispetto al

confronto con il modello *ED* sembra scomparire la differenza nei picchi di temperatura. Rimane però il fatto che in *Fluent* la fiamma è anticipata, ma i dati sperimentali in questo caso sembrano dare ragione al modello di *OpenFOAM*, il cui picco è posizionato praticamente allo stesso valore assiale di quello dei dati sperimentali. Anche analizzando gli altri profili radiali in appendice si nota comunque un buon accordo tra i due modelli e in molti casi *OpenFOAM* sembra rappresentare meglio i dati sperimentali. Resta infine da notare una particolarità riguardante la varianza della *mixture fraction*: questa grandezza tende a essere sovrastimata nella prima parte del dominio di calcolo e questo comportamento si è verificato per tutte le fiamme studiate. Questo però è un problema ben conosciuto del modello, legato alla soluzione dell'equazione di conservazione di questa variabile per fiamme a getto turbolento assialsimmetriche.

6.4.2 Fiamma Sandia $H_2/He2$

I profili assiali (fig. 6.26) mettono sostanzialmente in luce quanto detto per la fiamma precedente. Vediamo anche qui come spostandosi lungo l'asse i risultati peggiorino.

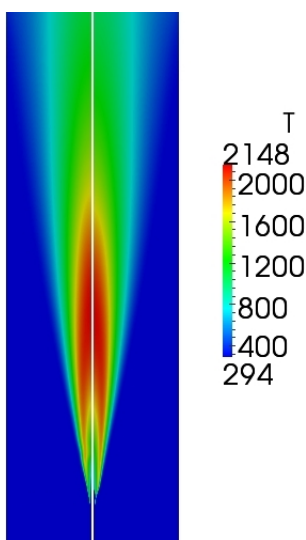


Figura 6.25: Mappe di temperatura per la fiamma $H_2/He2$. A sinistra è visualizzata la simulazione adiabatica, a destra quella non adiabatica

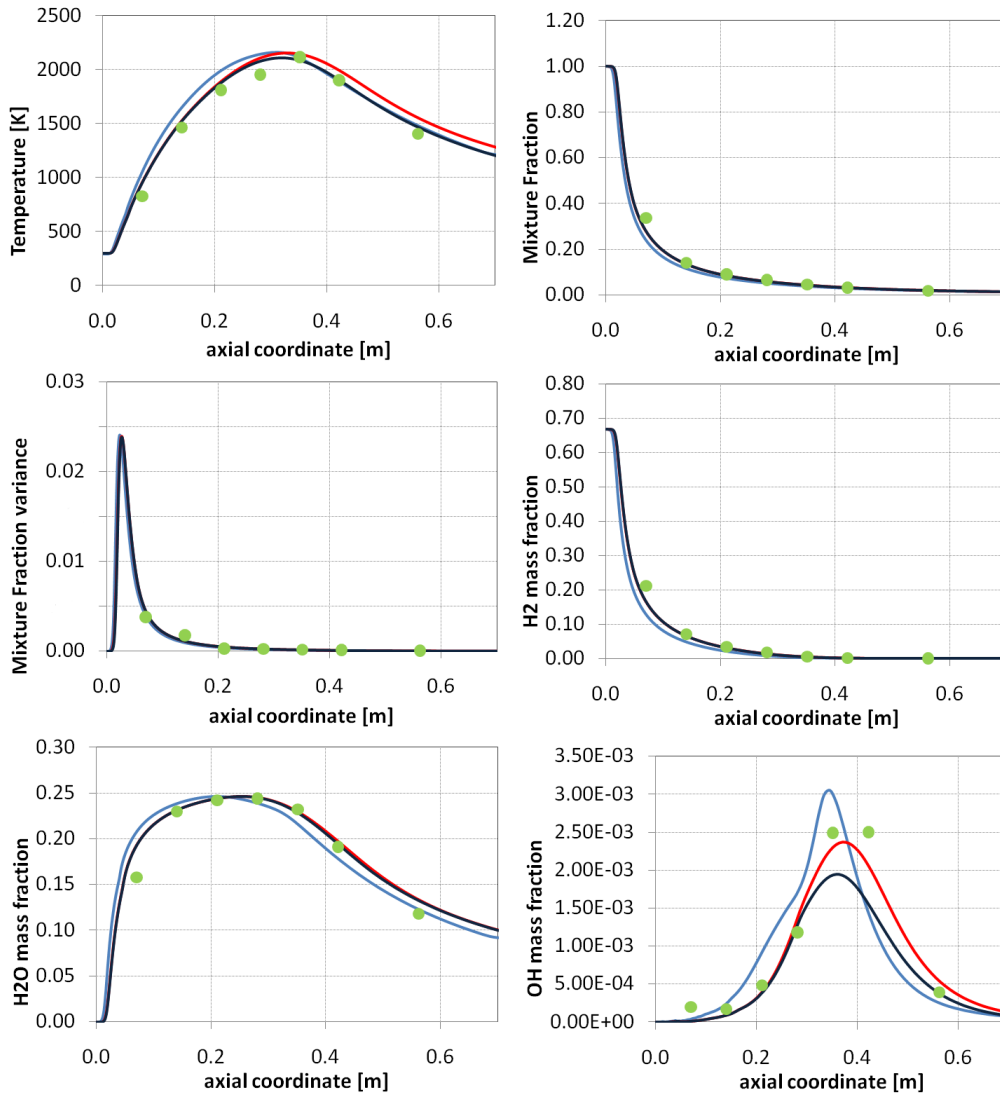


Figura 6.26: Fiamma H_2/He_2 . Profili assiali di temperatura, composizione, *mixture fraction* e varianza. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [2]

Questo si vede chiaramente anche andando ad analizzare la traversa posta a 422 mm e quella posta a 675 mm (fig. 6.27 e 6.28): si nota come l'andamento qualitativo della curva sia simile in entrambi i casi a quello dei dati sperimentali, ma il valore di partenza diverso rende il secondo grafico peggiore. Si è notato in alcuni casi (e si può verificare dai grafici in appendice) come sulla traversa più lontana alcune grandezze venissero addirittura rappresentate meglio dal modello *ED* piuttosto che dal modello *SLF*. E' però molto probabilmente casuale che in quel particolare punto dell'asse il modello *ED* assuma un valore

più simile al dato sperimentale e di conseguenza anche il grafico sulla traversa rispecchi questo dato.

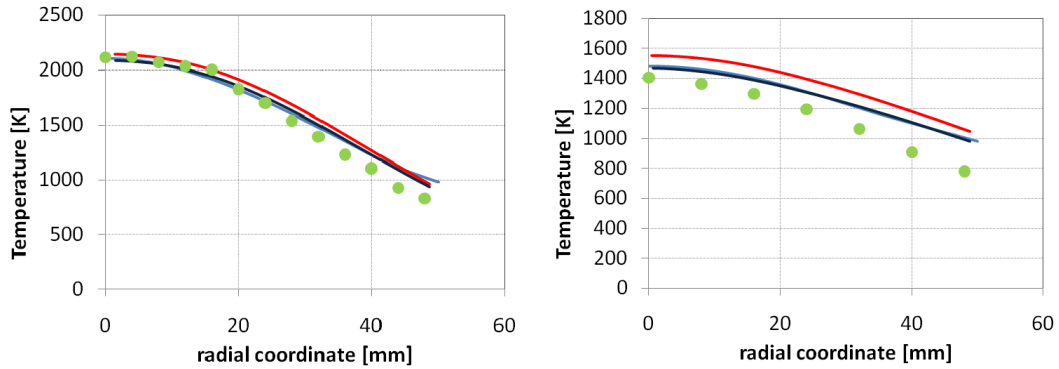


Figura 6.27: Fiamma $H_2/He2$. Profili radiali di temperatura calcolati a 352 mm (a sinistra) e 562.5 mm (a destra) dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [2]

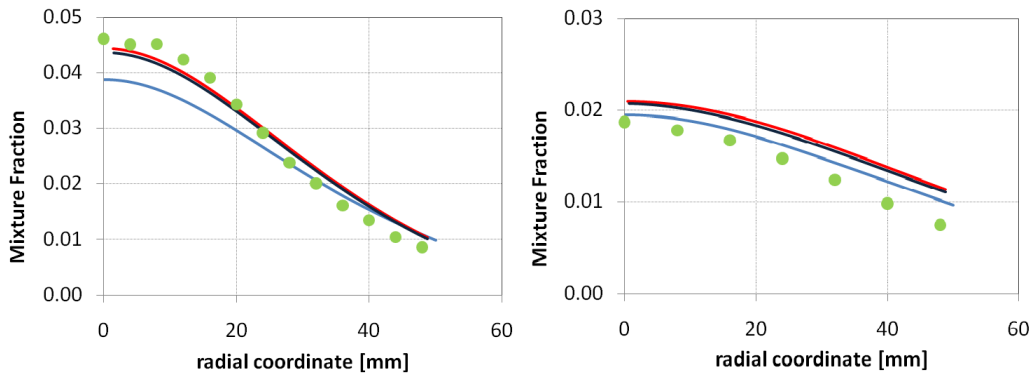


Figura 6.28: Fiamma $H_2/He2$. Profili radiali di *mixture fraction* calcolati a 352 mm (a sinistra) e 562.5 mm (a destra) dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [2]

6.4.3 Fiamma Sandia $H_2/He4$

Questa terza fiamma è quella in cui in alimentazione è contenuta una maggior percentuale di elio. Questo, come si nota dalla mappa di temperatura (fig. 6.29) confrontata con le precedenti, abbassa notevolmente la temperatura della fiamma e accorcia la fiamma stessa,

tanto che il picco di temperatura si posiziona a valori molto inferiori della coordinata assiale.

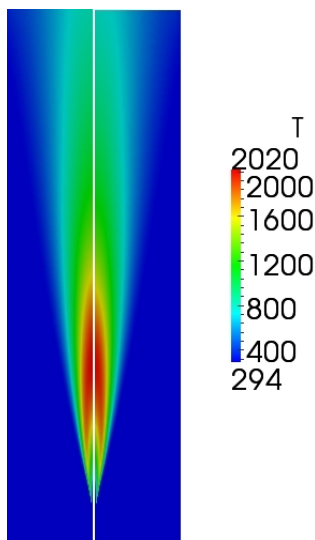


Figura 6.29: Mappe di temperatura per la fiamma $H_2/He4$. A sinistra è visualizzata la simulazione adiabatica, a destra quella non adiabatica

Inoltre, rispetto alle mappe precedenti, la differenza tra la simulazione adiabatica e quella non adiabatica è minima, in quanto le figure risultano quasi identiche, dimostrando che al diminuire della temperatura diminuisce anche il contributo dovuto all'irraggiamento.

Per questa fiamma sono presenti pochi dati sperimentali sull'asse di simmetria, perciò si è scelto di rappresentare qui i risultati oltre che lungo l'asse (fig. 6.30), anche in corrispondenza di una delle traverse proposte, in particolare quella posizionata a 281 mm dallo sbocco (fig. 6.31). Anche in questo caso si nota un buon accordo tra dati sperimentali e modelli e, come già detto precedentemente, in alcuni casi i dati vengono rappresentati meglio dal modello di *Fluent* (ad esempio nel caso della temperatura e della frazione massiva di H_2O), in altri casi da quello di *OpenFOAM* (come per le frazioni massive di OH e H_2)

6. RISULTATI

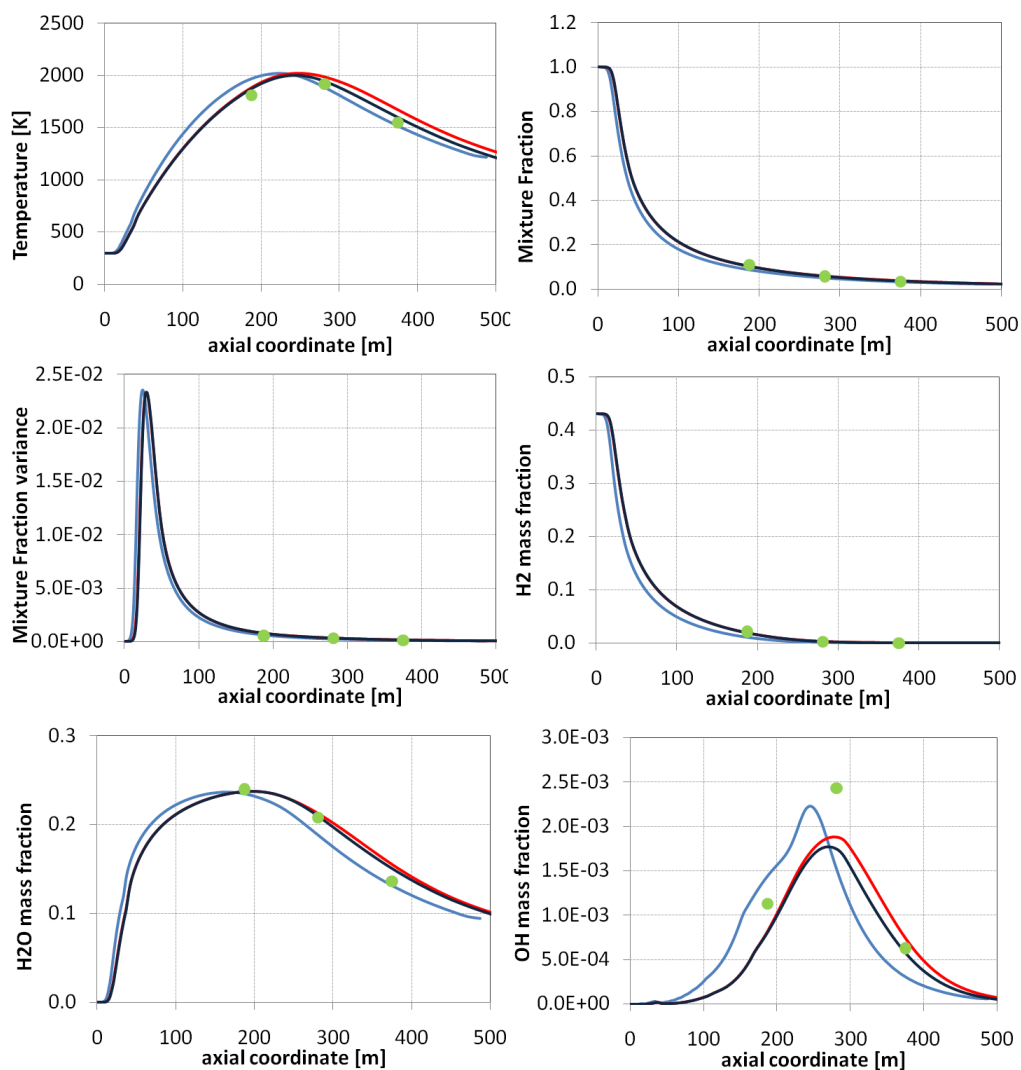


Figura 6.30: *Fiamma H_2/He_4 .* Profili assiali di temperatura, composizione, *mixture fraction* e varianza. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [2]

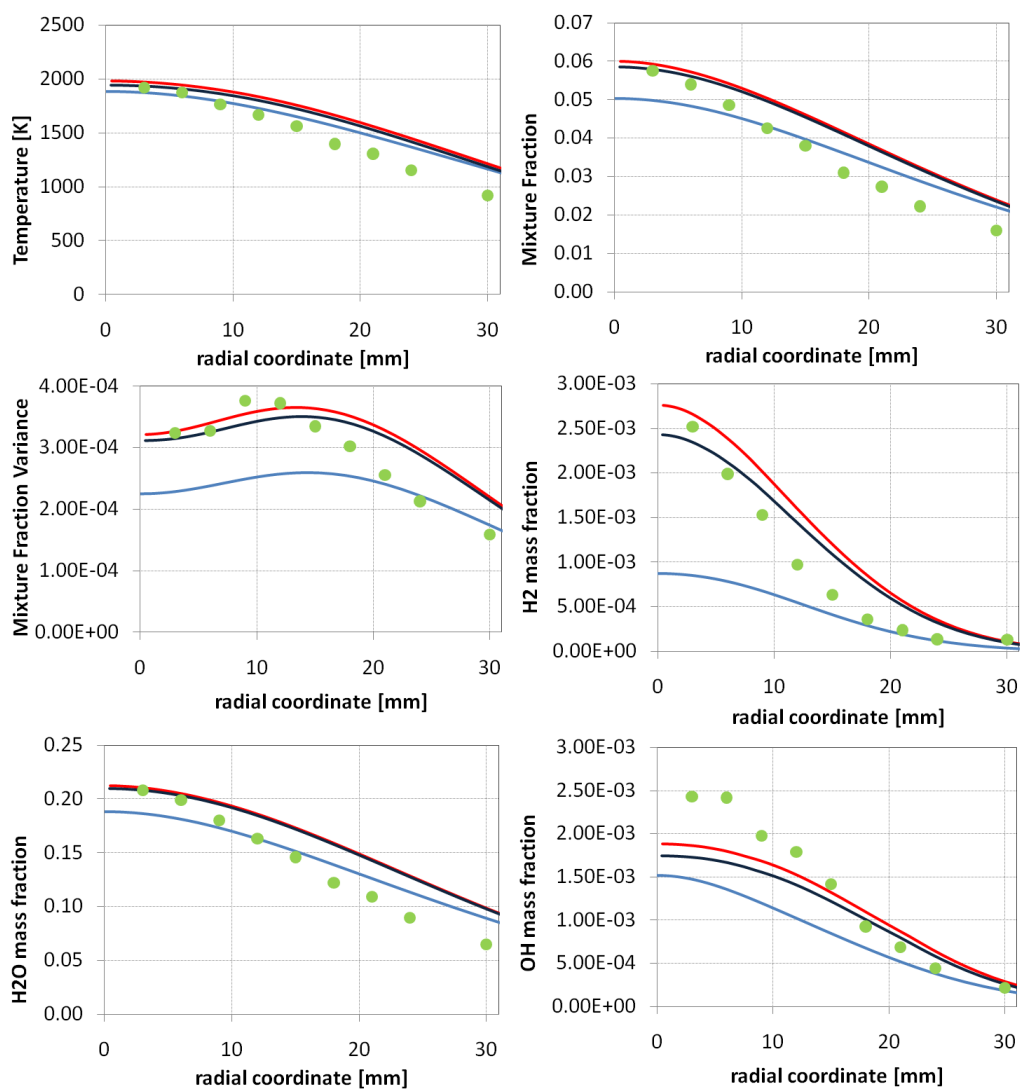


Figura 6.31: *Fiamma H₂/He₄*. Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 281 mm. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [2]

6.4.4 Fiamma Sandia CH₄/H₂/N₂ A

In figura 6.33 sono presentati i profili assiali per la prima delle due fiamme alimentate a metano e idrogeno. Ovviamente le grandezze presenti nei grafici sono aumentate in quanto con l'inserimento di un combustibile contenente carbonio anche i reagenti e i prodotti aumentano.

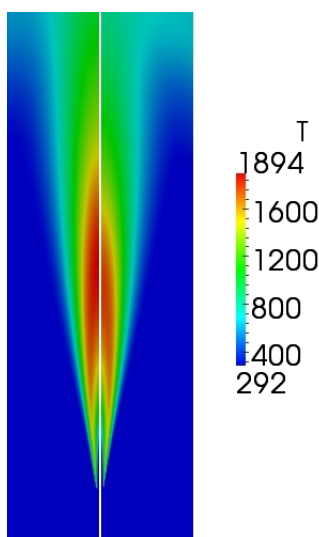


Figura 6.32: Mappe di temperatura della fiamma $CH_4/H_2/N_2$ A. A sinistra è visualizzata la simulazione adiabatica, a destra quella non adiabatica

Anche in questo caso occorre fare alcune osservazioni. Innanzitutto i risultati sono abbastanza buoni ma la simulazione non adiabatica sottostima di poco il picco di temperatura e soprattutto la curva è un poco spostata verso sinistra rispetto ai dati sperimentali. Probabilmente in questo caso un aggiustamento sul coefficiente C_1 del modello *k-epsilon* porterebbe a un miglioramento della situazione, in quanto l'aumento di questo coefficiente normalmente provoca un allungamento della fiamma accompagnato da un piccolo innalzamento nel picco della temperatura. Lo strano comportamento riscontrato in prossimità del picco dei grafici delle frazioni massive di CO e in minima parte di CO_2 potrebbe essere dovuto a una simulazione non arrivata a piena convergenza. In realtà sembra strano perchè le altre variabili calcolate non risentono di questo problema e in secondo luogo poichè anche la simulazione effettuata con *Fluent* possiede gli stessi andamenti.

Analizzando alcuni dei profili radiali si vede inoltre come la scomparsa dei reagenti, almeno per questa fiamma, venga spesso rappresentata in maniera migliore rispetto alla formazione dei prodotti della reazione stessa. Questo è visibile in figura 6.34 per una traversa abbastanza vicina allo sbocco, dove pertanto ci si potrebbero aspettare risultati migliori sulle frazioni massive dei prodotti. Anche in questo caso però le curve di *Fluent* sono addirittura peggiori, dimostrando una difficoltà nel rappresentare questi dati, forse collegata a quanto appena detto sul coefficiente del modello di turbolenza.

Infine va aggiunto che anche per questo tipo di fiamma il profilo di OH viene rappresentato meglio dalla curva della simulazione non adiabatica, confermando il difetto già messo in

luce nelle fiamme precedenti.

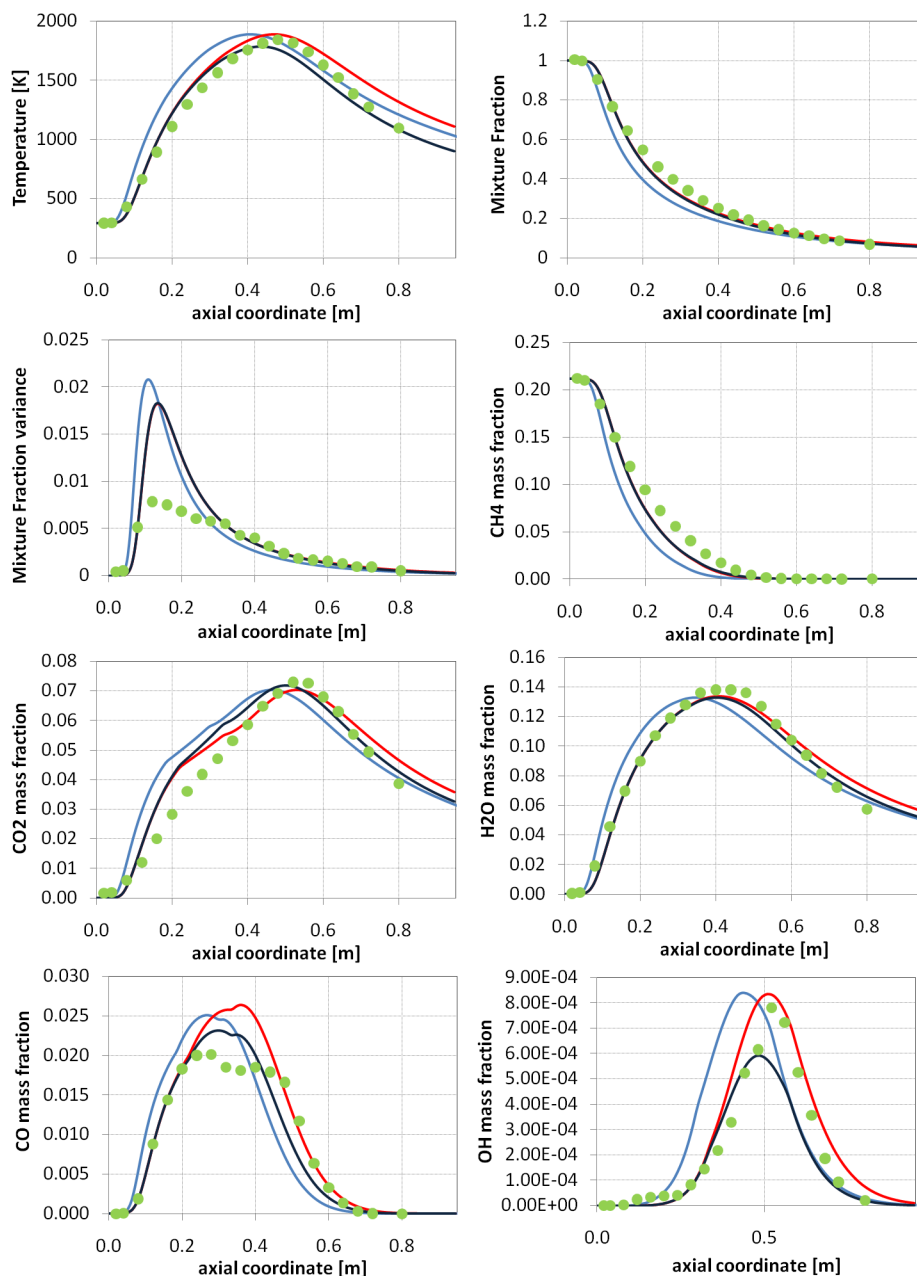


Figura 6.33: Fiamma $CH_4/H_2/N_2$ A. Profili assiali di temperatura, composizione, *mixture fraction* e varianza. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [47]

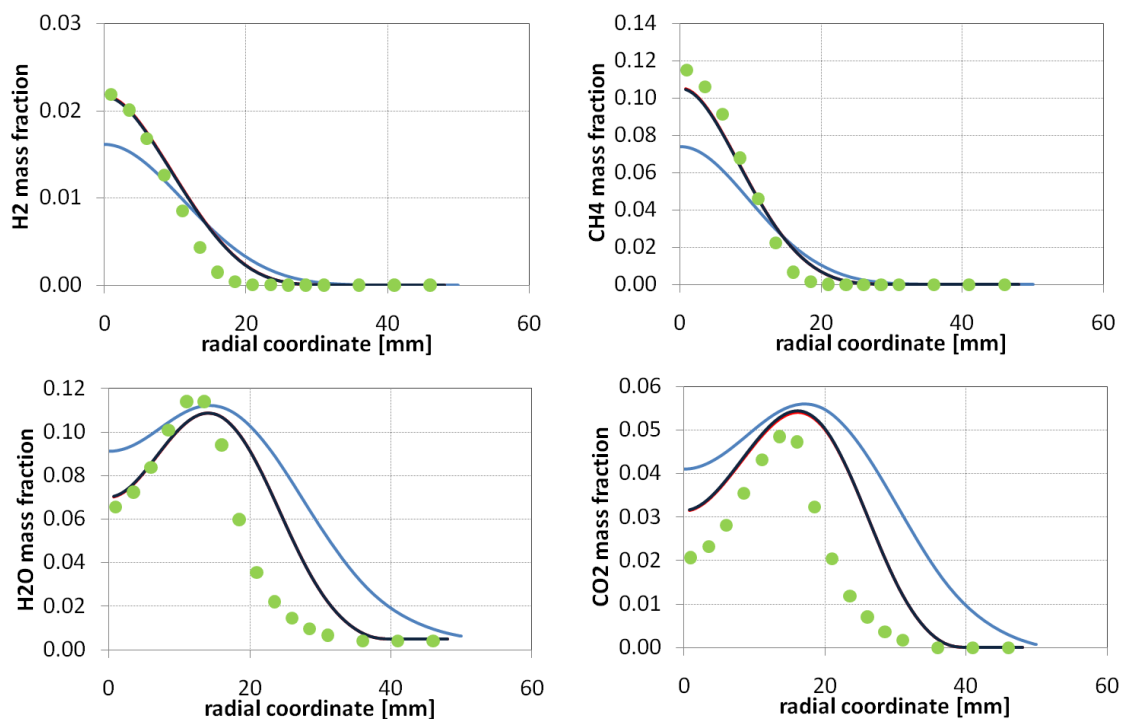


Figura 6.34: Fiamma $CH_4/H_2/N_2$ A. Profili radiali delle frazioni massive di reagenti e prodotti calcolati a 160 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [47]

6.4.5 Fiamma Sandia $CH_4/H_2/N_2$ B

Anche la seconda tipologia di fiamma ad alimentazione metano e idrogeno, i cui profili sono visualizzati in figura 6.36, conferma quanto visto precedentemente: il modello sottostima la temperatura e accorcia la fiamma. Infatti, traslando verso destra le intere curve dei modelli, si avrebbero risultati decisamente più in linea con i dati sperimentali. Va detto però che questa tendenza, come già visto sia nella convalida del modello *ED* che nelle fiamme a idrogeno, è sicuramente maggiore in *Fluent*, le cui curve sono nei grafici ancora più spostate verso sinistra. Da ultimo va detto che in questa fiamma, a differenza di tutte le precedenti, anche la simulazione adiabatica riesce a stimare il picco di concentrazione di *OH* con una certa precisione.

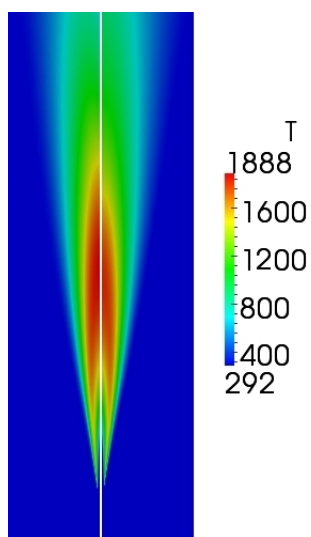


Figura 6.35: Mappe di temperatura della fiamma $CH_4/H_2/N_2$. A sinistra è visualizzata la simulazione adiabatica, a destra quella non adiabatica

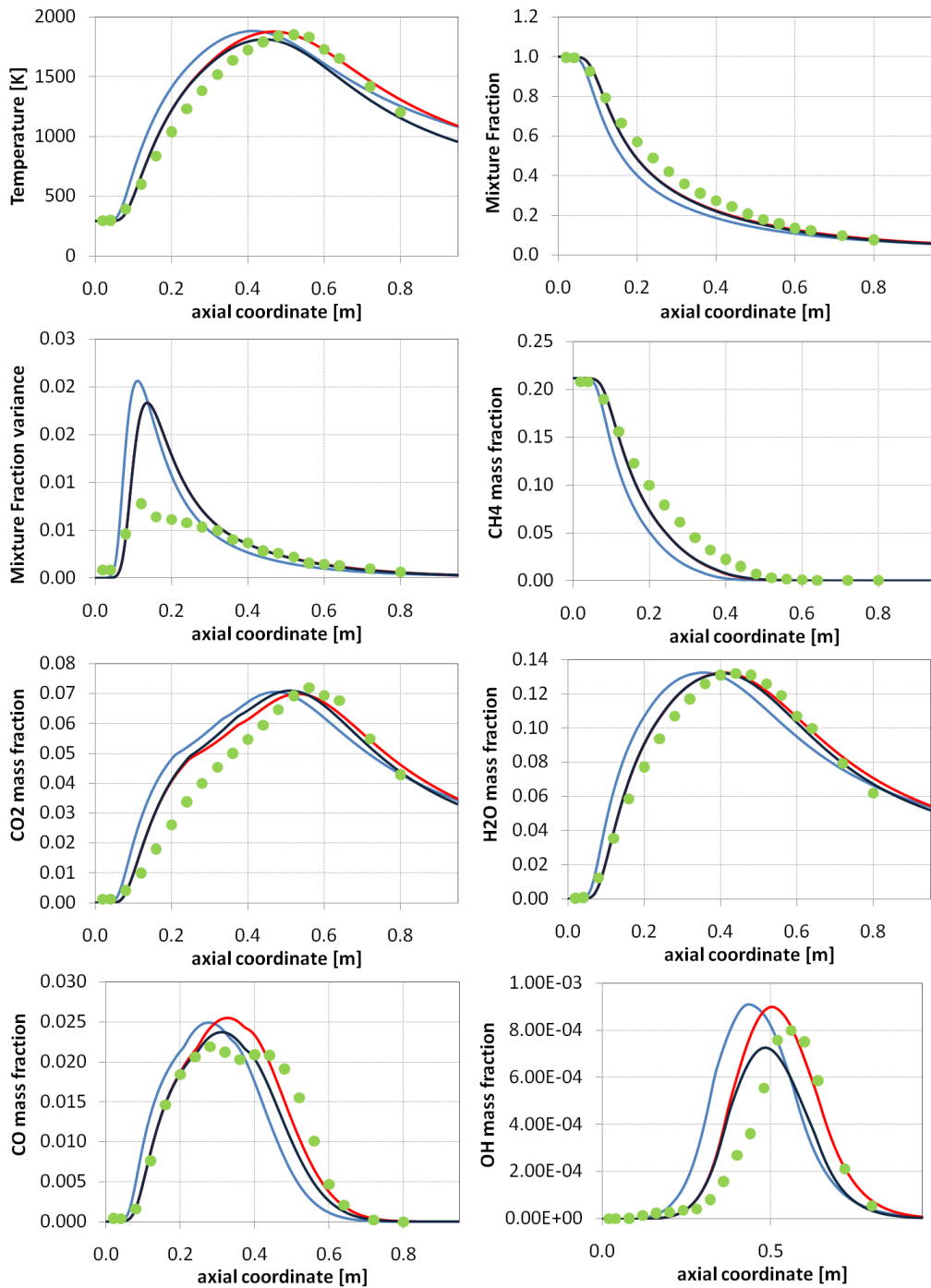


Figura 6.36: Fiamma $CH_4/H_2/N_2$ B. Profili assiali di temperatura, composizione, *mixture fraction* e varianza. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [47]

6.4.6 Fiamma Sandia $CO/H_2/N_2$ A

Questa fiamma alimentata con gas di sintesi è l'ultima fiamma semplice studiata in questo lavoro. Due sono gli aspetti da discutere: il primo è che le simulazioni adiabatica e non adiabatica danno risultati molto simili, mostrando un'influenza minore dell'irraggiamento rispetto alle fiamme precedenti.

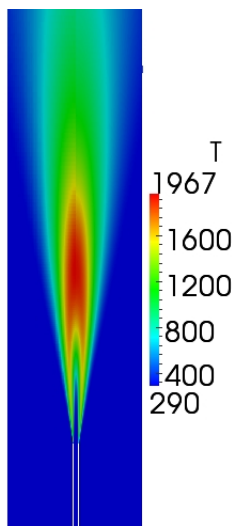


Figura 6.37: Mappe di temperatura della fiamma $CO/H_2/N_2$ A. A sinistra è visualizzata la simulazione adiabatica, a destra quella non adiabatica

Questo è visibile sia nel confronto tra le mappe di temperatura (fig. 6.37) sia analizzando il grafico della temperatura lungo l'asse (fig. 6.38), in cui la differenza è molto meno marcata. Il secondo aspetto è che analizzando i grafici assiali si nota come i dati sperimentali vengano rappresentati in maniera davvero ottima dal modello, anche nel caso della frazione massiva di OH . Questo ottimo accordo è visibile anche nei profili radiali riportati in appendice B.

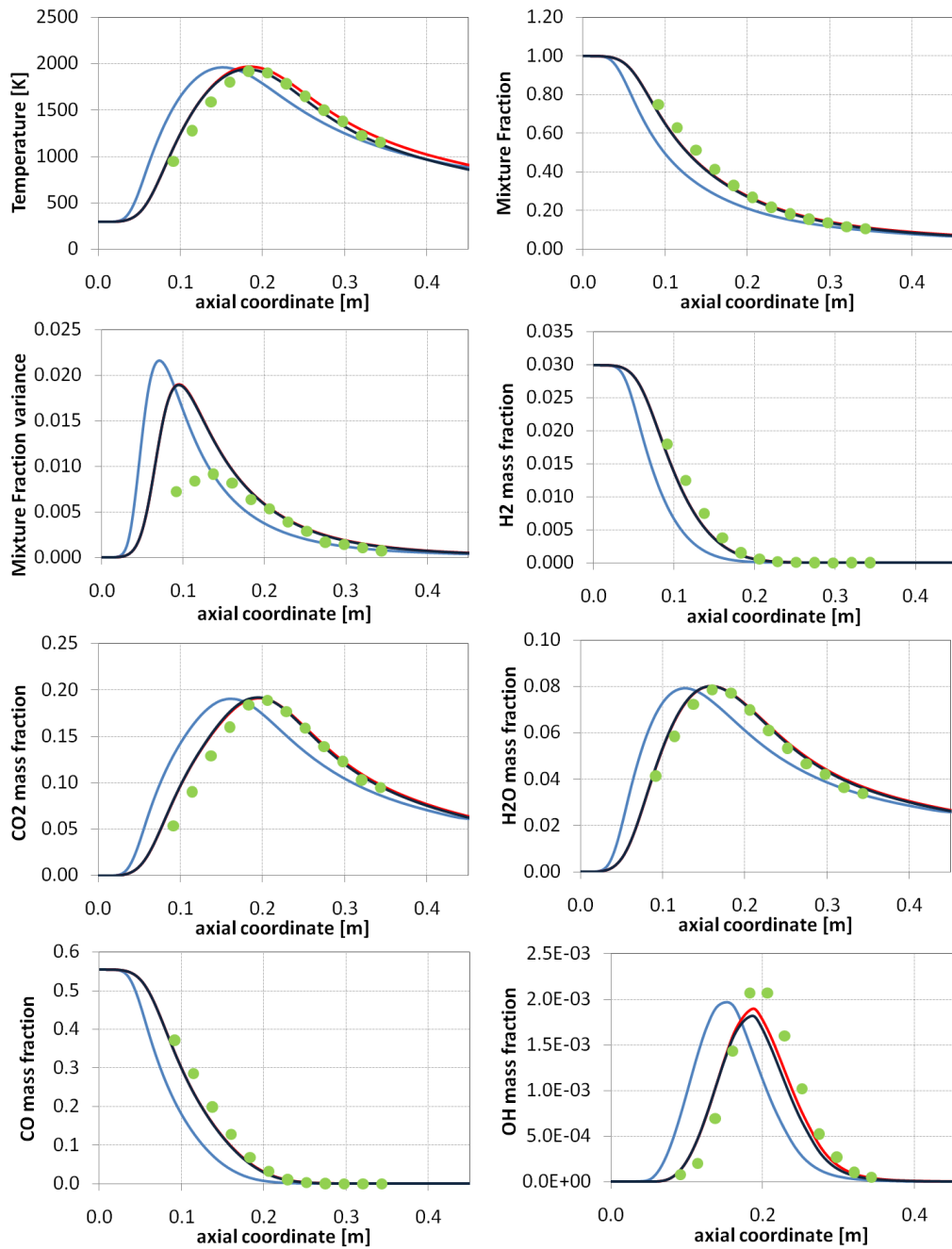


Figura 6.38: Fiamma $CO/H_2/N_2$ A. Profili assiali di temperatura, composizione, *mixture fraction* e varianza. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [4]

6.4.7 Fiamma Sandia *Piloted CH₄/Air D*

Come si vede nella mappa di temperatura in fig. 6.39 la fiamma con pilota riceve in alimentazione oltre ai reagenti anche dei fumi caldi di combustione. Questo accorgimento viene introdotto per far in modo che la fiamma rimanga ancorata al cannello di ingresso, come è stato spiegato nel paragrafo di descrizione della fiamma.

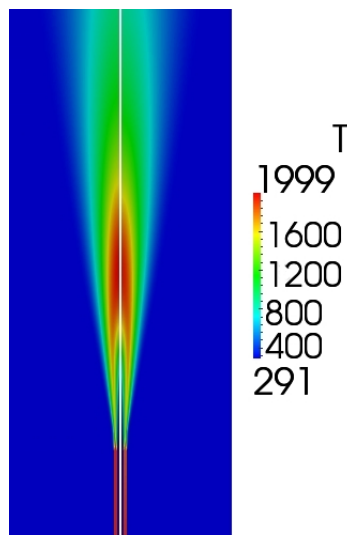


Figura 6.39: Mappe di temperatura della fiamma *Piloted CH₄/Air D*. A sinistra è visualizzata la simulazione adiabatica, a destra quella non adiabatica

Dai profili assiali (fig. 6.40) si nota uno spostamento della fiamma verso sinistra ancora più pronunciato che nei casi precedenti. Anche qui si renderebbe necessario un aggiustamento del coefficiente del modello di turbolenza. In questo caso le differenze nei profili assiali si ripercuotono, nei punti dove le differenze sono macroscopiche, anche sui profili radiali: ad esempio i profili di temperatura e *mixture fraction* visualizzati nella traversa a 216 mm (fig. 6.41) mostrano addirittura andamenti diversi tra i modelli e i dati sperimentali. Questo però può essere spiegabile in quanto già il valore di partenza sull'asse è completamente errato. Ancora una volta però anche le curve visualizzate con *Fluent* seguono sostanzialmente questo andamento. E' da notare infine, nella parte iniziale dei grafici, un comportamento strano della temperatura e delle frazioni massive. Questo comportamento potrebbe essere attribuibile a errori numerici, ma è più probabile che sia dovuto proprio alla presenza del pilota e al fatto che i profili assiali vengano lievemente disturbati dall'inserimento dei fumi caldi di combustione.

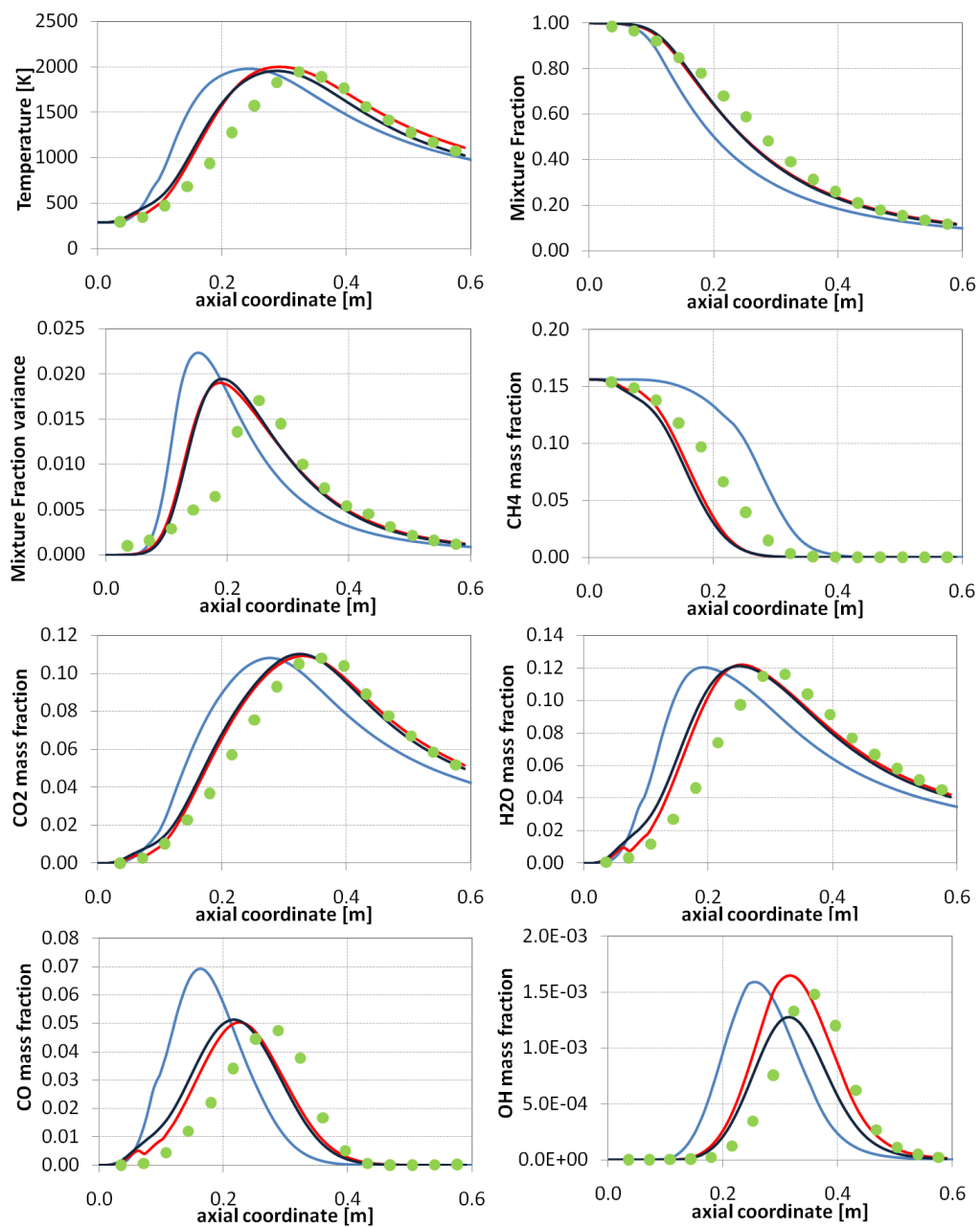


Figura 6.40: Fiamma *Piloted CH₄/Air D*. Profili assiali di temperatura, composizione, *mixture fraction* e varianza. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [3]

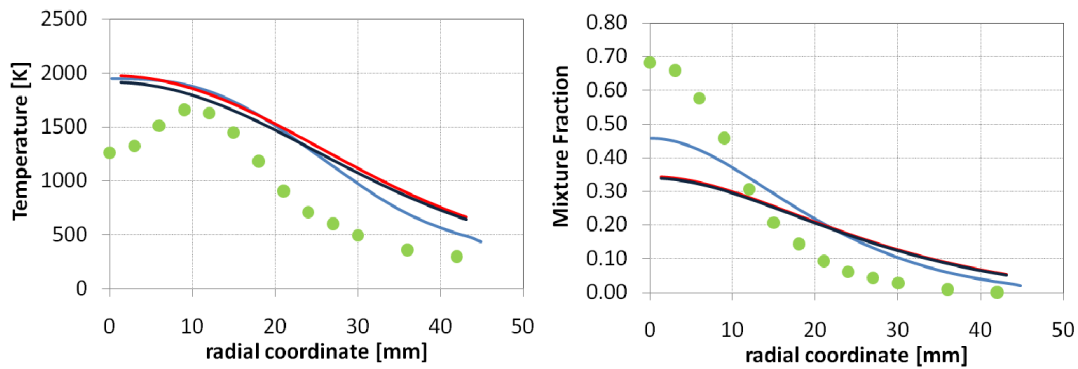


Figura 6.41: Fiamma *Piloted CH₄/Air D*. Profili radiali di temperatura e *mixture fraction* calcolati a 216 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [3]

6.4.8 Fiamma Sandia *Bluff-Body HM1*

La fiamma con *bluff-body*, come accennato in precedenza, è quella che ha dato maggiori problemi in fase di modellazione. Proprio per la sua particolarità e per la sua somiglianza alle fiamme reali esistono in letteratura molte pubblicazioni in cui questa fiamma è stata modellata, non sempre senza difficoltà (ad es. in [29, 33]). Il *bluff-body* crea un ricircolo nella zona appena sopra il cilindro di ceramica che permette come già detto l'ancoraggio della fiamma.

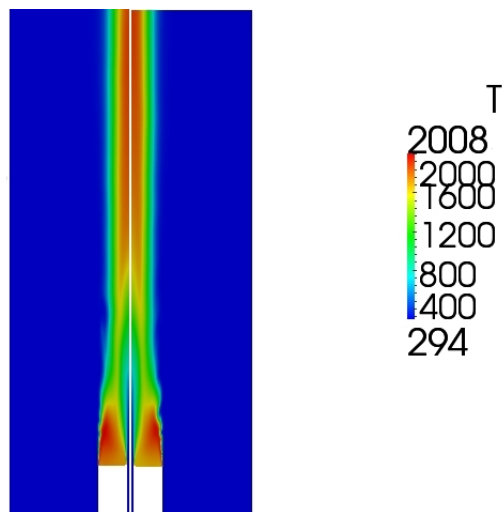


Figura 6.42: Mappe di temperatura della fiamma *Bluff-Body HM1*. A sinistra è visualizzata la simulazione adiabatica, a destra quella non adiabatica

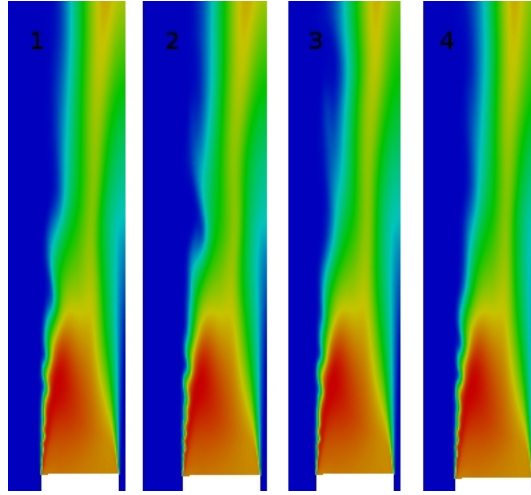


Figura 6.43: Ingrandimento del campo di temperatura nella zona del *bluff-body* in 4 step iterativi diversi

Il problema riscontrato sta nel fatto che la simulazione non riesce a raggiungere una convergenza perfetta, ma la fiamma, invece di stabilizzarsi, continua a oscillare. In figura 6.42 vediamo le mappe di temperatura delle due simulazioni, adiabatica e non, in cui si è riusciti a trovare un settaggio del caso tale per cui le oscillazioni fossero minime e quindi si potessero presupporre risultati accettabili. Andando però ad analizzare un ingrandimento della zona del *bluff-body* mostrato a diversi step iterativi (fig. 6.43), le oscillazioni sono molto evidenti. Questa difficoltà nel raggiungere la convergenza è sicuramente dovuta alle condizioni di turbolenza create dalla presenza del *bluff-body*, che generano instabilità, ma ha una particolare spiegazione, che risiede negli schemi numerici utilizzati. Infatti utilizzando *Fluent* il problema delle oscillazioni non si pone perchè gli schemi utilizzati di default da questo software sono molto dissipativi, proprio per favorire la robustezza a discapito dell'accuratezza. Gli schemi numerici usati da *OpenFOAM* sono meno dissipativi e perciò riescono a cogliere le fluttuazioni a frequenza più bassa delle variabili. Questo comportamento dipende anche dalla griglia utilizzata. Nel nostro lavoro si è notato che, utilizzando una griglia molto lasca, le fluttuazioni non sono presenti. Queste perciò diventano visibili con l'aumento del numero di celle e si può presupporre che per *mesh* molto fitte anche *Fluent* riesca a coglierle. Sarebbe interessante proseguire nello studio di questa fiamma andando ad analizzare nel dettaglio le oscillazioni per verificare se effettivamente il motivo di questo comportamento risieda negli schemi numerici utilizzati. Nel nostro lavoro, nonostante questi problemi, si è cercato di confrontare comunque in maniera macroscopica i dati sperimentali con i risultati ottenuti numericamente.

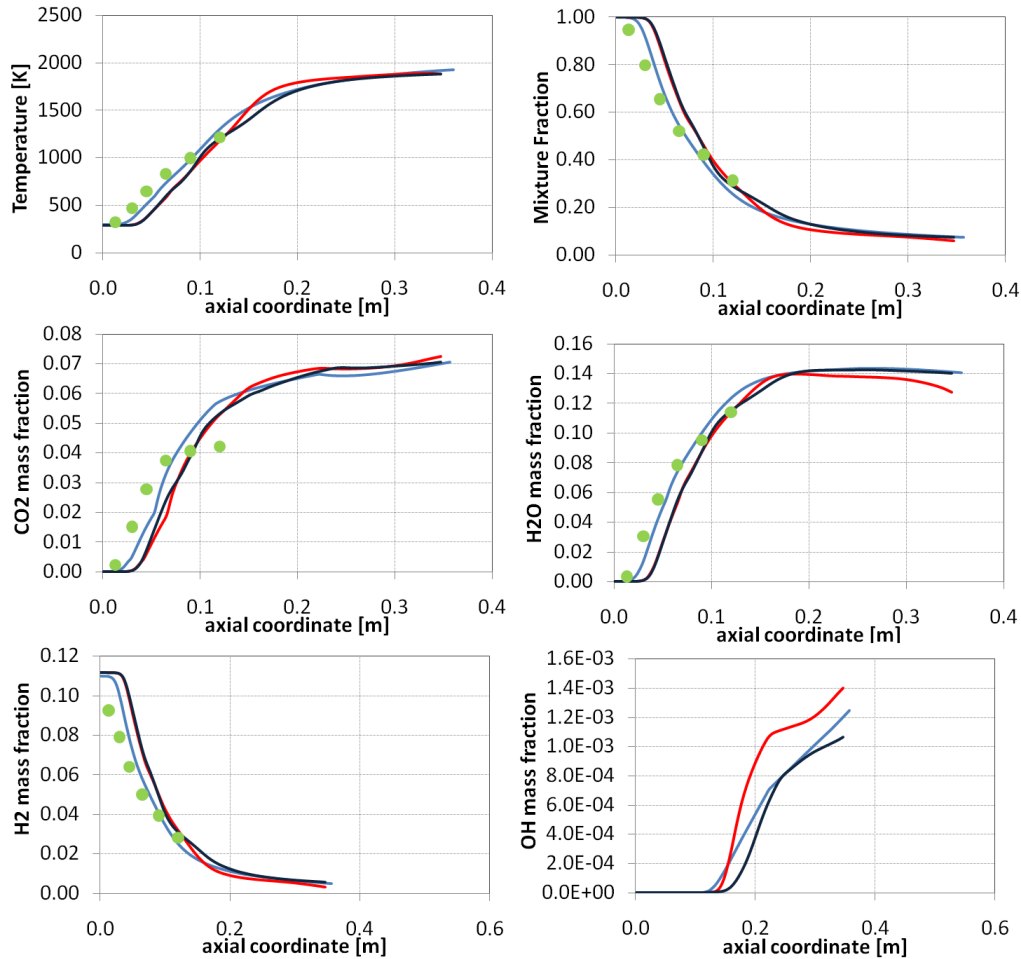


Figura 6.44: Fiamma *Bluff-Body HM1*. Profili assiali di temperatura, composizione, *mixture fraction* e varianza. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [1]

In particolare i profili assiali sono visualizzabili in fig. 6.44. Va detto però che è stato necessario in questo caso modificare il coefficiente C_1 del modello di turbolenza e portarlo a 1.6, perchè i risultati con il *k-epsilon standard* erano troppo differenti rispetto ai dati sperimentali. Anche in questo caso i profili sono decisamente ragionevoli e anche i radiali, alcuni dei quali visibili nelle figure 6.45 e 6.46, rispecchiano questo andamento. Oltre ai profili di temperatura, come sempre indicativi, sono stati aggiunti anche i profili di *OH*, in quanto i dati sperimentali di questa grandezza sull'asse di simmetria non sono disponibili.

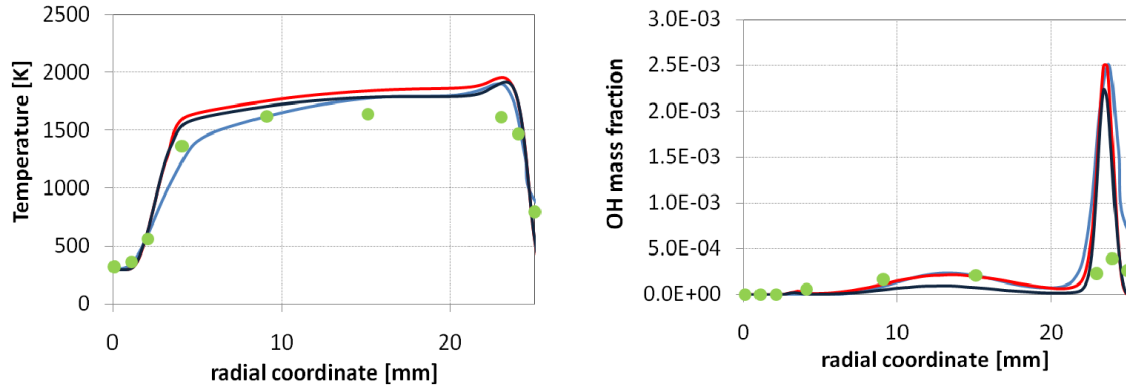


Figura 6.45: Fiamma *Bluff-Body HM1*. Profili radiali di temperatura e frazione massiva di *OH* calcolati a 13 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [1]

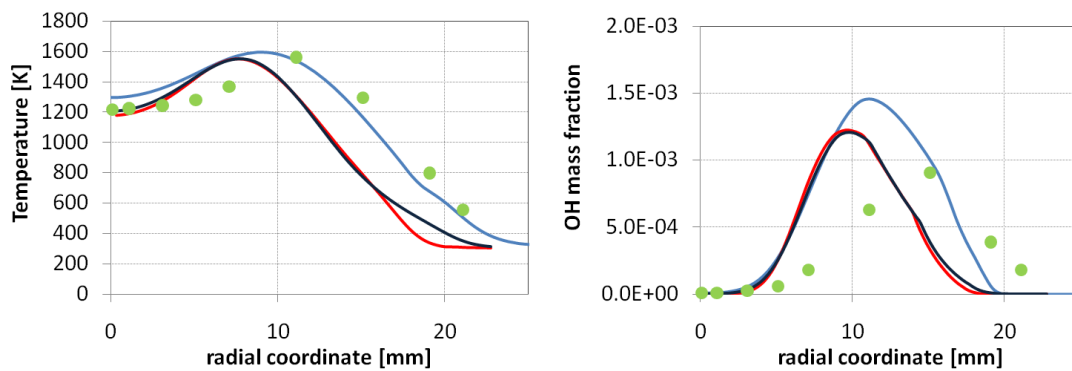


Figura 6.46: Fiamma *Bluff-Body HM1*. Profili radiali di temperatura e frazione massiva di *OH* calcolati a 120 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [1]

Conclusioni e futuri sviluppi

L'obiettivo di questo lavoro di tesi è stato quello di implementare e verificare il funzionamento di alcuni modelli in grado di simulare la combustione turbolenta di fiamme non premiscelate all'interno del software *OpenFOAM*. Questo software possiede alcuni punti di forza: è gratuito e il codice sorgente è reso disponibile e liberamente modificabile.

In questo lavoro è stata possibile l'implementazione di due modelli di combustione: il modello *Eddy Dissipation* e il modello *Steady Laminar Flamelets*. Entrambi sono stati convalidati attraverso l'uso del codice commerciale *Fluent* e di dati sperimentali. Il modello *ED*, molto semplice da implementare, è stato testato su due fiamme semplici, la prima alimentata a idrogeno e la seconda alimentata a metano e idrogeno, e i risultati sono in linea con quelli ottenuti da *Fluent*. Resta il fatto che, pur avendo ottenuto risultati sostanzialmente soddisfacenti, all'inizio del lavoro si pensava che l'accordo tra i due programmi fosse anche migliore. Probabilmente le piccole differenze sono dovute agli schemi numerici utilizzati dai programmi nella risoluzione delle equazioni differenziali, che non è stato possibile far coincidere o comunque ad alcuni accorgimenti usati da *Fluent* e che non è possibile conoscere essendo un programma commerciale con il codice sorgente nascosto all'utente. Il modello *ED*, confrontato coi dati sperimentali, conferma i suoi limiti: le fiamme risultanti possiedono infatti un picco di temperatura decisamente alto rispetto alla realtà, ma questo problema è insito nel modello, nel quale sostanzialmente la cinetica chimica non viene considerata, ma la velocità di reazione è data dalla velocità di miscelazione turbolenta.

Il modello *SLF* al contrario è un modello attraverso il quale si possono usare anche schemi cinetici molto dettagliati, in quanto la chimica della fiamma viene risolta in una fase di *preprocessing* e i risultati vengono raccolti in tabelle che sono poi consultate nella fase di simulazione fluidodinamica. In questo modo il costo computazionale viene abbassato e allo stesso tempo si possono ottenere dei risultati più attinenti alla realtà. Per convalidare il modello sono stati effettuati dei confronti su un numero maggiore di fiamme: oltre alle

due già viste per il modello *ED*, sono state simulate altre fiamme dello stesso tipo ma con differenti condizioni di alimentazione e inoltre una fiamma alimentata a *syngas*, una dotata di bruciatore pilota e una dotata di *bluff-body*. In totale il modello è stato convalidato su otto fiamme differenti. I risultati sono stati soddisfacenti: sebbene il confronto con *Fluent* risulti afflitto dalle piccole differenze sottolineate già per l'*ED*, in molti casi il modello implementato in *OpenFOAM* è riuscito a rappresentare i dati sperimentali addirittura meglio del codice commerciale. Comunque le curve calcolate tramite questo modello sono risultate per tutte le fiamme molto simili ai dati. I risultati però sono decisamente migliori per le fiamme alimentate ad idrogeno e a *syngas*. Quelle alimentate a metano infatti risultano sostanzialmente accorciate rispetto ai dati sperimentali e talvolta il picco di temperatura è un poco più basso. Sembra essere perciò necessario un aggiustamento nei parametri del modello $k - \epsilon$, che però si è deciso di non effettuare durante questo lavoro di tesi, il cui obiettivo era verificare il buon funzionamento del software e non ottenere risultati estremamente accurati. Merita un commento anche il comportamento del modello nelle zone della fiamma più lontane dal cannello di alimentazione. In queste zone, tipicamente vicine al contorno di *outlet* della griglia di calcolo, le soluzioni si fanno meno precise e questo si nota nei profili assiali, ma soprattutto nei profili radiali calcolati a dei valori sull'asse abbastanza lontani dallo sbocco. Questo è senz'altro dovuto all'imposizione di un contorno di uscita nella griglia di calcolo necessario allo svolgimento della simulazione. In realtà però la fiamma reale è inconfinata, e troncatura del dominio di calcolo è evidentemente un'approssimazione della fisica del problema.

Infine bisogna mettere in luce le difficoltà che si sono presentate nella modellazione della fiamma con bruciatore *bluff-body*. Questa fiamma, simulata con *OpenFOAM*, presenta infatti delle oscillazioni che impediscono di raggiungere la piena convergenza. Questo è dovuto probabilmente agli schemi numerici scarsamente dissipativi utilizzati dal programma che, con griglie abbastanza fitte, riescono a cogliere le fluttuazioni delle variabili che possiedono frequenza più bassa. Nonostante questo è stato possibile confrontare macroscopicamente i risultati numerici con i dati sperimentali e il confronto ha dato risposte positive. Pertanto anche il modello *SLF* può ritenersi convalidato.

I futuri sviluppi di questo lavoro potrebbero andare in due direzioni: da una parte si potrebbe cercare di implementare in *OpenFOAM* i restanti modelli di combustione, soprattutto l'*Eddy Dissipation Concept*. Esso è una derivazione dell'*ED*, ma decisamente più accurato e allo stesso tempo complesso e richiede tra l'altro l'utilizzo di un algoritmo denominato *ISAT*, la cui implementazione presenta notevoli difficoltà. Sarebbe inoltre utile verificare anche il funzionamento del modello *Pasr*, già presente in *OpenFOAM*, ma

utilizzabile solo in simulazioni transitorie. Si potrebbe pensare pertanto a un'estensione del modello ai casi stazionari.

Infine potrebbe essere interessante approfondire lo studio riguardante la fiamma con bruciatore *bluff-body*, andando ad analizzare nel dettaglio il problema della convergenza e cercando di verificare la validità della spiegazione data in questa tesi attraverso un'analisi approfondita delle fluttuazioni riscontrate.

Appendice A

Costruzione della libreria di *SLF* non adiabatiche

In questa appendice sono presentate e discusse le equazioni che governano le *SLF*. Inoltre sono mostrati alcuni dettagli sui metodi numerici sviluppati per la soluzione di tali equazioni e la metodologia adottata per la costruzione delle librerie.

A.1 Equazioni delle *flamelets* stazionarie

L'equazione di una *SLF* può essere derivata dalle equazioni che descrivono una fiamma a controdiffusione, utilizzando un'appropriata trasformazione dallo spazio fisico allo spazio della *mixture fraction* (cioè con f come variabile indipendente). La procedura è descritta in [41, 42] e non verrà riportata. In questo lavoro si è utilizzato un set di equazioni semplificate, in accordo con [40]. Qui vengono riassunte le equazioni di conservazione di specie ed energia in funzione della *mixture fraction*:

$$\frac{1}{2}\rho\chi\frac{\partial^2\omega_i}{\partial f^2} + \dot{\Omega}_i = 0 \quad i = 1, \dots, N_S \quad (\text{A.1})$$

$$\frac{1}{2}\rho\chi\frac{\partial^2 T}{\partial f^2} - \frac{1}{C_P} \sum_{i=1}^{N_S} \hat{H}_i \dot{\Omega}_i + \frac{1}{2C_P}\rho\chi\frac{\partial T}{\partial f} \left[\frac{\partial \hat{C}_P}{\partial f} + \sum_{i=1}^{N_S} \hat{C}_{P,i} \frac{\partial \omega_i}{\partial f} \right] = 0 \quad (\text{A.2})$$

Le condizioni al contorno per i lati del *fuel* e dell'ossidante sono:

$$\text{Oxidizer side } (f = 0) \begin{cases} \omega_k = \omega_{k,O} \\ T = T_O \end{cases} \quad (\text{A.3})$$

$$\text{Oxidizer side } (f = 1) \begin{cases} \omega_k = \omega_{k,F} \\ T = T_F \end{cases} \quad (\text{A.4})$$

Nell'equazione riportata sopra, χ è lo *scalar dissipation rate*, che deve essere modellato lungo l'asse della *flamelet*. Nel presente lavoro si assume la seguente espressione [26] che è un'estensione di un'espressione più semplice [38] ai flussi a densità variabile:

$$\chi(f) = \frac{a_S}{4\pi} \frac{3 \left(\sqrt{\frac{\rho_O}{\rho}} + 1 \right)^2}{2 \sqrt{\frac{\rho_O}{\rho}} + 1} \exp \left(-2 \left[\text{erfc}^{-1}(2f) \right]^2 \right) \quad (\text{A.5})$$

dove a_S è lo *strain rate* caratteristico, erfc^{-1} è il complementare dell'inverso della funzione di errore e ρ_O è la densità nel flusso di ossidante.

A.1.1 Estensione alla combustione non adiabatica

L'estensione alla combustione non adiabatica richiede l'introduzione del difetto di entalpia, che è definito come la differenza tra l'entalpia attuale e l'entalpia per una fiamma adiabatica:

$$\phi_H = \left[\hat{H} - f \left(\hat{H}_F - \hat{H}_O \right) \right] \quad (\text{A.6})$$

dove \hat{H}_O e \hat{H}_F denotano rispettivamente l'entalpia del flusso di *fuel* e ossidante. Un set di profili di *laminar flamelets*, a differenti *scalar dissipation rate*, devono essere calcolati ad alcuni valori prefissati di difetto di entalpia. I profili sono organizzati in librerie, e ognuna contiene riferimenti ai valori di χ_{st} , dai valori corrispondenti all'equilibrio fino a quelli corrispondenti alla fiamma estinta, più il valore assunto nel caso di pura miscelazione, e ogni libreria si riferisce a un diverso valore di difetto di entalpia. Assumendo l'ipotesi di diffusività uguale per tutte le specie, è abbastanza facile calcolare i nuovi profili di *flamelets* corrispondenti a un assegnato difetto di entalpia ϕ_H . Se lo stesso valore di difetto di entalpia è forzato sia dal lato del *fuel* che da quello dell'ossidante, il difetto di entalpia

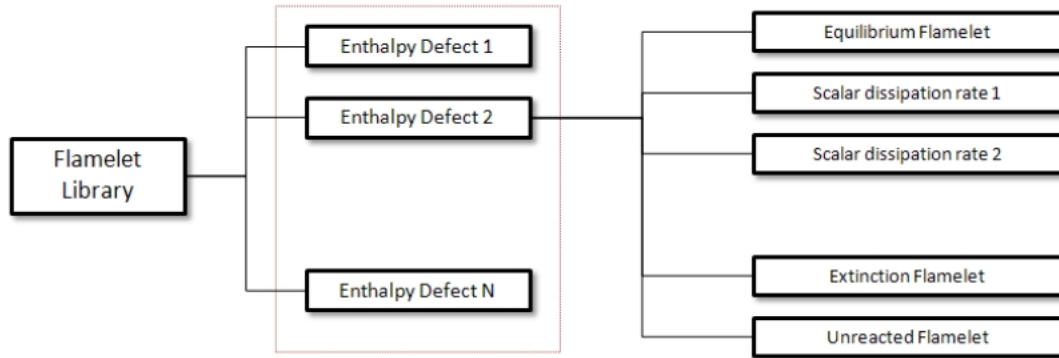


Figura A.1: Struttura di una libreria di *flamelets* non adiabatica

è uniforme lungo l'intera *flamelet*. Molti codici commerciali, tra cui *Fluent* utilizzano l'assunzione che i profili non vengano modificati dal difetto di entalpia, il che significa che per ogni difetto di entalpia nella libreria *flamelets* i profili delle specie vengono presi come gli stessi di quelli della *flamelet* adiabatica. Solo il profilo di temperatura viene aggiornato, usando un approccio semplificato che evita di ricalcolare un nuovo set di *flamelets* [8, 34]. Il principale svantaggio di questo tipo di approccio sta nel fatto che l'effetto delle perdite di calore sulle frazioni delle specie non è tenuto in conto. Utilizzando *OpenFOAM* è stato invece possibile usare l'approccio che per ogni difetto di entalpia ricalcola i profili sia delle specie che della temperatura da zero. Questo approccio è molto più accurato, ma richiede il calcolo di un numero molto grande di profili di *flamelets*. In figura A.1 è riportata l'organizzazione della libreria.

A.1.1.1 Metodo transitorio

Il metodo di risoluzione consiste in un sistema differenziale tra le equazioni di conservazione con condizioni al contorno per le variabili dipendenti. La soluzione numerica del sistema complessivo corrispondente alle equazioni A.1 e A.2 richiede particolare attenzione: in questa procedura tali equazioni vengono risolte in una forma transitoria, che è ottenuta aggiungendo le corrispondenti derivate in funzione del tempo, in modo da ottenere un sistema parabolico di equazioni differenziali alle derivate parziali piuttosto che un sistema di equazioni differenziali ordinarie. Le equazioni sono riportate di seguito e vengono risolte per un tempo sufficiente a raggiungere le condizioni stazionarie:

$$\rho \frac{\partial \omega_i}{\partial t} = \frac{1}{2} \rho \chi \frac{\partial^2 \omega_i}{\partial f^2} + \dot{\Omega}_i \quad i = 1, \dots, N_S \quad (\text{A.7})$$

$$\rho \frac{\partial T}{\partial t} = \frac{1}{2} \rho \chi \frac{\partial^2 T}{\partial f^2} - \frac{1}{C_P} \sum_{i=1}^{N_S} \hat{H}_i \dot{\Omega}_i + \frac{1}{2C_P} \rho \chi \frac{\partial T}{\partial f} \left[\frac{\partial \hat{C}_P}{\partial f} + \sum_{i=1}^{N_S} \hat{C}_{P,i} \frac{\partial \omega_i}{\partial f} \right] \quad (\text{A.8})$$

A.1.1.2 Discretizzazione alle differenze finite

Le equazioni appena scritte sono discretizzate attraverso una griglia spaziale non uniforme, in modo da ottenere un sistema di equazioni differenziali ordinarie. La discretizzazione adottata usa delle tecniche convenzionali di risoluzione alle differenze finite per *mesh* non uniformi:

$$\frac{\omega_{i,j}}{\partial t} = \frac{1}{2} \chi_j \frac{\frac{\omega_{i,j+1} - \omega_{i,j}}{x_{j+1} - x_j} - \frac{\omega_{i,j} - \omega_{i,j-1}}{x_j - x_{j-1}}}{\frac{1}{2}(x_{j+1} - x_{j-1})} + \frac{\dot{\Omega}_i}{\rho} \quad i = 1, \dots, N_S \quad (\text{A.9})$$

$$\rho \frac{\partial T_j}{\partial t} = \frac{1}{2} \chi_j \frac{\frac{T_{j+1} - T_j}{x_{j+1} - x_j} - \frac{T_j - T_{j-1}}{x_j - x_{j-1}}}{\frac{1}{2}(x_{j+1} - x_{j-1})} - \frac{\sum_{i=1}^{N_S} \hat{H}_{i,j} \dot{\Omega}_{i,j}}{\rho_j \hat{C}_{P,j}} + \frac{\chi_j}{2\hat{C}_{P,j}} \frac{\partial T}{\partial f} \left[\frac{\partial \hat{C}_P}{\partial f} \Big|_j^{cen} + \sum_{i=1}^{N_S} \hat{C}_{P,i} \frac{\partial \omega_i}{\partial f} \Big|_j^{cen} \right] \quad (\text{A.10})$$

in cui le derivate sono discretizzate usando lo schema centrale differenziato:

$$\frac{\partial \psi}{\partial x} \Big|_j^{cen} = \psi_{j-1} \frac{x_{j+1} - x_j}{(x_j - x_{j-1})(x_{j+1} - x_{j-1})} + \psi_j \frac{x_{j-1} - 2x_j + x_{j+1}}{(x_{j+1} - x_j)(x_j - x_{j-1})} + \psi_{j+1} \frac{x_j - x_{j-1}}{(x_{j+1} - x_j)(x_{j+1} - x_{j-1})} \quad (\text{A.11})$$

A.1.2 Costruzione della libreria *flamelet* non adiabatica

Dopo la generazione della *flamelet*, i profili della stessa sono approssimati con una assunta *PDF* associata e quindi tabulati per poter essere consultati nel codice *CFD*. Il valore istantaneo di ogni quantità ψ all'interno di una *flamelet* può essere espresso come una funzione della *mixture fraction*, dello *scalar dissipation rate* e del difetto di entalpia:

$$\psi = \psi(Z, \chi_{st}, \phi_H) \quad (\text{A.12})$$

Il corrispondente valore medio può essere ottenuto introducendo la *joint-PDF* $P(Z, \chi_{st}, \phi_H)$:

$$\bar{\psi} = \int_{-\infty}^{+\infty} \int_0^{\infty} \int_0^1 \psi(Z, \chi_{st}, \phi_H) \cdot P(Z, \chi_{st}, \phi_H) \cdot dZ \cdot d\chi_{st} \cdot d\phi_H \quad (\text{A.13})$$

Nell'approccio *presumed PDF* una forma per tale *PDF* è assegnata su base empirica. Da numerose evidenze sperimentali [38, 5] si denota che la *mixture fraction* e lo *scalar dissipation rate* sono statisticamente indipendenti. In accordo con [11], la *joint-PDF* può essere divisa nel prodotto di tre diverse *PDF* di una singola variabile:

$$P(Z, \chi_{st}, \phi_H) = P(Z) \cdot P(\chi_{st}) \cdot P(\phi_H) \quad (\text{A.14})$$

In accordo con [6], il risultato dell'integrazione rispetto alla *mixture fraction* è relativamente insensibile ai dettagli della forma della *PDF*, se questa viene scelta con basi fisiche. Normalmente vengono adottate una β -*PDF* o una *clipped Gaussian PDF*. L'integrazione rispetto a χ_{st} viene normalmente effettuata con una distribuzione *logaritmica normale* o semplicemente con una *Delta di Dirac* centrata sul valore medio locale. La *PDF* del difetto di entalpia è solitamente assunta essere anch'essa una *Delta di Dirac*.

A.1.2.1 β -*PDF*

La β -*PDF* è l'approccio più usato per modellare la *PDF* della *mixture fraction*, che è descritta dalla seguente funzione:

$$\tilde{P}_B(f^*; \underline{x}, t) = \frac{(f)^{\alpha-1}(1-f)^{\beta-1}}{\int_0^1 (f)^{\alpha-1}(1-f)^{\beta-1} df} \quad (\text{A.15})$$

in cui α e β sono parametri funzione del valore medio e della varianza della *mixture fraction*, calcolati dal solutore:

$$\alpha = \tilde{f} \cdot \left[\frac{\tilde{f}(1-\tilde{f})}{\tilde{f}''2} - 1 \right] \quad (\text{A.16})$$

$$\beta = (1-\tilde{f}) \cdot \left[\frac{\tilde{f}(1-\tilde{f})}{\tilde{f}''2} - 1 \right] \quad (\text{A.17})$$

In generale la distribuzione è asimmetrica, ma approssima la forma di una gaussiana simmetrica quando i rapporti $\frac{\sqrt{\tilde{f}''^2}}{\tilde{f}}$ e $\frac{\sqrt{\tilde{f}''^2}}{(1-\tilde{f})}$ sono piccoli.

A.1.2.2 *Clipped-Gaussian PDF*

La *clipped-Gaussian PDF* è basata sul *clipping* (letteralmente taglio) di una funzione gaussiana così che la probabilità è una quantità finita solo nella regione consentita della *mixture fraction*. Una *clipped-Gaussian PDF* contiene una distribuzione gaussiana $G(f)$ per la regione turbolenta $0 < f < 1$ e la funzione *Delta di Dirac* dove i fluidi sono smiscelati a $f = 0$ e $f = 1$. Nella forma secondo Favre risulta:

$$P(f) = \alpha_1 \cdot \delta(f) + (1 - \alpha_1 - \alpha_2) \frac{G(f)}{I_G} + \alpha_2 \cdot \delta(1 - f) \quad (\text{A.18})$$

in cui $\delta(f - f_0)$ è la *Delta di Dirac* centrata in $f = f_0$, di cui l'integrale assume valore unitario se $f = f_0$ e nullo negli altri casi. La variabile I_G è definita come:

$$I_G = \int_0^1 G(f) df \quad (\text{A.19})$$

I parametri α_1 e α_2 e la varianza g della distribuzione, che sono funzioni di \tilde{f} e \tilde{f}''^2 , possono essere valutate usando un metodo non costoso dal punto di vista computazionale suggerito in [14]:

$$g = \min \left[\tilde{f}''^2, \left(\frac{\tilde{f}}{2} \right)^2, \left(\frac{1 - \tilde{f}}{2} \right)^2 \right] \quad (\text{A.20})$$

$$G(f) = \begin{cases} \frac{1}{\sqrt{2\pi g}} e^{-\frac{(f-\tilde{f})^2}{2g}} & \frac{|f-\tilde{f}|}{\sqrt{g}} \leq 2 \\ 0 & \frac{|f-\tilde{f}|}{\sqrt{g}} > 2 \end{cases} \quad (\text{A.21})$$

$$\alpha_1 = \left(1 - \frac{\tilde{f}(1-\tilde{f}) - \tilde{f}''^2}{\tilde{f}(1-\tilde{f}) - g} \right) (1 - \tilde{f}) \quad (\text{A.22})$$

A.1.2.3 Distribuzione *log-normal*

La distribuzione *log-normal* è molto usata per descrivere la *PDF* dello *scalar dissipation rate*:

$$P(\chi_{st}) = \frac{1}{\sqrt{2\pi}} \frac{1}{\sigma \chi_{st}} e^{-\frac{1}{2\sigma^2} \left(\ln \frac{\chi_{st}}{\tilde{\chi}_{st}} - \frac{\sigma^2}{2} \right)^2} \quad (\text{A.23})$$

in cui si assume che $\sigma = 2$ (in accordo con i risultati sperimentali mostrati in [46])

A.1.2.4 Integrazione numerica

L'integrazione numerica rispetto al difetto di entalpia e allo *scalar dissipation rate* se anche a questo viene associata una *Delta di Dirac* è immediata, per via della semplicità della *PDF*:

$$\begin{aligned} \bar{\psi} &= \int_{-\infty}^{+\infty} \int_0^{\infty} \int_0^1 \psi(f, \chi_{st}, \phi_H) \cdot P(f) \cdot \delta(\chi_{st} - \tilde{\chi}_{st}) \cdot \delta(\phi_H - \tilde{\phi}_H) \cdot df \cdot d\chi_{st} \cdot d\phi_H = (\text{A.24}) \\ &= \int_0^1 \psi(f, \tilde{\chi}_{st}, \tilde{\phi}_H) \cdot P(f) \cdot df \end{aligned}$$

Al contrario l'integrazione numerica rispetto alla *mixture fraction* e allo *scalar dissipation rate* nel caso di *PDF log normale* è più difficoltosa e deve essere effettuata in maniera accurata e poco costosa:

$$\bar{\psi} = \int_0^{\infty} \int_0^1 \psi(f, \chi_{st}, \tilde{\phi}_H) \cdot P(f) \cdot P(\chi_{st}) \cdot df \cdot d\chi_{st} = \int_0^{\infty} P(\chi_{st}) \cdot d\chi_{st} \int_0^1 \psi(f, \chi_{st}, \tilde{\phi}_H) \cdot P(f) \cdot df \quad (\text{A.25})$$

A.1.2.5 Integrazione rispetto a f

Innanzitutto focalizziamoci sull'integrale $\int_0^1 \psi(f, \chi_{st}, \tilde{\phi}_H) \cdot P(f) \cdot df$ rispetto alla *mixture fraction*. Per la *clipped Gaussian* la soluzione numerica di tale integrale è abbastanza semplice, per il fatto che non ci sono singolarità e la soluzione si può ottenere usando le normali formule di quadratura [24]. Comunque mentre la formulazione della $\beta - PDF$ sembra semplice confrontata con quella della *clipped Gaussian*, lo sforzo computazionale

richiesto per determinare numericamente l'integrale può risultare molto grande. L'integrazione numerica della $\beta - PDF$ incontra numerosi problemi dovuti alla singolarità del problema sia nel lato dell'ossidante che del combustibile, che dipende dai parametri α e β , e un problema di *overflow* quando i parametri della *PDF* sono sufficientemente alti:

$$\int_0^1 \psi(f, \chi_{st}, \tilde{\phi}_H) \cdot P(f) \cdot df = \frac{\int_0^1 \psi(f, \chi_{st}, \tilde{\phi}_H) \cdot B(f) \cdot df}{\int_0^1 B(f) df} \quad (\text{A.26})$$

in cui $B(f) = f^{\alpha-1}(1-f)^{\beta-1}$. In [28] viene suggerito di valutare gli integrali al numeratore e al denominatore usando questa soluzione approssimata, che rimuove le singolarità ai contorni del dominio di integrazione:

$$\begin{aligned} \int_0^1 \Gamma(f)B(f)df &= \int_0^\varepsilon \Gamma(f)B(f)df + \int_\varepsilon^{1-\varepsilon} \Gamma(f)B(f)df + \int_{1-\varepsilon}^1 \Gamma(f)B(f)df \cong \quad (\text{A.27}) \\ &\cong \Gamma_{ox} \frac{\varepsilon^\alpha}{\alpha} + \int_\varepsilon^{1-\varepsilon} \Gamma(f)B(f)df + \Gamma_{fuel} \frac{\varepsilon^\beta}{\beta} \end{aligned}$$

in cui ε è un parametro (solitamente assunto pari a 10^{-6}) e $\Gamma(f)$ è una generica funzione che corrisponde a $\psi(f, \chi_{st}, \tilde{\phi}_H)$ per l'integrale al numeratore e a 1 per l'integrale al denominatore. L'integrale $\int_\varepsilon^{1-\varepsilon} \Gamma(f)B(f)df$ può ora essere calcolato numericamente attraverso le già citate formule di quadratura.

A.1.2.6 Integrazione rispetto a χ_{st}

L'integrazione rispetto a χ_{st} è più semplice da ottenere. In questo lavoro si utilizza l'approccio suggerito in [32]. Dall'equazione A.25, dopo l'integrazione rispetto a f si ottiene:

$$\int_0^\infty P(\chi_{st}) \cdot d\chi_{st} \int_0^1 \psi(f, \chi_{st}, \tilde{\phi}_H) \cdot P(f) \cdot df = \int_0^\infty H(\chi_{st}, \tilde{f}, \tilde{f}''^2, \tilde{\phi}_H) P(\chi_{st}) d\chi_{st} \quad (\text{A.28})$$

L'intervallo di integrazione in χ_{st} è divisa in L sottoinsiemi $[\chi_{l-\frac{1}{2}}, \chi_{l+\frac{1}{2}}]$ con $l = 1, 2, \dots, L$ e $\chi_{l-\frac{1}{2}} = 0$, così che χ_l può essere considerato un valore rappresentativo per l'intervallo

corrispondente. L'integrale è valutato attraverso le seguenti approssimazioni:

$$\int_0^\infty H(\chi_{st}, \tilde{f}, \tilde{f}''^2, \tilde{\phi}_H) P(\chi_{st}) d\chi_{st} = \sum_{l=1}^L \int_{\chi_{l-\frac{1}{2}}}^{\chi_{l+\frac{1}{2}}} H(\chi_{st}, \tilde{f}, \tilde{f}''^2, \tilde{\phi}_H) P(\chi_{st}) d\chi_{st} \cong \quad (\text{A.29})$$

$$\sum_{l=1}^L H(\chi_{st}, \tilde{f}, \tilde{f}''^2, \tilde{\phi}_H) \int_{\chi_{l-\frac{1}{2}}}^{\chi_{l+\frac{1}{2}}} P(\chi_{st}) d\chi_{st}$$

La soluzione approssimata dell'integrale riportata sopra non è costosa in termini computazionali ed è accurata anche con un numero limitato di punti nello spazio di χ_{st} , al contrario dei punti necessari nello spazio di f . Ora abbiamo bisogno di risolvere gli integrali:

$$\int_{\chi_{l-\frac{1}{2}}}^{\chi_{l+\frac{1}{2}}} P(\chi_{st}) d\chi_{st} = \int_{\chi_{l-\frac{1}{2}}}^{\chi_{l+\frac{1}{2}}} \frac{1}{\sqrt{2\pi}} \frac{1}{\sigma \chi_{st}} e^{-\frac{1}{2\sigma^2} \left(\ln \frac{\chi_{st}}{\tilde{\chi}_{st}} - \frac{\sigma^2}{2} \right)^2} d\chi_{st} \quad (\text{A.30})$$

in cui $P(\chi_{st})$ viene assunta essere una distribuzione *log-normal*, già riportata in A.23. Introducendo la variabile di integrazione $\theta = \frac{1}{\sqrt{2\sigma}} \left(\ln \frac{\chi_{st}}{\tilde{\chi}_{st}} - \frac{\sigma^2}{2} \right)$, l'integrale diventa:

$$\int_{\chi_{l-\frac{1}{2}}}^{\chi_{l+\frac{1}{2}}} \frac{1}{\sqrt{2\pi}} \frac{1}{\sigma \chi_{st}} e^{-\theta^2} d\chi_{st} = \frac{1}{\sqrt{\pi}} \int_{\theta_{l-\frac{1}{2}}}^{\theta_{l+\frac{1}{2}}} e^{-\theta^2} d\theta \quad (\text{A.31})$$

Questo integrale ha una soluzione analitica, data da:

$$\frac{1}{\sqrt{\pi}} \int_{\theta_{l-\frac{1}{2}}}^{\theta_{l+\frac{1}{2}}} e^{-\theta^2} d\theta = \frac{1}{2} \left[\text{erf}(\theta_{l+\frac{1}{2}}) - \text{erf}(\theta_{l-\frac{1}{2}}) \right] \quad (\text{A.32})$$

dove *erf* è la funzione d'errore e

$$\theta_{l\pm\frac{1}{2}} = \frac{1}{\sqrt{2\sigma}} \left(\ln \frac{\chi_{l\pm\frac{1}{2}}}{\tilde{\chi}_{st}} - \frac{\sigma^2}{2} \right) \quad (\text{A.33})$$

Appendice B

Risultati aggiuntivi

B.1 Modello *ED*

B.1.1 Fiamma Sandia $H_2/He0$

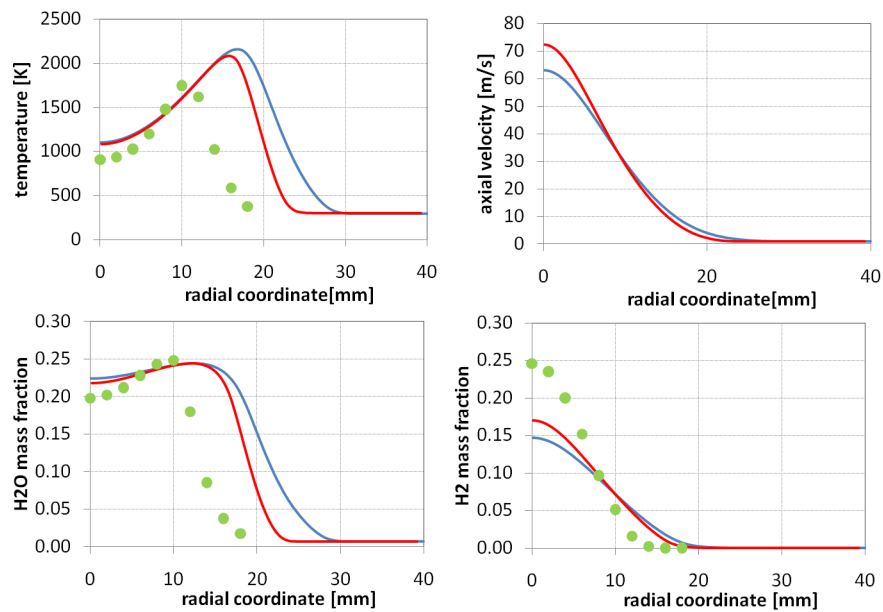


Figura B.1: Fiamma $H_2/He0$. Profili radiali di temperatura, velocità e composizione calcolati a 84 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent*, linea rossa *OpenFOAM*) e dati sperimentali (punti in verde) [2]

B. RISULTATI AGGIUNTIVI

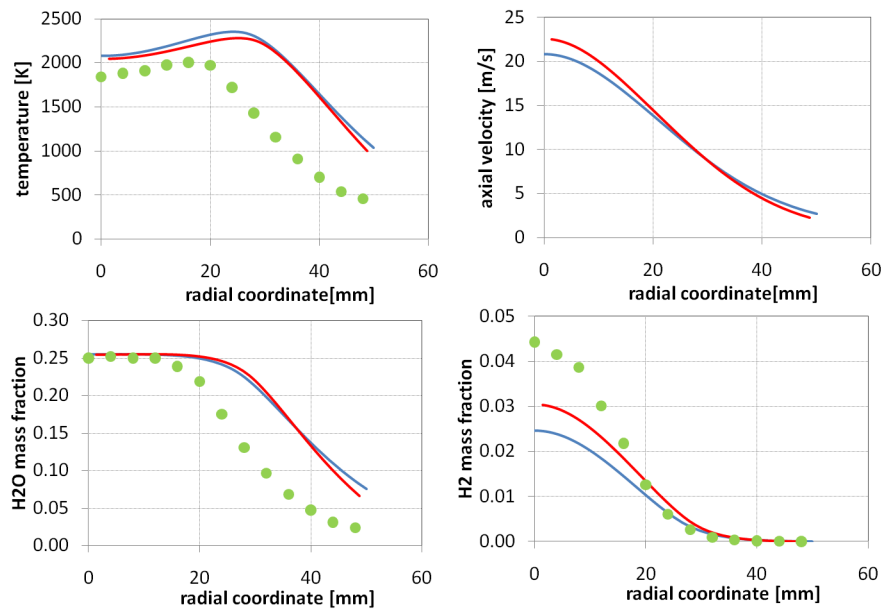


Figura B.2: Fiamma $H_2/He0$. Profili radiali di temperatura, velocità e composizione calcolati a 253 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent*, linea rossa *OpenFOAM*) e dati sperimentali (punti in verde) [2]

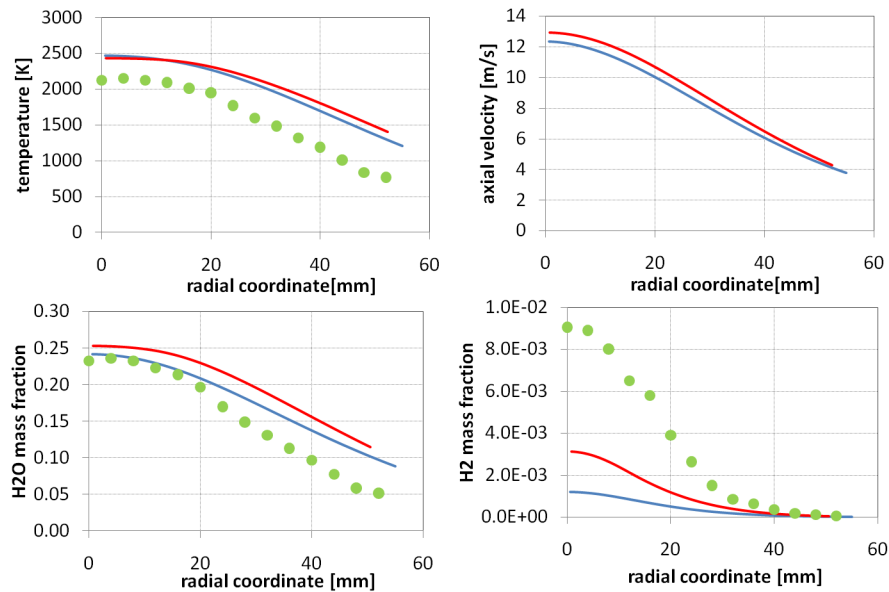


Figura B.3: Fiamma $H_2/He0$. Profili radiali di temperatura, velocità e composizione calcolati a 422 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent*, linea rossa *OpenFOAM*) e dati sperimentali (punti in verde) [2]

B. RISULTATI AGGIUNTIVI

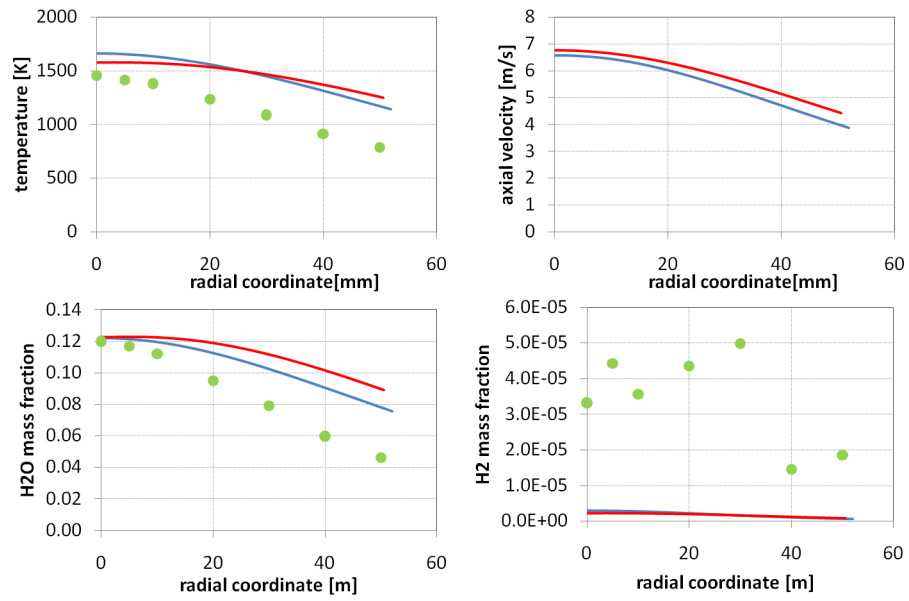


Figura B.4: Fiamma $H_2/He0$. Profili radiali di temperatura, velocità e composizione calcolati a 675 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent*, linea rossa *OpenFOAM*) e dati sperimentali (punti in verde) [2]

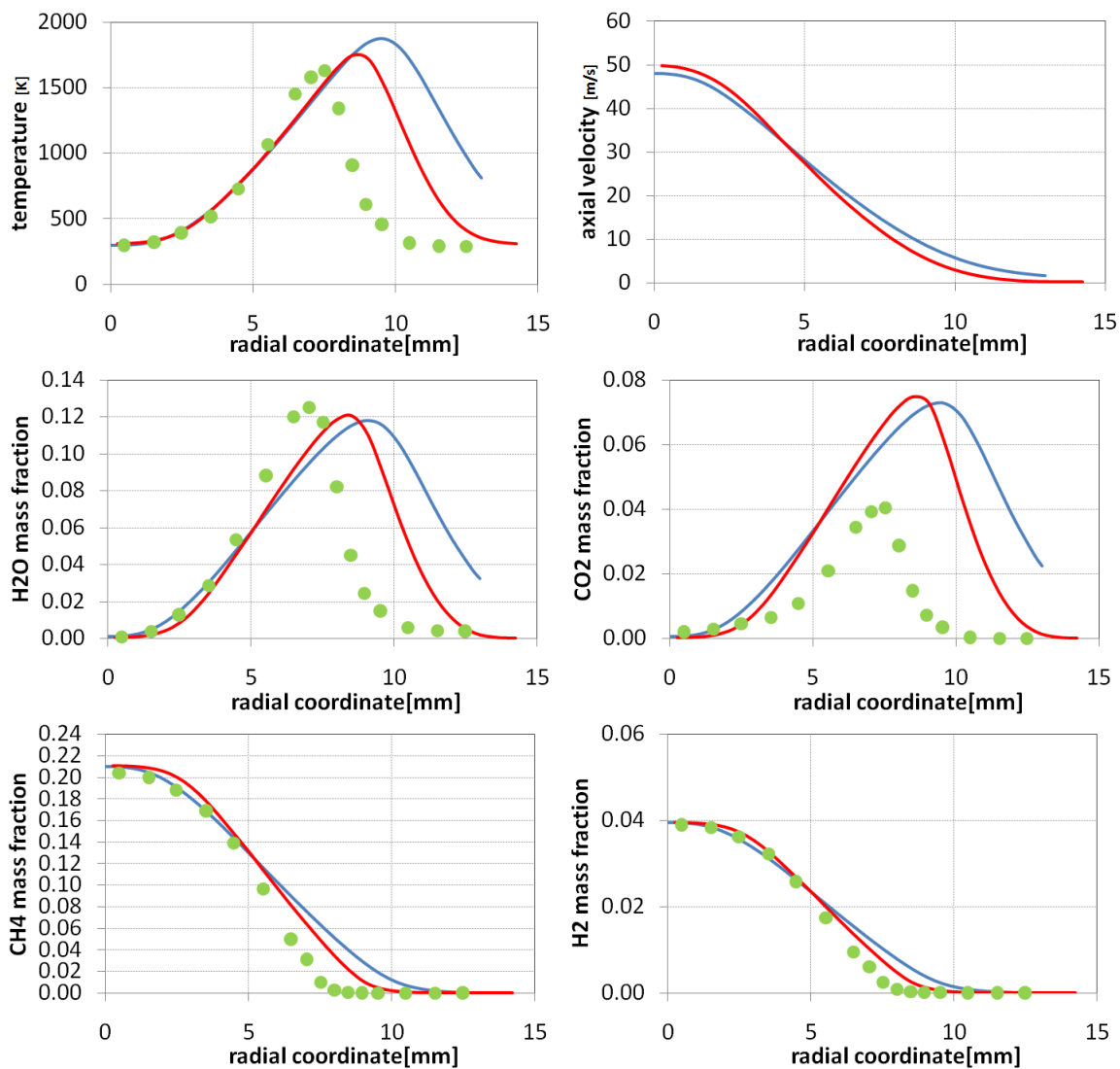
B.1.2 Fiamma Sandia $CH_4/H_2/N_2$ A

Figura B.5: Fiamma $CH_4/H_2/N_2$ A. Profili radiali di temperatura, velocità e composizione calcolati a 40 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent*, linea rossa *OpenFOAM*) e dati sperimentali (punti in verde) [47]

B. RISULTATI AGGIUNTIVI

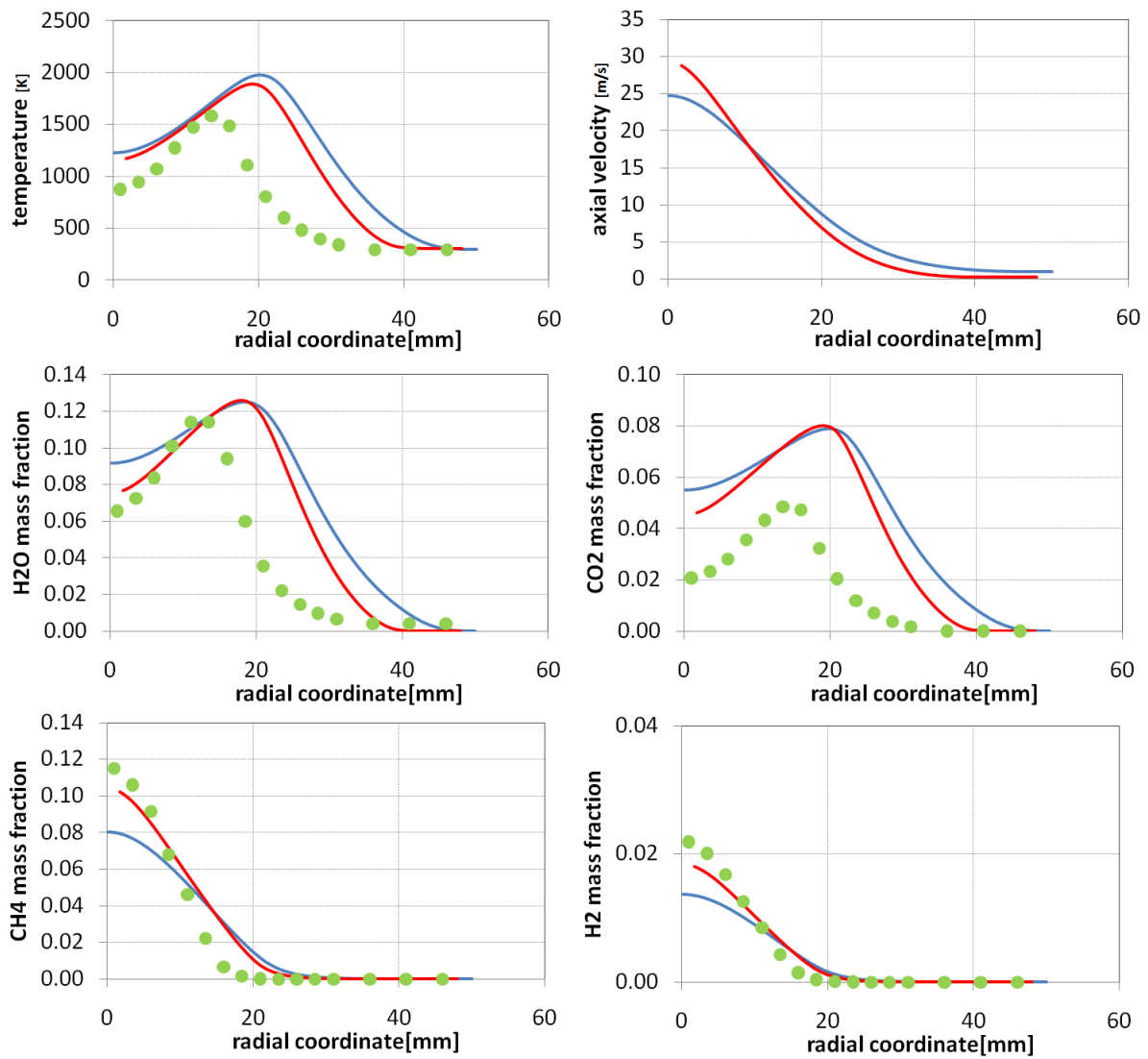


Figura B.6: Fiamma $CH_4/H_2/N_2$ A. Profili radiali di temperatura, velocità e composizione calcolati a 160 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent*, linea rossa *OpenFOAM*) e dati sperimentali (punti in verde) [47]

B. RISULTATI AGGIUNTIVI

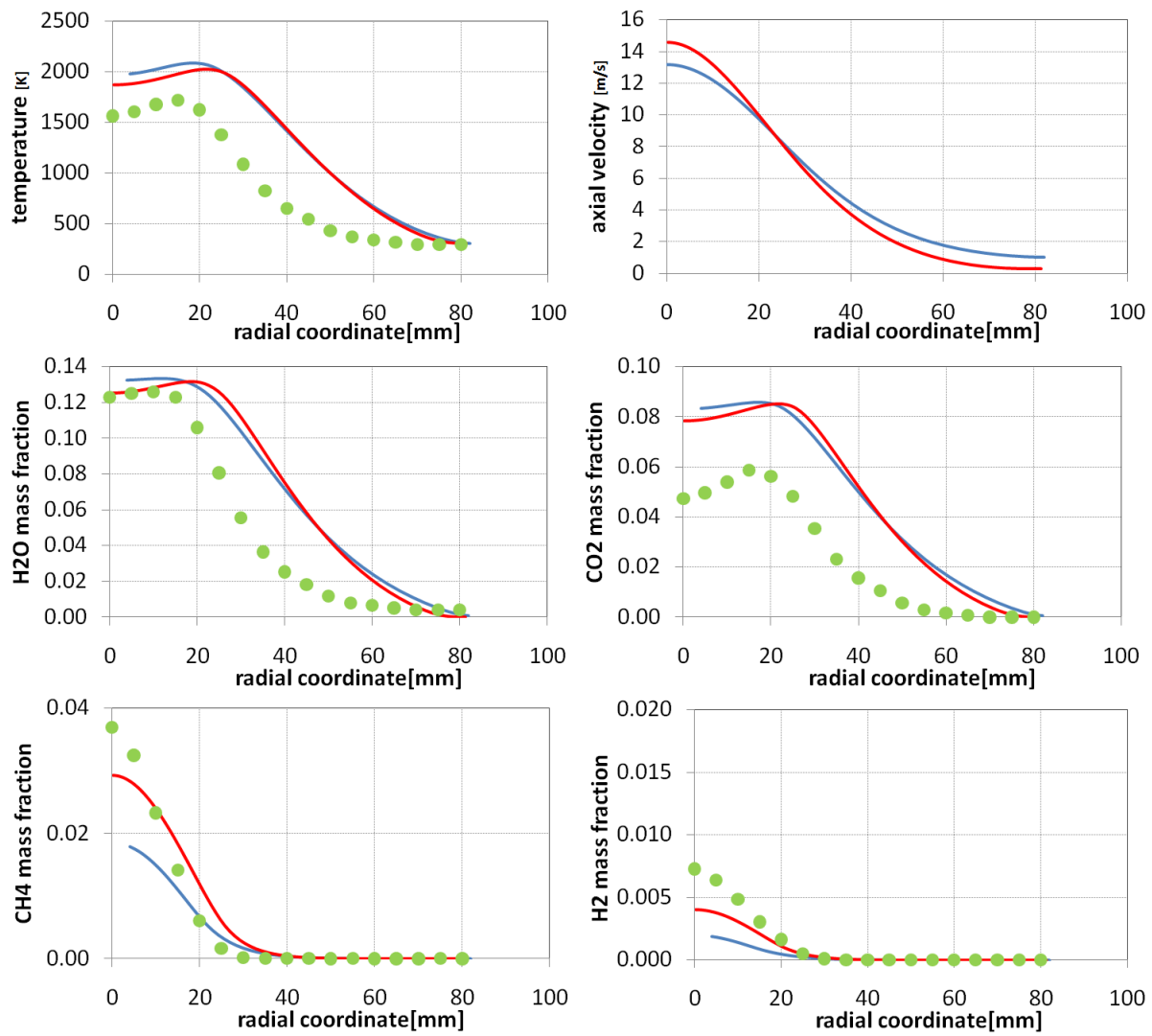


Figura B.7: Fiamma $CH_4/H_2/N_2$ A. Profili radiali di temperatura, velocità e composizione calcolati a 320 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent*, linea rossa *OpenFOAM*) e dati sperimentali (punti in verde) [47]

B. RISULTATI AGGIUNTIVI

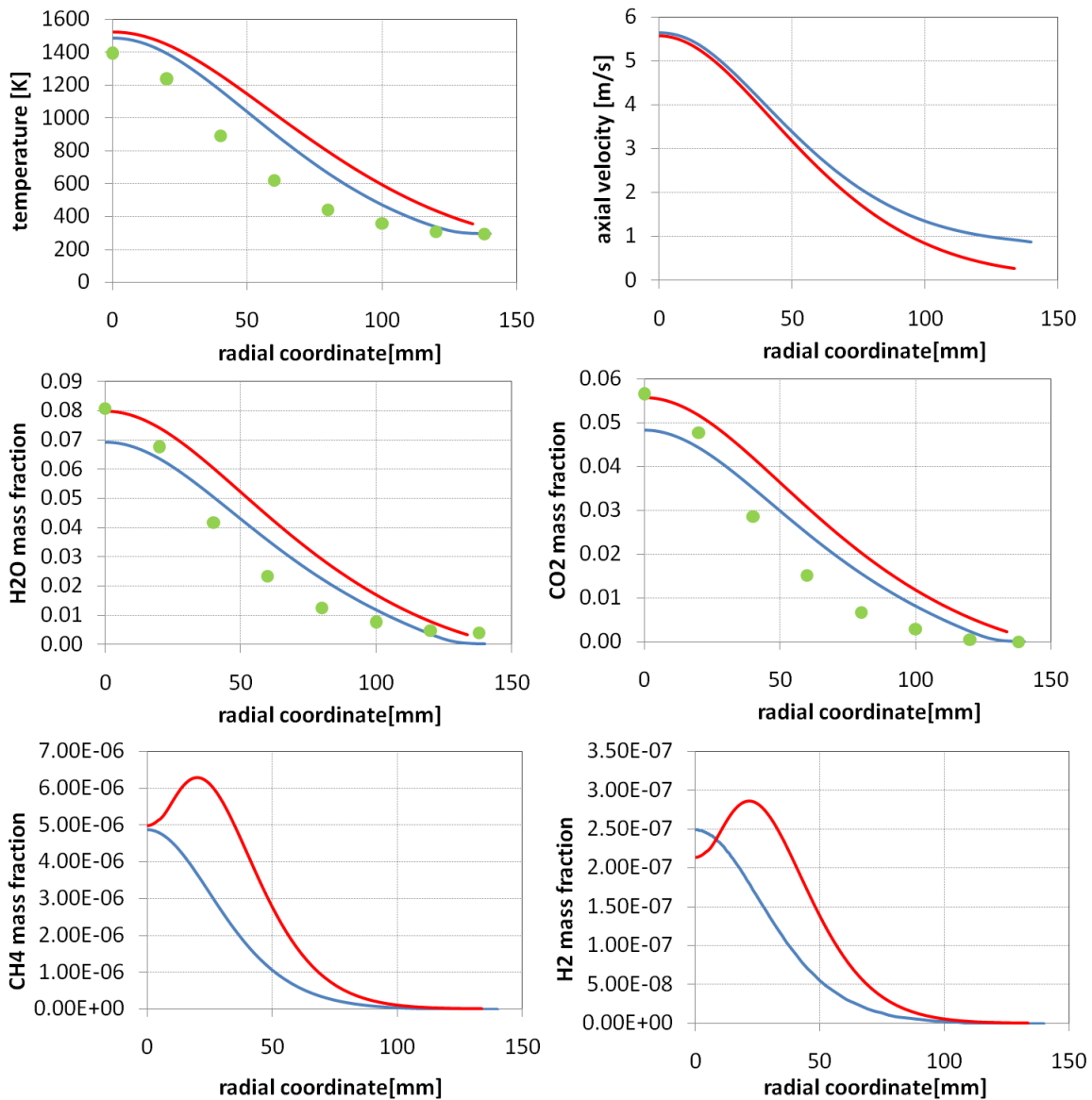


Figura B.8: Fiamma $CH_4/H_2/N_2$ A. Profili radiali di temperatura, velocità e composizione calcolati a 640 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent*, linea rossa *OpenFOAM*) e dati sperimentali (punti in verde) [47]

B.2 Modello *SLA*

B.2.1 Fiamma Sandia $H_2/He0$

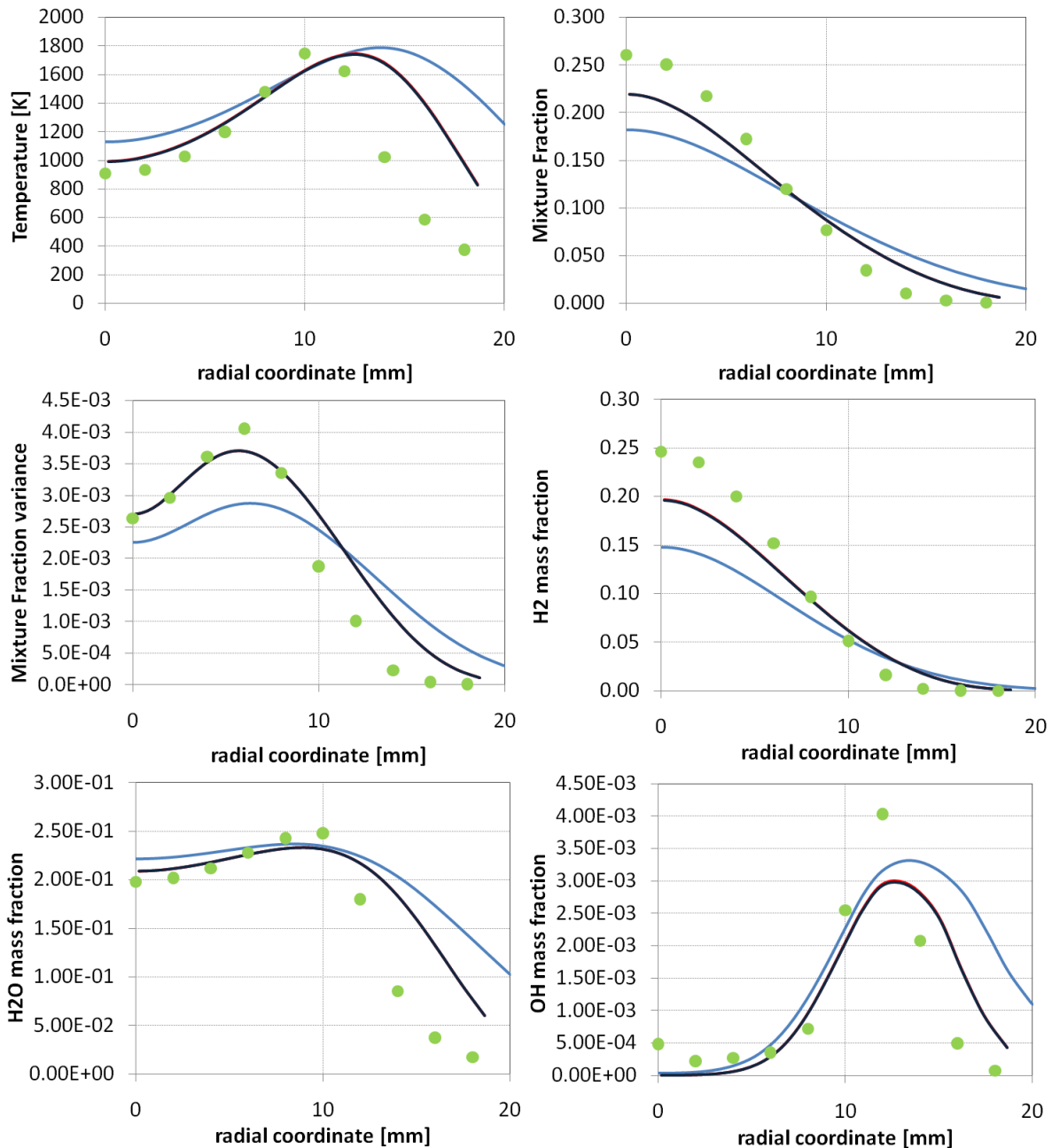


Figura B.9: Fiamma $H_2/He0$. Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 84 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [2]

B. RISULTATI AGGIUNTIVI

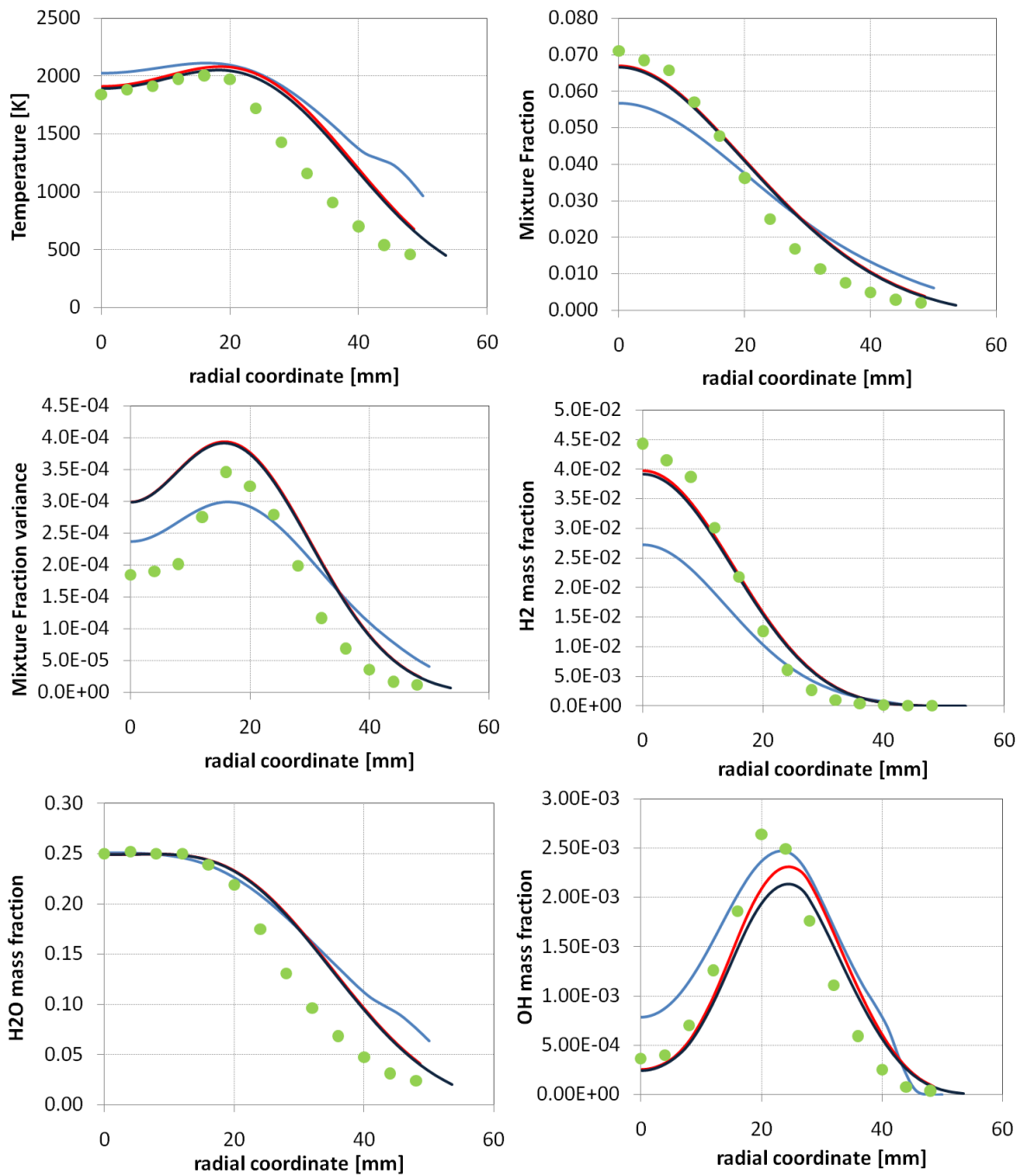


Figura B.10: Fiamma $H_2/He0$. Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 253 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [2]

B. RISULTATI AGGIUNTIVI

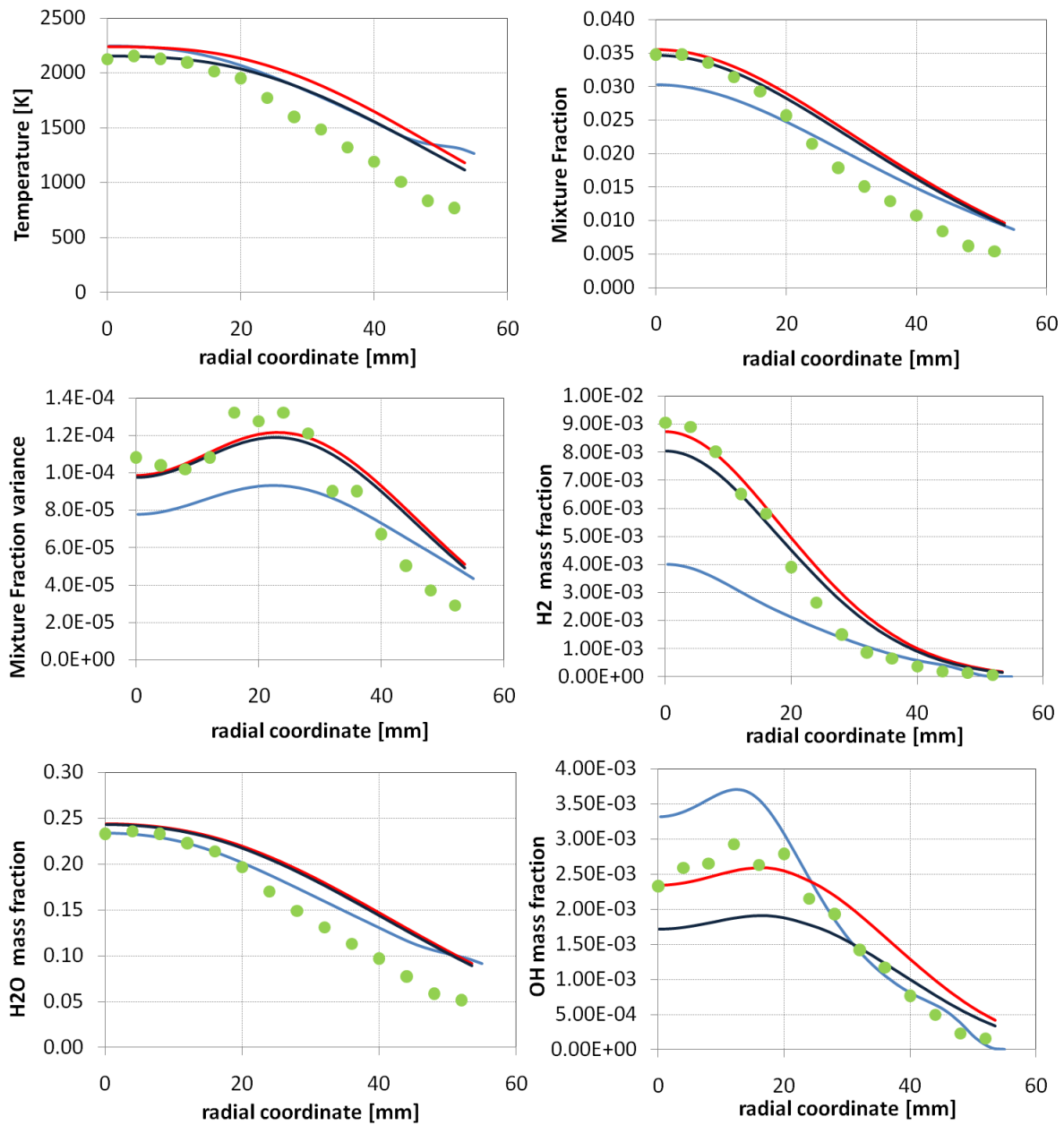


Figura B.11: Fiamma $H_2/He0$. Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 422 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [2]

B. RISULTATI AGGIUNTIVI

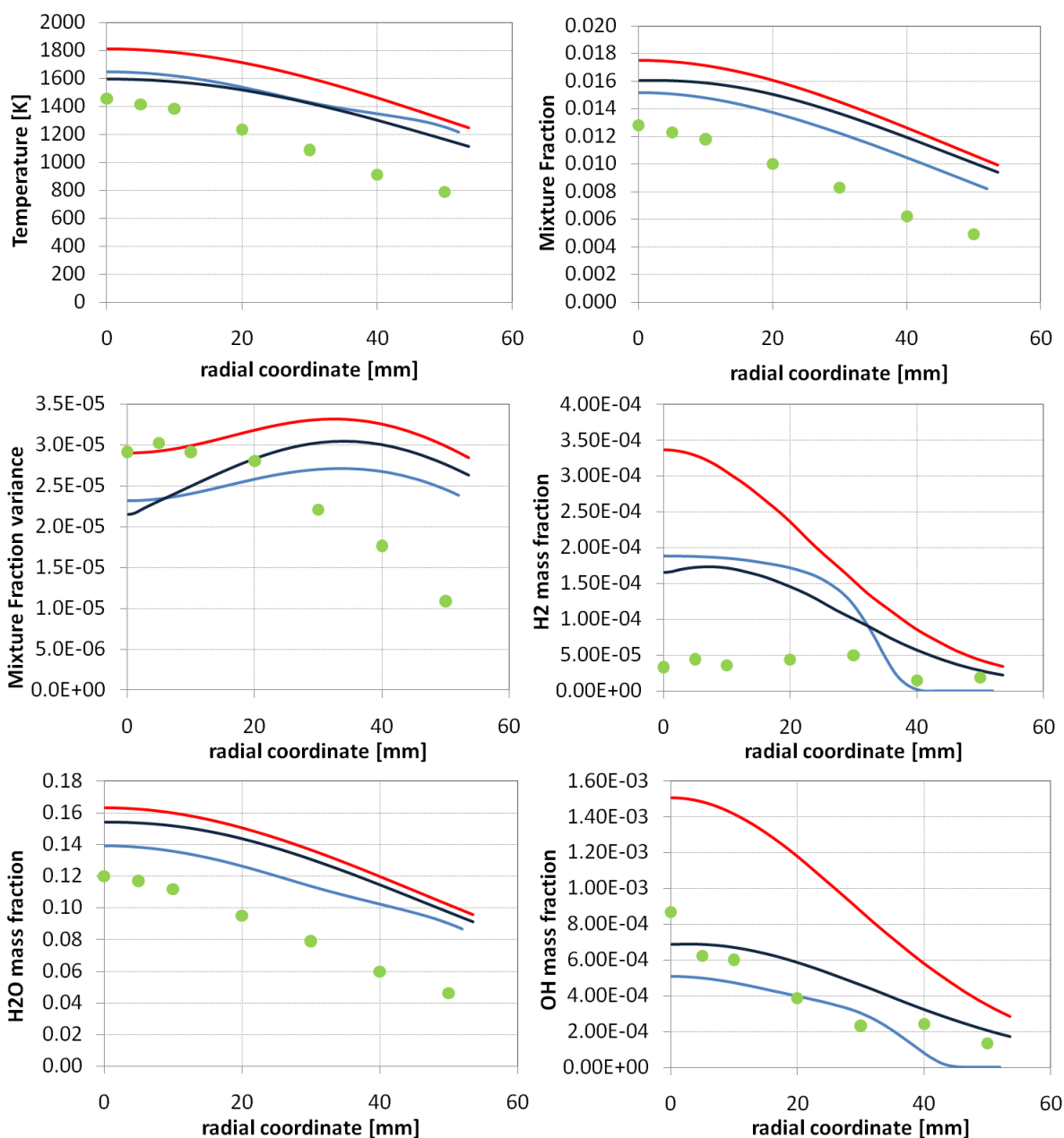


Figura B.12: Fiamma H_2/HeO . Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 675 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [2]

B.2.2 Fiamma Sandia H_2/He_2

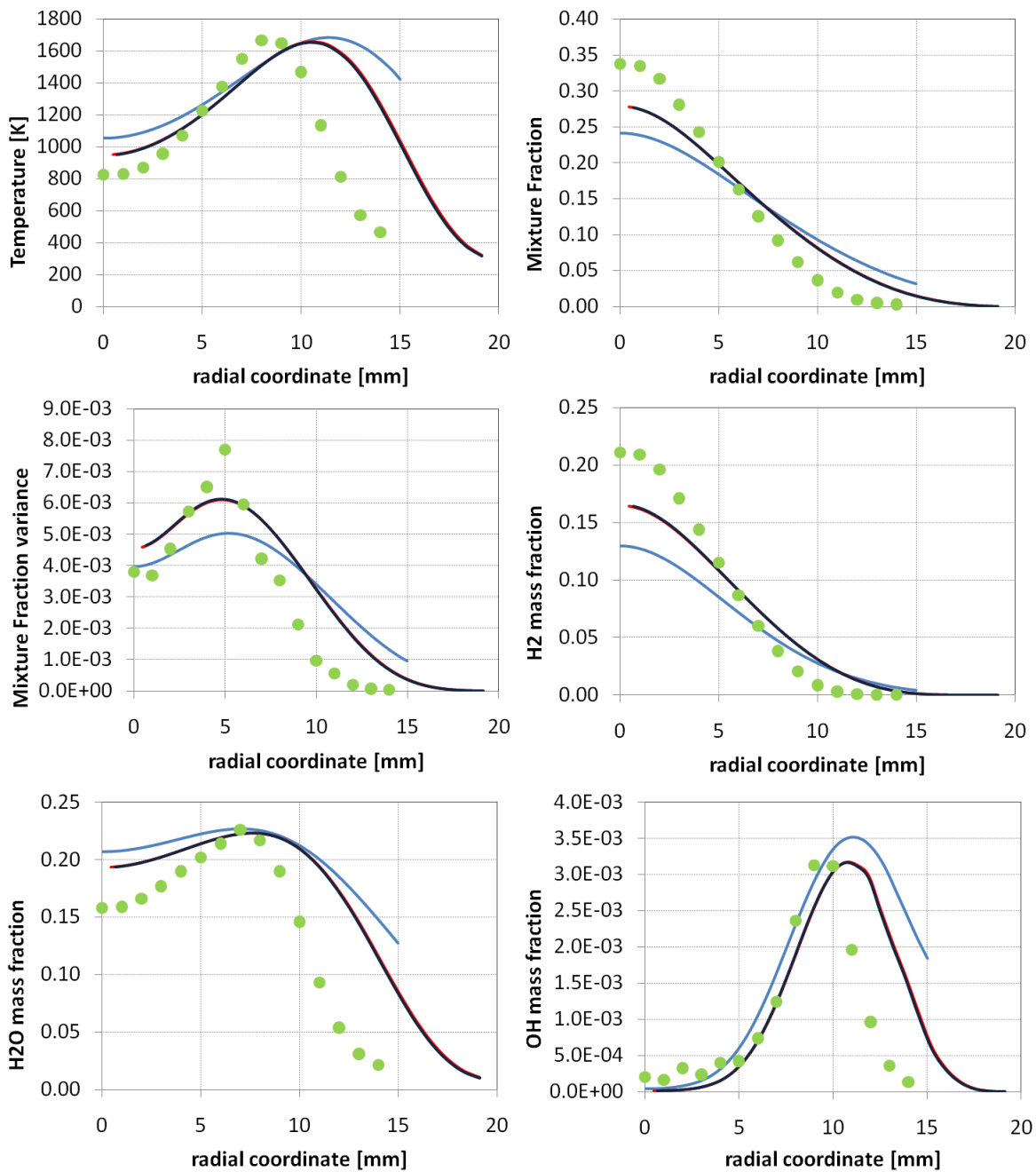


Figura B.13: Fiamma H_2/He_2 . Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 70 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [2]

B. RISULTATI AGGIUNTIVI

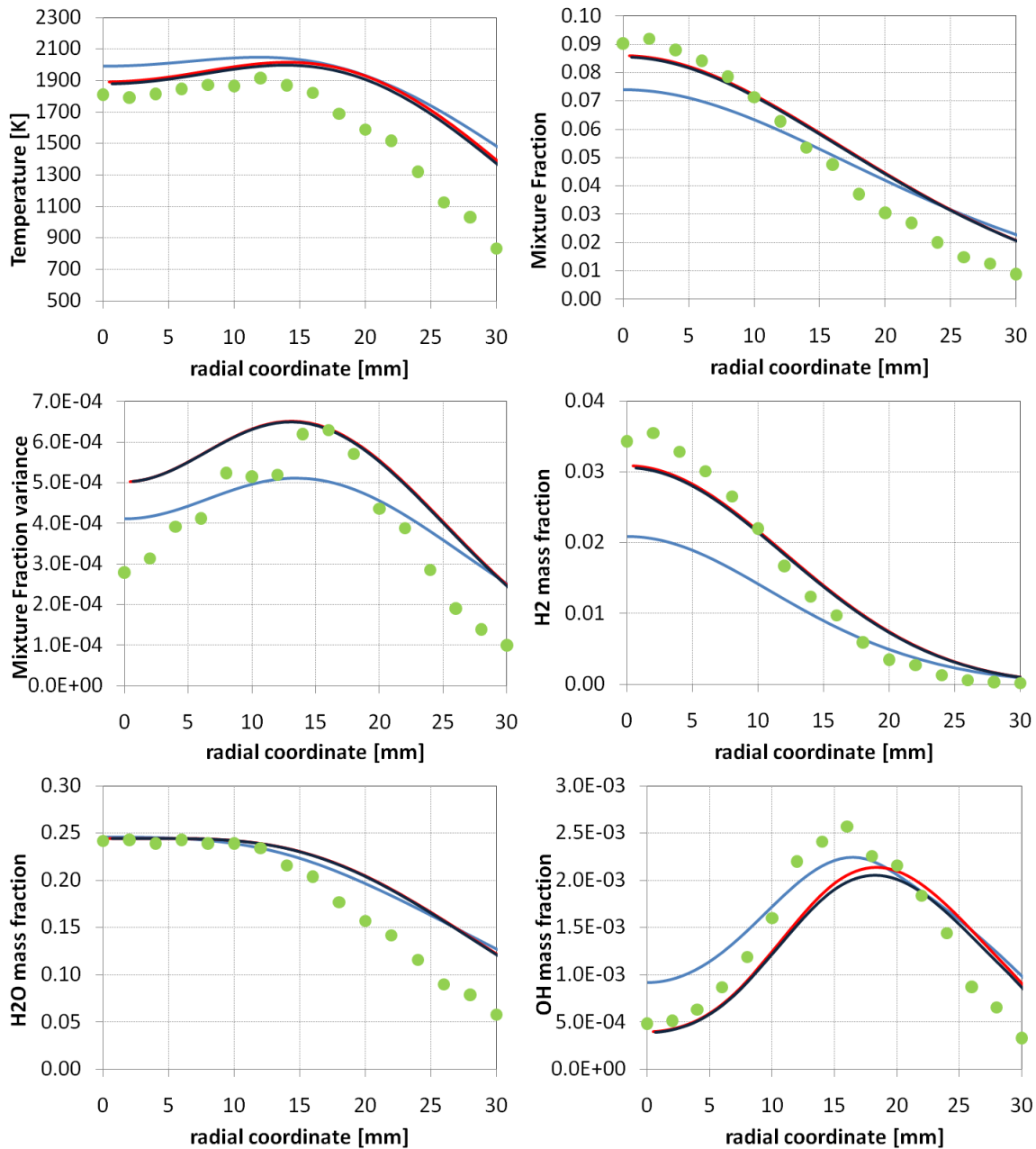


Figura B.14: Fiamma H_2/He_2 . Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 211 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [2]

B. RISULTATI AGGIUNTIVI

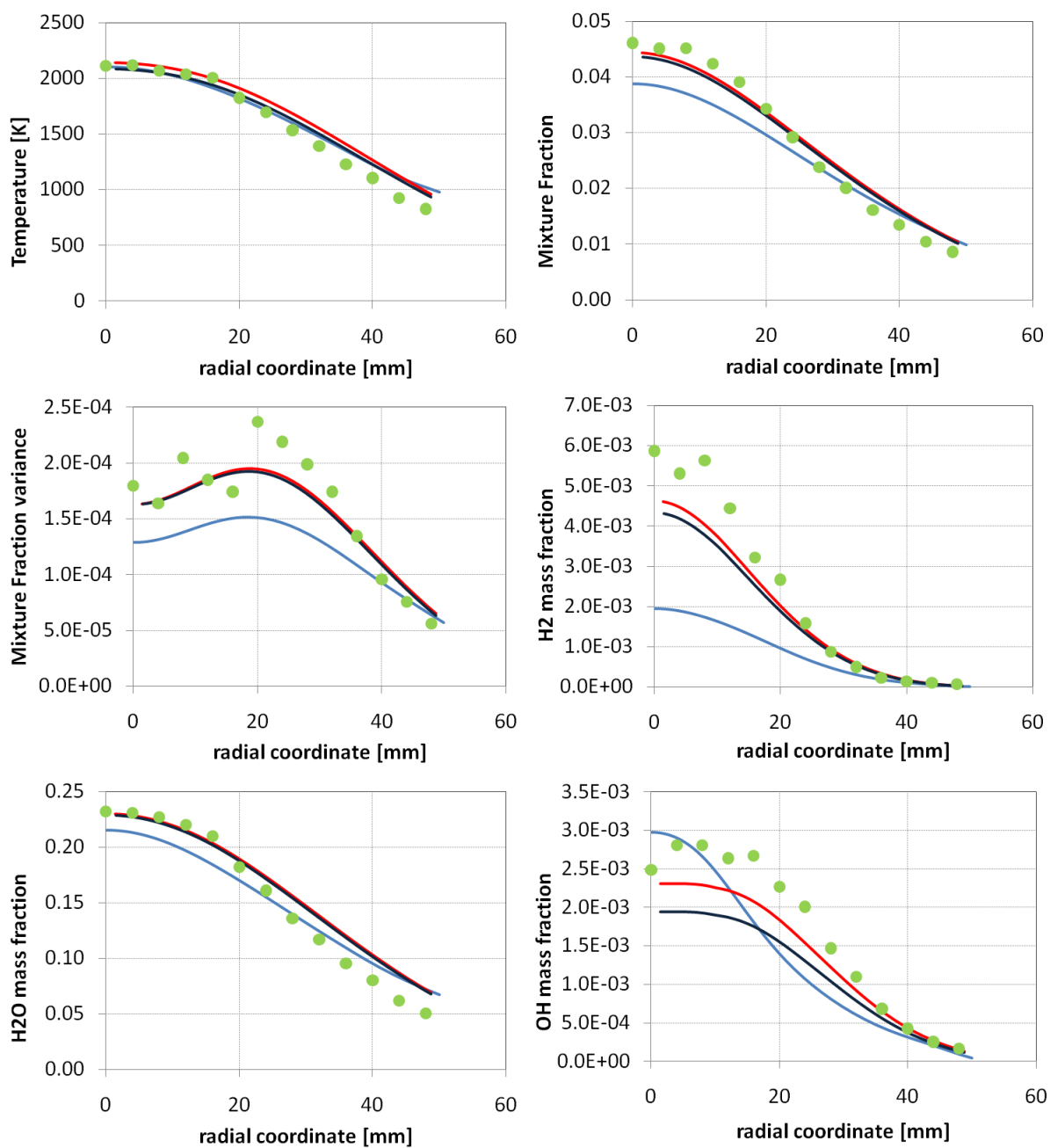


Figura B.15: Fiamma H_2/He_2 . Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 352 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [2]

B. RISULTATI AGGIUNTIVI

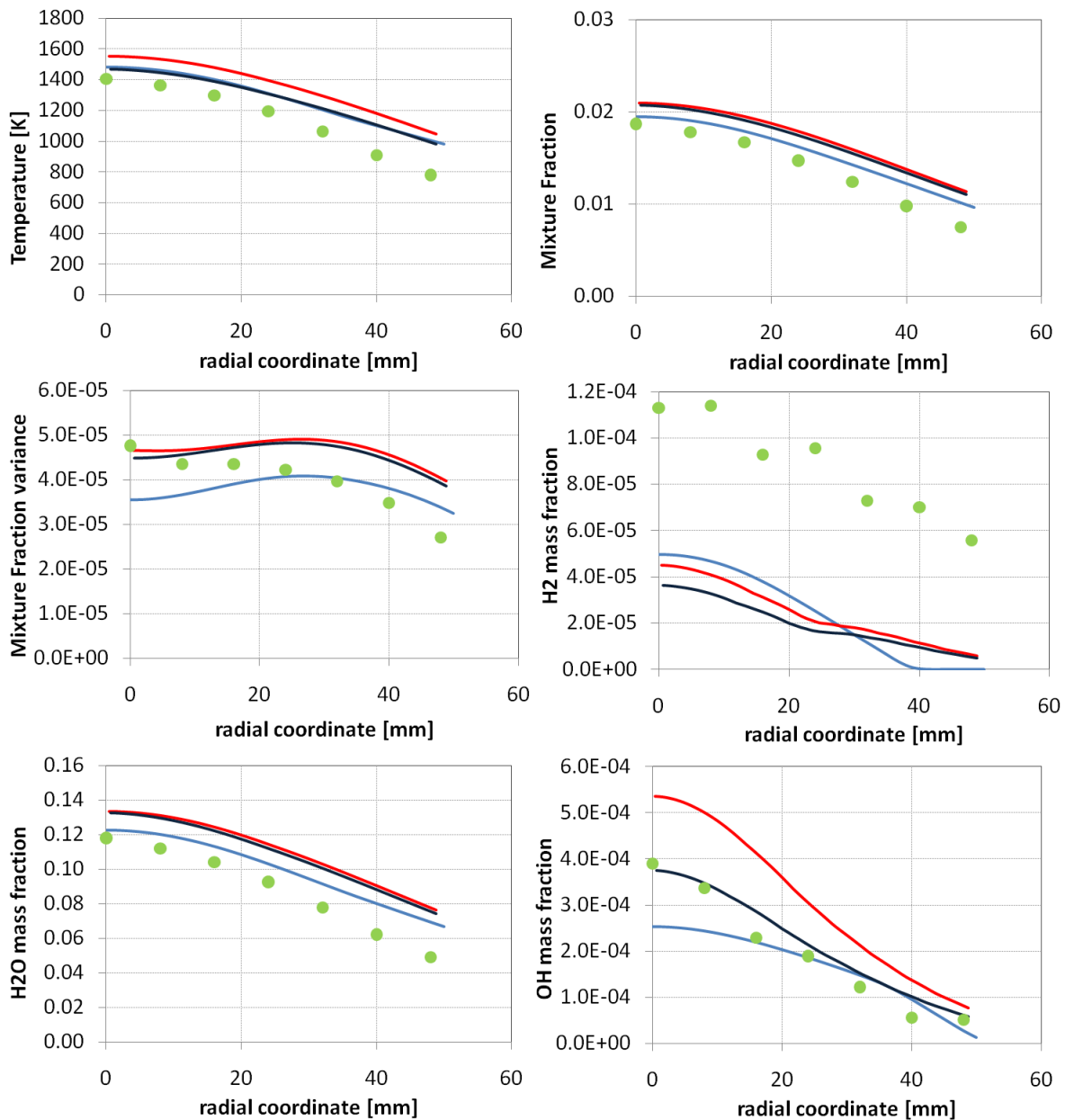


Figura B.16: Fiamma H_2/He_2 . Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 562.5 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [2]

B.2.3 Fiamma Sandia H_2/He_4

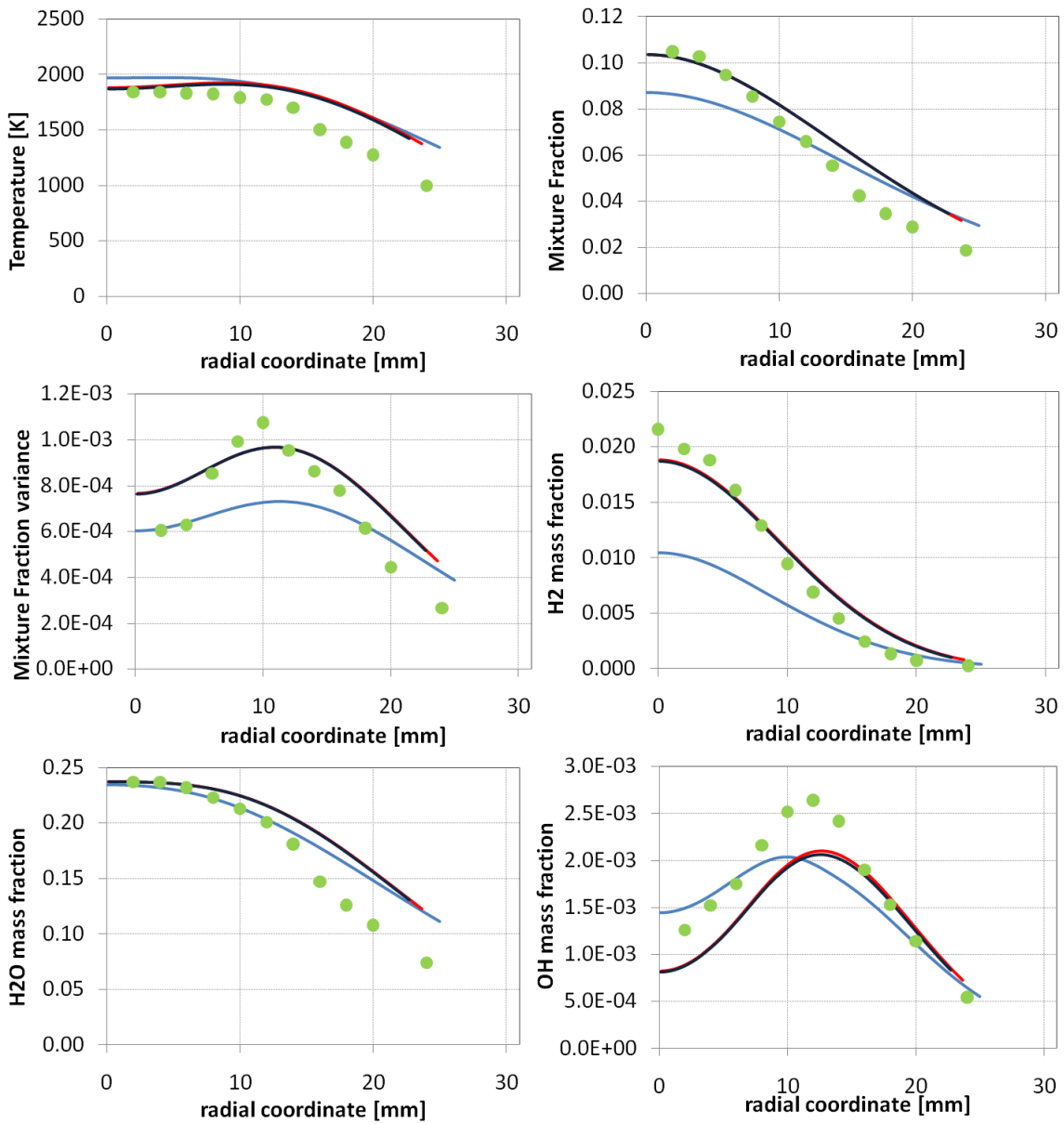


Figura B.17: Fiamma H_2/He_4 . Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 187.5 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [2]

B. RISULTATI AGGIUNTIVI

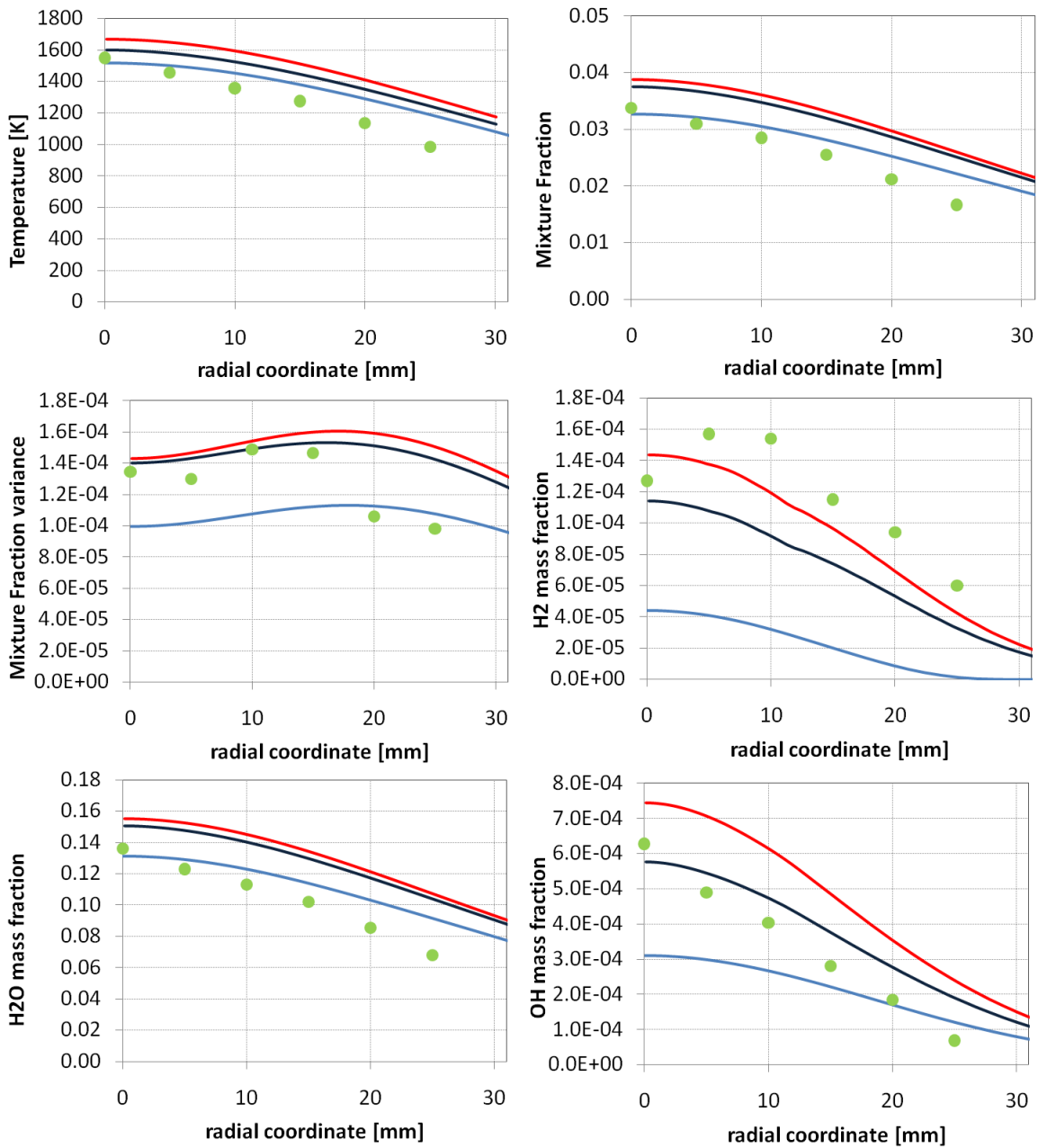


Figura B.18: Fiamma H_2/He_4 . Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 375 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [2]

B.2.4 Fiamma Sandia $CH_4/H_2/N_2$ A

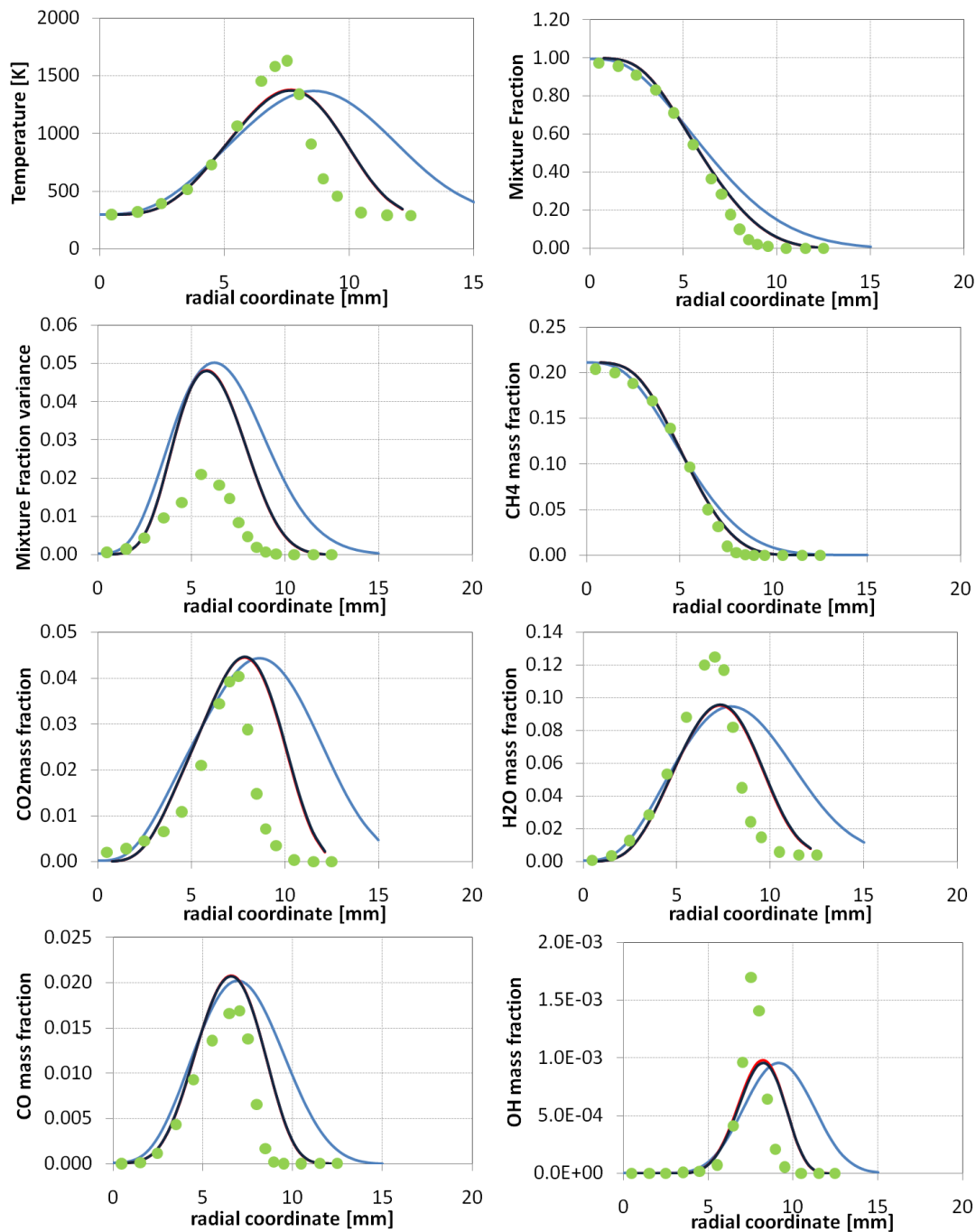


Figura B.19: Fiamma $CH_4/H_2/N_2$ A. Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 40 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [47]

B. RISULTATI AGGIUNTIVI

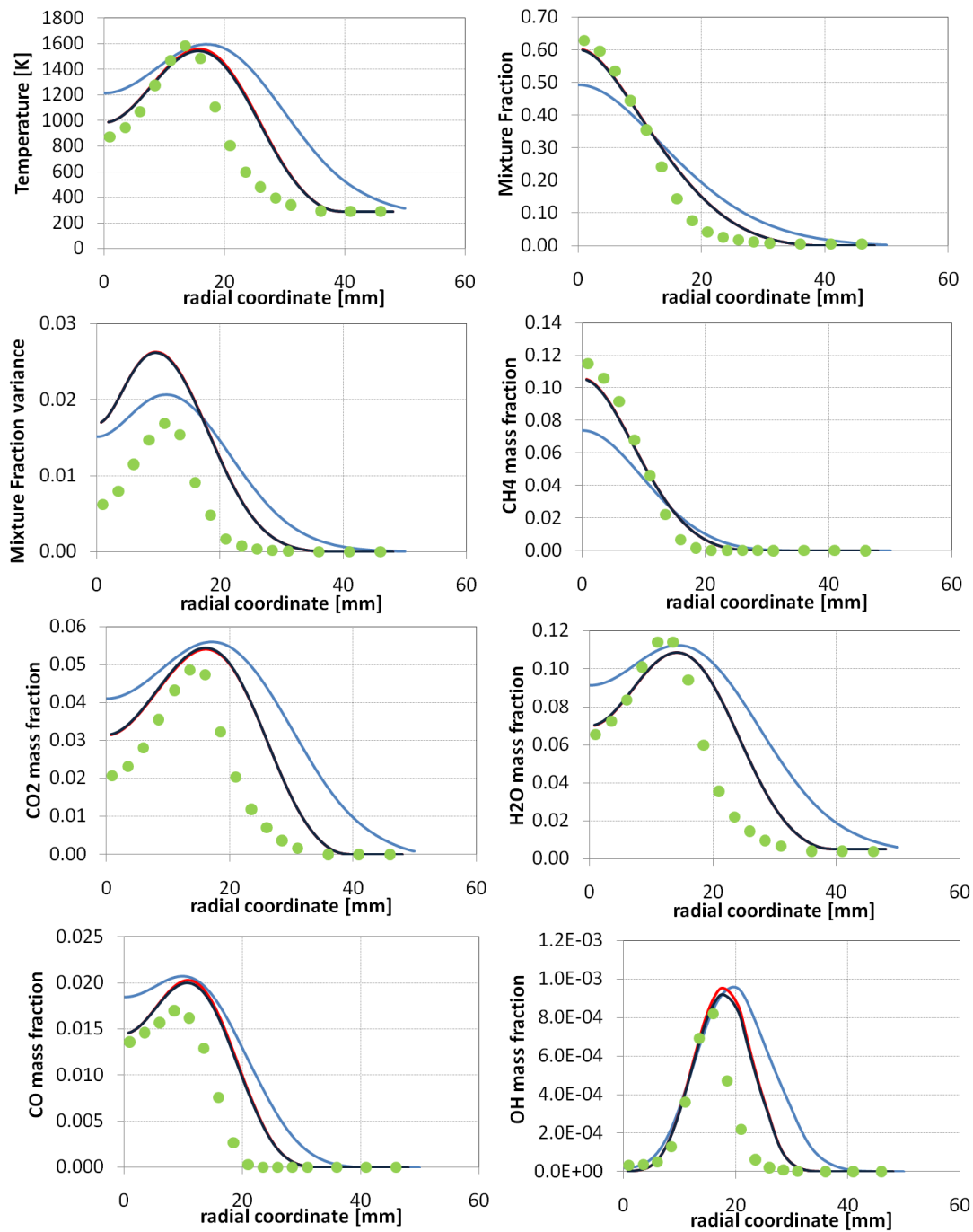


Figura B.20: Fiamma $CH_4/H_2/N_2$ A. Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 160 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [47]

B. RISULTATI AGGIUNTIVI

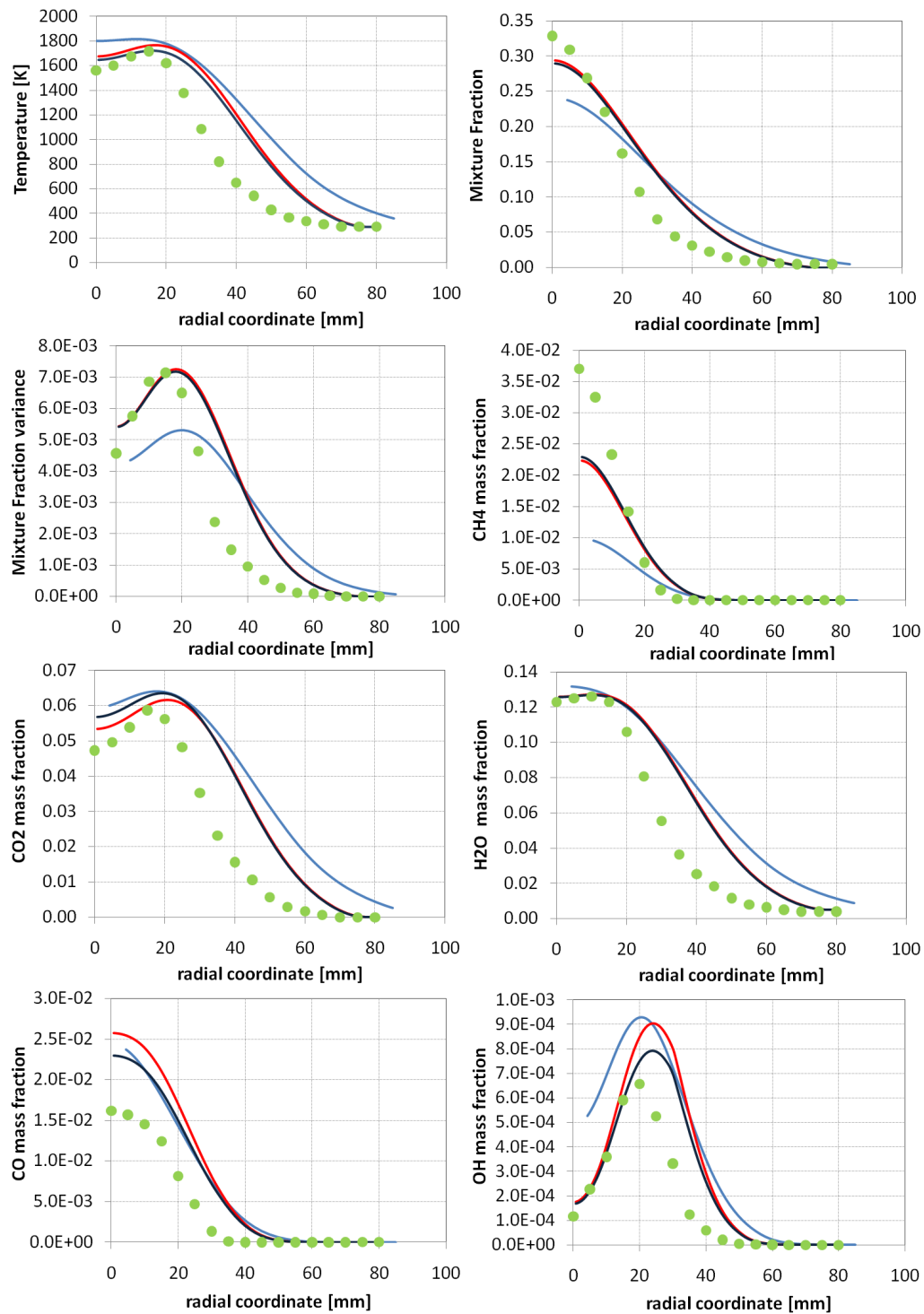


Figura B.21: Fiamma $CH_4/H_2/N_2$ A. Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 320 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [47]

B. RISULTATI AGGIUNTIVI

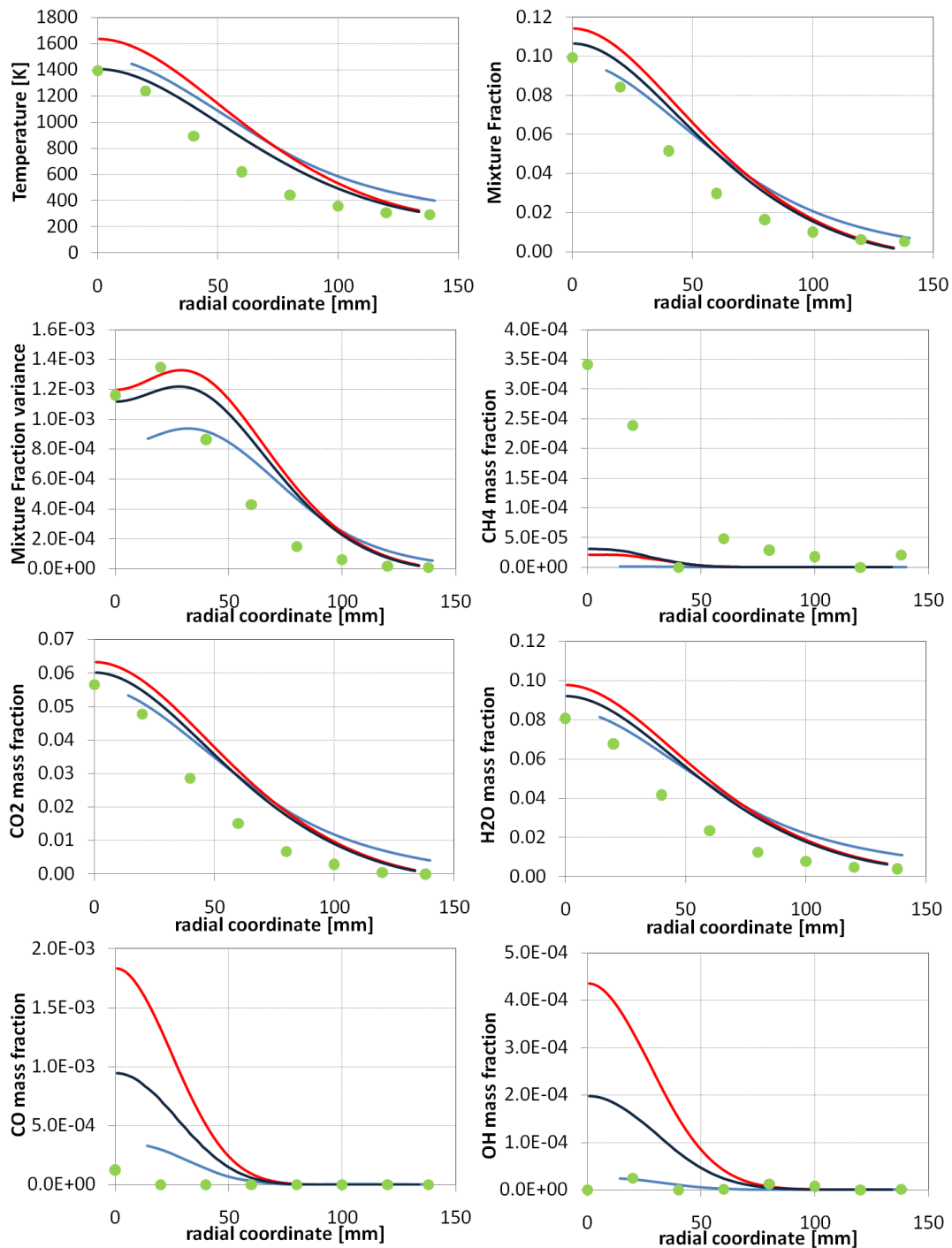


Figura B.22: Fiamma $CH_4/H_2/N_2$ A. Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 640 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [47]

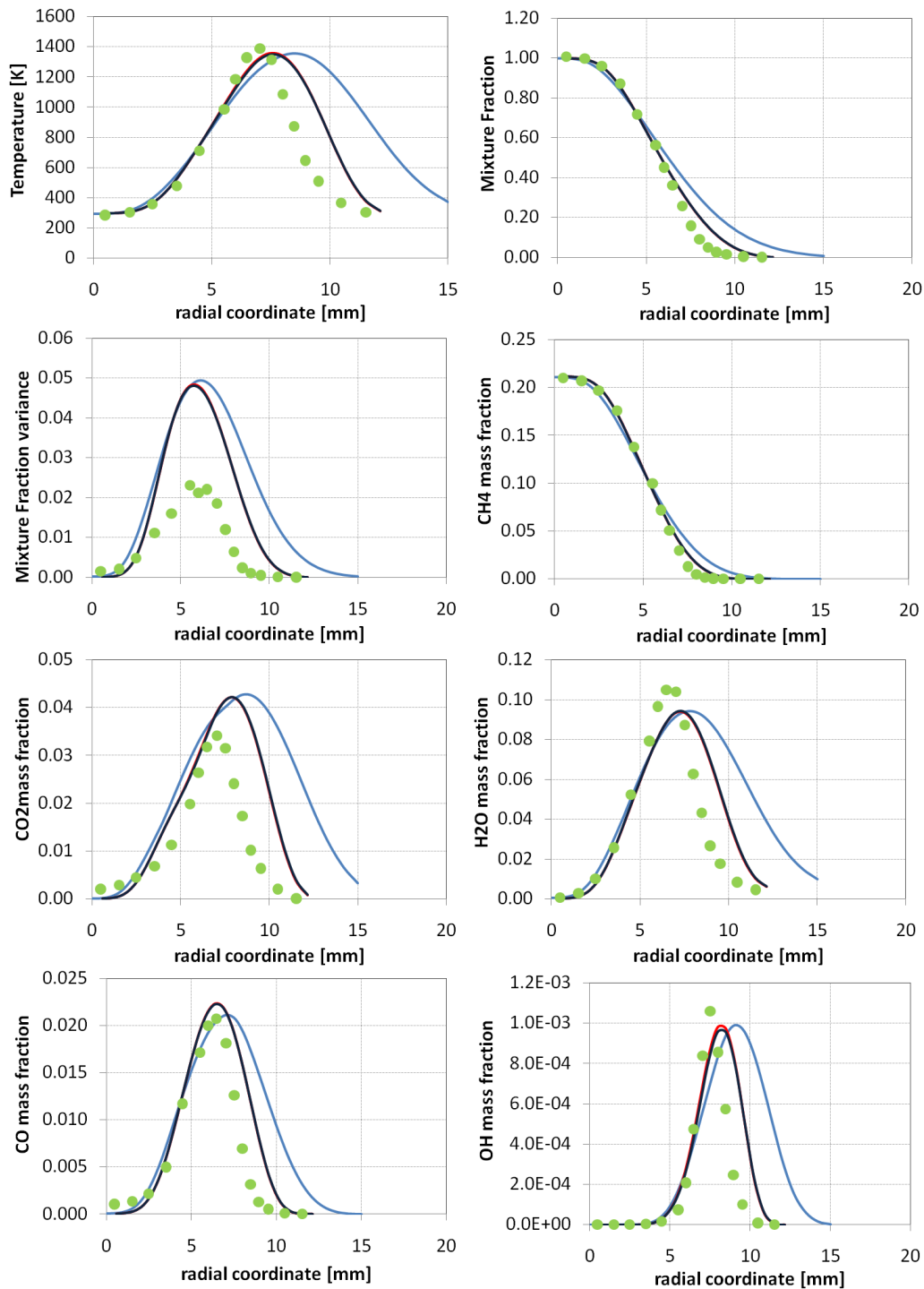
B.2.5 Fiamma Sandia $CH_4/H_2/N_2$ B

Figura B.23: Fiamma $CH_4/H_2/N_2$ B. Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 40 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [47]

B. RISULTATI AGGIUNTIVI

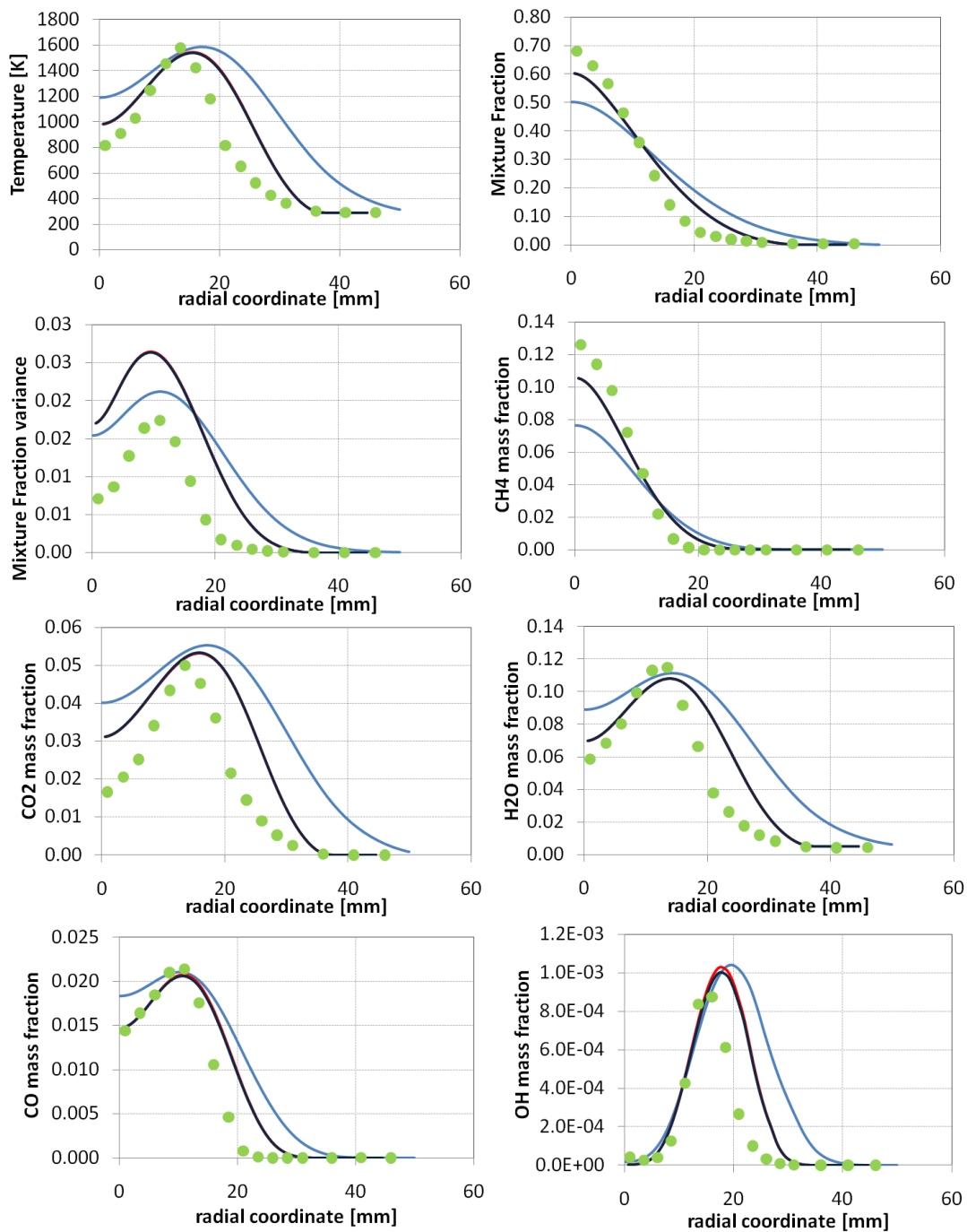


Figura B.24: Fiamma $CH_4/H_2/N_2$ B. Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 160 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [47]

B. RISULTATI AGGIUNTIVI

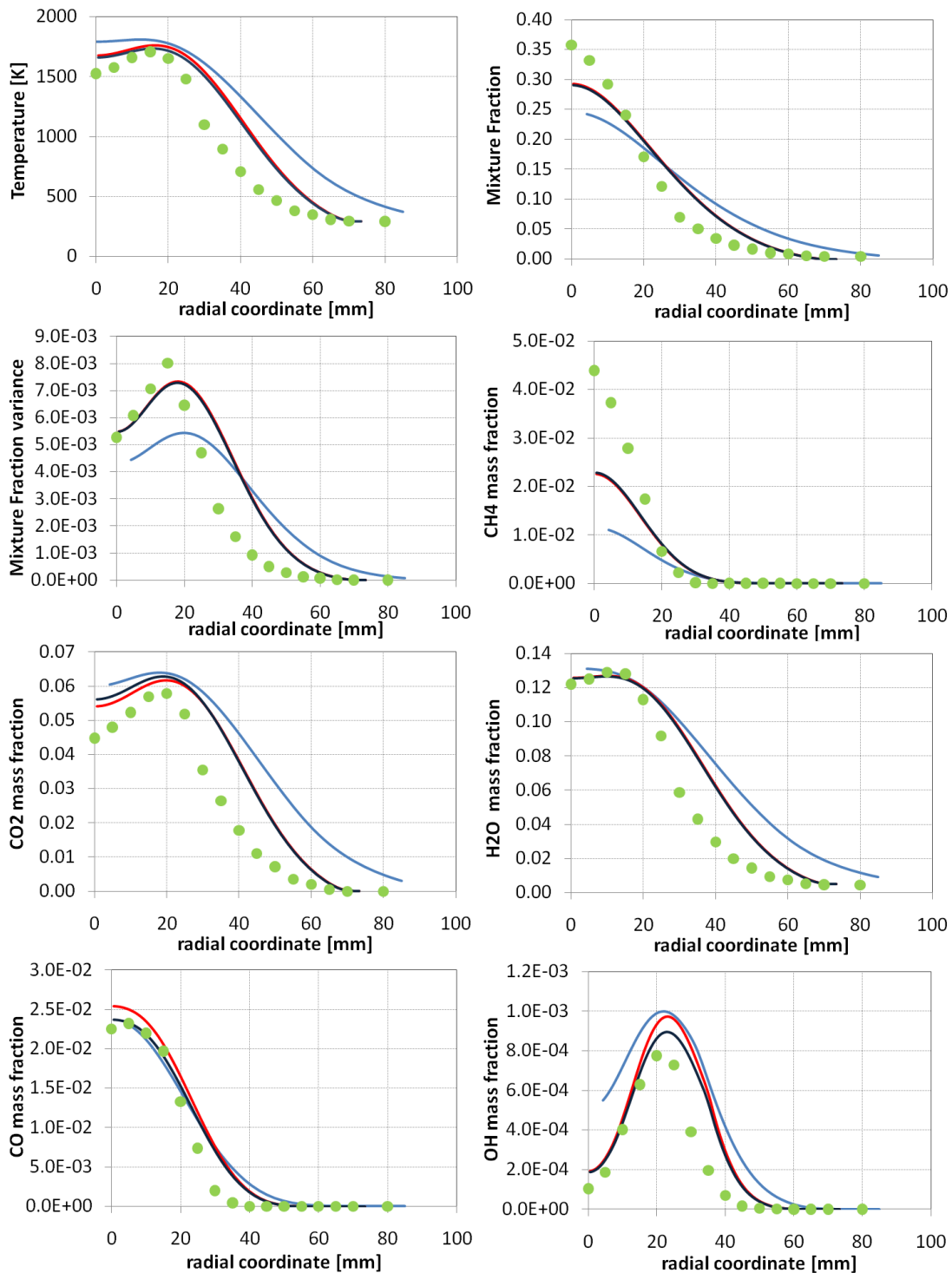


Figura B.25: Fiamma $CH_4/H_2/N_2$ B. Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 320 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [47]

B. RISULTATI AGGIUNTIVI

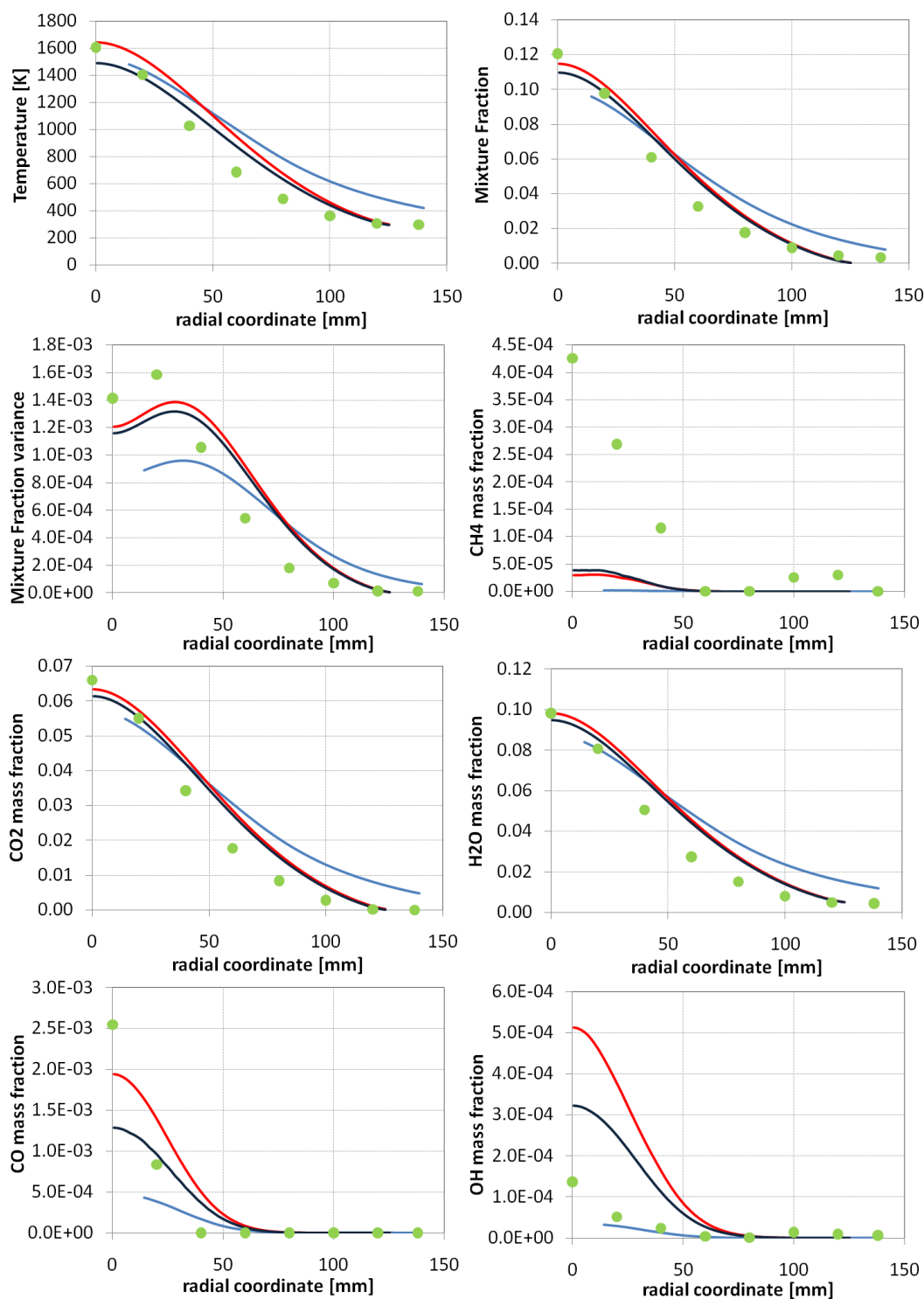


Figura B.26: Fiamma $CH_4/H_2/N_2$ B. Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 640 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [47]

B.2.6 Fiamma Sandia $CO/H_2/N_2$ A

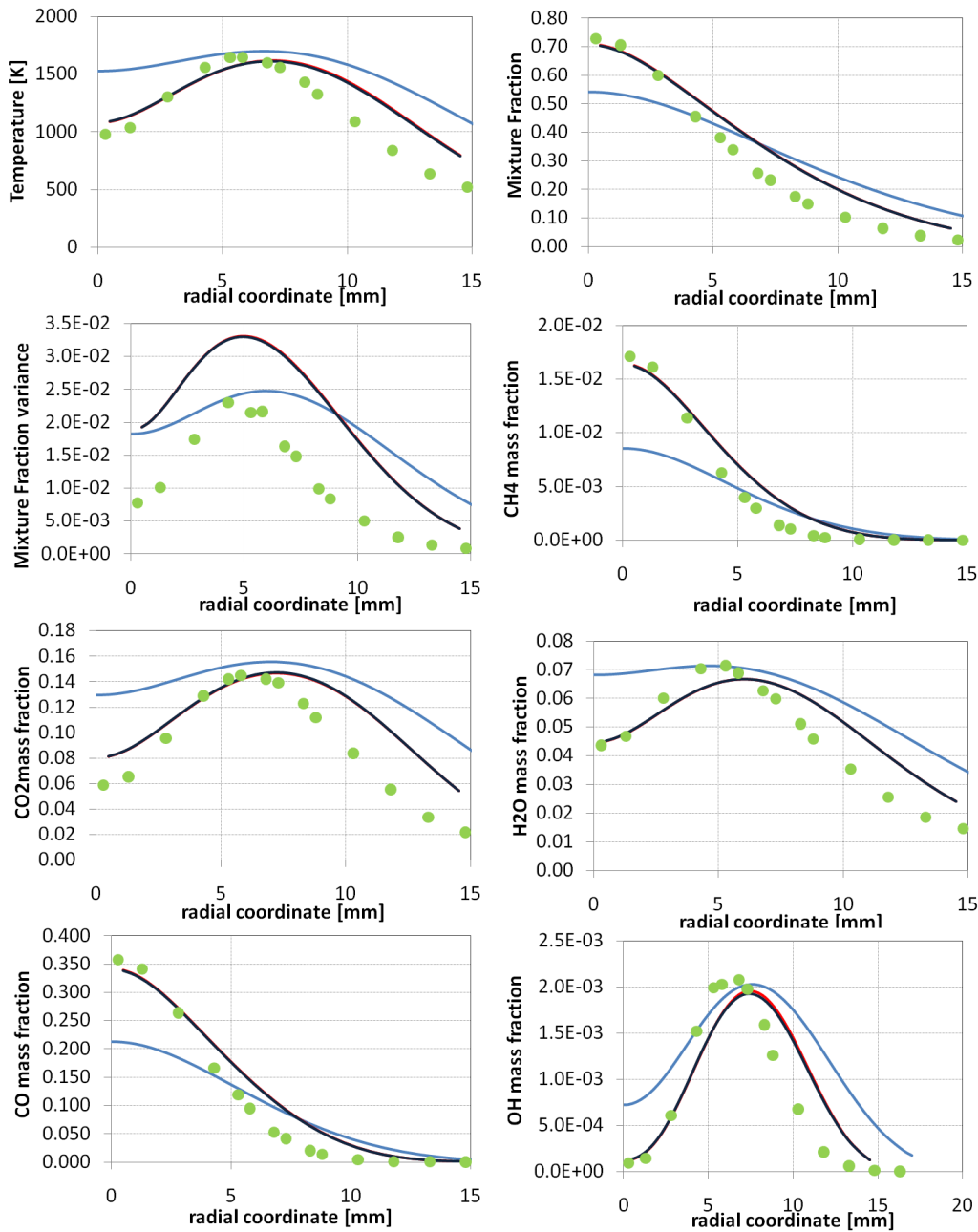


Figura B.27: Fiamma $CO/H_2/N_2$ A. Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 92 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [4]

B. RISULTATI AGGIUNTIVI

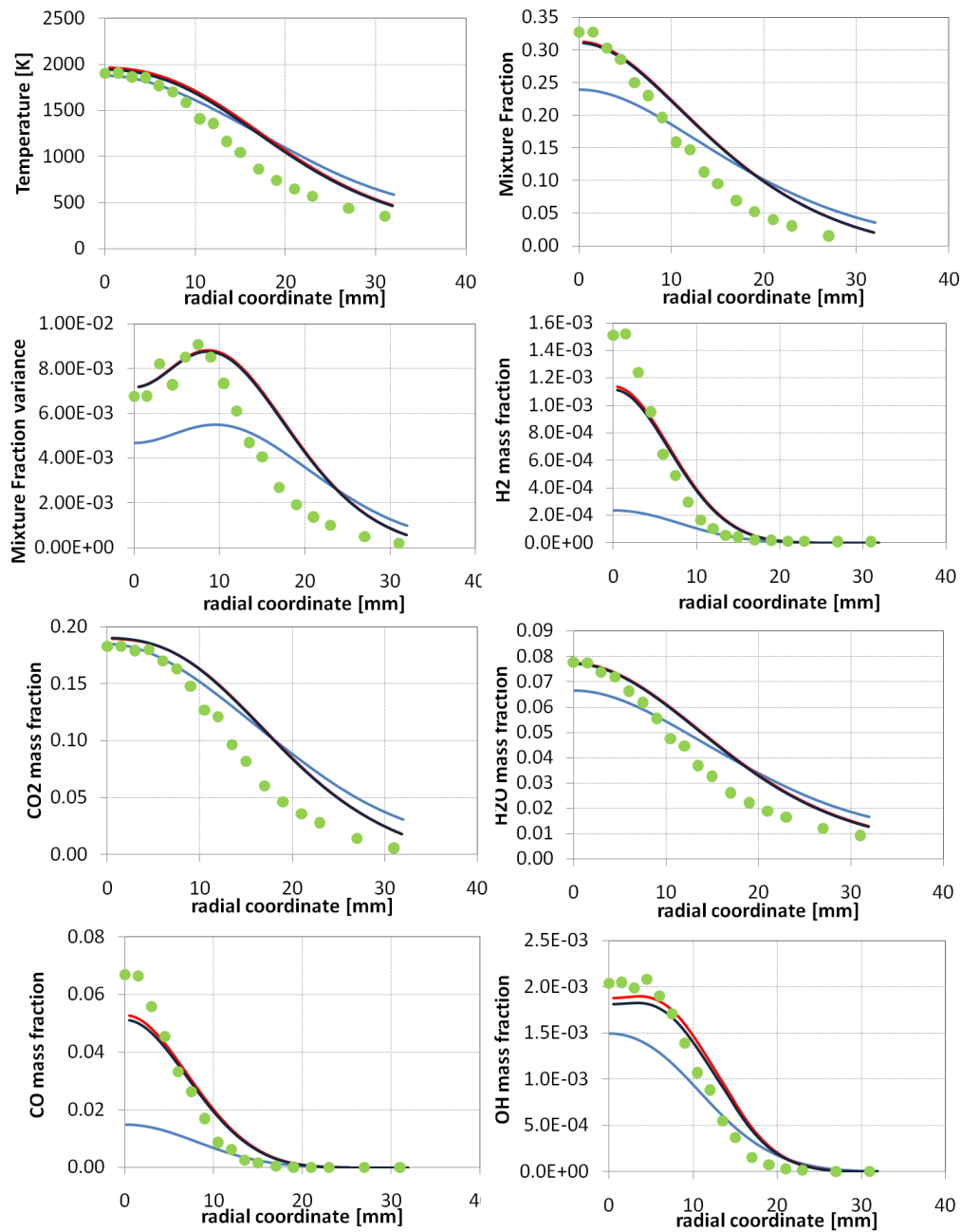


Figura B.28: Fiamma $CO/H_2/N_2$ A. Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 183 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [4]

B. RISULTATI AGGIUNTIVI

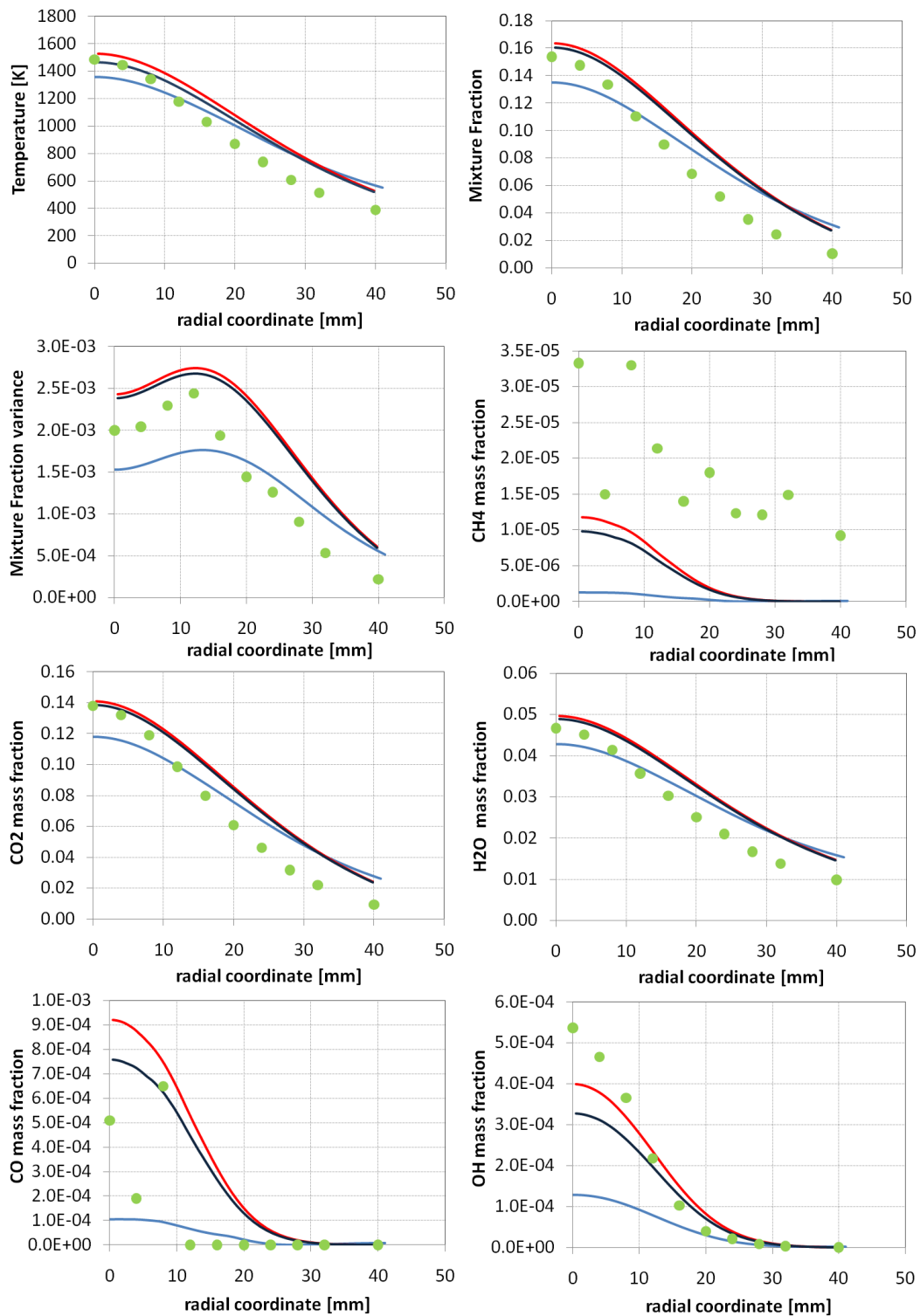


Figura B.29: Fiamma $CO/H_2/N_2$ A. Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 275 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [4]

B.2.7 Fiamma Sandia *Piloted CH₄/Air D*

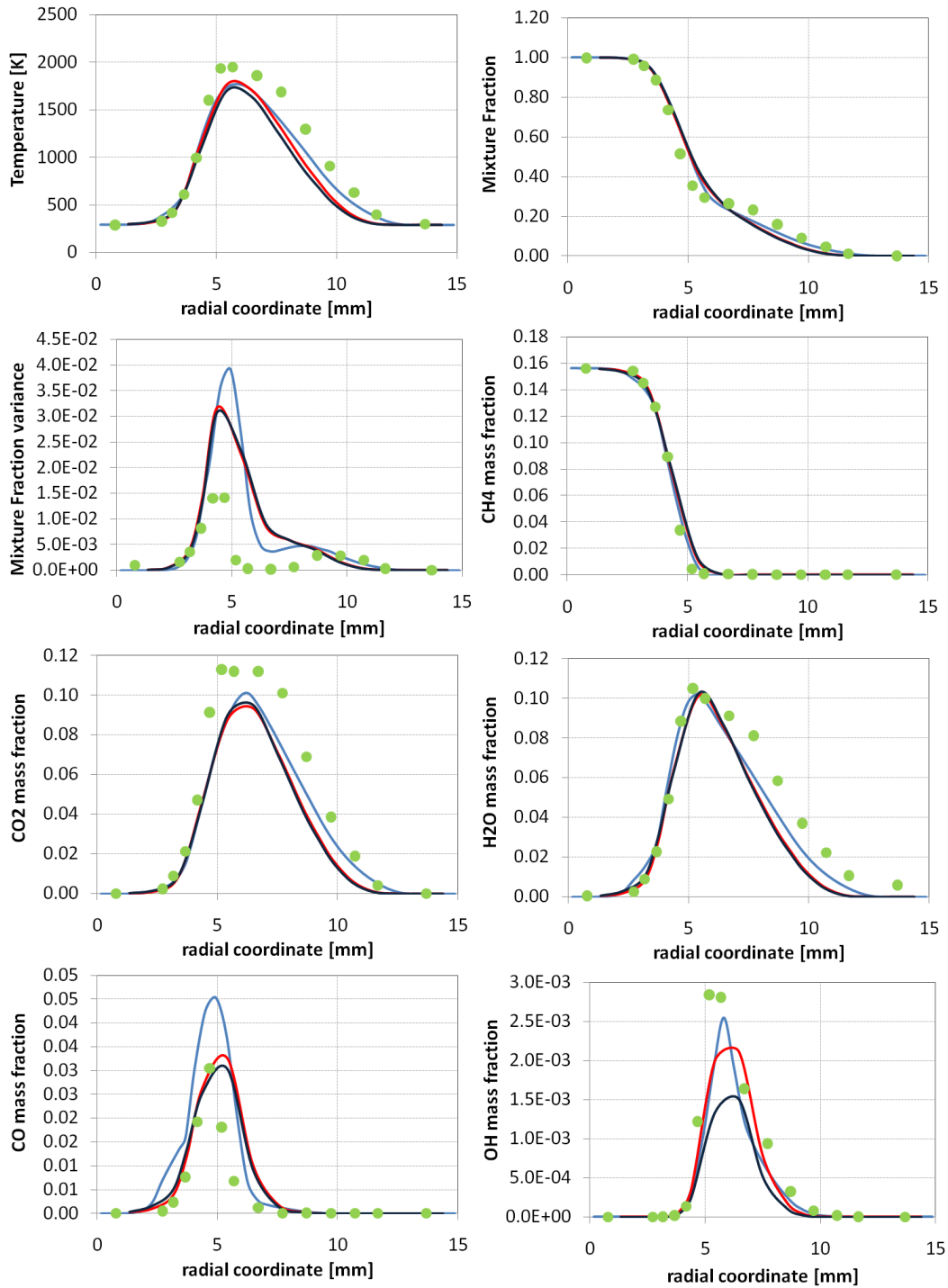


Figura B.30: Fiamma *Piloted CH₄/Air D*. Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 14 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde)[3]

B. RISULTATI AGGIUNTIVI

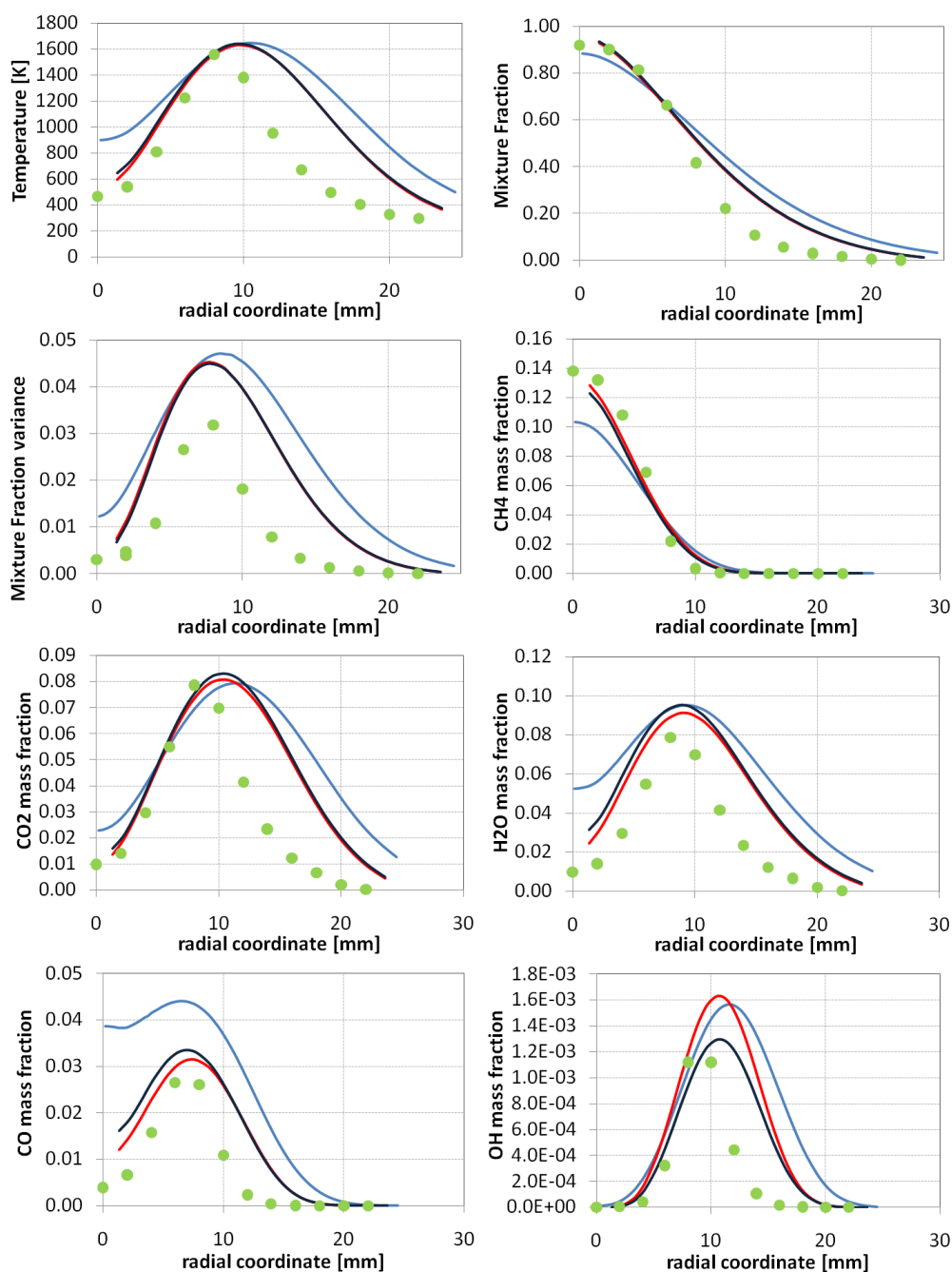


Figura B.31: Fiamma *Piloted CH₄/Air D*. Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 108 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [3]

B. RISULTATI AGGIUNTIVI

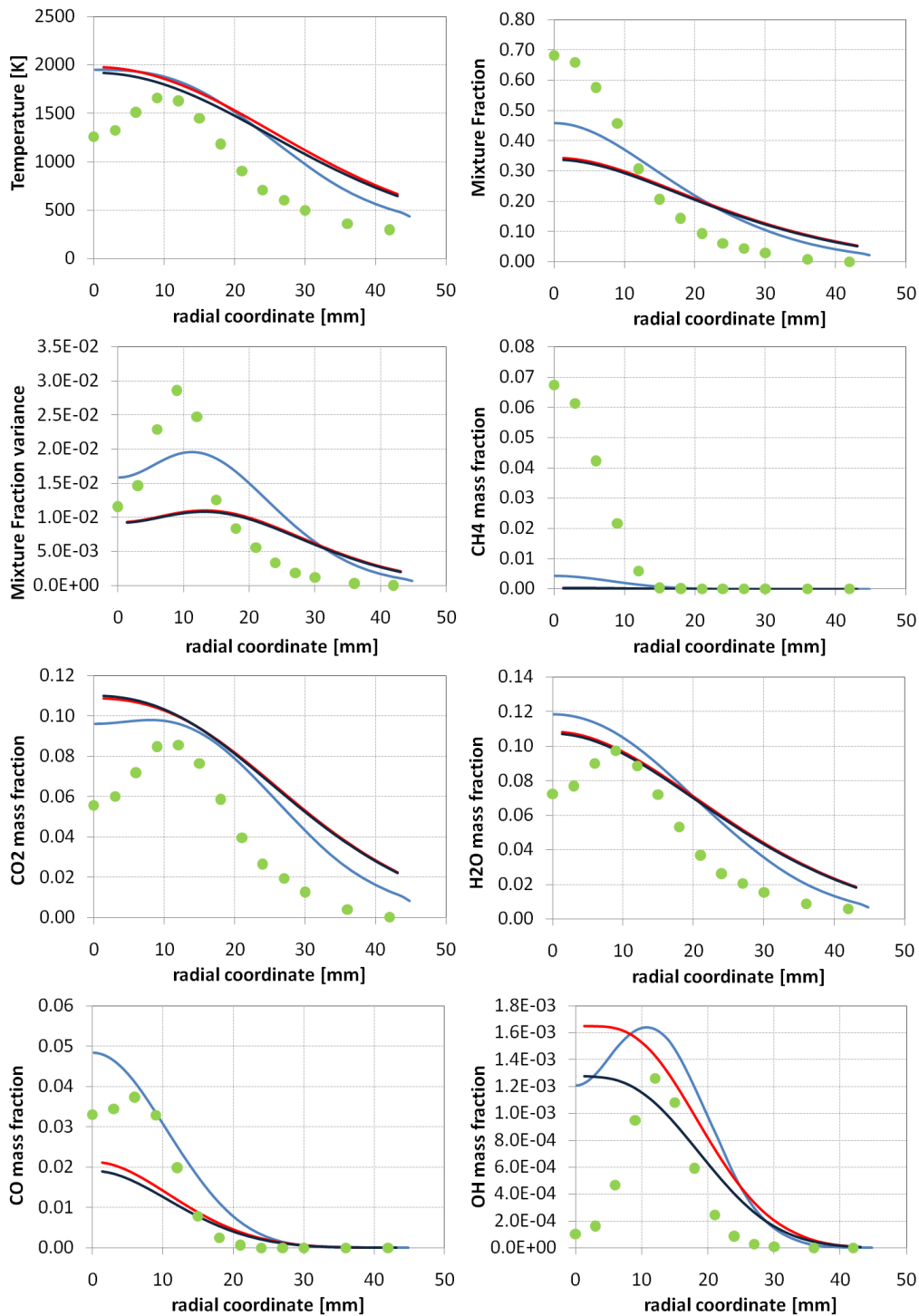


Figura B.32: Fiamma *Piloted CH_4/Air D*. Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 216 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [3]

B. RISULTATI AGGIUNTIVI

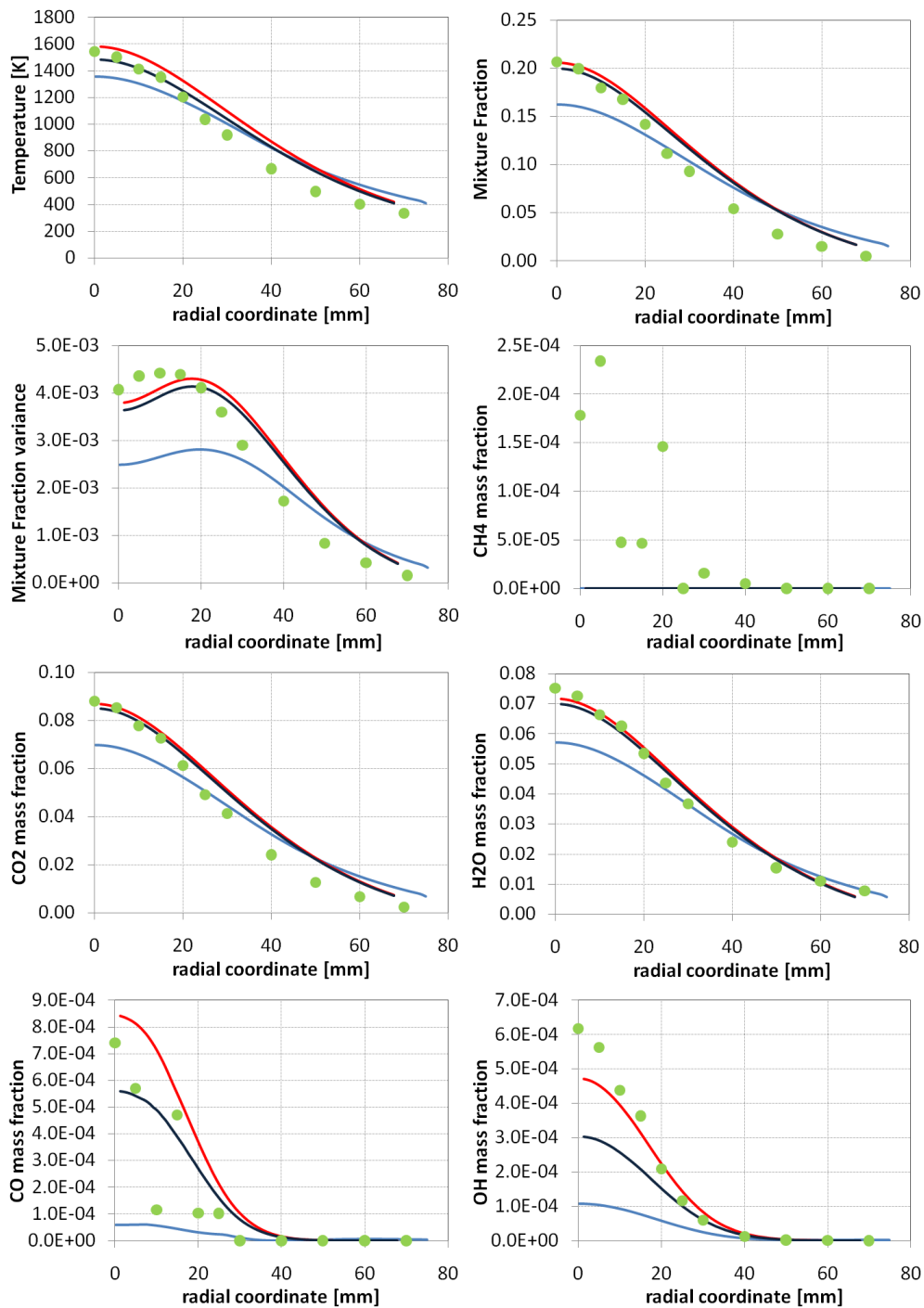


Figura B.33: Fiamma *Piloted CH₄/Air D*. Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 432 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [3]

B.2.8 Fiamma Sandia *Bluff-Body HM1*

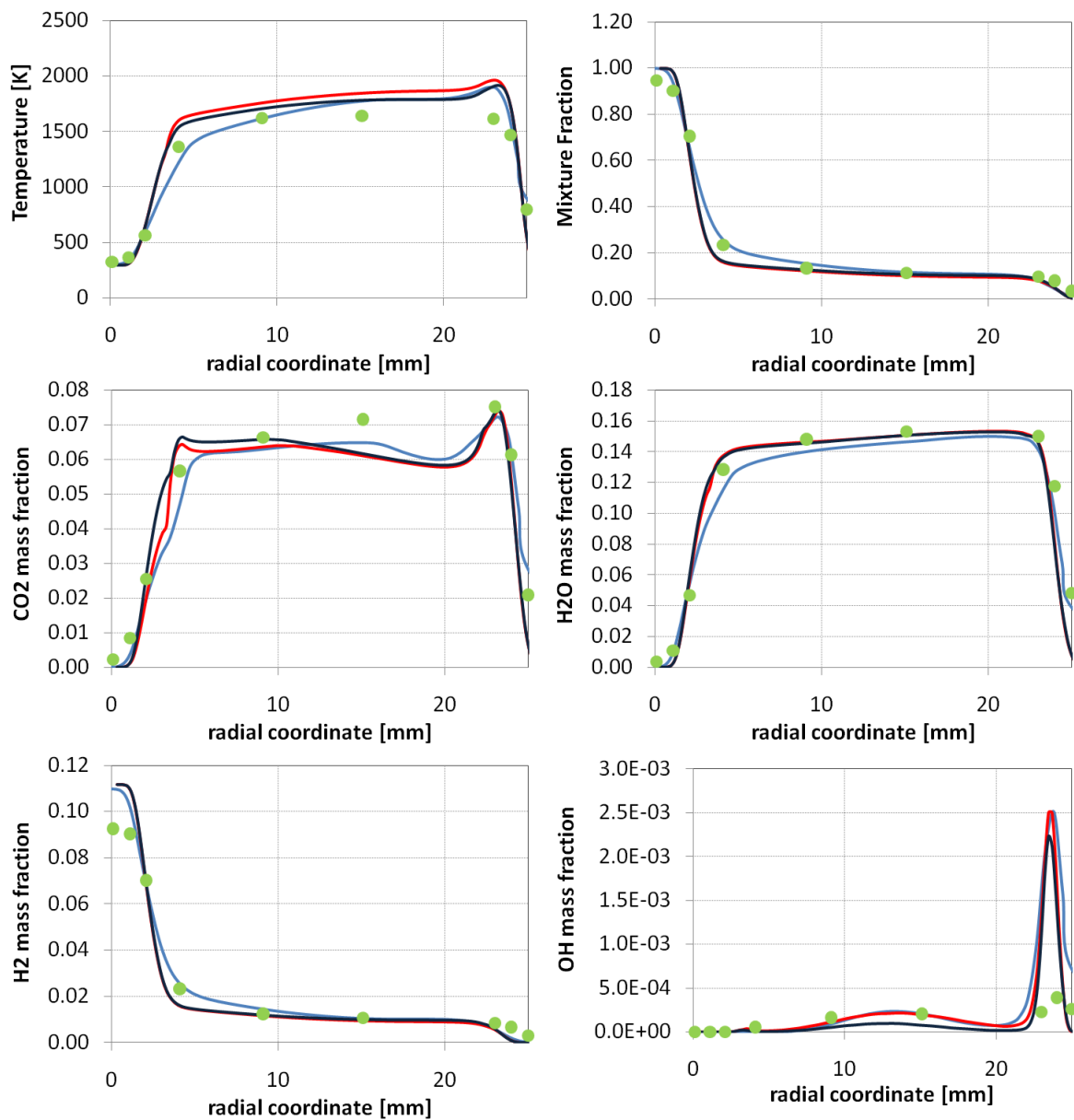


Figura B.34: Fiamma *Bluff-Body HM1*. Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 13 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [1]

B. RISULTATI AGGIUNTIVI

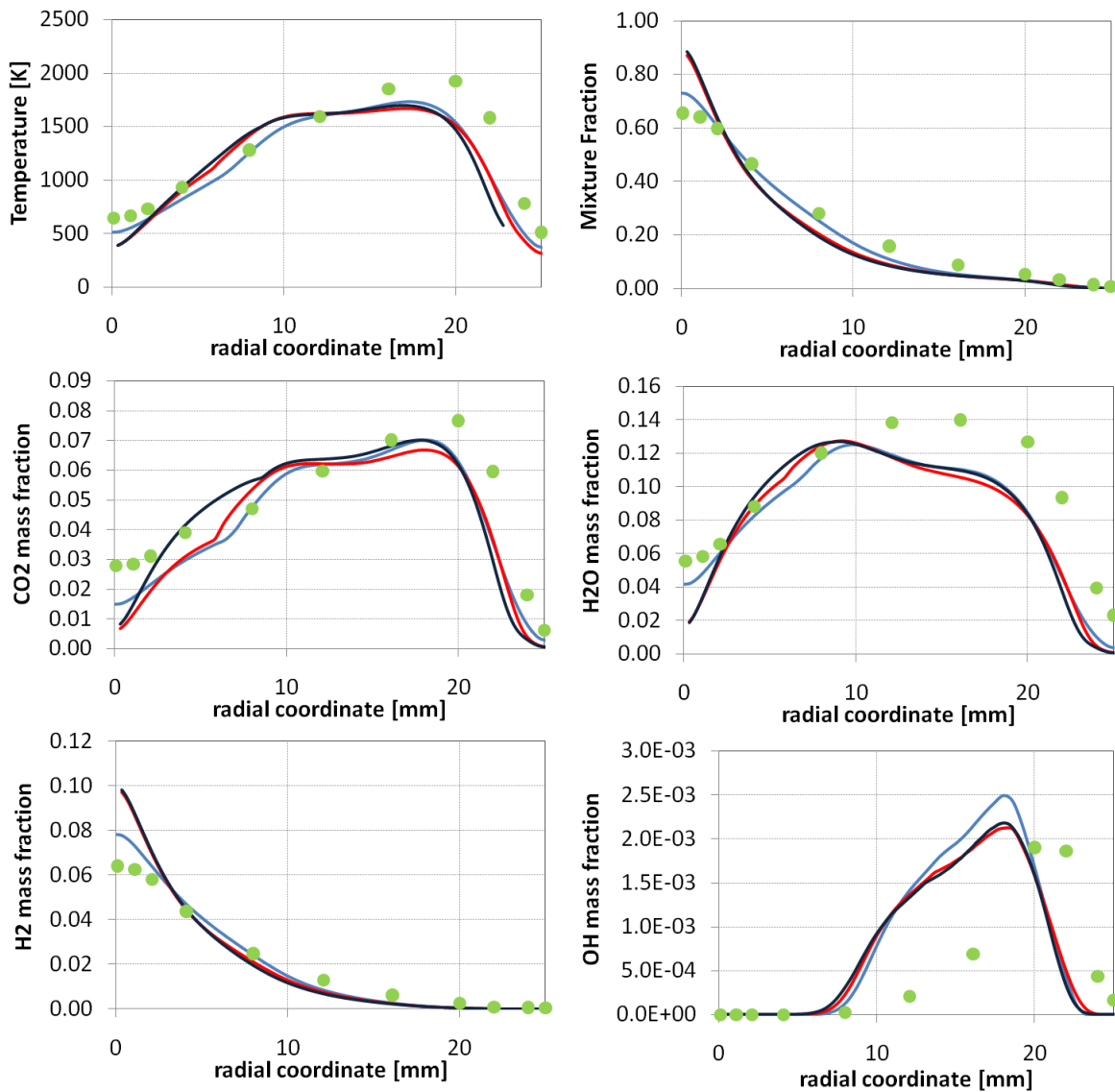


Figura B.35: Fiamma *Bluff-Body HM1*. Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 45 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [1]

B. RISULTATI AGGIUNTIVI

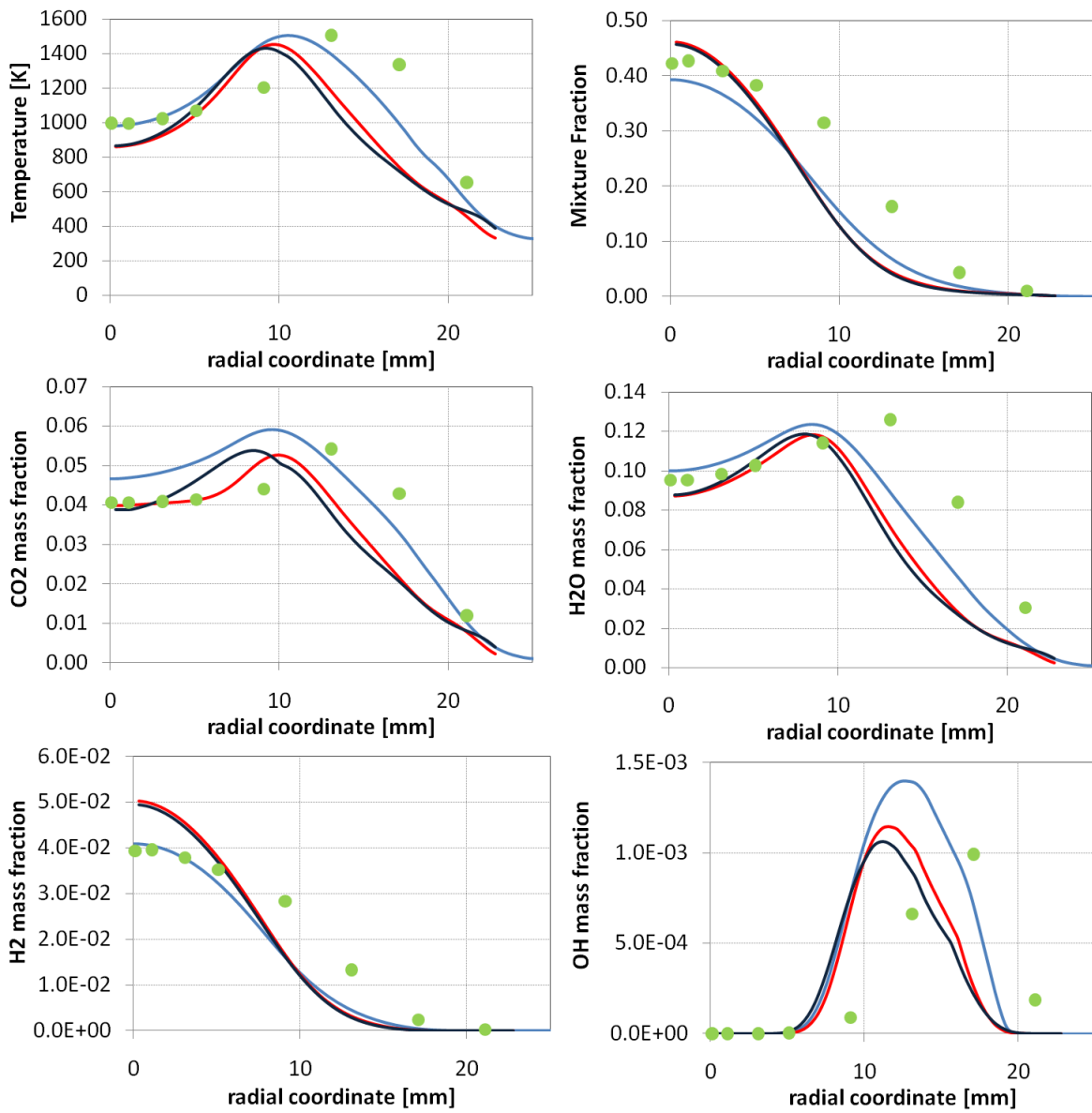


Figura B.36: Fiamma *Bluff-Body HM1*. Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 90 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [1]

B. RISULTATI AGGIUNTIVI

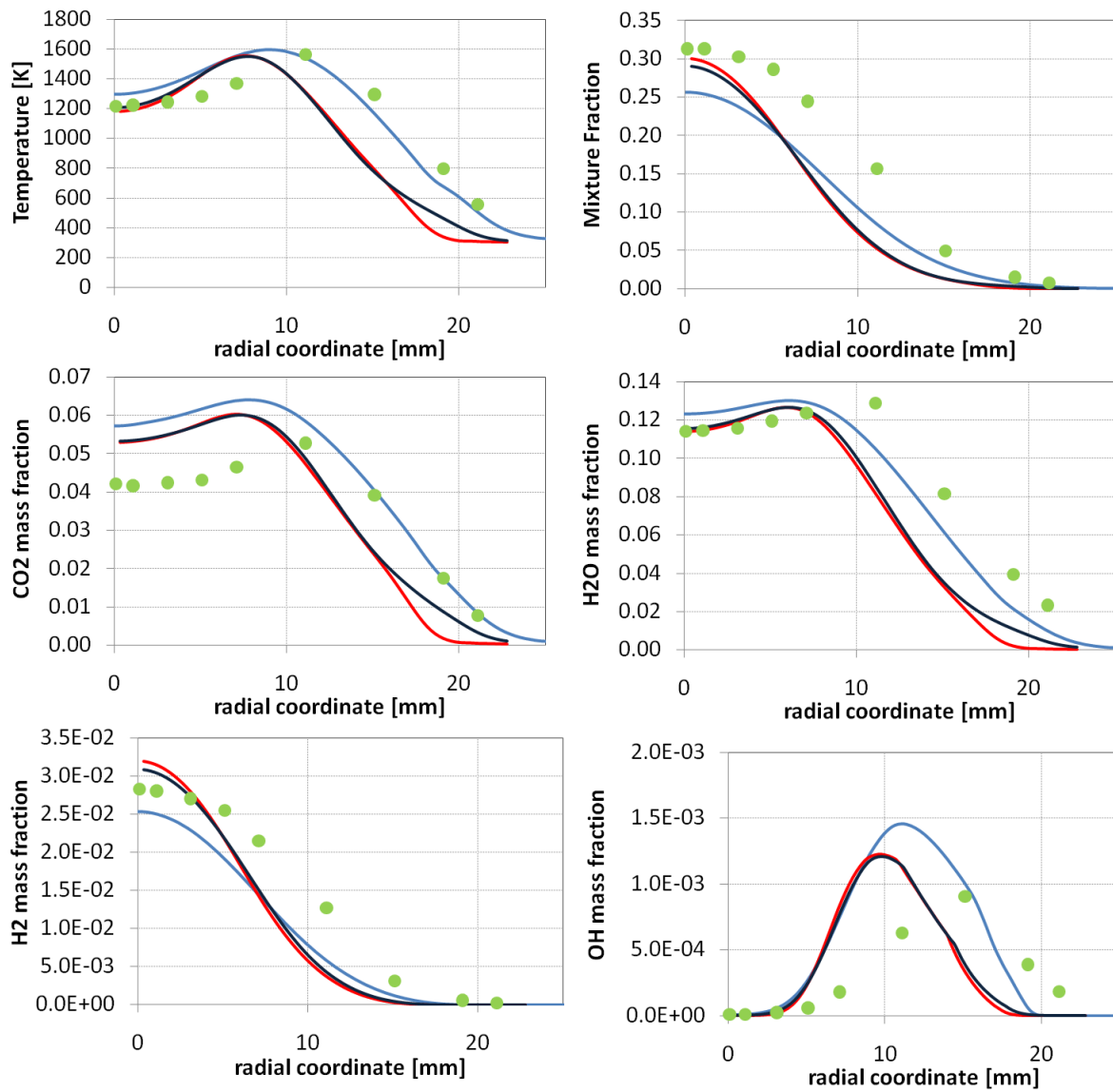


Figura B.37: Fiamma *Bluff-Body HM1*. Profili radiali di temperatura, composizione, *mixture fraction* e varianza calcolati a 120 mm dallo sbocco. Confronto tra modelli (linea azzurra *Fluent* adiabatico, linea rossa *OpenFOAM* adiabatico, linea blu *OpenFOAM* non adiabatico) e dati sperimentali (punti in verde) [1]

Elenco dei simboli

A	fattore preesponenziale
a_p	coefficiente di assorbimento
a_{ij}	numero di atomi dell'elemento j all'interno della specie i
C_i	concentrazione della specie i
$C_{1\varepsilon}$	costante del modello $k - \epsilon$
$C_{2\varepsilon}$	costante del modello $k - \epsilon$
C_μ	costante dei modelli di turbolenza
C_χ	costante per il calcolo dello <i>scalar dissipation rate</i>
C_P	calore specifico
C_{EBU}	costante del modello <i>Eddy Break Up</i>
Da	numero di Damkohler
$\underline{\underline{D}}$	tensore delle deformazioni
E_a	energia di attivazione
\hat{E}	energia totale specifica per unità di massa
\hat{E}_K	energia cinetica specifica
erf	funzione degli errori
f	<i>mixture fraction</i>

ELENCO DEI SIMBOLI

f''^2	varianza della <i>mixture fraction</i>
\vec{f}	forza di volume
g	varianza della distribuzione della <i>Clipped Gaussian PDF</i>
\hat{H}	entalpia specifica
\hat{H}_s	entalpia specifica sensibile
$\underline{\underline{I}}$	matrice identità
\vec{j}	flusso diffusivo molecolare
k	energia cinetica turbolenta
K	costante di velocità di reazione
L	macroscala della turbolenza
Le	numero di Lewis
l_d	microscala di Kolmogorov
l_{mix}	lunghezza di miscelazione turbolenta
m	portata massiva
M_w	peso molecolare
N_C	numero di componenti nel sistema
n_e	numero di elementi all'interno del sistema
\underline{n}	versore normale alla superficie
p	pressione
P	<i>probability density function (PDF)</i>
P_k	termine sorgente nell'equazione dell'energia cinetica
$\vec{Q}_S(\phi)$	flusso di ϕ attraverso la superficie S
$\vec{Q}_V(\phi)$	produzione di ϕ nel volume V

ELENCO DEI SIMBOLI

\dot{Q}	termine sorgente dell'equazione dell'energia
\vec{q}	flusso termico
r	rapporto massivo aria-combustibile
r_r	rapporto tra i prodotti dei coefficienti stechiometrici e dei pesi molecolari del <i>fuel</i> e dell'ossidante
R	costante dei gas perfetti
Re	numero di Reynolds
S	superficie
Sc_t	numero di Schmidt
$\underline{\underline{\tilde{S}}}$	tensore degli sforzi medio
T	temperatura
t	tempo
\vec{u}	velocità
\hat{U}	energia interna specifica
V	volume
\vec{V}_k	velocità di diffusione
W	peso atomico
x	coordinata cartesiana
X	frazione molare
Z_j	frazione massiva degli elementi
α	coefficiente della $\beta - PDF$
α_1, α_2	coefficienti della <i>Clipped Gaussian PDF</i>
β	coefficiente della $\beta - PDF$

ELENCO DEI SIMBOLI

γ_λ	frazione volumetrica della <i>fine structure</i>
δ	delta di Dirac
$\delta_{i,j}$	delta di Kronecker
Δ	delta
ε	velocità di dissipazione dell'energia cinetica turbolenta
θ	variabile di integrazione
λ	scala della turbolenza
λ_t	conducibilità termica
λ_d	diffusività termica
Γ	coefficiente di diffusione
μ	viscosità dinamica
μ_t	viscosità dinamica turbolenta
ν	viscosità cinematica
ν_i	coefficiente stechiometrico della specie i
ϕ	generica quantità
$\bar{\phi}$	media secondo Reynolds della generica quantità
ϕ'	componente fluttuante secondo Reynolds della generica quantità
$\tilde{\phi}$	media secondo Favre della generica quantità
ϕ''	componente fluttuante secondo Favre della generica quantità
φ	rapporto di equivalenza
π	pi greco
Π	produttoria
Re	numero di Reynolds

ELENCO DEI SIMBOLI

ρ	densità
σ	varianza della distribuzione normale logaritmica
σ_{SB}	costante di Stefan Boltzmann
σ_t	costante per il calcolo del flusso turbolento di entalpia
$\sigma_k, \sigma_\varepsilon$	costanti del modello <i>k - epsilon</i>
Σ	sommatoria
$\underline{\underline{\sigma}}$	tensore degli sforzi
τ_c	tempo caratteristico della chimica
τ_{mix}	tempo caratteristico della miscelazione turbolenta
τ^*	tempo di residenza
$\underline{\underline{\tau}}$	componente viscosa del tensore degli sforzi
χ	<i>scalar dissipation rate</i>
ψ	generica quantità
ω	frazione massiva
$\dot{\Omega}_k$	velocità di formazione massiva per unità di volume
∇	nabla
∂	derivata parziale
\int	integrale

Bibliografia

- [1] Aereomech, University of Sydney "<http://www.aeromech.usyd.edu.au/thermofluids>"
- [2] **Barlow R.**, "*Sandia H₂/He Flame: Scalar Data*", Sandia National Laboratories, 2003
- [3] **Barlow R., Frank J.**, "*Piloted CH₄/Air Flames C, D, E, and F – Release 2.1*", Sandia National Laboratories, 2007
- [4] **Barlow R., Fiechtner G. J., Carter C. D., Flury M.**, "*Sandia/ETH-Zurich CO/H₂/N₂ Flame Data - Release 1.1*", Sandia National Laboratories, 2003
- [5] **Bilger R. W.**, "*The structure of turbulent nonpremixed flames, Twenty-Second Symposium (International) on Combustion*", The Combustion Institute, p. 475-488, 1988
- [6] **Bilger R. W.**, "*Turbulent flows with nonpremixed reactants*", Turbulent Reacting Flows, Libby, P. A. and F. A. Williams (Editors), Springer-Verlag, Berlin, p. 65-113, 1980
- [7] **Bilger R. W.** "*Turbulent Jet Diffusion Flames*", Prog. Energy Combustion Science 1, 87-109, 1976
- [8] **Binninger B., Chan M. , Paczkko G., Hermann M.**, "*Numerical Simulation of Turbulent Partially Premixed Hydrogen Flames with the Flamelet Model*", Technical report, Advanced Combustion GmbH, Internal Report, 1998.
- [9] **Bird R. B., Stewart W. E., Lightfoot E. N.**, Transport phenomena (2nd Edition), Wiley, New York, 2002.

-
- [10] **Bowman C.T.** “*Kinetics of nitric oxide formation in combustion processes*”, 14th Symposium (International) on Combustion, The Combustion Institute, Pittsburg, PA, 729-738, 1973
- [11] **Bray K. N. C., Peters N.**, “*Laminar flamelets in turbulent flames*”, Turbulent Reacting Flows, Libby, P. A. and F. A. Williams (Editors), Academic Press, London, p. 78-84, 1994
- [12] **Buzzi Ferraris G.**, “*Metodi numerici e software in C++*”, Pearson Education Italia, 1998
- [13] **Castiglioni R., Croci F.**, “*Modellazione fluidodinamica di ossicombustione di gas naturale*”, tesi di laurea, Politecnico di Milano, 2007-2008
- [14] **Cleary, M. J.**, “*CMC Modelling of Enclosure Fires*”, The School of Aerospace, Mechanical and Mechatronic Engineering, The University of Sydney, Sydney, 2004
- [15] **Corbellini G.**, “*Simulazione numerica di flussi turbolenti reattivi*”, tesi di laurea, Politecnico di Milano, 2005-2006
- [16] **Covelli E., Dicatorato A.** “*Modellazione della combustione turbolenta non premiscelata*”, tesi di laurea, Politecnico di Milano, 2003-2004
- [17] **Cuoci A.** “*Pollutant formation in turbulent reactive flows: interaction between chemistry and turbulence*”, PhD Thesis, Politecnico di Milano, 2008
- [18] **Dente M., Ranzi E.** “*Principi dell'ingegneria chimica*”, 1977
- [19] DRM kinetic scheme, “<http://www.me.berkeley.edu/drm/drm19.dat>”
- [20] **Fenimore C.P.**, “*Formation of nitric oxide in premixed hydrocarbon flames*”. 13th Symposium (International) on Combustion, The Combustion Institute, Pittsburg, PA, 373-379, 1971
- [21] **Fluent Inc.** “*Fluent 6.3 User's guide*”, 2006
- [22] **Frassoldati A.** “*NO_x formation in turbulent diffusion flames: kinetic and fluid dynamics analysis*”, PhD Thesis, Politecnico di Milano, 2004
- [23] **Grosshandler W. L.**, “*RADCAL: a Narrow-Band Model for Radiation Calculations in a Combustion Environment*”, NIST technical note 1402, 1993.

-
- [24] **Hildebrand F. B.**, *“Introduction to Numerical Analysis (2nd Edition)”*, Dover Publications, 1987
- [25] **Jones W.P., Launder B.E.** *“The prediction of laminarization in a two-equation model of turbulence”*, Int. Journ. of Heat Mass Transfer, Vol 15, 301-304, 1972
- [26] **Kim J. S., Williams F. A.**, *“Extinction of Diffusion Flames with Non-Unity Lewis Number”*, Engineering Mathematics 31, 101-118, 1997
- [27] **Launder B. E., Spalding D. B.** *“The numerical computation of turbulent flows”*, Computer Methods in Applied Mechanics and Engineering, 3(2), 269-289, 1974
- [28] **Liu F., Guo H., Smallwood G. J., O. Gulder and M. D. Matovic**, *“A robust and accurate algorithm of the Beta-PDF integration and its application to turbulent methane-air diffusion combustion in a gas turbine combustor simulator”*, International Journal of Thermal Sciences 41, no. 8, 763-772, 2002
- [29] **Liu K., Pope S. B., Caughey D. A.**, *“Calculations of bluff-body stabilized flames using a joint probability density function model with detailed chemistry”*, Combustion and Flame 141, 89-117, 2005
- [30] **Magnussen B. F.** *“Modeling of NO_x and soot formation by the Eddy Dissipation Concept”*, International Flame Research Foundation, 1989
- [31] **Magnussen B. F., Hjertager B. H.**, *“On the mathematical modelling of turbulent combustion with special emphasis on soot formation and combustion”*, 16th Symposium (International) on Combustion, The Combustion Institute, 719-729, 1976
- [32] **Marracino B., Lentini D.**, *“Radiation modelling in non-luminous nonpremixed turbulent flames”*, Combustion Science and Technology 128, 23-48, 1997
- [33] **Merci B., Roekaerts D., Naud B., Pope S.B.**, *“Comparative study of micromixing models in transported scalar PDF simulations of turbulent nonpremixed bluff body flames”*, Combustion and Flame 146, 109-130, 2006
- [34] **Muller C., Breitbach M. H., Peters N.**, *“Partially premixed turbulent flame propagation in jet flames”*, Twenty-Fifth Symposium (International) on Combustion, The Combustion Institute, 1994, p.
- [35] OpenCFD Ltd, *“OpenFOAM 1.6 ProgrammersGuide”*, 2007

-
- [36] OpenCFD Ltd, “*OpenFOAM 1.6 UserGuide*”, 2008
- [37] OpenFOAM website, “<http://www.openFOAM.com>”
- [38] **Peters N.**, “*Laminar diffusion flamelet models in non premixed combustion*”, Progress Energy Combustion Science 10 , 319-339, 1984
- [39] **Peters N.** “*Turbulent Combustion, Cambridge University Press, Cambridge Monographs on Mechanics*”, 2000
- [40] **Pitsch H., Barths H., Peters N.**, “*Three-dimensional modeling of NO_x and soot formation in DI-Diesel engines using detailed chemistry based on the interactive flamelet approach*”, SAE Paper 962057, SAE, 1996
- [41] **Pitsch H., Peters N.**, “*A consistent flamelet formulation for non-premixed combustion considering differential diffusion effects*”, Combustion and Flame 114, p 26-40, 1998
- [42] **Poinsot T., Veynante D.**, “*Theoretical and numerical combustion*”, Edwards, Philadelphia, 2001
- [43] Polimi kinetic schemes, “<http://www.chem.polimi.it/CRECKModeling/kinetic.html>”
- [44] **Prandtl L.** “*Über ein neues Formelsystem für die ausgebildete Turbulenz*”, Nacr. Akad. Wiss. Göttingen, Math-Phys. Kl., 6-19, 1945
- [45] **Spalding D. B.**, “*Mixing and chemical reaction in steady confined turbulent flames*”, Thirteenth Symposium (International) on Combustion, The Combustion Institute, p. 649-657, 1971
- [46] **Sreenivasan K. R., Antonia A. R., Dahn H. Q.**, “*Temperature dissipation fluctuations in a turbulent boundary layer*”, Physics of Fluids 20 , 1238, 1977
- [47] TNF Workshop, “<http://www.ca.sandia.gov/TNF>, Sandia National Laboratories”
- [48] **Versteeg H., Malalasekera W.** “*An Introduction to Computational Fluid Dynamics: The Finite Volume Method*”, Longman, 1995
- [49] Wikipedia, “<http://www.wikipedia.com>”

BIBLIOGRAFIA

- [50] **Williams, F. A.**, "*Recent advances in theoretical descriptions of turbulent diffusion flames*", in S. N. B. Murthy, editor, *Turbulent mixing in Nonreactive and Reactive Flows*, p 189-208, Plenum Press, New York.
- [51] **Zeldovich J.** "*The oxidation of nitrogen in combustion and explosions*", *Acta Physicochimica, U.R.S.S.*, 21: 577-628, 1946