



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA



# Desarrollo y aplicación de un modelo CFD avanzado para chorros de combustible



Trabajo Final de Máster en Ingeniería  
Aeronáutica

Autor: Jaime Carrillo Cayuela

Tutor: José Manuel Pastor Enguidanos





# Resumen

El proyecto del que forma parte este trabajo final de máster tiene por objetivo final el estudio y la comprensión de la formación de chorros de combustible así como de su comportamiento para motores de combustión interna alternativos, para así intentar aumentar la eficiencia de la combustión, reducir emisiones y aumentar el propio rendimiento de los motores. Este trabajo, en concreto, se centra en el desarrollo y mejora de modelos de cálculo CFD para la atomización y formación del chorro de combustible. Se parte de un modelo de spray de combustible para cálculos RANS ya validado, se estudia la configuración del código además de los diferentes métodos y la formulación implementados, para ver cuáles son los puntos débiles. Una vez localizados se plantean diferentes planes de acción y se realizan en diferentes modificaciones con el fin obtener un resultado más fiel a la realidad y un modelo de cálculo con menor coste computacional. Al final, es sobre el modelo termodinámico utilizado para el cálculo de la temperatura de la mezcla aire-combustible a partir de su entalpía y sobre las ecuaciones del comportamiento del combustible en estado líquido dónde se practican los cambios más significativos para el modelo de cálculo, cumpliendo con los objetivos propuestos. La modificación del modelo y la comprobación de los avances se lleva a cabo en primer lugar con simulaciones RANS, después se aplican estas cambios sobre el modelo de simulación LES y se comparan entre ambos además de con un caso experimental. De esta manera, se aprecia la mejora en la solución que implica usar un modelo de cálculo LES frente a uno RANS.



# Agradecimientos

Este trabajo quiero agradeceréselo por su apoyo continuo, como, a mi familia: mis padres, mi hermano y mi abuela. Durante estos seis años estudiando en la universidad ninguno ha fallado, y mis padres han dado todo y más para que mi hermano y yo estudiemos sin preocupaciones. Y mi abuela tampoco, siempre pendiente de que mi hermano y yo contemos con recursos para poder disfrutar de estos años como estudiantes. Por tanto, muchas gracias a todos.

En segundo lugar, también quiero agradecer a mi tutor, José Manuel, toda la atención que me ha prestado durante el desarrollo de este trabajo. Además, tanto a él como a José María, debo dar las gracias por la oportunidad de colaborar en el proyecto y la confianza que depositaron en mí por ofrecerme continuar con ellos como doctorando, aunque yo lo declinara. Es gratificante que valoren tu trabajo.



# Tabla de contenidos

## Documento 1: Memoria

<b>1</b>	<b><i>Introducción</i></b>	<b>3</b>
1.1	Motivación del estudio	3
1.2	Objetivos y metodología	4
1.3	Estructura del proyecto	5
<b>2</b>	<b><i>Conceptos previos</i></b>	<b>7</b>
2.1	Proceso de inyección. Conceptos básicos.	7
2.1.1	Geometría de la tobera de inyección	8
2.1.2	Caracterización hidráulica de la tobera	9
2.2	Desarrollo del chorro diésel	10
2.2.1	Atomización	11
2.2.2	Coalescencia	14
2.2.3	Evaporación	16
2.3	Caracterización y parámetros del chorro	17
2.4	Modelos de aproximación para la simulación	20
2.4.1	Aproximaciones para la definición de interfaz	21
2.4.2	Modelos euleriano-lagrangianos	22
2.4.3	Modelos eulerianos multi-fluido	23
2.4.4	Modelos de flujo homogéneo	24
<b>3</b>	<b><i>Descripción del modelo e implementación</i></b>	<b>25</b>
3.1	Descripción general del modelo	25
3.2	Ecuaciones fundamentales e implementación	27
3.2.1	Ecuaciones que gobiernan el modelo	27
3.3	Desarrollo de los modelos	38
3.3.1	Modelo de turbulencia	38
3.3.2	Modelo de evaporación	42
3.3.3	Modelo termodinámico del combustible líquido	45
3.3.4	Ecuación de estado del combustible líquido	47
<b>4</b>	<b><i>Metodología, configuración del caso y resultados</i></b>	<b>49</b>
4.1	Definición del problema y condiciones de contorno	49

4.2	Metodología y malla	51
4.3	Análisis de los resultados	53
4.3.1	Resultados RANS	53
4.3.2	Resultados LES	72
5	Conclusiones y futuros trabajos	83
6	Bibliografía	87

## Documento 2: Pliego de condiciones

I	Objeto del pliego de condiciones	91
II	Condiciones generales	91
II.I	Condiciones generales facultativas	91
III	Condiciones particulares	92
III.I	Condiciones de hardware	92
III.II	Condiciones de software	93

## Documento 3: Presupuesto

I	Objeto del presupuesto y consideraciones	97
II	Desglose del presupuesto	97
II.I	Coste de la mano de obra	97
II.II	Amortización del equipo informático y licencias	99
II.III	Costes asociados a la simulación de los casos	99
II.IV	Costes de materiales fungibles	99
III	Resumen del presupuesto	100

## Documento 4: Anexos

I	Tutorial de OpenFOAM	103
II	Programas para agilizar el proceso de cálculo	111
III	Programas de post-procesado	123

# Índice de figuras

Figura 2.1. Esquema de sistema Common-rail.	7
Figura 2.2. Tipos de tobera, microsac y VCO.	8
Figura 2.3. Área y velocidad efectivas en tobera.	9
Figura 2.4. Estructura del chorro diésel.	10
Figura 2.5. Regímenes de atomización.	13
Figura 2.6. Mecanismos de atomización secundaria.	14
Figura 2.7. Mecanismos de colisión y coalescencia.	15
Figura 2.8. Mapa de tipos de colisión y coalescencia.	16
Figura 2.9. Descripción macroscópica del chorro.	18
Figura 4.1. Dominio del caso RANS.	50
Figura 4.2. Temperatura de la mezcla según composición. Modelo original.	56
Figura 4.3. Temperatura de la mezcla según composición. Control por entalpía.	56
Figura 4.4. Temperatura de la mezcla según composición. Control por temperatura.	56
Figura 4.5. Combustible líquido según composición. Modelo original.	57
Figura 4.6. Combustible líquido según composición. Control por entalpía.	57
Figura 4.7. Combustible líquido según composición. Control por temperatura.	57
Figura 4.8. Combustible gaseoso según composición. Modelo original.	58
Figura 4.9. Combustible gaseoso según composición. Control por entalpía.	58
Figura 4.10. Combustible gaseoso según composición. Control por temperatura.	58
Figura 4.11. Enthalpía según composición de la mezcla. Modelo original.	59
Figura 4.12. Enthalpía según composición de la mezcla. Control por entalpía.	59
Figura 4.13. Enthalpía según composición de la mezcla. Control por temperatura.	59
Figura 4.14. Tasa de evaporación/coalescencia según composición. Modelo original.	60
Figura 4.15. Evaporación/coalescencia según composición. Control por entalpía.	60
Figura 4.16. Evaporación/coalescencia según composición. Control por temperatura.	60
Figura 4.17. Error relativo de entalpía para los tres modelos.	61
Figura 4.18. Error relativo de entalpía para los modelos controlados por entalpía y temperatura.	61
Figura 4.19. Mapa de temperatura para el modelo controlado por la temperatura.	62
Figura 4.20. Mapa de temperatura para el modelo original.	63
Figura 4.21. Mapa de temperatura para el modelo controlado por entalpía.	63
Figura 4.22. Mapa de temperatura para el modelo controlado por temperatura.	63
Figura 4.23. Mapa de combustible líquido para el modelo original.	64
Figura 4.24. Mapa de combustible líquido para el modelo controlado por entalpía.	65
Figura 4.25. Mapa de combustible líquido para el modelo controlado por temperatura.	65
Figura 4.26. Mapa de combustible gaseoso para el modelo original.	65
Figura 4.27. Mapa de combustible gaseoso para el modelo controlado por entalpía.	66
Figura 4.28. Mapa de combustible gaseoso para el modelo controlado por temperatura.	66
Figura 4.29. Penetración del chorro: original, control por entalpía y por temperatura.	66
Figura 4.30. Temperatura según composición. Control por temperatura NIST.	68
Figura 4.31. Combustible líquido según composición. Control por temperatura NIST.	68
Figura 4.32. Combustible líquido según composición. Control por temperatura NIST.	69
Figura 4.33. Enthalpía según composición. Control por temperatura NIST.	69
Figura 4.34. Evaporación/coalescencia según composición. Control temperatura NIST.	69
Figura 4.35. Error relativo de entalpía para los modelos controlados por temperatura.	70
Figura 4.36. Mapa de temperature. Control por temperatrua NIST.	70
Figura 4.37. Mapa de combustible líquido. Control por temperatura NIST.	70
Figura 4.38. Mapa de combustible gaseoso. Control por temperatua NIST.	71
Figura 4.39. Penetración del chorro para los modelos controlados por temperatura.	71
Figura 4.40. Temperatura según composición. Modelo LES. Promedio.	74

Figura 4.41. Combustible líquido según composición. Modelo LES. Promedio.	74
Figura 4.42. Combustible líquido según composición. Modelo LES. Instante 1.9 ms.	74
Figura 4.43. Combustible gaseoso según composición. Modelo LES.	75
Figura 4.44. Combustible gaseoso según composición. Modelo LES. Instante 1.9 ms.	75
Figura 4.45. Penetración del chorro. Modelo LES.	76
Figura 4.46. Mapa de temperatura [K]. Modelo LES.	76
Figura 4.47. Penetración del chorro para el caso experimental y el modelo LES.	78
Figura 4.48. Contorno del chorro para el caso experimental y el modelo LES.	78
Figura 4.49. Velocidad normalizada. Eje longitudinal y plano normal a 25 mm.	79
Figura 4.50. Velocidad normalizada. Eje longitudinal y plano normal a 45 mm.	79
Figura 4.51. Contornos de velocidad para el modelo LES.	80
Figura 4.52. Combustible gaseoso. Eje longitudinal y plano normal a 25 mm.	81
Figura 4.53. Combustible gaseoso. Eje longitudinal y plano normal a 45 mm.	81
Figura 4.54. Contornos de combustible gaseoso para el modelo LES.	82
Figura I.1. Ejemplo 1 de archivo "blockMeshDict".	105
Figura I.2. Ejemplo 2 de archivo "blockMeshDict".	106
Figura I.3. Ejemplo del fichero "sampleDict".	110

## Índice de tablas

Tabla 3.1. Coeficientes del modelo de turbulencia de alta densidad $k-\varepsilon$	40
Tabla 3.2. Coeficientes.	47
Tabla 3.3. Coeficientes.	48
Tabla 4.1. Parámetros geométricos de la tobera 675.	50
Tabla 4.2. Condición nominal de Spray A definida por la ECN.	50
Tabla 4.3. Longitud líquida. Original, control por entalpía y control por temperatura.	64
Tabla 4.4. Tiempo de cálculo para 1ms.	67
Tabla 4.5. Longitud líquida. Original y modelos controlados por temperatura.	71
Tabla 4.6. Tiempo de cálculo para 1ms de tiempo físico, segundo cambio en el modelo.	72
Tabla 4.7. Longitud líquida. Original, modelo RANS y modelo LES.	76
Tabla 4.8. Longitud líquida. Original, experimental, LES.	77
Tabla II. 1. Coste anual de un ingeniero para la empresa.	98
Tabla II. 2. Coste global de los ingenieros implicados en el proyecto.	98
Tabla II. 3. Coste por amortización de los equipos y las licencias.	99
Tabla II. 4. Coste asociado a la simulación de los casos de estudio.	99
Tabla II. 5. Coste de materiales fungibles.	100
Tabla II. 6. Resumen del presupuesto	100



# Documento 1:

## Memoria



# 1 Introducción

## 1.1 Motivación del estudio

El motor de combustión interna alternativo (MCI) es uno de los mayores inventos desde la revolución industrial. Desde su aparición, el afán de los ingenieros por su desarrollo y el cada vez mayor conocimiento de la termodinámica y la combustión han permitido avanzar mucho en materia de motores de combustión interna, consiguiendo así el uso eficiente y práctico de los mismos. Es usado en todo tipo de aparatos (bombas, cortacéspedes, generadores eléctricos...), así como en muchísimos medios de transporte (tractores, aeronaves ligeras, barcos, vehículos terrestres...). De hecho, el sector del transporte ha sido dominado por este tipo de motor durante un siglo debido a su eficiencia térmica y la gran potencia que pueden desarrollar.

Su creciente uso impulsó la economía y la industrialización del mundo moderno y supuso dar el salto importante en el nivel de consumo de combustibles fósiles. Al principio, la gente no estaba concienciada con este último dato, por lo que se trataban de diseñar motores que alcanzaran cada vez una mayor potencia. Sin embargo, la situación ha cambiado en estas últimas décadas especialmente: la demanda por conseguir que los aparatos con motores combustión sean más limpios y eficientes es cada vez mayor. En este contexto, el diésel tiene una importancia sustancial.

Dentro de los MCIs, los motores de encendido por compresión (MEC) son los más eficientes y su uso en los medios de transporte está muy extendido. No obstante, las regulaciones medio ambientales y los requisitos en cuanto al consumo de combustible se han vuelto muy restrictivas en los últimos años, especialmente en la Unión Europea. Además los precios de los combustibles fósiles, bastante usados por los medios de transporte, ha ido creciendo paulatinamente y no se espera que bajen en los años venideros. Por tanto, es necesario conseguir tanto una máxima eficiencia en el consumo de combustible como una gran reducción de las emisiones contaminantes en los motores de combustión interna para mantener su viabilidad como fuente de potencia primario para los vehículos. A razón de alcanzar esta meta, entender el proceso de inyección del combustible, la posterior formación de la mezcla de aire-combustible y su evaporación es esencial, ya que juegan un papel muy importante en la combustión y la formación de contaminantes. De lo contrario, una mezcla pudiera no ser adecuada y existir zonas con fracciones de combustible demasiado ricas o demasiado pobres, ocasionando un mal rendimiento del motor e incrementando la contaminación. Ergo, una predicción precisa de este proceso es muy útil.

En el presente, los motores diésel son sistemas tan sofisticados que cualquier mejora requiere un gran esfuerzo. Afortunadamente, la llegada de los ordenadores ha desencadenado la aparición de línea de investigación en la ciencia y la ingeniería llamada simulación numérica. Las simulaciones computacionales se han convertido en una parte fundamental del proceso de diseño de sistemas de combustión y lo dotan también de una mayor agilidad. Asimismo, las simulaciones también proporcionan información adicional sobre problemas subyacentes, difíciles o imposibles de obtener de forma experimental; lo cuál posibilita el estudio de diferentes fenómenos complejos (transferencia de calor, flujos multifase, turbulencia...) y el mismo proceso de combustión. Aun así, como ya se ha señalado antes, es imposible tener una predicción adecuada del proceso de combustión y generación de contaminantes sin una correcta simulación del proceso de inyección. En este sentido, el modelo de chorro diésel es todavía un reto a batir por los complejos procesos que lo envuelven; como la atomización primaria o la cavitación en la tobera de inyección.

Por tanto, el desarrollo de modelos avanzados de chorro para la representación de alta fidelidad de procesos como los mencionados antes es crucial en la investigación, mejora y diseño de futuros motores de combustión.

## 1.2 Objetivos y metodología

Uno de los objetivos fundamentales de este proyecto es la simulación numérica del chorro diésel en condiciones similares a las del cilindro de un motor, pues este trabajo se centra en condiciones inertes de inyección. Esto implica considerar la atomización y los procesos de transporte y evaporación en la formación y el desarrollo del chorro diésel. Una predicción fiable de parámetros característicos como la penetración del chorro o la longitud líquida, junto con la distribución de combustible es un importante indicador de los modelos de chorro.

Para alcanzar este objetivo, la implementación y validación de modelos avanzados de chorro en una plataforma CFD son las tareas principales de este trabajo. Ante la necesidad de un modelo avanzado capaz de reproducir de forma precisa los procesos físicos del chorro diésel, los modelos lagrangianos de partículas sufren tratando de resolver este tipo de problemas y no son capaces de incluir los efectos de la tobera en la simulación, por lo que parece más adecuado trabajar con un modelo euleriano para cumplir las metas propuestas principales.

El otro objetivo principal es, partiendo de un modelo de chorro diésel ya hecho, conseguir otro modelo que reduzca los costes computacionales y permita una mayor

velocidad de cálculo, llegando a ser un modelo más funcional que el anterior. Para ello se deben realizar algunos cambios en los modelos elegidos para representar ciertas propiedades de los compuestos estudiados en la inyección así como cambiar determinados procesos de cálculo. Como entorno computacional para dichos modelos, se han usado tanto cálculos RANS como LES, con el objetivo último de plantear un buen modelo para cálculos LES.

Así pues, el nuevo modelo nace del anterior, y como ya se ha dicho, se introducen modificaciones respecto a los modelos físicos de diferentes propiedades del diésel y variaciones en determinados procesos de cálculo. Dado que el modelo de chorro diésel anterior para cálculos RANS ya se había validado con un caso experimental, el nuevo modelo se ejecutaba primero en RANS y se validaba y comparaba con el antiguo: tanto resultados de la simulación como velocidad de cálculo.

Una vez se comprobaba que el nuevo modelo preparado para cálculos RANS correctamente, se comenzaba la implementación del modelo para cálculos LES, dónde se realizaban la validación directamente con el caso experimental.

El software utilizado para implementar el código de los modelos ha sido un software libre de simulación CFD: OpenFOAM. Este software usa el lenguaje de programación C++, por lo que los modelos han sido escritos en este lenguaje.

El caso experimental utilizad para validar el modelo de chorro diésel forma parte de la base de datos de una red de distintas instituciones llamada Engine Combustion Network y de la cuál es miembro el Departameto de Máquinas y Motores Térmicos.

Para tratar los datos generados durante la simulación se han utilizado dos programas de software diferentes, Paraview y MATLAB.

### 1.3 Estructura del proyecto

El documento de este Trabajo Final de Máster queda estructurado en cuatro secciones: la sección principal o memoria y tres secciones secundarias, pliego de condiciones, presupuesto y anexos.

En la sección principal quedan explicados los fundamentos teóricos del trabajo, la implementación de los modelos, el caso de estudio y su análisis y las conclusiones:

- Capítulo 1: Es el actual capítulo y proporciona información básica sobre el proyecto desarrollado y sus objetivos, además de presentar la estructura del documento completo.

- Capítulo 2: Consiste en una revisión bibliográfica sobre el proceso de inyección y el chorro diésel, explicando conceptos básicos sobre ambos, además de los parámetros que caracterizan el chorro y son útiles para validar los modelos. También se discute acerca de los modelos lagrangianos y eulerianos.
- Capítulo 3: Describe por completo el modelo euleriano desarrollado y las ventajas que ofrece y las ecuaciones que gobiernan las ecuaciones de transporte y su implementación en el código. Por último expone los modelos físicos utilizados para el combustible así como el modelo de turbulencia.
- Capítulo 4: Se describe el caso de estudio: condiciones de contorno y malla; también analiza los resultados de la simulación
- Capítulo 5: Recoge las conclusiones resultantes del análisis desarrollado en el capítulo anterior.
- Capítulo 7: Bibliografía.

En el pliego de condiciones queda recogida la normativa y el seguimiento técnico de la realización del proyecto.

En la tercera sección del proyecto queda explicado el presupuesto así como el desglose de todos los costes que acarrea este trabajo.

Por último, la sección de anexos contiene información explicativa sobre el manejo del software OpenFOAM para los modelos realizados y el caso ejecutado. También incluye los scripts externos al código de OpenFOAM que permiten agilizar el proceso de cálculo y el tratamiento de los datos resultantes tras la simulación.

## 2 Conceptos previos

### 2.1 Proceso de inyección. Conceptos básicos.

El sistema de inyección es el encargado de proporcionar el combustible al interior del motor y de preparar la mezcla air-combustible para la combustión. Entonces, tanto la eficiencia de la combustión como la formación de emisiones contaminantes son, en gran medida, dependientes de la actuación del sistema de inyección. Según como se introduzca el combustible en la cámara de combustión, los sistemas de inyección se pueden clasificar en dos tipos: inyección indirecta (IDI) inyección directa (DI). Sin embargo, en los últimos años los sistemas de inyección directa han sido los más utilizados porque alcanzan los niveles de menor consumo y mayor eficiencia. Hoy día el más usado es el sistema *Common-rail*.

En la Figura 2.1 se muestra una representación esquemática de un sistema de inyección *Common-rail*. Entre la bomba de alta presión y los inyectores hay un elemento intermedio. Usando este conducto común para el combustible es posible proporcionar una presión de inyección constante y controlada. Los inyectores se controlan electrónicamente, lo que permite un gran control sobre el inicio y la duración del proceso de inyección. Luego, el proceso puede ser dividido en varias etapas para reducir el problema del ruido, mejorar la combustión y limitar las emisiones contaminantes. Además, estos sistemas pueden trabajar con elevados valores de presión (mayores a 200 MPa) que mejoran la atomización del combustible, el proceso de evaporación y la mezcla con el gas del interior de la cámara.

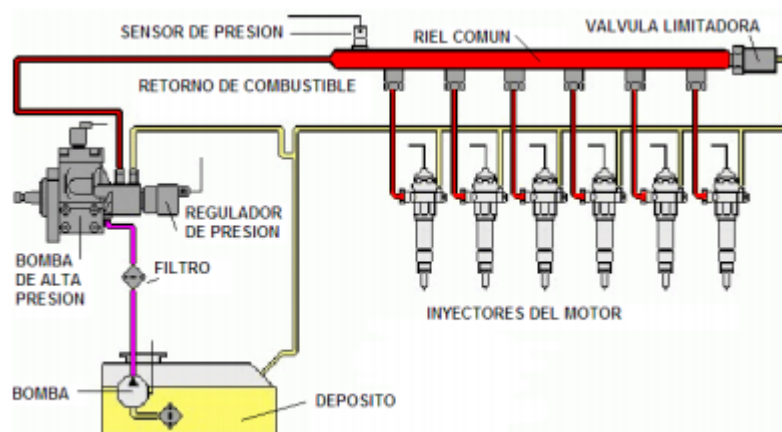


Figura 2.1. Esquema de sistema *Common-rail*.

En este trabajo, la simulación del flujo interno de combustible en el inyector (dónde se puede dar el problema de cavitación) no es objeto de estudio, sólo se centra el chorro diésel una vez sale de la tobera de inyección. Ahora bien, sí se debe tener en cuenta la cantidad de movimiento que introduce en el volumen de control el sistema de inyección para conocer su efecto en el caso de estudio, por ello es necesario reconocer algunos parámetros y características de las toberas de inyección que se usan en la descripción del flujo a la salida de la misma.

### 2.1.1 Geometría de la tobera de inyección

Las características de la tobera de inyección son parámetros clave para la formación del chorro de combustible y juegan un papel muy importante en el comportamiento del chorro dentro de la cámara de combustión. Los dos tipos de toberas más usados en la actualidad son: VCO (válvula de orificio cerrado) y microsac. En las toberas VCO, la posición de la aguja influye mucho en el desarrollo del flujo de combustible durante los transitorios del inyector de apertura y cierre, mientras que las toberas del tipo microsac consiguen minimizar este efecto. No obstante, estas últimas toberas presentan un problema, se queda un remanente de combustible debido al pequeño saco que tiene. Las toberas microsac, en condiciones que se alejan de las diseño, pueden dar un bajo rendimiento para la combustión, propiciando el incremento de hidrocarburos sin quemar y la formación de hollín. En la Figura 2.2 se puede ver la geometría de ambos tipos de tobera.

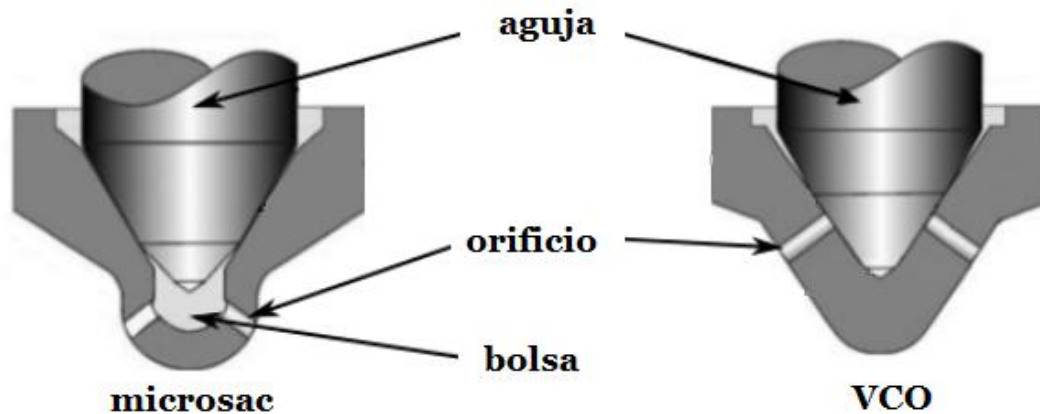


Figura 2.2. Tipos de tobera, microsac y VCO.

Los orificios de las toberas son los responsables de la aceleración del flujo inyectado, por lo que su geometría es clave para las características del chorro. Algunos de los principales parámetros geométricos de las toberas son: el diámetro de entrada a la tobera ( $D_i$ ), el diámetro de salida ( $D_o$ ) o la longitud de la misma ( $L$ ). A partir de estos parámetros se pueden definir variables que ayudan a caracterizar las toberas, como el término *k-factor*, que se calcula según la Ecuación 2.1:



$$k - factor = \frac{D_i - D_o}{10 [\mu m]}$$

*Ecuación 2.1*

### 2.1.2 Caracterización hidráulica de la tobera

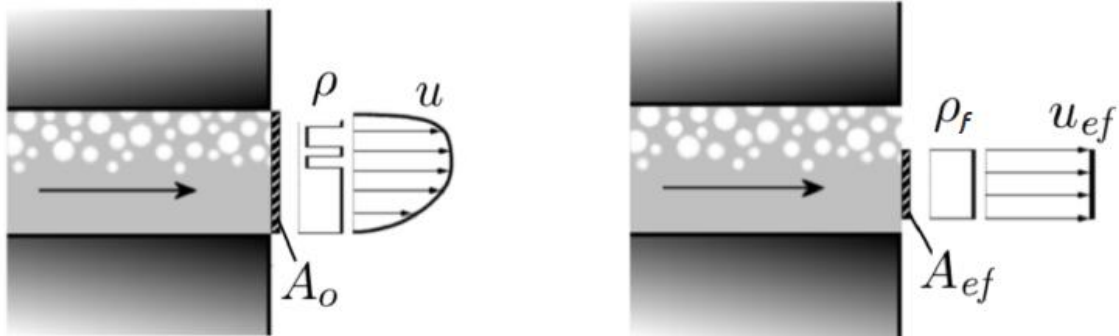
La complejidad del estudio del flujo de combustible a través del orificio de la tobera de inyección yace en las pequeñas dimensiones de la geometría del orificio y en altas velocidades que alcanza el flujo. Esta es la razón por la cuál el análisis del flujo interno en los canales del inyector se suele realizar a través de la parametrización del orificio de salida. Uno de los parámetros usados para ello es el coeficiente de descarga ( $C_d$ ), es el cociente entre el gasto másico de combustible real y el teórico (Ecuación 2.3). El gasto teórico se calcula partiendo de la ecuación de Bernoulli (Ecuación 2.2).

$$u = \sqrt{\frac{2\Delta P}{\rho_f}} ; \dot{m}_{f,th} = \rho_f u A_o$$

*Ecuación 2.2*

$$C_d = \frac{\dot{m}_f}{\dot{m}_{f,th}} = \frac{\dot{m}_f}{A_o \sqrt{2 \rho_f \Delta P}}$$

*Ecuación 2.3*



*Figura 2.3. Área y velocidad efectivas en tobera.*

El coeficiente de descarga es una medida de la eficiencia del orificio al entregar una cantidad de gasto másico. En su definición quedan incluidas las pérdidas de presión por fricción con las paredes y las debidas el perfil de velocidad no uniforme a la salida de la tobera. Considerando todo esto, es posible definir un perfil de velocidad uniforme que asegure la tasa de gasto másico real (Figura 2.3). El valor de este perfil de velocidad es la velocidad efectiva ( $u_{ef}$ ) y la sección por la cuál fluye el combustible es el área efectiva ( $A_{ef}$ ).

$$\dot{m}_f = \rho_f u_{ef} A_{ef}$$

*Ecuación 2.4*

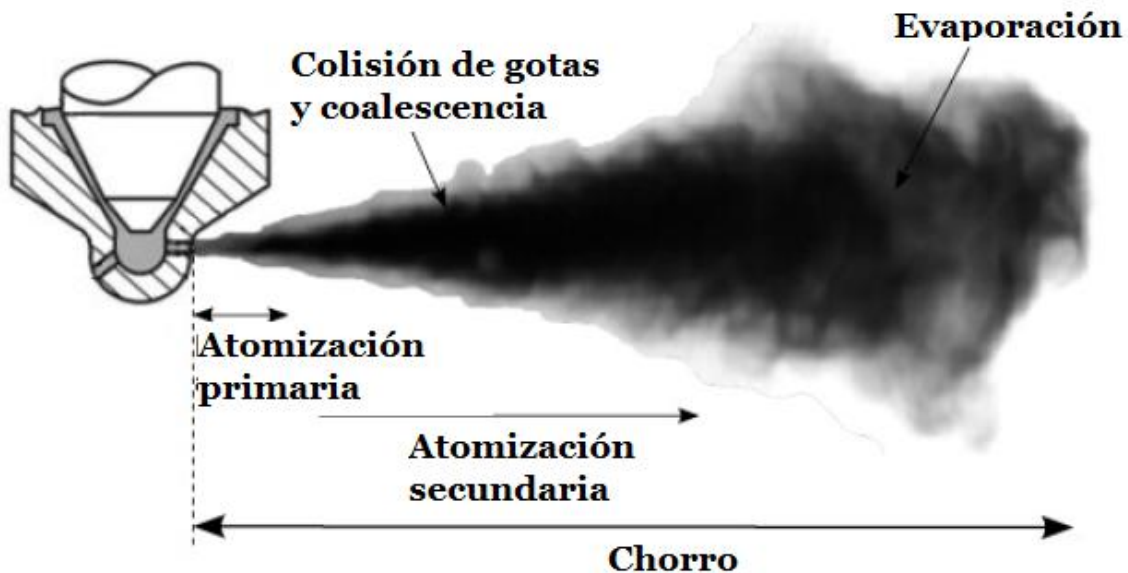
Por consiguiente, las pérdidas incluidas en el coeficiente de descarga se pueden dividir en dos: la reducción del área efectiva respecto al área geométrica debido al perfil de velocidad no uniforme y cuya relación viene dada por el coeficiente de área ( $C_a$ ), y las pérdidas de presión, que se incluyen en el coeficiente de velocidad ( $C_v$ ).

$$C_d = C_a C_v = \frac{A_{ef}}{A_0} \frac{u_{ef}}{u}$$

*Ecuación 2.5*

## 2.2 Desarrollo del chorro diésel

La formación del chorro diésel tiene lugar cuando el combustible líquido se mezcla con el gas de su alrededor en la cámara de combustión. Esta mezcla es consecuencia de varios procesos físicos, los cuales han sido estudiados durante años debido a la gran importancia que tiene este proceso en la combustión de la mezcla. Si bien se conoce la importancia de este fenómeno, no se termina de entender en su totalidad como se produce, ya que es extremadamente complejo.



*Figura 2.4. Estructura del chorro diésel.*

Por lo general, los fenómenos físicos que dirigen la formación del chorro de combustible son: atomización y coalescencia, el momento de transferencia entre las gotitas de combustible y el aire que las rodea, la interacción turbulenta y la evaporación.

En la Figura 2.4 se puede ver un esquema del chorro de combustible. A la salida de la tobera de inyección, los primeros milímetros del chorro consisten en un núcleo líquido, donde aún no se ha producido la segmentación del mismo en gotitas. A partir de ahí, la inestabilidad interna del líquido, causada por efectos inerciales junto con la turbulencia, propicia la separación de este núcleo y la aparición de “ramas líquidas” de combustible llamadas ligamentos. Esta primera etapa es conocida como *atomización primaria*. Tras esto, la tensión superficial de los mismos ligamentos y las fuerzas aerodinámicas (debidas a velocidades relativas entre gas y líquido) conducen a la aparición de gotas. Esta segunda segregación es llamada *atomización secundaria*. Al final del proceso existe una distribución de tamaños de gotas no homogénea. Una vez las gotas están formadas, se mezclan con el gas de alrededor y el combustible se evapora si las condiciones son adecuadas.

### 2.2.1 Atomización

La atomización del combustible líquido es el fenómeno que tiene lugar después de que la inyección haya sido inicializada. Consiste en la transformación del núcleo líquido de combustible en un conjunto de pequeñas gotas, incrementando la superficie de interacción entre combustible y aire y, consecuentemente, favoreciendo la evaporación del diésel. La ruptura del chorro líquido se debe a la interacción a escala microscópica que implica a la interacción aerodinámica y las fuerzas de tensión superficial de las gotas. Las fuerzas aerodinámicas tratan de separar el líquido mientras que las de tensión superficial tratan de mantenerlo cohesionado. Esta interacción entre ambos tipos de fuerzas es dependiente de las condiciones de inyección y la geometría del inyector. De acuerdo con el diferente nivel de interacción entre las fuerzas aerodinámicas y las de tensión superficial, se distinguen dos fases diferentes en el proceso de atomización:

- Atomización primaria. Esta etapa de la atomización aparece en los primeros milímetros del chorro y es la responsable de la desintegración del núcleo líquido de combustible. Esta disgregación es generada por la acción de fuerzas internas desencadenadas por inestabilidades inerciales, la turbulencia del proceso, el perfil irregular de velocidades a la salida del inyector y la cavitación en el interior del mismo.
- Atomización secundaria. Tras la formación, durante la atomización primaria, de “ligamentos” de combustible líquido o gotas con forma alargada, éstas siguen poseyendo una velocidad elevada, por lo que las fuerzas aerodinámicas actúan de forma importante sobre ellas y conducen a su dispersión en gotas de menor tamaño.

### Atomización primaria

Los procesos de atomización han sido estudiados tanto experimental como teóricamente durante. En ellos se entiende que para fluidos no viscosos inyectados con velocidades bajas, el proceso de atomización está dominado principalmente por las fuerzas de tensión superficiales; mientras que con fluidos viscosos la interacción con las fuerzas aerodinámicas adquiere importancia. Según estos estudios, el proceso de atomización para un chorro cilíndrico puede llevarse a cabo en diferentes regímenes:

- Régimen de Rayleigh. Se observa para velocidades de inyección muy bajas. De acuerdo con este caso, las perturbaciones en la superficie del chorro inyectado, causadas por el efecto de la tensión superficial, deforman el hilo de líquido hasta su separación en gotas cuyos tamaños son uniformes y mayores que el diámetro del orificio de salida del inyector.
- Régimen inducido por interacción aerodinámica I. Como consecuencia del incremento de velocidad, el efecto de estas fuerzas, cuyo origen reside en la velocidad relativa entre combustible y el gas de su entorno, comienza a adquirir importancia. De esta manera, el efecto de las fuerzas de fricción es muy similar al de las tensiones superficiales y facilitar la formación de gotas. Al igual que en el régimen anterior, estas gotas aparecen alejadas de la salida del inyector aguas abajo y con un diámetro similar al orificio de salida de la tobera.
- Régimen inducido por interacción aerodinámica II. Al incrementar la velocidad del chorro inyectado el proceso de atomización se vuelve más eficiente. Las perturbaciones iniciales se amplifican a la vez que crecen las fuerzas aerodinámicas y la separación del chorro de combustible en gotas aparece más cerca de la tobera de inyección. En este caso, el tamaño medio de las gotas formadas es mucho más pequeño que el diámetro del orificio del inyector.
- Régimen de atomización incompleta. La desintegración del chorro tiene lugar inmediatamente después del orificio de salida del inyector. Existe una pequeña separación entre el orificio y el comienzo de la atomización que depende de la velocidad del chorro, a mayor velocidad menor distancia. En este régimen se pueden dar dos casos dependientes del núcleo líquido del chorro. Éstos se denominan como régimen de atomización incompleta o completa, y la diferencia entre ambos reside en la presencia o ausencia del núcleo de combustible líquido. El tamaño de las gotas formadas es muchísimo más pequeño que el diámetro del orificio de salida del inyector.

La Figura 2.5 escenifica los regímenes de atomización mencionados en la explicación anterior.

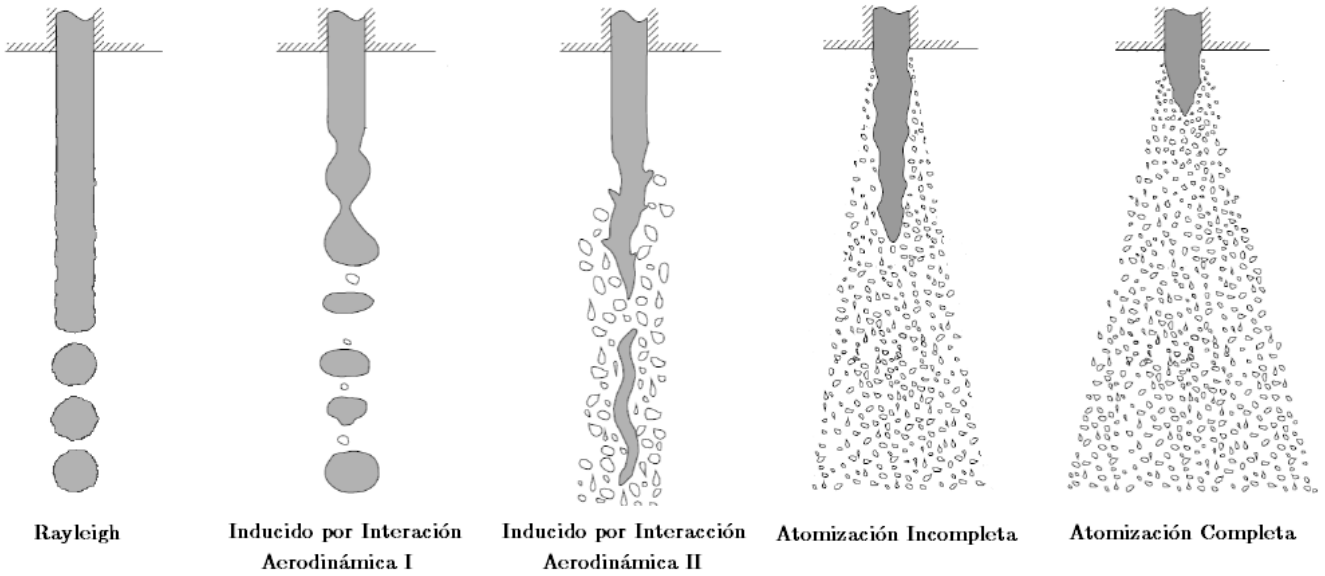


Figura 2.5. Regímenes de atomización.

### Atomización secundaria

En el caso de la atomización secundaria, las fuerzas aerodinámicas producen ciertas deformaciones en la superficie de las gotas que pueden dirigir a la división de éstas en gotas de menor tamaño. Esto sucede cuando las fuerzas de tensión superficial que mantienen la cohesión de las gotas crecen excesivamente. La tensión superficial depende la curvatura de la propia superficie, por lo que las gotas de menor tamaño presentan fuerzas de cohesión mayores. En el estudio de la atomización secundaria, para cuantificar la magnitud de estas fuerzas se usa el número de Weber, el cuál representa la relación entre las fuerzas aerodinámicas y la tensión superficial y se calcula según describe la Ecuación 2.6:

$$We = \frac{u_{rel}^2 D_{gota} \rho_{gas}}{\sigma}$$

*Ecuación 2.6*

dónde  $u_{rel}$  es la velocidad relativa entre combustible y gas,  $D_{gota}$  el diámetro de la gota,  $\rho_{gas}$  la densidad del gas y  $\sigma$  la tensión superficial.

A lo largo de los años, varios estudios experimentales han mostrado que dependiendo del número de Weber existen diferentes mecanismos para la división de las gotas en otras más pequeñas. Está aceptada una clasificación de los mecanismos de separación que los agrupa en cinco tipos diferentes: vibracional ( $We \sim 12$ ), de bolsa ( $We < 20$ ), de bolsa y cinta ( $We < 50$ ), de banda ( $We < 100$ ) y catastrófico ( $We > 100$ ). La Figura 2.6 ilustra estos mecanismos de separación:

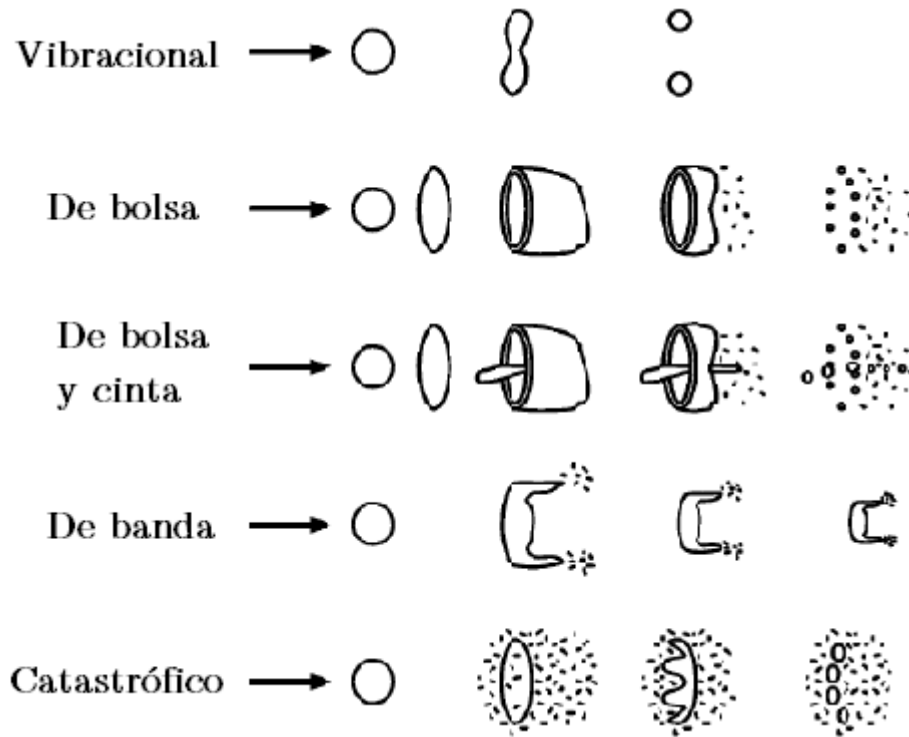


Figura 2.6. Mecanismos de atomización secundaria.

En condiciones de operación del motor, todos los mecanismos de atomización secundaria tienen lugar, pero predomina el mecanismo de atomización catastrófica ya que la mayor parte del proceso de separación del chorro en gotas tiene lugar cerca de la tobera, y en esa zona el número de Weber suele ser muy alto.

### 2.2.2 Coalescencia

A la vez y en la misma localización donde tiene lugar el proceso de atomización secundaria, el fenómeno de coalescencia y colisión de gotas también está presente. La coalescencia consiste en la comunión de dos o más gotas por el hecho de estar en contacto, formando así una misma gota de mayor tamaño. La probabilidad de la colisión entre las gotas depende de la velocidad y la dirección relativa entre las mismas además de la concentración local de combustible. Por tanto, las colisiones son más frecuentes en las regiones con alta densidad de combustible, como el eje longitudinal del chorro.

El proceso de colisión entre gotas es extremadamente complejo y depende de diferentes características geométricas de las gotas y las condiciones ambientales. Hay cuatro parámetros dimensionales que dirigen este acontecimiento: los números de Reynolds y Weber, el cociente entre los diámetros de las gotas colindantes ( $D_{g,2}/D_{g,1}$ , siendo  $D_{g,1}$  el diámetro de la gota grande y  $D_{g,2}$  el diámetro de la gota pequeña) y el parámetro de impacto B (cuyo valor varía entre 0 y 1 dependiendo del tipo de colisión).

$$Re = \frac{u_{rel} D_{g,1} \rho_{fuel}}{\mu_{fuel}}$$

Ecuación 2.7

$$We = \frac{u_{rel}^2 D_{g,21} \rho_{fuel}}{\sigma}$$

Ecuación 2.8

El resultado de la colisión puede ser clasificado en cinco tipos diferentes: rebote, coalescencia, separación reflexiva, separación alargada y colisión fatal. En la Figura 2.7 quedan representados estos regímenes de colisión. Ahora falta determinar el criterio para determinar cuándo ocurre cada tipo de colisión. A día de hoy se acepta que este criterio queda definido por el número de Weber y el parámetro de impacto, según muestra la Figura 2.8. Ahora bien, el hecho de aumentar la densidad del gas favorece la colisión por rebote, mientras que una atmósfera con alto contenido de combustible en estado gaseoso propicia la coalescencia. Además, el número y el tamaño de las nuevas gotas resultantes de una colisión son altamente dependiente del cociente entre diámetros de las gotas iniciales.

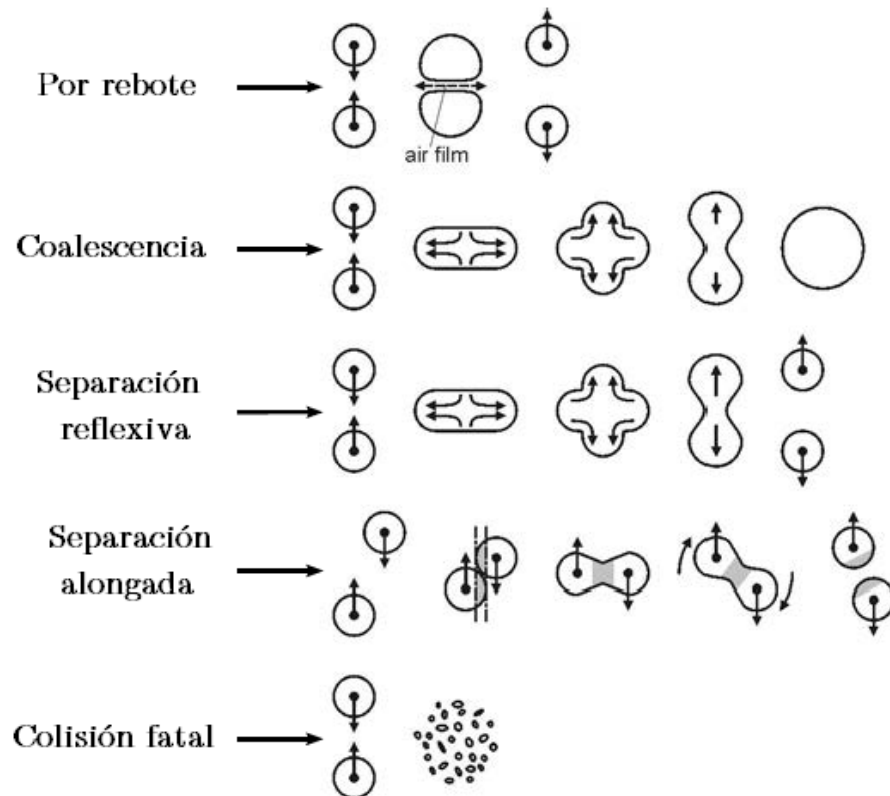


Figura 2.7. Mecanismos de colisión y coalescencia.

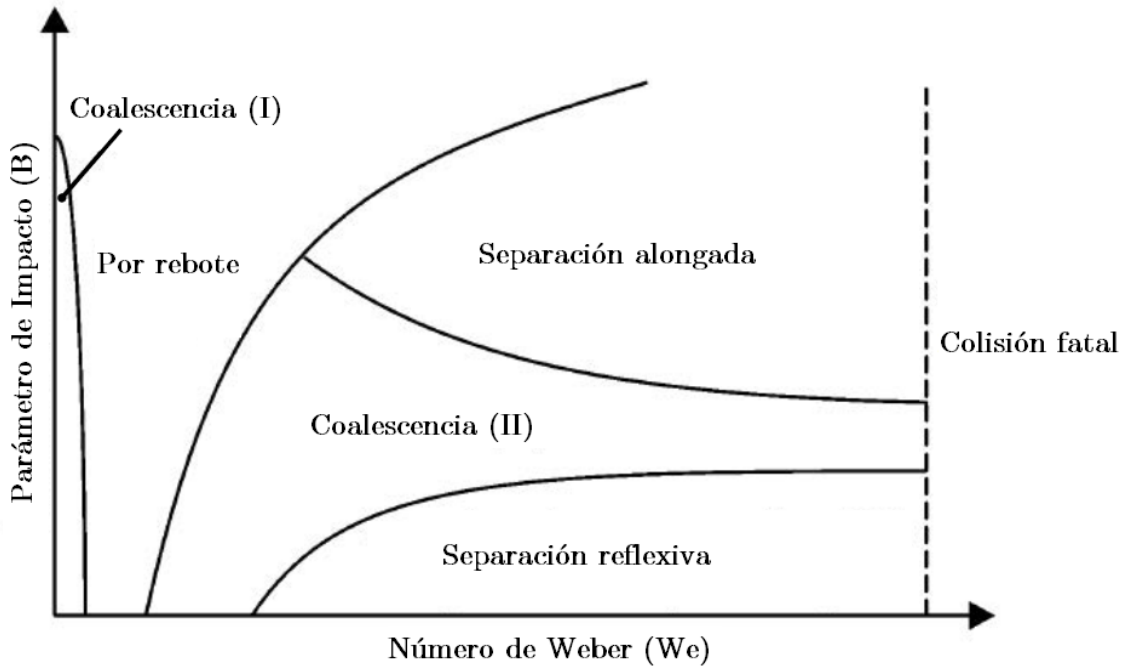


Figura 2.8. Mapa de tipos de colisión y coalescencia.

### 2.2.3 Evaporación

Al terminar el proceso de atomización, el combustible inyectado se ha transformado en una distribución de pequeñas gotas que intercambia energía con el gas que lo rodea presente en la cámara de combustión. Cuando la temperatura del gas es lo suficientemente elevada, éste emite un flujo de calor que recibe el combustible a través de la superficie de las gotas, ya sea por conducción o por convección. Este hecho supone un incremento de la temperatura de las gotas y, consecuentemente, de la presión de vapor. Este fenómeno dirige al combustible en estado líquido a su evaporación y hace que la mezcla local se aproxime a condiciones de saturación adiabática.

Al examinar en profundidad el proceso de evaporación, es posible encontrar tres tipos de fenómenos diferentes que determinan la evolución de las gotas de combustible en condiciones de operación del motor:

- Deceleración de las gotas debido a la resistencia aerodinámica.
- Transmisión de calor desde el gas a las gotas.
- Pérdida de masa de combustible evaporado por parte de las gotas.

Dado que las gotas de combustible viajan a través del aire, su velocidad se ve disminuida ante la presencia de la resistencia aerodinámica, consiguiendo así más tiempo para intercambiar calor con el ambiente. Además, dado que la superficie de combustible en contacto con el aire es la mayor al comienzo de la evaporación, el calor absorbido por las gotas es alto y la tasa de evaporación crece rápidamente. El calor transferido aumenta la temperatura, que eleva la presión de vapor y también la tasa de evaporación. El tamaño



de las gotas comienza a decrecer, pero la superficie expuesta al gas se reduce cuadráticamente, por lo que el flujo de calor transmitido cambia drásticamente y la velocidad de evaporación alcanza su máximo. La temperatura de las gotas comienza a alcanzar un valor asintótico, pues su temperatura está mucho más cerca del valor de la temperatura del aire que el de la inyección, asimismo, la superficie de las gotas decrece hasta que finalmente desaparece. Cabe mencionar una diferencia entre el calor emitido por el aire y el recibido por las gotas líquidas. Las gotas absorben menor calor que el transferido por el aire ya que parte de este último es requerido por el cambio de fase de líquido a vapor y se lo queda el combustible gaseoso.

Desde estudios experimentales se observa que hay una distancia desde el inyector, denominada longitud líquida, a partir de la cuál el combustible líquido desaparece y sólo queda combustible en estado gaseoso. Esta longitud líquida parece depender de diferentes parámetros que afectan al desarrollo del chorro, tales como el diámetro de la tobera, la temperatura y la densidad del gas en la cámara de combustión y propiedades termodinámicas del combustible inyectado. Sin embargo, la presión de inyección no parecer tener efecto sobre dicha longitud.

Se puede decir que los procesos de evaporación de chorro diésel inyectado directamente, en condiciones de funcionamiento del motor, están controlados principalmente por la mezcla. La energía que permite la evaporación del combustible se transfiere desde al aire que entra al cilindro como consecuencia de la mezcla turbulenta y, por tanto, el combustible líquido se evapora hasta alcanzar una fracción característica de la mezcla. Además, el proceso de difusión de masa, momento y energía entre fases parece no ser tan fundamental como la mezcla turbulenta, lo que refuerza la hipótesis de equilibrio termodinámica local del chorro en condiciones críticas de evaporación.

## 2.3 Caracterización y parámetros del chorro

Una vez quedan explicados el proceso de atomización y la formación del chorro, es importante conocer los parámetros que se usan para estudiar chorros. El chorro diésel puede caracterizarse desde dos puntos de vista diferentes: macroscópico, el cual estudia la forma global del mismo chorro (principalmente penetración, ángulo y longitud líquida), y microscópica, centrada en el estudio de variables locales (distribuciones de velocidad y concentración y tamaños de gotas).

Antes de analizar en detalle los distintos parámetros que caracterizan el chorro, uno puede definir dos zonas desde un punto de vista macroscópico. La primera es la zona estacionaria, que se extiende desde el principio hasta el 60-70% de la longitud total del

chorro y que tiene principalmente una forma cónica. La segunda es la zona transitoria y se sitúa a la vanguardia del chorro, dónde la interacción aerodinámica y la turbulencia la dotan de una forma irregular y llena de volutas. La Figura 2.9 es una representación esquemática del chorro diésel y la distinción de las mencionadas zonas.

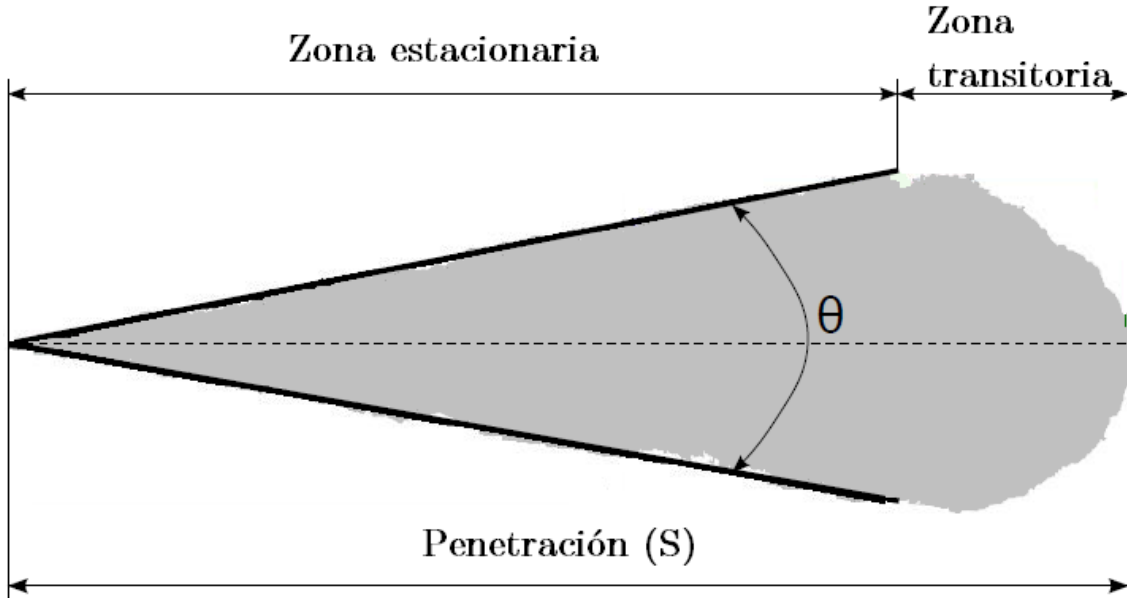


Figura 2.9. Descripción macroscópica del chorro.

### Penetración del chorro

La penetración del chorro es la distancia axial más lejana alcanzada sólo por la punta delantera del mismo. Estudiar la penetración de la punta del combustible inyectado es fundamental, no sólo porque es un indicador de la eficiencia de la mezcla aire-combustible sino porque también determina si el chorro puede llegar a impregnar, o no, de combustible la pared del cilindro de combustión, un factor clave para el proceso de combustión y la producción de emisiones contaminantes.

Este parámetro depende principalmente de la cantidad de movimiento del chorro, la densidad del gas ambiental, el ángulo de dispersión del combustible y, obviamente, el tiempo de inyección. Numéricamente queda definido como la distancia entre el orificio de inyección hasta la posición axial dónde se tiene una fracción másica de combustible total menor a 0'001.

### Ángulo de inyección del combustible

Debido a que resulta ser un indicador del grado de mezcla, al igual que la penetración, el ángulo que alcanza el combustible al ser inyectado ( $\theta$ ) es uno de los parámetros más estudiados. Se define normalmente como el ángulo incluido entre dos líneas que se ajustan a ambos lados del contorno del chorro (Figura 2.9). Este es también un parámetro

fundamental que dirige la interacción entre el aire y el combustible, por tanto, tiene una gran influencia en los fenómenos de la evaporación del combustible y de la combustión. Después de un breve transitorio al comienzo de la inyección, este ángulo se estabiliza en cierto valor.

### Longitud líquida

En condiciones de vaporización, la longitud líquida es la distancia máxima alcanza por el combustible en estado líquido antes de que termine de evaporarse. Es un parámetro que se mide experimentalmente durante la totalidad del tiempo de inyección, por lo que al principio coincide con la penetración del chorro de combustible, hasta que éste se estabiliza en un valor concreto y se mantiene prácticamente constante hasta el final de la inyección. La longitud líquida caracteriza el proceso de mezclado y depende de ciertos parámetros, como el tipo de combustible, la geometría del inyector o la diferencia de presión entre la cámara de combustión y la de inyección.

Numéricamente es descrita como la distancia a partir de la cuál la fracción másica de combustible líquido es menor a 0'01.

### Tamaño de gota

La eficiencia de la mezcla de aire-combustible es un factor importantísimo en el estudio del chorro diésel, y la caracterización de la distribución de concentración permite su propia evaluación. Por esta razón, muchos investigadores analizan estos parámetros. La mayoría de este tipo de estudios se basan en la similitud entre el chorro diésel y el chorro de gas que se puede tener a la salida de una tobera usada para inyectar, y son aceptados para describir el comportamiento estacionaria del chorro a una distancia relativamente lejana del orificio del inyector, dónde el chorro está completamente atomizado. De acuerdo con esta similitud, el chorro diésel presenta perfiles de velocidad y concentración de combustible similares en todas las secciones.

El uso de modelos gaussianos está muy extendido a la hora de describir estos perfiles. Los perfiles de concentración de combustible se definen con ayuda del número de Schmidt (que consiste en el cociente entre la difusión de cantidad de movimiento y la difusión de masa); sin embargo, establecer la ley de evolución axial de velocidad supone un reto mayor porque depende de complejos fenómenos físicos como la atomización y la coalescencia de las gotas de combustible.

Actualmente se acepta que los perfiles de concentración y de velocidad sobre el eje longitudinal se caracterizan distinguiendo entre dos zonas. La primera está relacionada con la longitud líquida, dónde las variables de concentración y velocidad permanecen constantes hasta cierta distancia; en la segunda ambos perfiles son inversamente proporcionales a la distancia hasta el orificio de inyección.

## 2.4 Modelos de aproximación para la simulación

A día de hoy, el uso de modelos computacionales, para predecir el comportamiento del chorro de combustible o incluso la actuación completa del motor, es de vital importancia debido al hecho de que se pueden evaluar diferentes configuraciones del caso sin hacer uso de caras pruebas experimentales. Una primera clasificación de estos modelos puede hacerse en función de su complejidad, dividiendo éstos en dos grupos: los modelos termodinámicos y los modelos fluido-dinámicos.

Los primeros son los más sencillos y se fundamentan en la primera ley de la termodinámica y la conservación de masa. Pueden ser *cero dimensionales*, incapaces de predecir cualquier aspecto geométrico del fluido, o *fenomenológicos (uno dimensionales)*, como aquéllos que tienen en cuenta la forma del chorro. En este último caso, el chorro puede quedar considerado como un chorro gaseoso, semi-estacionario y uno dimensional (desprecia efectos locales y el tamaño de las gotas) o puede ser discretizado en paquetes que modelen subprocesos locales, como la atomización y la evaporación.

Los modelos termodinámicos son útiles para hallar una estimación inicial rápida de la solución del caso planteado, pero los modelos fluido-dinámicos son más completos y precisos. Éstos resuelve el conjunto de ecuaciones que gobiernan el flujo fluido: continuidad, cantidad de movimiento, energía, etc. Debido a la complejidad del problema, es necesario plantear el problema en un entorno de dinámica de fluidos computacional (CFD) con diferentes algoritmos numéricos para la discretización del dominio y la solución.

El tratamiento del flujo turbulento es propio de cada modelo CFD y, consecuentemente, es también un criterio de clasificación:

- Simulación numérica directa (DNS). Con este planteamiento se describe el flujo turbulento sin usar ningún modelo, se resuelven las ecuaciones de Navier-Stokes directamente. Ésta es su mayor ventaja, pero establecer una malla lo suficiente fina para resolver todas las escalas turbulentas es obligatorio junto con un paso temporal muy bajo, y ello conlleva un enorme coste computacional (que se ve incrementado con el número de Reynolds) que supone el principal obstáculo para las actuales potencias de cálculo computacional. Por tato, esta metodología queda relegada sólo al estudio académico de casos simples y queda fuera del marco de la industria.
- Simulación de turbulencia de gran tamaño (large eddy simulation. LES). Esta segunda aproximación numérica de resolución de turbulencia trata de ser un compromiso entre la resolución directa y el modelado. Las escalas grandes de flujo turbulento se resuelven directamente mientras que las escalas más pequeñas de turbulencia se modelan. Un importante efecto de esto es la disminución del coste

computacional respecto a DNS. El tamaño de celda la malla es normalmente el factor que dicta (filtro espacial) a partir de qué tamaño se resolverá la turbulencia directamente y hasta qué tamaño esta turbulencia será modelada (por debajo del tamaño de celda). Así, muchos detalles del flujo turbulento pueden ser calculados. Ésta es la razón principal para el incremento del uso de este tipo de planteamiento. Un ligero contratiempo es el hecho de que se tienen que lanzar varias simulaciones LES para el mismo problema si se quiere obtener una significativa media de valores para comparar con los datos experimentales.

- Navier-Stokes promediado con Reynolds (Reynolds averaged Navier-Stokes, RANS). En la computación RANS se modela el rango completo de escalas turbulentas. Son resueltas las ecuaciones del problema para los valores medios del flujo, promediados con el Reynolds a partir de la evaluación instantánea de las mismas ecuaciones. El coste computacional es relativamente bajo incluso en problemas grandes y complejos, por lo que es aún el método de aproximación CFD más usado. La desventaja principal es el hecho de que nuevas incertidumbres aparecen durante el promedio, lo que hace necesario resolver ecuaciones adicionales para superar este problema. Con este fin, se han desarrollado una variedad de modelos de turbulencia tratando de describir lo más fielmente posible el efecto de la turbulencia.

Como un breve resumen, las simulaciones DNS reproducen todas las fluctuaciones turbulentas presentes en el problema, las simulaciones LES modelan las fluctuaciones rápidas y pequeñas de turbulencia y calcula las de mayor tamaño, y, por último, las RANS representan los valores medios del flujo a la vez que modela todos los fenómenos turbulentos.

En este trabajo, se utiliza la metodología RANS para comprobar inicialmente que los modelos y las ecuaciones usadas para resolver el caso están bien implementadas en el código y permiten realizar la simulación del problema. Tras esto, se plantea el caso para ser resuelto mediante cálculos LES y obtener resultados más precisos a cambio de realizar con un coste computacional más elevado.

Además de la clasificación según los distintos tipos de metodología usada para la turbulencia, los modelos multi-fase se pueden clasificar en otros tres grupos dependiendo del tratamiento de la interfaz entre fases: métodos de rastreo de la interfaz, métodos de guardado de la interfaz y métodos de interfaz suavizada.

### 2.4.1 Aproximaciones para la definición de interfaz

Lidiar con flujos que presentan superficies libres es una dura tarea, pues sus contornos se mueven con el mismo fluido. La posición del contorno sólo se conoce el instante inicial, por tanto, su localización en instantes posteriores debe ser determinada como parte de la

solución. Se han desarrollado muchos métodos para encontrar la forma de estas superficies, y éstos se pueden agrupar en tres tipos principales:

- Métodos de rastreo de la interfaz. Estos métodos tratan la superficie libre como una interfaz acentuada cuyo movimiento se puede seguir. En este tipo de método, el dominio del flujo es discretizado usando una aproximación por diferencias finitas y la interfaz se representa explícitamente por una red independiente y no estructurada que se mueve a través de la malla estacionaria. Ya que la interfaz se deforma continuamente, es necesario reestructurar su malla conforme los cálculos avanzan. Este método destaca por su elevada precisión pero conlleva un alto coste computacional (debido al hecho de tener que generar una red dependiente de espacio y del tiempo para la superficie) que resulta ser una gran desventaja.
- Métodos de grabado de la interfaz. En este caso, el cálculo se realiza en una malla fija y la forma de la superficie libre se determina computando la fracción de cada celda cercana a la interfaz que es rellenada parcialmente por la misma superficie libre. Esto se consigue introduciendo partículas sin masa en la superficie libre en el instante inicial y siguiendo su movimiento, lo cual es llamado esquema de marcador y celda (Marker-and-Cell scheme, MAC). Otra opción es resolver una ecuación de transporte para la fracción de celda ocupada por la fase líquida, también conocido como método de volumen de fluido (Volume-of-Fluid method, VOF).
- Métodos de interfaz difusa. En estos métodos la superficie de la interfaz no es grabada o rastreada, pero sí modelada usando expresiones algebraicas para establecer ecuaciones de transporte adicionales. El mayor exponente de este tipo de modelos aplicados a las simulaciones de chorro de combustible es la ecuación que modela la densidad de área de la interfaz principal.

Por último, los modelos CFD pueden dividirse en tres grupos más: modelos euleriano-lagrangianos, modelos eulerianos multi-fluido y modelos de flujo homogéneo.

### 2.4.2 Modelos euleriano-lagrangianos

El modelo euleriano-lagrangiano es el planteamiento clásico para estudios computacionales de chorro de combustible, lo cuál se debe a su alta eficiencia para predecir parámetros globales del chorro (penetración y ángulo) y porque todos los fenómenos que tienen lugar en la cámara de combustión (evaporación, combustión, emisión, etc.) puede ser fácilmente incluidos en la simulación. Este tipo de modelos fue originalmente desarrollado para chorros dispersos, con un límite máximo en la fracción de volumen en torno a 0.1, para asegurar el acople y la solución de todas las ecuaciones. Sin embargo, su eficiencia ha extendido su uso a chorros densos, como el chorro diésel.

En este modelo se combinan dos aproximaciones numéricas: por su lado, la fase gaseosa se resuelve usando una descripción euleriana mientras que para el líquido se

aplica una descripción lagrangiana. Esta última metodología está basada en la discretización del líquido en partículas (*parcelas*) computaciones que son inyectadas en el flujo y trazadas desde el inicio de la inyección. Cada *parcela* representa un grupo de gotas de combustible con las mismas características que no interactúan entre ellas y son usadas para describir estadísticamente el chorro. En este tratamiento estadístico de las *parcelas*, conocido como Modelo Discreto de Gotas (Discrete Droplets Model, DDM), las ecuaciones que lo gobiernan se resuelven aplicando el método de *Monte-Carlo*. Esta aproximación reduce el coste computacional, especialmente en comparación con el Modelo Continuo de Gotas (Continuum Droplets Model, CDM) donde cada gota partículas queda representada y calculada.

La interacción entre ambas fases se consigue a través de términos fuente, ponderados con la fracción vacía y añadidos a las ecuaciones del modelo. Estos términos fuente son calculados haciendo uso de diferentes sub-modelos que representan los procesos presentes en el chorro desde el comienzo de la inyección hasta la evaporación (atomización, colisión y coalescencia de las gotas, dispersión turbulenta, evaporación, etc.) y, como consecuencia, juegan un papel importante en la eficiencia del modelo. Otros parámetros clave en este tipo de simulaciones son la malla, (muy dependiente de su tamaño y estructura), y el paso temporal, por el hecho de que una resolución espacial inadecuada para los gradientes de velocidad y concentración provoca una difusión numérica excesiva. En consecuencia, la resolución de la malla no debe ser demasiado gruesa, y al mismo tiempo tampoco debe ser demasiado fina debido a la limitación de la fracción vacía. Esta es la razón principal de por qué el modelado de chorros diésel en CFD dirige a una resolución insuficiente en la zona del chorro más cercana a la tobera.

### 2.4.3 Modelos eulerianos multi-fluido

En un modelo euleriano multi-fluido, las fases de gas y líquido son tratadas como un continuo inter-penetrante en la infraestructura euleriana con campos de velocidad y temperatura diferentes. La fase gaseosa es considerada como la fase primaria, mientras que la líquida se considera como fase secundaria o dispersa. Ambas son caracterizadas a través de las fracciones volumétricas, y por definición, las fracciones volumétricas de todas las fases deben sumar la unidad.

Las ecuaciones que gobiernan el problema del modelo multi-fluido se pueden deducir de condicionalmente del conjunto promediado de las ecuaciones locales de conservación instantánea del flujo de una sola fase. Sin embargo, dado que los campos promediados de una sola fase no son independientes de la otra, se necesitan términos de interacción entre ambas fases para la transferencia de masa, cantidad de movimiento y energía entre las fases en las interfaces.

Los métodos usados para computar estos flujos son similares a aquellos usados por flujos de una fase, excepto por la adición de los términos de interacción y las condiciones

de contorno y, por supuesto, el hecho de tener que resolver el doble de ecuaciones al tener dos fases. Ya que los procesos de transferencia de cada fase son expresados por sus propias ecuaciones de equilibrio, el modelo puede predecir más cambios e interacciones entre fases más detallados que un modelo de mezcla o un modelo homogéneo. Debido a sus características, este tipo de modelos ofrecen una gran eficiencia y un enorme detalle en el análisis del fenómeno transitorio, olas de propagaciones y los cambios en los regímenes de flujo teniendo en cuenta la interacción dinámica y de no equilibrio entre fases. Por el contrario, cuando ambas fases están altamente acopladas estos modelos presentan complicaciones innecesarias y un elevado coste computacional para su aplicación práctica.

#### 2.4.4 Modelos de flujo homogéneo

La otra opción son los modelos de un único fluido. Una partícula computacional no es una partícula física real, como las *parcelas* lagrangianas. Por igual, una fase computacional en un cálculo CFD multi-fase no es igual a una fase en el sentido físico, una fase computacional representa una masa en movimiento a una misma velocidad.

La aproximación de flujo homogéneo es el modelo CFD multi-fase más simples. El concepto básico de este modelo consiste en considerar la mezcla como un todo, debiendo escribirse las ecuaciones del caso para el equilibrio de la mezcla de masas, cantidad de movimiento y energía en términos de propiedades de la mezcla. Estas tres ecuaciones de conservación macroscópica de la mezcla se suplementan entonces con una ecuación de difusión que tiene en cuenta los cambios de concentración. Se asume, entonces, equilibrio local entre las fases tanto para la continuidad, transporte y dispersión; además, todas las fases comparten la misma presión. Así pues, en cada punto las partículas se moverían con la velocidad de deslizamiento relativa a la fase gaseosa. Esto permite calcular los componentes de la velocidad del líquido a través de ecuaciones algebraicas.

Las principales ventajas comparadas con los modelos previos son: mayor velocidad de aproximación a la solución puesto que hay un número de ecuaciones a resolver significativamente menor, una descripción adecuada del tamaño de las partículas y otras distribuciones, así como del efecto de la turbulencia en la fase dispersa de la mezcla. Las desventajas que más pesan para este método son: la necesidad de utilizar pasos temporales y las limitaciones que implica para estudiar flujos donde las fases no están completamente acopladas.



## 3 Descripción del modelo e implementación

### 3.1 Descripción general del modelo

De acuerdo con la clasificación descrita en la sección 2.4, la estrategia de modelado elegida puede distinguirse como un modelo completamente euleriano de flujo homogéneo, el cual emplea un método de interfaz difusa para representar la superficie de contacto entre las fases de líquido y gas. Este tipo de aproximación se llama modelo de atomización  $\Sigma - Y$  y fue inicialmente propuesto por Vallet y Borghi (1999).

En primer lugar, se exponen las características generales del modelado del chorro de combustible y también se informará sobre los conceptos claves y fundamentales usados en dicho modelo.

Después de esta parte introductoria, serán explicadas las ecuaciones fundamentales de transporte empleadas en el modelo, prestando especial atención en su implementación dentro del código del software OpenFOAM. Así mismo, también se desarrollarán algunos sub-modelos aplicados para completar el modelo de chorro o mejorar su precisión y representar mejor el comportamiento físico del combustible, como el modelo de turbulencia o el modelo termodinámico del n-dodecano.

#### Aproximación del modelo de chorro de combustible

A pesar del gran interés práctico en la forma de fluir el chorro desde el inyector de combustible, pues el diseño del inyector es un factor crítico en el diseño del motor de combustión interna, la región más cercana a la tobera continúa siendo un reto para los modeladores del chorro. Este espacio ópticamente denso dentro de los primeros milímetros del inyector es solamente analizable aplicando técnicas especiales de diagnóstico, como la radiografía por rayos x. Por lo contrario, la región donde el chorro está más disperso es mucho más fácil de analizar, pues el diagnóstico óptico es bueno y da resultados fiables.

Por esta razón, modelar la atomización primaria es un reto particular. Aunque se ha prestado atención a numerosos modelos de atomización primaria, estos modelos no eran necesariamente predictivos. Esta parte del chorro no abarca el rango completo de fracciones de volumen líquido, de 0 a 1. Además, el núcleo líquido del chorro está tan cargado de combustible que el rastreo de la traza de las partículas lagrangianas falla en esta parte, porque casi todos los modelos de resistencia, colisión, separación y vaporización existentes se basan en la suposición de que el chorro disperso está formado

por gotas de forma esférica. La aproximación lagrangiana clásica presenta en este punto una clara desventaja que rara vez puede resolverse.

Por el contrario, los recientes modelos eulerianos han mostrado potencial para capturar las rápidas interacciones líquido-gas de esta zona del chorro. Estas aproximaciones hacen énfasis en la turbulencia de la mezcla de gas y líquido y consideran que el proceso de atomización y el consecuente transporte de energía y masa en la interfaz gota-aire no limitan el proceso de evaporación del chorro diésel. Además, también sostienen que las condiciones que tienen lugar en el interior del cilindro durante el proceso de inyección pueden ser supercríticas o estar muy cerca de ellas. Incluso si las condiciones del combustible no están exactamente sobre el punto crítico, las elevadas temperaturas y presiones indican que existe tensión superficial sobre áreas muy reducidas y que se alcanzan números de Weber extremadamente altos. Bajo estas condiciones, la interfaz gas/líquido desaparece y el modelo de aerosol se convierte por completo en un ejercicio de modelar una mezcla turbulenta. Luego, plantear un completo tratamiento euleriano de los chorros de inyección, haciendo énfasis en la mezcla turbulenta, parece presentar una descripción más apropiada del proceso físico.

Estos modelos eulerianos, en la región dispersa del chorro aguas abajo, a menudo describen la fase líquida del mismo chorro dentro del marco de referencia lagrangiano. No obstante, el presente modelo de aerosol se define por completo como euleriano ya que la fase lagrangiana puede no ser necesaria para simulaciones de combustible diésel en ambiente de alta presión. Además de esto, la transición al entorno lagrangiano no es simple y podría desencadenar algunos problemas relacionados, por ejemplo, con la conservación de la masa, la optimización del cálculo computacional en paralelo y la selección de un criterio correcto para la inicialización del tamaño de las gotas y sus velocidades durante esta transición.

#### Modelos fundamentales

Los conceptos clave y fundamentales adoptados por el modelo aquí presentado quedan recogidos brevemente en los siguientes puntos. El modelo se basa en cuatro suposiciones básicas:

- Separación de escalas de flujo. La tensión superficial y la viscosidad actúan sólo en las escalas más bajas del flujo (asociadas con la gran curvatura de la interfaz gas-líquido y los grandes gradientes de velocidad), debido a esto y a los valores de Weber y de Reynolds de las gotas del chorro, el comportamiento de las grandes escalas del flujo debe ser independiente de la tensión superficial y la viscosidad para números de Reynolds y Weber “infinitos”. Esto implica que la actuación de las grandes escalas del flujo será dependiente solamente de la variación de densidad. Como resultado, y asumiendo que el flujo del inyector opera a altos números de Reynolds y Weber, es posible suponer que una separación de las

características del flujo a gran escala, como el transporte de masa debido al proceso de atomización, que tiene lugar a escalas más pequeñas de flujo.

- Campo de velocidad medio. Aunque las fluctuaciones de velocidad del flujo a escalas pequeñas son impredecibles, el campo de velocidad medio del flujo puede ser predicho usando determinadas suposiciones como las usadas en los modelos de turbulencia basados en el Reynolds medio. La mezcla líquido-gas es tratada como un pseudo-fluido de densidad variable y que fluye con un campo de velocidad particular.
- Dispersión del líquido. La dispersión de la fase líquida dentro de la fase gaseosa se modela como un flujo líquido difusivo y turbulento, el cuál captura el efecto de la velocidad relativa entre ambas fases.
- Geometría del chorro líquido. La geometría media de las estructuras líquidas puede ser caracterizada modelando el área media de la interfaz gas-líquido por unidad de volumen.

## 3.2 Ecuaciones fundamentales e implementación

En este apartado se describe brevemente la implementación de las ecuaciones del modelo de aerosol en la plataforma OpenFOAM. La plataforma CFD OpenFOAM usa el lenguaje de programación C++, el cuál está orientado a objetos. Éste software proporciona diferentes aplicaciones como “solvers” (métodos para resolver determinados problemas) o herramientas de post-proceso de datos y un amplio rango de modelos (modelos de turbulencia, termodinámicos, etc.) implementados como “clases”. Las ecuaciones del modelo implementadas se resumen dentro de este capítulo y quedan presentadas con una notación de índice usual así como con la correspondiente sintaxis que usa OpenFOAM. Se puede realizar una comparación directa entre ambas notaciones, ya que los términos de cada ecuación están escritos en el mismo orden para las dos.

### 3.2.1 Ecuaciones que gobiernan el modelo

#### Ecuación de transporte para la ley de conservación de masa

La conservación de masa para un fluido compresible se representa con la ecuación de continuidad

$$\frac{\partial \bar{\rho}}{\partial t} + \frac{\partial \bar{\rho} \tilde{u}_i}{\partial x_i} = 0$$

*Ecuación 3.1*

dónde la densidad del fluido se representa por  $\rho$  y la velocidad en la componente  $i$  por  $u_i$ . Así mismo, esta ecuación en OpenFOAM se implementa como:

```
fvScalarMatrix rhoEqn
(
  fvm::ddt(rho)
  + fvm::div(phi, rho)
);
rhoEqn.solve();
```

La ecuación “rhoEqn” es definida como tipo “fvScalarMatrix” ya que la densidad es un escalar. Las operaciones matemáticas como la derivada respecto del tiempo y la divergencia (usadas en la expresión exterior como “fvm::ddt” y “fvm::div” respectivamente) son funciones propias de OpenFOAM, lo cuál facilita la lectura de las ecuaciones en la sintaxis de OpenFOAM. Finalmente, el término “phi” se corresponde con la velocidad del flujo en las caras de las celdas de la malla.

### Ecuación de transporte para la fracción másica de combustible líquido

Para seguir la dispersión de la fase líquida se utiliza una función indicadora que toma el valor de la unidad para la fase líquida y cero para la fase gaseosa. La fracción másica de volumen líquido media se denomina como  $\bar{Y}$  o en la sintaxis de OpenFOAM como “Ybar”, y la fracción másica de combustible media como  $\tilde{Y} = \overline{\rho_{luq}Y}/\bar{\rho}$ . La ecuación de transporte queda entonces como:

$$\frac{\partial \bar{\rho}\tilde{Y}}{\partial t} + \frac{\partial \bar{\rho}\tilde{u}_i\tilde{Y}}{\partial x_i} = - \frac{\partial \bar{\rho}\tilde{u}'_i\tilde{Y}'}{\partial x_i} - S_{evap}$$

*Ecuación 3.2*

La fracción másica de líquido es una propiedad que se conserva, sin embargo, aparece un término fuente en la ecuación de transporte, el segundo término en el lado derecho de la Ecuación 3.2, a tener en cuenta en el proceso de evaporación. Este término fuente será explicado más adelante, en la sección 3.3.2 de este mismo capítulo.

Ignorando el término fuente, el lado derecho de la Ecuación 3.2 se corresponde con el término de difusión turbulenta del flujo líquido, que recoge el efecto de la velocidad relativa entre las dos fases y es un parámetro fundamental de este tipo de modelos. El término  $u'$  denota la fluctuación turbulenta de velocidad ponderada con la densidad, y el término  $Y'$  la fluctuación turbulenta de la fracción másica de líquido. El modelado de este término puede estar basado en diferentes hipótesis, desde suponer un gradiente a resolver una ecuación de transporte. En este proyecto, el modelo se basa en suponer un gradiente para este término turbulento, ya que en otros trabajos (basados en

comparaciones con cálculos DNS) se ha probado que una ley de gradiente es suficiente en chorros diésel. De esta manera, aquí se formula:

$$\bar{\rho} \overline{u'_i \tilde{Y}'} = - \frac{\mu_t}{Sc} \frac{\partial \tilde{Y}}{\partial x_i}$$

*Ecuación 3.3*

dónde  $\mu_t$  es la viscosidad turbulenta y  $Sc$  el número de Schmidt. Mientras que la aproximación empleada asume que la cantidad de movimiento de la mezcla líquido-gas resuelto puede ser caracterizado por una velocidad ponderada particular, la velocidad de deslizamiento entre líquido y gas se puede expresar explícitamente como la derivada de Demoulin y queda reflejada en la Ecuación 3.4:

$$u_{i|l} - u_{i|g} = \frac{1}{\tilde{Y}(1 - \tilde{Y})} \overline{u'_i \tilde{Y}'}$$

*Ecuación 3.4*

Finalmente, la Ecuación 3.2 se convierte en la Ecuación 3.5:

$$\frac{\partial \bar{\rho} \tilde{Y}}{\partial t} + \frac{\partial \bar{\rho} \tilde{u}_i \tilde{Y}}{\partial x_i} - \frac{\partial}{\partial x_i} \left( \frac{\mu_t}{Sc} \frac{\partial \tilde{Y}}{\partial x_i} \right) = S_{evap}$$

*Ecuación 3.5*

Esta ecuación en la sintaxis de OpenFOAM se expresa como:

```
fvScalarMatrix YEqn
(
  fvm::ddt(rho, Y)
  + fvm::div(rhoPhi, Y)
  - fvm::laplacian(turbulence->mut()/Sc, Y)
  ==
  - Sevap
);
YEqn.solve();
```

En una implementación de volumen finito, los flujos “rhoPhi” son los componentes normales al flujo másico entre las caras de las celdas. La ecuación “YEqn” es del tipo “fvScalarMatrix” debido a que la fracción másica de líquido es también un escalar. Otro operador de OpenFOAM aparece implementado en la ecuación, el operador laplaciano (“fvm::laplacian”). Por último, el término “YEqn.solve();” llama a la solución de la ecuación.

### Ecuación de transporte de la cantidad de movimiento

La velocidad se resuelve a partir de la cantidad de movimiento:

$$\frac{\partial \bar{\rho} \tilde{u}_j}{\partial t} + \frac{\partial \bar{\rho} \tilde{u}_i \tilde{u}_j}{\partial x_i} + \frac{\partial}{\partial x_i} \left( \bar{\rho} \overline{u'_i u'_j} - \mu \frac{\partial \tilde{u}_j}{\partial x_i} \right) = - \frac{\partial \tilde{p}}{\partial x_j}$$

*Ecuación 3.6*

En esta ecuación, el tercer término del lado izquierdo es el correspondiente a las tensiones debidas al Reynolds ( $\bar{\rho} \tilde{u}_i \tilde{u}_j$ ) junto con las tensiones normal y cortante ( $\mu \frac{\partial \tilde{u}_j}{\partial x_i}$ ).

La ecuación en sintaxis de OpenFOAM:

```
fvVectorMatrix UEqn
(
    fvm::ddt(rho, U)
    + fvm::div(rhoPhi, U)
    + turbulence->divDevRhoReff(U)
    ==
    - fvc::grad(p)
);
UEqn.solve();
```

La ecuación “UEqn” es definida como “fvVectorMatrix” ya que la velocidad sí es una magnitud vectorial. Otro útil uso de OpenFOAM es el término correspondiente a la tensión de Reynolds (“turbulence->divDevRhoReff(U)”), en cada modelo turbulento es esta función la que permite su fácil implementación.

### Ecuación de transporte de la energía

La ley de conservación de la energía establece que la energía total de un sistema aislado permanece constante, ésta se conserva a lo largo del tiempo y tampoco se crea o se destruye pero sí se transforma de una forma a otra. Es más, en la mayoría de problemas multi-fase con gran adición de calor, los efectos de la energía mecánica o cinética pueden ser despreciados, y el único efecto importante a tener en cuenta es la difusión en el transporte de energía térmica entre fases por su diferencia de entalpía. Como consecuencia, se lleva a cabo la distinción entre energía térmica y mecánica, despreciando esta última. Aquí se considera la variable  $h$  como la entalpía estática de la mezcla, y aparece en la Ecuación 3.7. En esta ecuación,  $\alpha_{ef}$  representa la difusividad térmica efectiva compuesta a partir de la contribución laminar y turbulenta con la difusividad térmica turbulenta  $\alpha_t = \mu_t / Pr$ . Se suele usar un valor constante para el número turbulento de Prandtl,  $Pr$ . Y el término  $\tau_{ij} \frac{\partial \tilde{u}_j}{\partial x_i}$  constituye la disipación viscosa.

$$\frac{\partial \bar{\rho} \tilde{h}}{\partial t} + \frac{\partial \bar{\rho} \tilde{u}_i \tilde{h}}{\partial x_i} - \frac{\partial}{\partial x_i} \left( \alpha_{ef} \frac{\partial \tilde{h}}{\partial x_i} \right) = \frac{\partial \tilde{p}}{\partial t} + \frac{\partial \tilde{p}}{\partial x_i} + \tau_{ij} \frac{\partial \tilde{u}_j}{\partial x_i}$$

*Ecuación 3.7*

En la implementación de OpenFOAM la difusividad térmica turbulenta efectiva se calcula como “turbulence->alphaEff()”, otra función de OpenFOAM implementada en todos los modelos turbulentos. Por tanto, los dos primeros términos del lado derecho de la Ecuación 3.7 se corresponden con la derivada total de la presión, evaluable con el operador “DDT” y el último término de la implementación en OpenFOAM es la disipación viscosa:

```
fvVectorMatrix hEqn
(
    fvm::ddt(rho, h)
    + fvm::div(rhoPhi, h)
    - fvm::laplacian(turbulence->alphaEff(), h)
    ==
    + fvc::DDt(phi, p)
    + turbulence->muEff()*
    ((fvc::grad(U) + fvc::grad(U)().T()) && fvc::grad(U))
);
hEqn.solve();
```

Para determinar la evolución de la temperatura de la entalpía transportada, se considera inicialmente el calor específico para presión constante ( $c_{p,k}$ ), tanto de la fase líquida como gaseosa, para calcular la temperatura a partir de la entalpía:

$$\tilde{h} = \tilde{Y} \cdot h_{f,l}(T) + (1 - \tilde{Y}) \cdot h_g(T) = \tilde{Y} \cdot c_{p,l} \cdot T + (1 - \tilde{Y}) \cdot c_{p,g} \cdot T$$

*Ecuación 3.8*

$$T = \frac{\tilde{h}}{\tilde{Y} \cdot c_{p,l} + (1 - \tilde{Y}) \cdot c_{p,g}}$$

*Ecuación 3.9*

De esta manera, se desarrolla un proceso iterativo, que es explicado en el apartado 3.3.3 de este capítulo, y así calcular la evolución de la temperatura mientras se considera la dependencia de la temperatura como el calor específico a presión constante ( $c_{p,l}(T)$ ), con el fin de mejorar la precisión y representar más fielmente el comportamiento del aerosol.

Ecuación de transporte para la densidad de área de interfaz

La escala de atomización pequeña se modela resolviendo la ecuación de transporte para la evolución de la densidad de área de la superficie de la interfaz entre fases,  $\Sigma$ . Esta densidad de superficie se describe como la cantidad de superficie por unidad de volumen en un instante y situación espacial dados. No es fácil establecer una ecuación de equilibrio para esta magnitud, incluso si es de forma abierta. En este trabajo se asume la existencia de un gradiente para el término de la difusión turbulenta del flujo, donde  $D_\Sigma$  es un coeficiente de difusión aceptable, obtenido a partir del número de Schmidt y la viscosidad cinemática turbulenta ( $D_\Sigma = \nu_t/Sc_\Sigma$ ):

$$\frac{\partial \tilde{\Sigma}}{\partial t} + \frac{\partial \tilde{u}_i \tilde{\Sigma}}{\partial x_i} - \frac{\partial}{\partial x_i} \left( D_\Sigma \frac{\partial \tilde{\Sigma}}{\partial x_i} \right) - a \tilde{\Sigma} + b \tilde{\Sigma}^2 - S_{\Sigma evap} - S_{\Sigma init} = 0$$

*Ecuación 3.10*

dónde la inversa de la escala temporal  $a$  y el coeficiente  $b$  pueden entenderse como la generación de superficie debida al crecimiento de las inestabilidades del fluido y la destrucción de superficie por la coalescencia de las gotitas respectivamente. Sin embargo, la forma más común para combinar estos dos términos fuente es recuperar un valor de equilibrio ( $\tilde{\Sigma}_{eq}$ ):

$$\frac{\partial \tilde{\Sigma}}{\partial t} + \frac{\partial \tilde{u}_i \tilde{\Sigma}}{\partial x_i} - \frac{\partial}{\partial x_i} \left( D_\Sigma \frac{\partial \tilde{\Sigma}}{\partial x_i} \right) - c_\Sigma \tilde{\Sigma} \left( 1 - \frac{\tilde{\Sigma}}{\tilde{\Sigma}_{eq}} \right) - S_{\Sigma evap} - S_{\Sigma init} = 0$$

*Ecuación 3.11*

El término  $S_{\Sigma evap}$  representa la contribución de evaporación en el término fuente, el cual es explicado en la sección 3.3.2 de este capítulo. Finalmente, el último término del lado izquierdo de la ecuación anterior ( $S_{\Sigma init}$ ) es un término de inicialización que también es discutido más adelante.

Según la sintaxis de OpenFOAM, la Ecuación 3.11 tiene el aspecto siguiente:

```
fvScalarMatrix SigmaEqn
(
  fvm::ddt(Sigma)
  + fvm::div(phi, Sigma)
  - fvm::laplacian(nut/ScSigma, Sigma)
  - fvm::Sp(Csigma, Sigma)
  + fvm::Sp(Csigma*Sigma/SigmaEq, Sigma)
  - fvm::Sp(SsigmaEvap, Sigma)
  - Sinit
);
SigmaEqn.solve();
```



El término  $\tilde{\Sigma}_{eq}$  es la densidad de superficie de equilibrio o superficie crítica para la cuál la densidad de superficie local es establecida por un radio de gota adecuado para el equilibrio ( $r_{eq}$ ):

$$\tilde{\Sigma}_{eq} = \frac{3 \bar{\rho} \tilde{Y}}{\rho_l r_{eq}}$$

*Ecuación 3.12*

Se puede ver que todos los términos fuente implicados en la Ecuación 3.11 son proporcionales a la densidad de superficie de la interfaz líquido-gas  $\Sigma$ . Consecuentemente, si no hay interfaz tampoco hay producción de combustible gaseoso. Luego, una adecuada inicialización debe llevarse a cabo mediante el término  $S_{\Sigma init}$ . Con este propósito, se considera un valor mínimo de la densidad de superficie de interfaz en cualquier celda que no esté completamente llena de gas o de líquido puro. Este valor se estima como  $V^{-1/3}$  dónde  $V$  es el volumen de la celda. El término fuente sólo toma valores positivos si el campo de la interfaz es menor que la densidad de superficie mínima:

$$S_{\Sigma init} = \frac{\Sigma_{min} - \Sigma}{\Delta t} pos(\Sigma_{min} - \Sigma)$$

*Ecuación 3.13*

dónde  $pos$  es un operador booleano de OpenFOAM:

$$pos(x) = \begin{cases} 1 & \text{si } x > 0 \\ 0 & \text{si } x \leq 0 \end{cases}$$

Junto con el promedio de la fracción de masa líquida, la densidad de área en la interfaz puede emplearse para obtener parámetros relacionados con el tamaño de las gotas de la atomización: el diámetro local de las gotas ( $D_{32}$ ) o la densidad del número de gotas.

$$D_{32} = \frac{6 \bar{\rho} \tilde{Y}}{\rho_l \tilde{\Sigma}}$$

*Ecuación 3.14*

### Ecuación de densidad

Asumiendo que las dos fases conforman una mezcla inmiscible, la fracción másica promedio de líquido se relaciona con la densidad a través de la siguiente ecuación:

$$\frac{1}{\bar{\rho}} = \frac{\tilde{Y}}{\rho_l} + \frac{1 - \tilde{Y}}{\rho_g}$$

*Ecuación 3.15*

Una ecuación de estado se asigna entonces a cada fase. La del gas se rige según la ley de los gases ideales,

$$\rho_g = \frac{p}{R_g T}$$

*Ecuación 3.16*

dónde  $R_g$  es la constante específica de la mezcla de gases presente en la cámara de combustión, que se obtiene dividiendo la constante universal de los gases ideales ( $R_u = 8.314472 \text{ J/mol K}$ ) entre el peso molecular (PM) de la mezcla. De esta manera se tienen en cuenta todos los gases de la mezcla.

Por otro lado, para la fase líquida se implementan dos opciones. La primera se corresponde con la hipótesis de que la compresibilidad del líquido es lineal, denominada como  $\psi_l$ :

$$\rho_l = \rho_{l,o} + \psi_l (p - p_o)$$

*Ecuación 3.17*

dónde  $\rho_{l,o}$  y  $p_o$  hacen referencia a la densidad y presión, respectivamente, sobre las cuales la ecuación de estado se linealiza. Esta ecuación de estado se usa en aplicaciones donde el combustible es un componente puro. Para la simulación del combustible se ha implementado la misma ecuación avanzada de estado para líquidos que emplea el NIST (National Institute of Standards and Technology de EEUU).

#### Ecuación de presión, Acople presión-velocidad

Para obtener una ecuación que permita resolver la presión, la ecuación de la cantidad de movimiento deber satisfacer la ecuación de continuidad, lógicamente. Para ello, se ha adoptado el algoritmo de Jasak (1996) de una única fase de flujo incompresible y se ha ampliado para que se pueda usar con flujos compresibles, multi-fase y de temperatura variable, a la vez que también se ha debido reescribir la ecuación de la cantidad de movimiento en forma semi-discretizada:

$$a_p U_p = H(U) - \nabla p$$

*Ecuación 3.18*

dónde la matriz de cantidad de movimiento ha sido dividida, separando su diagonal  $a_p$  del resto de la matriz  $H(U)$ . Reordenando la Ecuación 3.17, interpolando entre las caras de las celdas y aplicando realizando la divergencia a ambos lados, se tiene:

$$\nabla U_p = \nabla \left( \frac{H(U)}{a_p} \right) - \nabla \left( \frac{1}{a_p} \nabla p \right)$$

*Ecuación 3.19*

Para un flujo compresible y multi-fase, la divergencia de la velocidad es distinta de cero y puede dividirse entre los efectos de compresibilidad, expansión térmica y mezcla turbulenta con tan sólo aplicar la regla de la cadena a la ecuación de la continuidad.

$$\nabla U_p = -\frac{1}{\rho} \frac{D\rho}{Dt} = -\frac{1}{\rho} \frac{\partial \rho}{\partial P} \frac{DP}{Dt} - \frac{1}{\rho} \frac{\partial \rho}{\partial T} \frac{DT}{Dt} - \frac{1}{\rho} \frac{\partial \rho}{\partial Y} \frac{DY}{Dt} + \frac{\rho - \rho_{EOS}}{\delta_t k_r \rho}$$

*Ecuación 3.20*

Para obtener una ecuación de transporte de presión totalmente cerrada, cada uno de estos términos debe ser manipulado de una forma que sea numéricamente estable.

**Efectos de la compresibilidad.** La divergencia debida a la compresibilidad para elevados números de Mach es originada por el término:

$$-\frac{1}{\rho} \frac{\partial \rho}{\partial P} \frac{DP}{Dt}$$

*Ecuación 3.21*

La derivada de la densidad respecto de la presión es compresibilidad isotérmica efectiva de la mezcla aire-líquido y se define como  $\psi$ . Su valor se obtiene derivando la Ecuación 3.15 respecto a la presión y realizando un par de simplificaciones, llegando a la expresión:

$$\frac{\partial \rho}{\partial P} = \frac{\tilde{Y} \rho}{\rho_l} \frac{\partial \rho_l}{\partial P} + \frac{(1 - \tilde{Y})\rho}{P} = \rho \left( \frac{\tilde{Y}}{\rho_l} \frac{\partial \rho_l}{\partial P} + \frac{1 - \tilde{Y}}{P} \right)$$

*Ecuación 3.22*

La derivada total de la presión es

$$\frac{DP}{Dt} = \frac{\partial P}{\partial t} + u \nabla P$$

*Ecuación 3.23*

Y por tanto, el término de la compresibilidad queda como sigue:

$$-\frac{1}{\rho} \frac{\partial \rho}{\partial P} \frac{DP}{Dt} = -\left( \frac{\tilde{Y}}{\rho_l} \frac{\partial \rho_l}{\partial P} + \frac{1 - \tilde{Y}}{P} \right) \left( \frac{\partial P}{\partial t} + u \nabla P \right)$$

*Ecuación 3.24*

**Efecto de la expansión térmica.** Para cuantificar el efecto del cambio de la temperatura sobre la densidad se utiliza un método similar al anterior. El término debido a la temperatura es

$$-\frac{1}{\rho} \frac{\partial \rho}{\partial T} \frac{DT}{Dt}$$

*Ecuación 3.25*

La derivada de la Ecuación 3.15, en combinación de la Ecuación 3.16 y tras realizar un par de simplificaciones, resulta:

$$\frac{\partial \rho}{\partial T} = \frac{\tilde{Y} \rho \partial \rho_l}{\rho_l \partial T} - \frac{(1 - \tilde{Y}) \rho}{T} = \rho \left( \frac{\tilde{Y} \partial \rho_l}{\rho_l \partial T} - \frac{1 - \tilde{Y}}{T} \right)$$

*Ecuación 3.26*

La derivada total de la temperatura es

$$\frac{DT}{Dt} = \frac{\partial T}{\partial t} + u \nabla T$$

*Ecuación 3.27*

Finalmente, la expresión correspondiente a la expansión térmica puede escribirse como:

$$-\frac{1}{\rho} \frac{\partial \rho}{\partial T} \frac{DT}{Dt} = - \left( \frac{\tilde{Y} \partial \rho_l}{\rho_l \partial T} + \frac{1 - \tilde{Y}}{T} \right) \left( \frac{\partial T}{\partial t} + u \nabla T \right)$$

*Ecuación 3.28*

**Efecto de la mezcla multi-fase turbulenta.** A lo largo de una línea de corriente del flujo, la mezcla turbulenta de alta densidad líquida con baja densidad de gas puede tener un gran peso en la divergencia del campo de velocidades, y este efecto se contabiliza a través del término:

$$-\frac{1}{\rho} \frac{\partial \rho}{\partial \tilde{Y}} \frac{D\tilde{Y}}{Dt}$$

*Ecuación 3.29*

La derivada parcial de la densidad respecto de la fracción másica es:

$$\frac{\partial \rho}{\partial \tilde{Y}} = -\rho^2 \left( \frac{1}{\rho_l} - \frac{1}{\rho_g} \right)$$

*Ecuación 3.30*

La derivada total de la fracción másica de líquido:

$$\frac{D\tilde{Y}}{Dt} = \frac{\partial \tilde{Y}}{\partial t} + u \nabla \tilde{Y}$$

*Ecuación 3.31*

y esta ecuación se relaciona con la ecuación de transporte de  $\tilde{Y}$  (Ecuación 3.5). Tomando sólo el lado izquierda de esta ecuación y aplicando la regla de la cadena:

$$\frac{\partial \rho \tilde{Y}}{\partial t} + \nabla(\rho u \tilde{Y}) = \rho \frac{\partial \tilde{Y}}{\partial t} + \tilde{Y} \frac{\partial \rho}{\partial t} + \tilde{Y} \nabla \rho u + \rho u \nabla \tilde{Y} = \rho \left( \frac{\partial \tilde{Y}}{\partial t} + u \nabla \tilde{Y} \right) + \tilde{Y} \left( \frac{\partial \rho}{\partial t} + \nabla \rho u \right)$$

*Ecuación 3.32*

En esta expresión, el primer término del lado derecho representa la derivada total de  $\tilde{Y}$  y el segundo, el interior del paréntesis, se corresponde con la ecuación de la continuidad. Por tanto, este último término es cero, y la expresión queda como sigue:

$$\frac{\partial \rho \tilde{Y}}{\partial t} + \nabla(\rho u \tilde{Y}) = \rho \frac{D\tilde{Y}}{Dt}$$

*Ecuación 3.33*

y la derivada total de  $\tilde{Y}$ , partiendo de la Ecuación 3.5, es:

$$\frac{D\tilde{Y}}{Dt} = \frac{1}{\rho} \left( \frac{\partial}{\partial x_i} \left( \frac{\mu_t}{Sc} \frac{\partial \tilde{Y}}{\partial x_i} \right) - S_{evap} \right)$$

*Ecuación 3.34*

Finalmente, el término total de la mezcla turbulenta puede escribirse como:

$$-\frac{1}{\rho} \frac{\partial \rho}{\partial \tilde{Y}} \frac{D\tilde{Y}}{Dt} = \left( \frac{1}{\rho_l} - \frac{1}{\rho_g} \right) \left( \frac{\partial}{\partial x_i} \left( \frac{\mu_t}{Sc} \frac{\partial \tilde{Y}}{\partial x_i} \right) - S_{evap} \right)$$

*Ecuación 3.35*

Sustituyendo cada uno de estos términos de vuelta en la Ecuación 3.19 y la Ecuación 3.20, la ecuación de transporte final para la presión resulta:

$$\begin{aligned} \nabla \left( \frac{1}{a_p} \nabla p \right) - \nabla \left( \frac{H(U)}{a_p} \right) &= \left( \frac{\tilde{Y}}{\rho_l} \frac{\partial \rho_l}{\partial P} + \frac{1 - \tilde{Y}}{P} \right) \left( \frac{\partial P}{\partial t} + u \nabla P \right) + \left( \frac{\tilde{Y}}{\rho_l} \frac{\partial \rho_l}{\partial T} - \frac{1 - \tilde{Y}}{T} \right) \left( \frac{\partial T}{\partial t} + u \nabla T \right) + \\ &+ \left( \frac{1}{\rho_l} - \frac{1}{\rho_g} \right) \left( \frac{\partial}{\partial x_i} \left( \frac{\mu_t}{Sc} \frac{\partial \tilde{Y}}{\partial x_i} \right) - S_{evap} \right) + \frac{\rho - \rho_{EOS}}{\delta_t k_r \rho} \end{aligned}$$

*Ecuación 3.36*

Este planteamiento de las ecuaciones y la forma de resolverse se inspiran en el acople presión-velocidad de los métodos PISO. Cabe señalar que este acople entre ecuaciones se realiza usando una aproximación implícita segregada, en la cuál las ecuaciones son formuladas para cada dependiente variable y resueltas secuencialmente, con la posibilidad de realizar iteraciones dentro del sistema de ecuaciones hasta que se alcance la convergencia de los resultados.

La ecuación, según la sintaxis que sigue OpenFOAM, se implementa tal que así:

```
fvScalarMatrix pEqn
(
  fvm::laplacian(rUAf, p)
  - fvc::div(phi)
  ==
  (Ybar/rholiq*devRholiqP+(1-Ybar)/p)*
  (fvm::ddt(p)+fvm::div(phi, p)-fvm::Sp(fvc::div(phi), p))
  + (Ybar/rholiq*devRholiqT-(1-Ybar)/T)*
  (fvm::ddt(T)+fvm::div(phi, T)-fvm::Sp(fvc::div(phi), T))
  + Prho*(-fvm::laplacian(turbulence->mut())/Sc, Y)+Sevap)
  + (rhoEOS-rho)/runTime.deltaT()/rhoRelaxationFactor/rho
);
pEqn.solve();
```

El término “rUAf” se corresponde con el coeficiente diagonal de la ecuación de la cantidad de movimiento, “devRholiqP” y “devRholiqT” son las derivadas parciales de la densidad líquida calculadas numéricamente, e “Ybar” es la fracción de volumen líquido. Por su parte, “runTime.deltaT()” es el valor del paso temporal y “rhoRelaxationFactor” es la constante  $k_r$ , un factor de relajación para facilitar el cálculo.

### 3.3 Desarrollo de los modelos

La implementación de modelos que representen lo más fielmente las características de los compuestos que participan en la formación del chorro diésel es de vital importancia para llegar a una solución muy aproximada de dicho fenómeno. Aquí se describen los demos de turbulencia, evaporación y la ecuación de estado del combustible líquido.

#### 3.3.1 Modelo de turbulencia

Como ya se mencionó en el capítulo 2, en este trabajo se realizan tanto aproximaciones RANS como LES para representar el efecto de la turbulencia, y es la manera en la que se calcula la que marca las mayores diferencias entre ambos tipos de cálculo. En OpenFOAM hay varios modelos de turbulencia disponibles basados en la suposición de viscosidad turbulenta ( $k$ - $\epsilon$  estándar, RNG  $k$ - $\epsilon$ ,  $k$ - $\epsilon$  realizable). Sin embargo, se han desarrollado e implementado dos modelos de turbulencia especial para este tipo de fenómenos (high density ratio) que no están recogido en OpenFOAM. Ambos modelos

son explicados a continuación: el primero de ellos ha sido desarrollado para el modelo de cálculo RANS, mientras que el segundo se aplica en el modelo de cálculo LES.

### Modelo de turbulencia high density ratio k-ε

Este modelo de turbulencia, aplicado sólo para el modelo RANS, consiste en una forma modificada del modelo k-ε que permite solventar las limitaciones existentes del modelo k-ε estándar para la atomización del chorro: primero, permite manejar un rango de densidades más amplio que el inicial, y segundo, elimina la difusión molecular del gradiente de densidad para pequeña escalas de turbulencia.

Ahora bien, éste modelo continúa basándose en la hipótesis de vórtices de turbulencia viscosos y se compone de dos ecuaciones de transporte para la energía cinética turbulenta  $k$  (Ecuación 3.37) y su tasa de disipación  $\varepsilon$  (Ecuación 3.38) más una tercera para la viscosidad turbulenta  $\mu_t$  (Ecuación 3.39):

$$\frac{\partial \bar{\rho} k}{\partial t} + \frac{\partial \bar{\rho} \tilde{u}_j k}{\partial x_j} = \frac{\partial}{\partial x_j} \left[ \left( \mu + \frac{\mu_t}{\sigma_j} \right) \frac{\partial k}{\partial x_j} \right] + P_k - \overline{u_i''} \frac{\partial \bar{p}}{\partial x_i} - \bar{\rho} \varepsilon$$

*Ecuación 3.37*

$$\frac{\partial \bar{\rho} \varepsilon}{\partial t} + \frac{\partial \bar{\rho} \tilde{u}_j \varepsilon}{\partial x_j} = \frac{\partial}{\partial x_j} \left[ \left( \mu + \frac{\mu_t}{\sigma_j} \right) \frac{\partial \varepsilon}{\partial x_j} \right] + C_{\varepsilon 1} \frac{\varepsilon}{k} \left( P_k - \overline{u_i''} \frac{\partial \bar{p}}{\partial x_i} \right) - C_{\varepsilon 2} \bar{\rho} \frac{\varepsilon^2}{k} - C_{\varepsilon 3} \frac{\varepsilon}{k} \bar{\rho} \varepsilon \frac{\partial \tilde{u}_j}{\partial x_j}$$

*Ecuación 3.38*

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

*Ecuación 3.39*

Las ecuaciones anteriores cuentan con un término adicional para contabilizar la generación de turbulencia debida a las grandes fluctuaciones de densidad entre los fluidos pesados y ligeros por el efecto del gradiente presión al que están sometidos:  $\overline{u_i''} \frac{\partial \bar{p}}{\partial x_i}$ . Las fluctuaciones de la velocidad promedio, Ecuación 3.37 y Ecuación 3.38, vienen dadas por la expresión:

$$\overline{u_i''} = \bar{\rho} \widetilde{u_i' Y'} \left( \frac{1}{\rho_l} - \frac{1}{\rho_g} \right)$$

*Ecuación 3.40*

y tras usar la Ecuación 3.3 y la Ecuación 3.39 se obtiene la expresión final de la fluctuación de velocidad promedio:

$$\overline{u_i''} = \bar{\rho} \frac{C_\mu}{Sc} \frac{k^2}{\varepsilon} \left( \frac{1}{\rho_l} - \frac{1}{\rho_g} \right) \frac{\partial \tilde{Y}}{\partial x_i}$$

*Ecuación 3.41*

El término de producción  $P_k$  se define como:

$$P_k = -\bar{\rho} \overline{u_i' u_j'} \frac{\partial \tilde{u}_i}{\partial x_i}$$

*Ecuación 3.42*

Los coeficientes del modelo k- $\epsilon$  para alta densidad quedan recogidas en la siguiente tabla, junto con el número de Schmidt:

$C_\mu$	$C_{\epsilon 1}$	$C_{\epsilon 2}$	$C_{\epsilon 3}$	$Sc$
0.09	1.44	1.92	-0.33	0.9

*Tabla 3.1. Coeficientes del modelo de turbulencia de alta densidad k- $\epsilon$*

### Modelo de turbulencia basado en viscosidad, modelo $\sigma$

La mayoría de los modelos de turbulencia usados para simulaciones LES (Large Eddy Simulations, Simulaciones de Gran Vorticidad) hacen uso de la hipótesis de viscosidad turbulenta, ya que ésta conlleva un bajo coste computacional. Lógicamente, esta hipótesis se aplica para escalas inferiores al tamaño de las celdas de la malla (subgrid-scale, SGS). Un modelo bastante utilizado en problemas de flujo turbulento y no homogéneo, como el estudiado en el presente proyecto, es el modelo dinámico de Smagorinsky.

Este modelo de turbulencia, modelo  $\sigma$ , al igual que el de Smagorinsky, está apoyado en la hipótesis de viscosidad turbulencia aplicada a escalas de turbulencia de tamaño inferior al de las celdas de la malla y presenta muchas similitudes con él; pero propone un operador diferencial basado en valores singulares del tensor del gradiente de velocidad como base para un modelo mejorado de viscosidad turbulenta para escalas turbulentas inferiores al tamaño de las celdas de la malla. Este modelo consigue un comportamiento ligeramente más fiel a la realidad que el del de Smagorinsky con menor coste computacional y es más fácil de implementar. Así mismo, también presenta un gran potencial para flujos turbulentos no homogéneos con contornos de pared.

Se obtiene esta mejora con respecto al modelo de Smagorinsky ya que formula sus ecuaciones de acuerdo con tres objetivos deseables (desde un punto de vista físico y computacional) que un modelo de turbulencia debería lograr y que Smagorinsky no consigue al completo:

- Comportamiento cúbico cercano a los contornos sólidos.
- Generar cero viscosidad turbulenta para escalas inferiores al tamaño de malla para flujos de dos componentes o bidimensionales.
- Generar cero viscosidad turbulenta para escalas inferiores al tamaño de malla para expansiones o compresiones axi-simétricas o isotrópicas del flujo.

En este caso, el tensor de escalas inferiores al tamaño de malla (SGS tensor) se escribe:



$$\tau_{ij}^{SGS} - \frac{1}{3} \tau_{kk}^{SGS} \delta_{ij} = 2 \bar{\rho} \nu_{SGS} \left( \tilde{S}_{ij} - \frac{1}{3} \tilde{S}_{kk} \delta_{ij} \right)$$

Ecuación 3.43

$$\tilde{S}_{ij} = \frac{1}{2} (\tilde{g}_{ij} + \tilde{g}_{ji}), \quad \tilde{g}_{ij} = \frac{\partial \tilde{u}_i}{\partial x_j}$$

Ecuación 3.44

En las ecuaciones anteriores,  $\tilde{S}_{ij}$  y  $\tilde{g}_{ij}$  son los tensores de esfuerzos y del gradiente de velocidad para las escalas resueltas. La viscosidad de las sub-escalas turbulentas se define:

$$\nu_{SGS} = (C_m \Delta)^2 D_m(\mathbf{u})$$

Ecuación 3.45

dónde  $C_m$  es la constante del modelo,  $\Delta$  es la longitud característica para sub-escala en cuestión (en la práctica es el tamaño de la malla) y  $D_m$  es el operador diferencial del modelo, homogéneo con la frecuencia y función del campo de velocidad resuelto  $\mathbf{u} = u_i$ . El operador más ampliamente utilizado es el de gradiente de esfuerzos, que conduce al modelo de Smagorinsky:  $D_m = D_s = \sqrt{2S_{ij}S_{ij}}$  y  $C_m = C_s \approx 0.18$ .

Para desmarcarse de Smagorinsky y cumplir con los tres objetivos anteriores, este modelo estático plantea un desarrollo diferente para obtener el operador diferencial y la constante del modelo. En este caso, el operador se construye considerando el tensor de esfuerzos y se formula:

$$D_m = D_\sigma = \frac{\sigma_3(\sigma_1 - \sigma_2)(\sigma_2 - \sigma_3)}{\sigma_1^2}$$

Ecuación 3.46

dónde las variables  $\sigma_i$  deben cumplir con la condición de ser positivos y mayores que cero,  $\sigma_1 \geq \sigma_2 \geq \sigma_3 \geq 0$ , para cumplir con los tres objetivos descritos antes y son los valores del tensor del gradiente de velocidad,  $\mathbf{g} = (\tilde{g}_{ij})$ , y resultan de realizar la raíz cuadrada de los valores propios de la matriz:  $\mathbf{G} = \mathbf{g}^t \mathbf{g}$ . Para cumplir con las tres propiedades anteriores, por los efectos de la viscosidad y los esfuerzos, el campo de velocidad en la pared, siendo “y” la normal a la pared, se define como:

$$u_1 = a \cdot y + O(y^2), \quad u_2 = b \cdot y^2 + O(y^3), \quad u_3 = c \cdot y + O(y^2)$$

Ecuación 3.47

Por tanto, el tensor del gradiente de velocidad y la matriz  $\mathbf{G}$  se escriben como sigue:

$$\mathbf{g} = \begin{bmatrix} a_x y + O(y^2) & a + O(y) & a_z y + O(y^2) \\ b_x y^2 + O(y^3) & 2b y + O(y^2) & b_z y^2 + O(y^3) \\ c_x y + O(y^2) & c + O(y) & c_z y + O(y^2) \end{bmatrix}$$

Ecuación 3.48

$$\mathbf{G} = \begin{bmatrix} (a_x^2 + c_x^2)y^2 + O(y^3) & (aa_x + cc_x)y + O(y^2) & (a_x a_z + c_x c_z)y^2 + O(y^3) \\ (aa_x + cc_x)y + O(y^2) & (a + c) + O(y) & (aa_z + cc_z)y + O(y^2) \\ (a_x a_z + c_x c_z)y^2 + O(y^3) & (aa_z + cc_z)y + O(y^2) & (a_z^2 + c_z^2)y^2 + O(y^3) \end{bmatrix}$$

Ecuación 3.49

Por otro lado, la constante del modelo se obtiene de forma experimental y tiene un valor cercano a 1.35. Finalmente, quedan el operador diferencial del modelo y la viscosidad de las subescalas descritas tal que:

$$D_m = D_\sigma = \frac{\sigma_3(\sigma_1 - \sigma_2)(\sigma_2 - \sigma_3)}{\sigma_1^2}, \quad \nu_{SGS} = (C_\sigma \Delta)^2 D_\sigma$$

Ecuación 3.50

### 3.3.2 Modelo de evaporación

El modelo de evaporación ha sido desarrollado alrededor de las características de la tecnología actual de los motores. Los elevados empuje y presión de inyección junto con el pequeño diámetro del orificio de salida de la tobera dan lugar a un régimen de atomización completa del chorro de combustible muy cerca de la tobera. El proceso de atomización origina gotitas tan pequeñas que, en condiciones reales del motor, el transporte de masa y energía en la superficie de las mismas gotas deja de ser una limitación para la evaporación y mezcla del chorro diésel. Muchos resultados experimentales de diferentes autores han mostrado que la máxima longitud líquida queda determinada por la densidad y la temperatura del aire dentro del cilindro así como también por el diámetro del inyector y el combustible. Estos experimentos confirman la hipótesis de que el proceso de vaporización queda limitado por el dosado. Las gotas de combustible alcanzan el equilibrio dinámico con el aire que lo rodea muy cerca de la tobera, lo que conduce a una tasa local de transferencia de cantidad de movimiento, masa y energía muy veloz. Por tanto, las gotas de combustible se evaporan mientras hay suficiente aire que las caliente y las haga evaporar. Consecuentemente, la evaporación del diésel se ajusta adecuadamente a la aproximación flujo homogéneo local, para la cuál el equilibrio local es tanto térmico como dinámico.

Para contabilizar la evaporación del chorro del modelo CFD, se deben añadir una ecuación de transporte adicional para la fracción másica de combustible gaseoso y un procedimiento para calcular el término fuente  $S_{evap}$  de la Ecuación 3.5. La ecuación de transporte se puede escribir de forma similar a la conservación del combustible líquido:

$$\frac{\partial \bar{\rho} \tilde{Y}_v}{\partial t} + \frac{\partial \bar{\rho} \tilde{u}_i \tilde{Y}_v}{\partial x_i} - \frac{\partial}{\partial x_i} \left( \frac{\mu_t}{Sc} \frac{\partial \tilde{Y}_v}{\partial x_i} \right) = S_{evap}$$

Ecuación 3.51

De nuevo, el gradiente de turbulencia estándar se usa para cerrar esta ecuación de conservación, y se debe señalar que el término fuente de evaporación tiene el efecto contrario al término fuente de la fracción másica de líquido de la Ecuación 3.2. Este término, sea sumidero o fuente dependiendo de si se trata de la ecuación de conservación del combustible líquido o gaseoso, se calcular como la tasa necesaria para alcanzar las condiciones de saturación adiabática locales. Y se formula como:

$$S_{evap} = \frac{Y_{v,sat} - \tilde{Y}_v}{\tau_{evap}}$$

*Ecuación 3.52*

dónde  $\tilde{Y}_v$  es la fracción másica de combustible gaseoso local,  $Y_{v,sat}$  el valor de la fracción másica de combustible gaseoso en condiciones de saturación adiabática y  $\tau_{evap}$  un tiempo de relajación igual al paso temporal de cálculo computacional.

Para calcular el término fuente de evaporación asumiendo mezcla adiabática, la región de líquido o vapor del chorro se supone que debe tender hacia condiciones de saturación adiabática, y para determinar el equilibrio líquido-vapor, se considera la ley idea de Raoult. Las dos principales hipótesis de esta ley son:

- La fase de vapor se comporta como un gas ideal.
- La fase líquida se comporta como una solución ideal (hay que decir que en este estudio sólo ha sido considera un componente líquido).

Usando esta ley es posible relacionar la composición delas fases líquida y vapor, específicamente la fracción molar de vapor para condición de saturación ( $X_{v,sat}$ ) a través de la presión local de vapor y la presión total del sistema ( $p$ ). Matemáticamente, la ley de Raoult para un único componente en una solución ideal establece:

$$p_v = X_{v,sat} \cdot p$$

*Ecuación 3.53*

La secuencia de cálculos comienza con la temperatura local, que se usa para obtener la presión de vapor y, entonces,  $X_{v,sat}$  es evaluada para hallar  $Y_{v,sat}$ . La presión de vapor local antes mencionada se determina a través de la correlación de Lee y Kesler:

$$\ln \frac{p_v}{p_c} = f_1(T_r) + \omega f_2(T_r)$$

*Ecuación 3.54*

siendo  $p_c$  la presión crítica del combustible,  $T_r$  la temperatura relativa del combustible respecto a la crítica ( $T_r = T/T_c$ ) y  $f_i(T_r)$  definida como:

$$f_1(T_r) = 5.92714 - \frac{6.09648}{T_r} - 1.28862 \cdot \ln(T_r) + 0.169347 \cdot T_r^6$$

*Ecuación 3.55*

$$f_2(T_r) = 15.2518 - \frac{15.6875}{T_r} - 13.4721 \cdot \ln(T_r) + 0.43577 \cdot T_r^6$$

*Ecuación 3.56*

La fracción másica de combustible gaseoso en equilibrio de saturación está relacionada con la presión de vapor mediante:

$$Y_{v,sat} = X_{v,sat} \frac{PM_f}{PM_{mix}} = \frac{PM_f}{PM_{mix} + \left(\frac{p}{p_v} - 1\right) \cdot PM_{aire}}$$

*Ecuación 3.57*

y se tiene que  $PM_f$  es el peso molecular del combustible,  $PM_{aire}$  el peso molecular del aire ambiental y  $PM_{mix}$  el peso molecular de la mezcla total.

Así mismo, la evaporación del combustible también provoca cambios en la superficie de la interfaz líquido-gas, y como resultado, aparece un nuevo término fuente en la Ecuación 3.11. Este término se modela como:

$$S_{\Sigma evap} = \frac{2 \Sigma}{3 \bar{Y}} S_{evap}$$

*Ecuación 3.58*

Por último, la suma de combustible en fase gaseosa lleva a la aparición de una nueva contribución en la Ecuación 3.20 si se aplica la regla de la cadena a la ecuación de continuidad:

$$-\frac{1}{\rho} \frac{\partial \rho}{\partial Y_v} \frac{DY_v}{Dt}$$

*Ecuación 3.59*

De forma similar a los pasados seguidos al derivar la fase líquida, de la Ecuación 3.30 a la Ecuación 3.35, para la fase gaseosa del combustible se tiene:

$$-\frac{1}{\rho} \frac{\partial \rho}{\partial Y_v} \frac{DY_v}{Dt} = \left(\frac{1}{\rho_v} - \frac{1}{\rho_g}\right) \left(\frac{\partial}{\partial x_i} \left(\frac{\mu_t}{Sc} \frac{\partial Y_v}{\partial x_i}\right) + S_{evap}\right)$$

*Ecuación 3.60*

En este caso,  $\rho_v$  es la densidad del combustible en estado gaseoso, que se rige por la ley de los gases ideales.

### 3.3.3 Modelo termodinámico del combustible líquido

El objetivo último de este modelo es obtener la temperatura del combustible a partir de la ecuación de la entalpía de la mezcla, asumiendo que existe equilibrio termodinámico local en las celdas. La ecuación de la entalpía se expresa como sigue:

$$h(T) = \tilde{Y} \cdot h_{f,l}(T) + \tilde{Y}_v \cdot h_{f,v}(T) + (1 - \tilde{Y} - \tilde{Y}_v) \cdot h_a(T)$$

*Ecuación 3.61*

Aquí  $h_{f,l}(T)$ ,  $h_{f,v}(T)$  y  $h_a(T)$  representan la entalpía del combustible en estado líquido, en estado gaseoso y la entalpía del gas ambiental respectivamente. Estas entalpías son calculadas usando sus respectivos calores específicos a presión constante ( $c_{p,i}$ ), que a su vez son función de la temperatura  $T$  y un conjunto de coeficientes de las tablas termodinámicas JANAF. La NASA utiliza funciones polinómicas de 7 coeficientes para calcular el calor específico:

$$C_p^0 = R \cdot (a_1 + a_2 T + a_3 T^2 + a_4 T^3 + a_5 T^4)$$

*Ecuación 3.62*

Dado que el calor específico a presión constante resulta de derivar la entalpía respecto de la temperatura a presión constante, integrando la ecuación anterior respecto a la temperatura, la entalpía también se puede calcular como:

$$h^0 = RT \cdot (a_1 + a_2 T + a_3 T^2 + a_4 T^3 + a_5 T^4)$$

*Ecuación 3.63*

Estas expresiones pueden ser aplicadas directamente para calcular la entalpía del gas ambiental  $h_a$  y su calor específico a presión constante  $c_{p,a}$ , pero no sucede así para el combustible. En este caso, se debe utilizar la ecuación de Rowlinson-Bondi (1969), según la cuál el calor específico del combustible líquido y su entalpía se puede calcular tal que así:

$$c_{p,l}(T) = C_p^0 + R \cdot (1.45 + 0.45(1 - T_r)^{-1}) + \\ + R \cdot (0.25\omega[17.11 + 25.2(1 - T_r)^{-1/3} T_r^{-1} + 1.742(1 - T_r)^{-1}])$$

*Ecuación 3.64*

$$h_{f,l}(T) = h^0 + \int_{298}^T R \cdot (1.45 + 0.45(1 - T_r)^{-1})dT + \\ + \int_{298}^T R \cdot (0.25\omega[17.11 + 25.2(1 - T_r)^{-1/3} T_r^{-1} + 1.742(1 - T_r)^{-1}])dT$$

*Ecuación 3.65*

El combustible gaseoso se calcula por medio de la entalpía de vaporización  $\Delta H_v$ , algunas veces llamada como el calor latente de vaporización, y que se obtiene de la correlación:

$$\Delta H_v(T) = R T_c + (7.08(1 - T_r)^{0.354} + 10.95\omega(1 - T_r)^{0.456})$$

*Ecuación 3.66*

Por tanto, la entalpía del combustible gaseoso se evalúa utilizando dos expresiones diferentes dependiendo de si se tiene o no combustible líquido en la celda:

- En presencia de combustible líquido:

$$h_{f,v}(T) = h_{f,l}(T) + \Delta H_v(T)$$

*Ecuación 3.67*

- En ausencia de combustible líquido:

$$h_{f,v}(T) = h_{f,l}(T_{evap}) + \Delta H_v(T_{evap}) + \int_{T_{evap}}^T C_p^0 dT$$

*Ecuación 3.68*

Entonces, el algoritmo para calcular la temperatura a partir de la entalpía de la mezcla consiste en un método iterativo, en el que se toma la temperatura del instante anterior como la temperatura de la primera iteración ( $T_0$ ) y entonces ésta se va actualizando hasta encontrar un nuevo valor de temperatura ( $T$ ) que proporcione una diferencia con la temperatura de la iteración anterior por debajo de un mínimo. El proceso usado para actualizar la temperatura se corresponde con el método Newton-Raphson.

$$T = T_0 + \frac{h(T) - h(T_0)}{dh(T_0)}$$

*Ecuación 3.69*

En la ecuación anterior  $h(T)$  es la entalpía obtenida en la Ecuación 3.7 y,  $h(T_0)$  y  $dh(T_0)$  la entalpía y la derivada de la Ecuación 3.61 evaluadas para la temperatura inicial de la iteración ( $T_0$ ).

Hay que señalar que, a diferencia del modelo de turbulencia, implementado en el propio código para la computación del caso de estudio con el software OpenFOAM, el modelo termodinámico del combustible es parcialmente calculado durante el lanzamiento del caso. Mediante un software externo, MATLAB, se tabulan los valores de entalpía (y sus derivadas) para el aire, el combustible en estado líquido y el combustible en estado gaseoso según la temperatura. El código implementado en OpenFOAM sólo debe leer las tablas, interpolar datos en función de la temperatura y actualizarla. De esta manera se ahorra mucho tiempo y coste computacional durante el proceso de cálculo del caso.

### 3.3.4 Ecuación de estado del combustible líquido

En este punto del proyecto, se debe hablar de dos vías diferentes. Por un lado, se han introducido un cambio importante en el nuevo modelo respecto al anterior en cuanto a cálculos RANS se refiere, mientras que para los cálculos LES se ha mantenido la ecuación de estado del combustible líquido implementada en primera instancia. Esta ecuación de estado se obtiene a través de la correlación de Hankinson y Thomson (HBT, 1979), para densidades de líquidos saturados, y tiene en cuenta la presión y la temperatura del mismo. El volumen del líquido saturado sería:

$$V_s = V^* V_R^0 (1 - \omega \cdot V_R^\delta)$$

*Ecuación 3.70*

Aquí,  $V^*$  es componente característico del volumen cuyo valor está muy cerca del volumen crítico, y  $\omega$  es el factor acéntrico que mejora la ecuación de estado de Soave para las presiones de vapor de componentes puros. Ambos valores son constantes y están tabulados para diferentes componentes. Los valores  $V_R^0$  y  $V_R^\delta$  se calculan como sigue:

$$V_R^0 = 1 + a(1 - T_r)^{1/3} + b(1 - T_r)^{2/3} + c(1 - T_r) + a(1 - T_r)^{4/3}$$

*Ecuación 3.71*

$$V_R^\delta = \frac{e + f \cdot T_r + g \cdot T_r^2 + h \cdot T_r^3}{T_r - 1.00001}$$

*Ecuación 3.72*

Estas ecuaciones sólo pueden ser usadas para un rango determinado de temperaturas. En concreto, la Ecuación 3.71 es válida cuando  $T_r \in (0.25, 0.95)$ , y la Ecuación 3.72 cuando  $T_r \in (0.25, 1)$ . Las constantes de la  $a$  hasta la  $h$  valen:

$a$	$b$	$c$	$d$	$e$	$f$	$g$	$h$
-1.52816	1.43907	-0.81446	0.19045	-0.29612	0.38691	-0.04272	-0.04806

*Tabla 3.2. Coeficientes.*

El volumen molar del líquido es el volumen de líquido saturado con una corrección:

$$V = V_s \left( 1 - c' \ln \frac{\beta + p}{\beta + p_v} \right)$$

*Ecuación 3.73*

Las constantes  $\beta$  y  $c$  se calculan usando:

$$\beta = P_c (-1 + a'(1 - T_r)^{1/3} + b'(1 - T_r)^{2/3} + c'(1 - T_r) + a'(1 - T_r)^{4/3})$$

*Ecuación 3.74*

$$e' = \exp(f' + g' \cdot \omega + h'\omega^2)$$

*Ecuación 3.75*

$$c' = j' + k' \cdot \omega$$

*Ecuación 3.76*

Los valores de la presión crítica también se encuentran tabulados para muchos compuestos, así como las constantes para el modelo.

$a'$	$b'$	$d'$	$f'$	$g'$	$h'$	$j'$	$k$
-9.07021	62.45326	-135.110	4.79594	0.25005	1.14188	0.08615	0.03445

*Tabla 3.3. Coeficientes.*

Una vez se tiene el volumen molar, ya solo queda calcular la densidad del combustible líquido usando su peso molecular:

$$\rho_l = \frac{PM_f}{V}$$

*Ecuación 3.77*

Como se ha comentado al inicio del documento, el caso de estudio ha sido tanto calculado en RANS como LES. Así pues, para los cálculos RANS el modelo planteado se desmarca en este punto del modelo anterior, pues se han tomado y tabulado (según la presión y la temperatura) las propiedades termo-físicas como fluido del combustible usado en el caso de estudio (dodecano) que proporciona el Instituto Nacional de Estándares y Tecnología de EEUU (National Institute of Standards and Technology, NIST). Esto se debe a que tabulando ciertas propiedades e interpolando en vez de calcularlas a cada iteración supone ahorro de tiempo de cálculo y coste computacional, además de que el ajuste de propiedades que proporciona es más adecuado, en teoría, para las temperaturas que se tienen dentro del cilindro de combustión que el de las propiedades que proporcionaba el modelo anterior.

Por otro lado, debido a la gran cantidad de tiempo que requiere ejecutar las simulaciones LES, se ha decidido no utilizar la ecuación de estado del NIST en este tipo de simulación y tan sólo estudiar su efecto con los casos de estudio RANS. Por tanto, los casos ejecutados como LES cuentan con el nuevo modelo termodinámico (sección anterior) y la ecuación de estado del combustible líquido anterior.



## 4 Metodología, configuración del caso y resultados

El presente proyecto tiene por objetivo estudiar la formación del chorro diésel inmediatamente después de su inyección en cámaras de combustión, en un ambiente no reactivo pero sí evaporativo; pues el verdadero interés recae en el análisis del proceso de atomización del chorro, y así aplicar estos conocimientos para una mejor formación de la mezcla aire-combustible en el futuro.

Ya que el estudio se realiza ejecutando un modelo de cálculo CFD, es necesario validar la solución procedente de dicho modelo con un caso experimental. En este caso, la información experimental es cedida por una organización denominada ECN, Engine Combustion Network. Es una red conformada por diferentes instituciones de investigación que comparten sus propias indagaciones relacionadas con la combustión en motores.

### 4.1 Definición del problema y condiciones de contorno

El fenómeno a estudiar es la formación del chorro diésel (atomización, coalescencia, evaporación) en el interior de un cilindro de combustión, por lo que el volumen de control, y el dominio del problema por ende, es el propio interior de un cilindro de 80 mm de largo y 25 mm de radio. En el ejemplo calculado en LES, el radio ha sido reducido a 15 mm por simplificar, ya que el chorro de combustible no llegaba a las paredes y se queda lejos de tocarlas.

Siendo más concretos, para el caso ejecutado con cálculos RANS, el volumen de control se reduce a una cuña del cilindro con un ángulo de  $10^\circ$ . En este caso, el filo de la cuña coincidente con el eje longitudinal del cilindro se corresponde con un eje de simetría para la simulación, los planos de la cuña que contienen este eje se considerarían planos cíclicos y las paredes propias del cilindro, paredes isoterma y sin deslizamiento. Para los cálculos LES, sí se tiene la totalidad del volumen del cilindro como dominio, y por tanto, las paredes que rodean el volumen de control son las del cilindro, consideradas isoterma y sin deslizamiento. En la Figura 4.1 se puede ver el dominio del caso RANS.

En cuanto a los compuestos utilizados, se tiene que el gas inerte del interior del cilindro es el nitrógeno, a una temperatura de 900K y una densidad de  $22.8\text{kg/m}^3$ , y en el instante inicial posee velocidad nula. El combustible inyectado es el dodecano, y se inyecta a una temperatura de 363K y una presión de 150MPa. Además, se adopta un

perfil uniforme de velocidad de inyección del combustible que varía con el tiempo. Esta variación de velocidad fue obtenida a partir de medidas experimentales de gasto másico, en las que se variaba la densidad del combustible y el área de inyección.

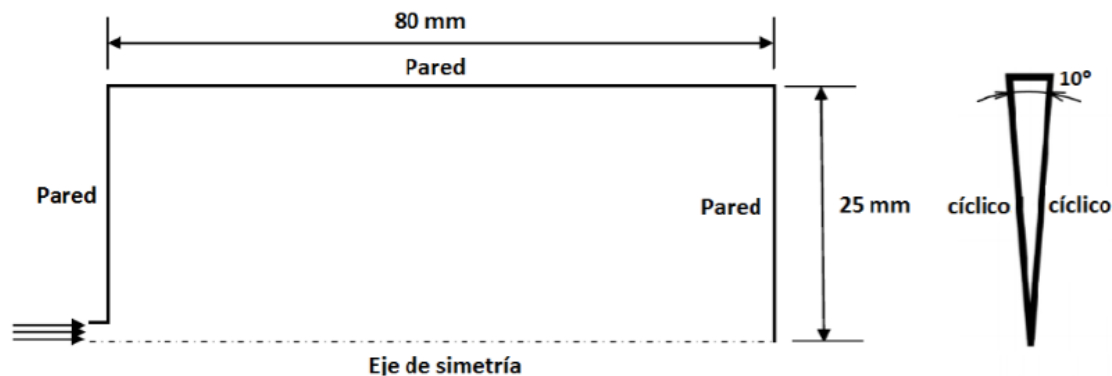


Figura 4.1. Dominio del caso RANS.

En lo referente al inyector usado para los cálculos, resulta ser un inyector axial con un solo orificio correspondiente al Serial# 210675, a partir de aquí será llamada como tobera 675, cuyas características quedan recogidas en esta tabla:

Serie	$D_0$ [ $\mu\text{m}$ ]	$D_i$ [ $\mu\text{m}$ ]	$L/D_0$ [-]	$r/D_0$ [-]	Factor k	Offset a la salida [ $\mu\text{m}$ ]
210675	89.4	116	11.5	0.23	2.7	53

Tabla 4.1. Parámetros geométricos de la tobera 675.

Para evaluar y validar el modelo aplicado para la simulación del chorro de combustible, tanto en RANS como en LES, la solución obtenida debe ser comparada con un caso que reproduzca este fenómeno físico en las mismas condiciones. Por ello se han utilizado los resultados de la base de datos del “Spray A” proporcionado por la ECN. Es un caso como el descrito para la simulación: el combustible es inyectado en entorno quiescente de nitrógeno (gas inerte para que no haya reacción). Las condiciones de esta prueba son como las del modelo: presión de inyección de 150MPa, gas ambiental en el interior del cilindro a 900K y de densidad 22.8kg/m<sup>3</sup>. En la Tabla 4.2 se recogen las condiciones de esta prueba:

Combustible	n-dodecano
Temperatura del combustible	363 K
Presión de inyección del combustible	150 MPa
Composición del gas ambiental	100% N <sub>2</sub>
Temperatura del gas ambiental	900 K
Presión del gas ambiental	~ 60 MPa
Densidad del gas ambiental	22.8 kg/m <sup>3</sup>

Tabla 4.2. Condición nominal de Spray A definida por la ECN.

## 4.2 Metodología y malla

Como ya se ha mencionado antes, este proyecto consiste en el estudio de la formación del chorro diésel en cámaras de combustión de cilindros alternativos, además de en mejorar el modelo de cálculo computacional de chorro diésel y reducir tanto el coste computacional como el tiempo de ejecución del mismo. Para ello se revisó la totalidad del código del modelo y se escogieron los aspectos con potencial para reducir su coste computacional en gran medida: el modelo termodinámico y la ecuación de estado, ambos, del combustible líquido. En el capítulo anterior han quedado explicadas los cambios introducidos, pero también se llevaron a cabo tareas de depuración del código, como eliminar variables redundantes o simplificar ciertas estructuras internas del mismo.

Una vez se había revisado el código y determinado qué cambiar en él, se procedía a reescribir esta parte del modelo para los cálculos RANS, ya que el tiempo de ejecución en este tipo de cálculo es mucho menor que el tiempo requerido para ejecutar en caso calculado en LES, se lanzaba el caso de prueba y se validaban o no los resultados. Con el fin de configurar el nuevo modelo para cálculos LES, la solución obtenida no sólo debía ser validada si no que el tiempo de cálculo del nuevo modelo también debía ser significativamente menor que su predecesor. Por tanto, si los resultados del caso de prueba eran válidos y el tiempo de ejecución también se veía reducido para el nuevo modelo modificado, estos cambios se implementaban para los cálculos LES y se volvían a validar. Se debe puntualizar que el modelo inicial de formación del chorro para cálculos RANS ya había sido validado para un proyecto anterior con los datos experimentales indicados en la sección 4.1 de este capítulo, por tanto, para probar la validez de los nuevos modelos de cálculo RANS tan sólo se han comparado las soluciones proporcionadas por los nuevos modelos RANS con la solución del antiguo. La solución correspondiente al modelo de cálculo LES, en este trabajo, sí ha sido comparada tanto con el caso experimental de la ECN como con la solución del cálculo RANS para su validación.

Así pues, la primera modificación introducida en el modelo es la correspondiente al modelo termodinámico del combustible líquido, de lo que sí se tiene resultados tanto para los cálculos RANS como los LES. El segundo cambio introducido es el relacionado con la ecuación de estado del combustible líquido, para el cuál sólo se tienen resultados RANS del caso de estudio. Debido al hecho de que la ejecución y validación del caso calculado como LES requiere largos periodos de tiempo, y ante la escasez de horas disponibles en súper computadores, no se han podido lanzar y obtener resultados del caso para cálculos LES con ambas modificaciones del código, tan sólo para la primera de ellas.

Por otro parte, lanzar los casos y obtener resultados precisos requiere una malla adecuada, la cuál tenga en su dominio el volumen que se quiera simular y cuyas celdas posean un tamaño apto para recoger todas las propiedades y fluctuaciones del volumen

fluido, imprescindible dar solución a los sistemas de ecuaciones de conservación planteadas en cada celda. En este aspecto, para los cálculos LES, el tamaño de estas celdas dictará el límite entre las escalas turbulentas que serán modeladas y las que serán calculadas mediante las ecuaciones generales Navier-Stokes.

Normalmente se debe llevar a cabo un estudio de independencia de malla para comprobar que la solución resultante no depende de la estructura de la malla y su resolución. En este trabajo la malla ha sido dada, pues también ha sido utilizada y validada en proyectos anteriores, de manera que en este documento no se trata este análisis de malla. Ahora bien, como se tienen dos tipos de cálculo diferentes, también se tienen dos mallas distintas:

- El caso calculado en RANS es poseedor de la malla más simple. La simulación LES consiste en un estudio tridimensional, mientras que, por simplicidad y agilizar el análisis de los cambios efectuados en el modelo, las simulaciones RANS son un estudio bidimensional y axi-simétrico del problema. La única forma de conseguir un estudio así con el software usado, OpenFOAM, es crear una malla que vaya desde un extremo al otro del cilindro y se extienda desde el eje de revolución hasta la pared del mismo, dotándola con una sola capa de celdas que le proporcione cierto espesor.
- La malla del caso calculado como LES es la de mayor tamaño, su volumen de control es la totalidad del cilindro de combustión. La razón para ello son las fluctuaciones presentes en la solución, mencionadas en el párrafo anterior, propias de los cálculos LES, pues se han de recoger todas estas fluctuaciones para ver la solución completa. Luego, estas fluctuaciones pueden ser tratadas y promediadas en el tiempo una vez que el chorro se haya estabilizado dentro del volumen de control.

En ambos casos, las mallas comparten ciertas características: la resolución de la malla no es uniforme, de esta manera se pueden obtener celdas de menor tamaño y resultados de mayor precisión en las zonas donde sea necesario. Para este problema, la zona con mayor resolución y celdas de menor tamaño es la región próxima al inyector, ya que es aquí donde se tienen los mayores gradientes para las variables. Conforme aumenta la distancia al inyector o al eje longitudinal del cilindro, el tamaño de las celdas crece y disminuye la precisión, ya que así se reduce también el coste computacional. Por tanto, las dos mallas presentan una resolución no uniforme.

Con todo esto, la malla hecha para el caso ejecutado en RANS cuenta con una distribución de 495 celdas a lo largo del eje longitudinal y de 90 celdas en dirección radial, así como con un ángulo de cuña de  $10^\circ$  y un volumen total de  $4.4 \text{ cm}^3$ . Mientras tanto, la malla con la que ha trabajado el caso para cálculos LES distribuye 522 celdas a lo largo del eje longitudinal, de 105 en la dirección radial y 72 celdas en la dirección azimutal, siendo su volumen total el mismo que el del cilindro:  $56.5 \text{ cm}^3$ .

## 4.3 Análisis de los resultados

De acuerdo con la metodología descrita para la implementación de cada una de las modificaciones del modelo inicial de la formación del chorro de combustible tras la inyección, esta sección recoge y analiza los diferentes resultados obtenidos durante todo el proyecto que han conducido al modelo actual. Así pues, esta sección quedará dividida en dos sub-apartados, uno destinado al análisis de los resultados RANS y otro al de los resultados LES:

- Por un lado, se realizará el análisis del modelo para cálculos RANS, tanto con la modificación pertinente al modelo termodinámico del combustible como con el cambio en la ecuación de estado del combustible líquido. En ambos casos se comparará el coste computacional y temporal del modelo con respecto a su versión anterior, así como ciertos parámetros característicos del chorro. Además, para la modificación del modelo termodinámico de cálculo de temperatura a partir de entalpía, se estudiarán dos vertientes, que serán expuestas más adelante.
- En cuanto a los resultados procedentes del modelo de cálculo LES, cómo ya ha sido explicado, por motivos de tiempo y falta de horas en súper computadoras, tan sólo se ha podido obtener resultados del modelo de formación de chorro para cálculos LES que incluye la variación del proceso de cálculo de temperatura del modelo termodinámico. Aun así, esta solución será comparada con la solución de los cálculos RANS y con los datos experimentales cedidos por la ECN.

### 4.3.1 Resultados RANS

En el modelo de formación de chorro de combustible de cálculo RANS se han llevado a cabo dos variaciones: la implementación del proceso de cálculo de la temperatura a partir de la entalpía que proporciona el modelo termodinámico y la ecuación de estado del combustible líquido.

En el apartado 3.3.3, del capítulo 3, queda explicado el modelo termodinámico y cómo calcular la temperatura de la mezcla a partir de la entalpía de la misma mezcla y un proceso de cálculo iterativo. Este proceso iterativo resulta ser la aplicación del método Newton-Raphson, y la Ecuación 3.69 muestra la expresión que determina la nueva temperatura con cada iteración. Sin embargo, antes de decir que este método de cálculo quedara así, se estudiaron dos propuestas:

- Una de ellas, y con la cuál se quedó este método, planteaba truncar este proceso y adoptar la temperatura para la mezcla según está explicado en el apartado 3.3.3 del capítulo 3. Es decir, la temperatura adoptada tras este proceso queda definida por el error de la nueva temperatura calculada en la última iteración de cálculo con respecto a la temperatura de la iteración anterior.

- El otro sistema, descartado tras analizar los resultados de ambos planteamientos, optaba por adoptar el error de la entalpía asociada a la nueva temperatura respecto a la entalpía asociada a la temperatura de la iteración anterior como el elemento determinante para trunca el cálculo iterativo y adoptar una temperatura para la mezcla.

Los resultados de ambos planteamientos se compararon con los del modelo inicial: se analizaron ciertos parámetros característicos del chorro y el tiempo de ejecución del caso. Como ahora se mostrará, los resultados eran muy parecidos, por lo que el factor determinante para elegir un sistema u otro fue el tiempo necesario para la ejecución del caso de estudio.

#### Comparativa de los nuevos modelos termodinámicos con el modelo original

Al ser un estudio computacional dónde uno de los modelos ya ha sido validado, se podrían analizar muchos parámetros del chorro de combustible ente los tres modelos del fenómeno, pero por simplificar, y una vez se ha estabilizado el chorro de combustible, se han representado varias variables frente a la composición de la mezcla (suma de las fracciones másicas de combustible en estado líquido y estado gaseoso): temperatura, entalpía, fracción másica de combustible líquido, fracción másica de combustible gaseoso, tasa de evaporación y coalescencia o el error relativo de entalpía entre la proporcionada por la Ecuación 3.7 y la calculada por la Ecuación 3.61 del apartado 3.3.3 a la hora de obtener la temperatura. También se ha representado la longitud líquida y cortes longitudinales del cilindro dónde se muestra la distribución espacial de algunas variables como la temperatura o la fracción másica de combustible.

Al representar (de la Figura 4.2 a la Figura 4.16) las variables de temperatura, entalpía y la fracción másica de combustible (y su derivada) tanto en estado líquido como gaseoso frente a la composición de la mezcla, es decir, frente a la suma de las fracciones másicas de combustible líquido y combustible gaseoso (pues sólo se pueden encontrar tres componentes en la mezcla: combustible líquido, combustible gaseoso y nitrógeno como gas ambiental); se puede constatar que los tres modelos, tanto el original como los dos nuevos, tienen el mismo comportamiento y representan el mismo modelo de forma casi idéntica. Esto es un buen indicador de que los nuevos modelos dan una solución próxima a la realidad:

- En la representación de las fracciones másicas de combustible líquido y gaseoso se aprecian varios aspectos. El primero de ellos es que en las zonas dónde la mezcla está solo compuesta por combustible, dicho combustible se encuentra solamente en estado líquido pues se correspondería a la zona de la tobera de inyección. Así mismo, al alejarse de esta zona, el combustible se mezcla con el aire, se reduce el nivel de combustible en la composición local de la mezcla, comienza a evaporarse y va creciendo la cantidad de combustible gaseoso frente a la reducción de combustible líquido. La cantidad de combustible en la mezcla va descendiendo hasta un punto

de inflexión dónde desaparece por completo el combustible líquido y tan sólo queda combustible gaseoso, que a la vez encuentra su máximo en la mezcla gas ambiental-combustible. Este punto de inflexión se tiene para un nivel de combustible en la mezcla cercano al 33%. A partir de este punto, el nivel de combustible gaseoso descendiendo hasta ser nulo.

- En el caso de la temperatura, se aprecia su descenso conforme aumenta el nivel de combustible en la mezcla. Esto es lógico, pues la temperatura del gas ambiental se sitúa en los 900K dentro del cilindro como condición inicial, mientras que el combustible es inyectado a 363K, una temperatura bastante inferior. Es fácil ver un cambio brusco de pendiente en la representación de la temperatura, y justo se da en el mismo punto de inflexión que en el caso la fracción másica de combustible.
- De la misma manera que ocurre con la temperatura, la entalpía estática de la mezcla disminuye al aumentar la cantidad de combustible, ya que éste parte de una temperatura bastante inferior. No obstante, si en la temperatura la pendiente cambiaba de forma brusca, en este caso la pendiente de la curva parece ser constante.
- También resulta de interés representar la derivada de la fracción másica de combustible gaseoso. Esta derivada significa que existe evaporación si su valor es positivo o que hay coalescencia si su valor es negativo, obteniendo así combustible en estado gaseoso o nuevas gotitas de combustible líquido. Se distinguen dos zonas claramente diferenciadas: una zona de coalescencia para composiciones de mezcla ricas en combustible (especialmente en combustible líquido) y otra de evaporación, con mayores valores absolutos y entorno al punto de inflexión mencionado, dónde tiene lugar la evaporación del combustible. Al tener valores tan elevados para la tasa de evaporación, el propio modelo corrige este aspecto con la aparición de la zona de coalescencia.

Salvo en el caso de la entalpía estática de la mezcla, aparece un punto de inflexión muy claro cuyo valor es común denominador para el resto de variables. Este punto registra la mayor tasa de evaporación y de combustible en estado gaseoso, así como un cambio significativo en la pendiente de la temperatura. Por tanto, debe representar un área del chorro de gran interacción entre gas ambiental y combustible, dónde deja de haber, en la práctica, combustible líquido y todo el que queda permanece en estado gaseoso.

Ahora bien, en la representación de la temperatura del modelo original aparece una línea inusual estanca en la temperatura crítica del combustible para baja presencia de dodecano en la mezcla, dividiendo la curva de temperatura en dos. Esto se debe a la existencia de un truncamiento en el modelo original, que en los dos nuevos ha sido suprimido tras corregir ciertos puntos del modelo, y otorgando así mayor validez al resultado de las ecuaciones.

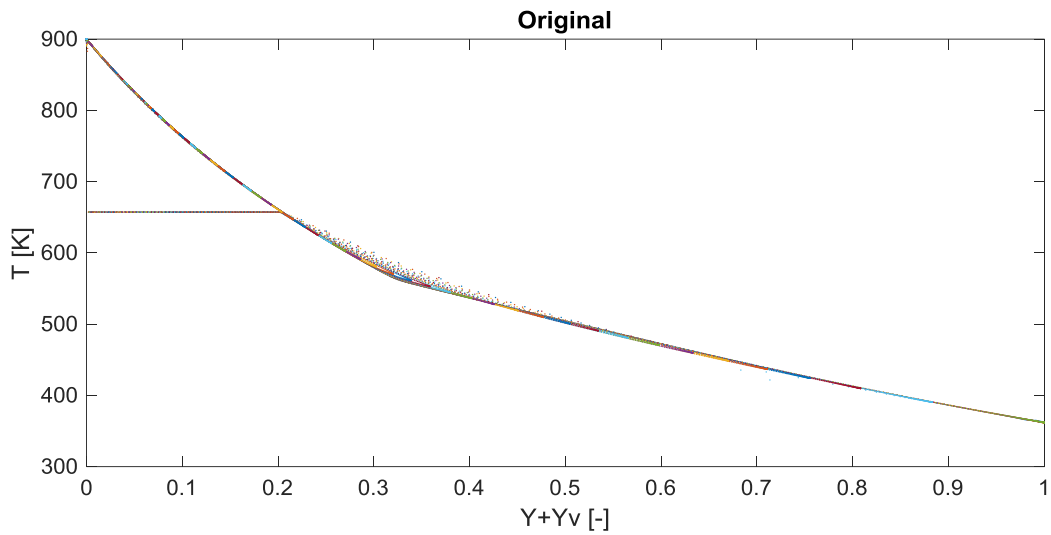


Figura 4.2. Temperatura de la mezcla según composición. Modelo original.

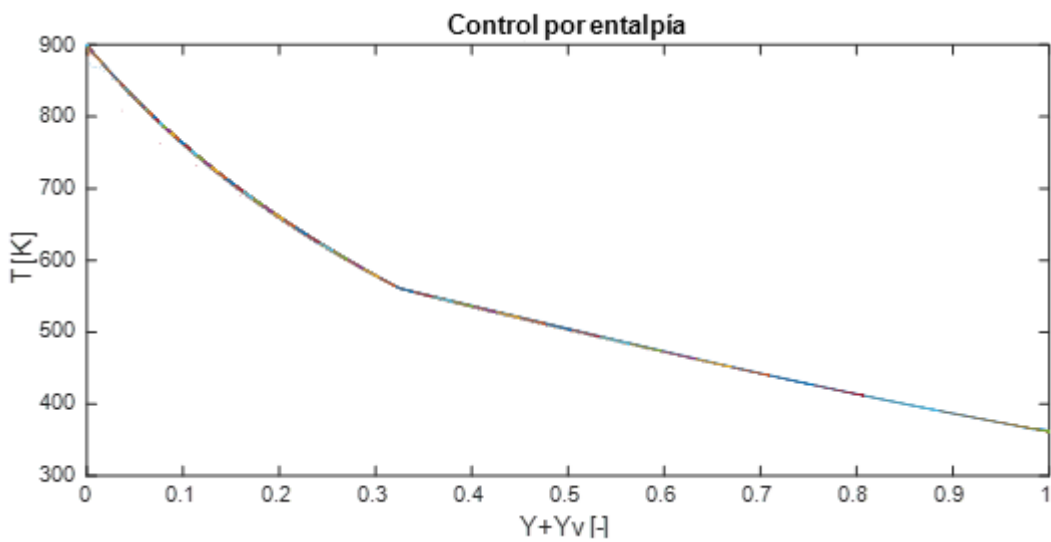


Figura 4.3. Temperatura de la mezcla según composición. Control por entalpía.

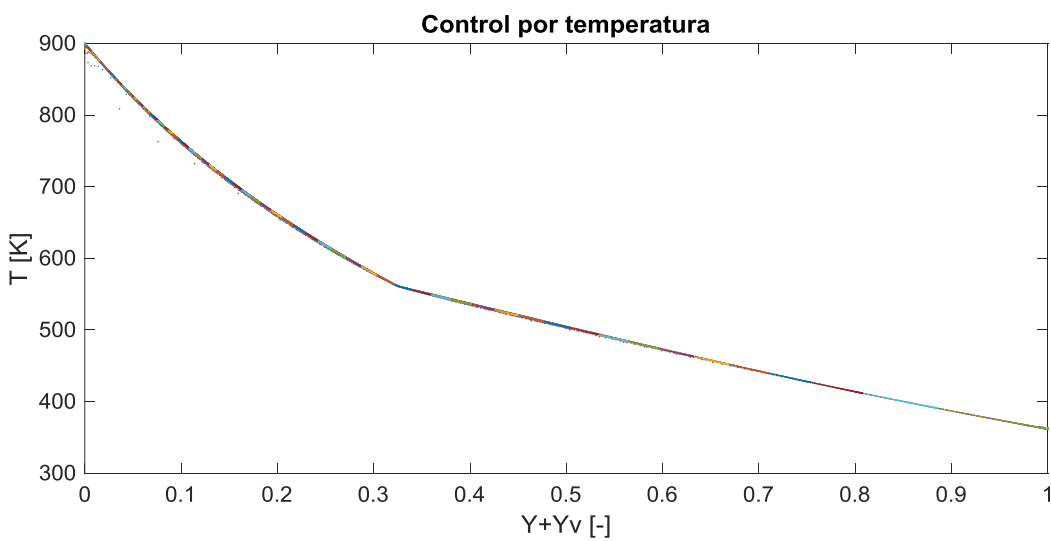


Figura 4.4. Temperatura de la mezcla según composición. Control por temperatura.



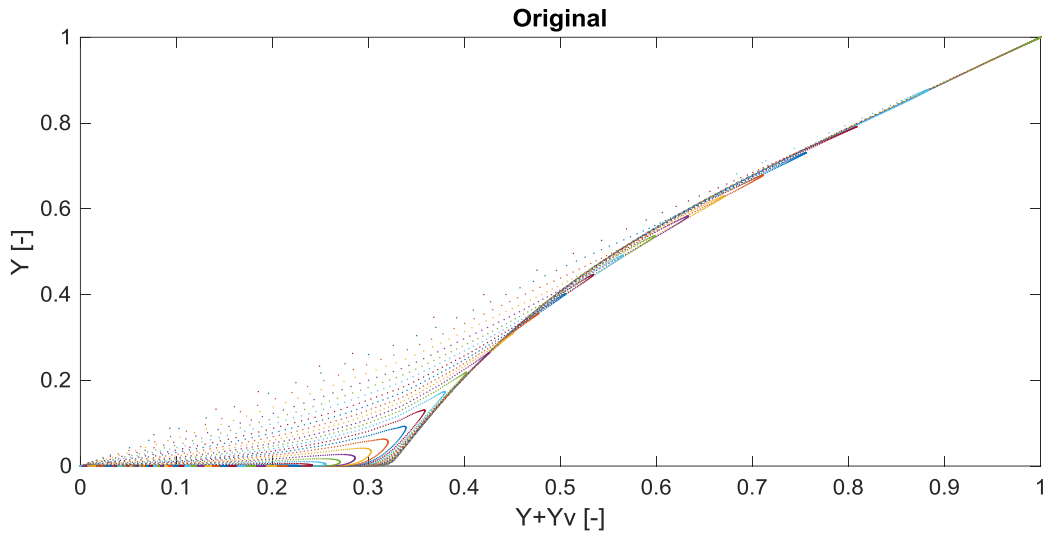


Figura 4.5. Combustible líquido según composición. Modelo original.

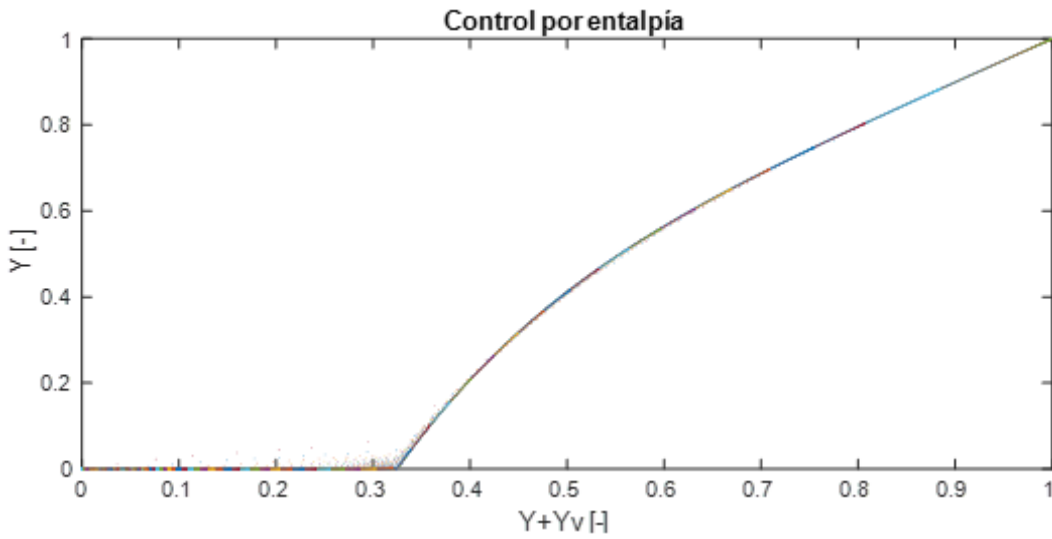


Figura 4.6. Combustible líquido según composición. Control por entalpía.

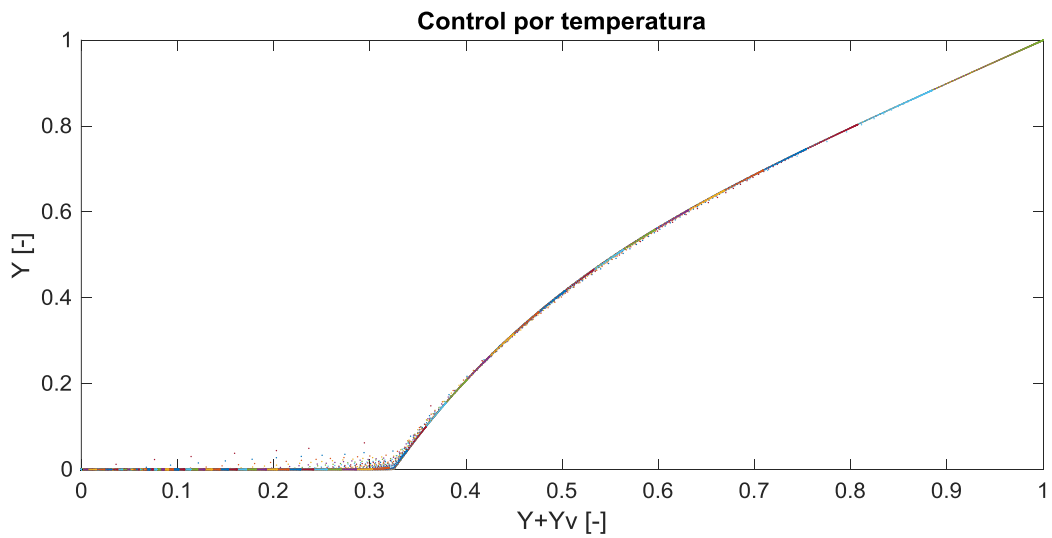


Figura 4.7. Combustible líquido según composición. Control por temperatura.

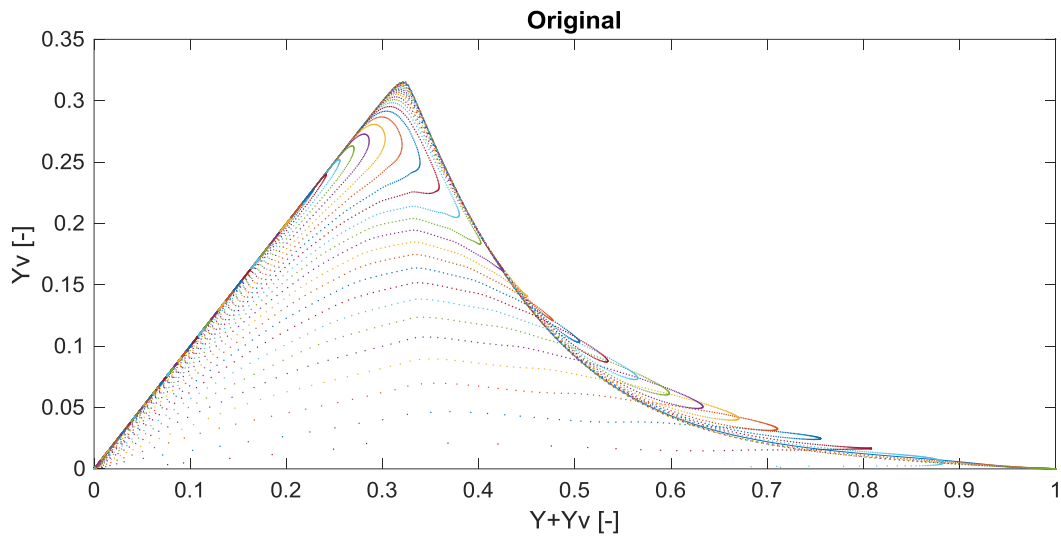


Figura 4.8. Combustible gaseoso según composición. Modelo original.

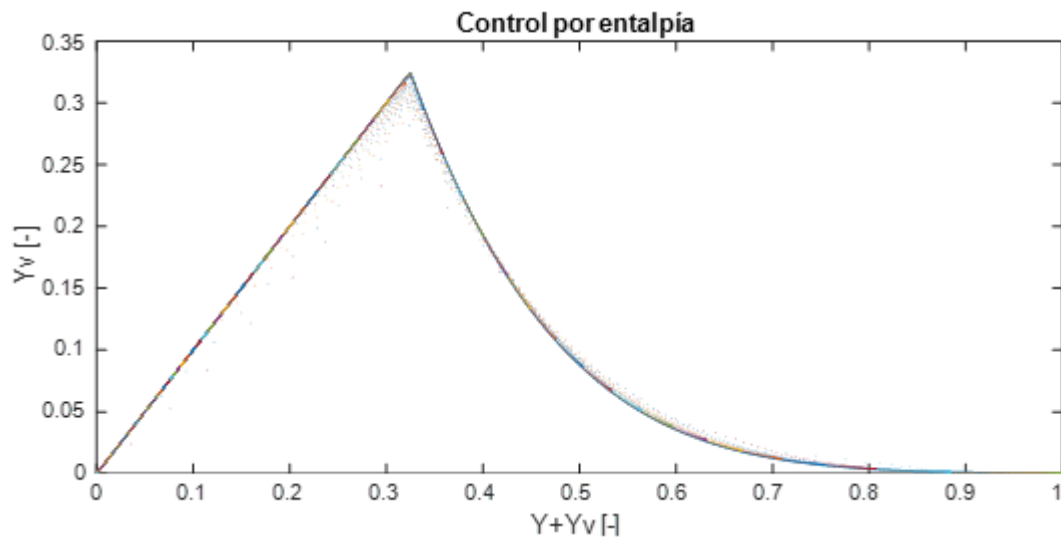


Figura 4.9. Combustible gaseoso según composición. Control por entalpía.

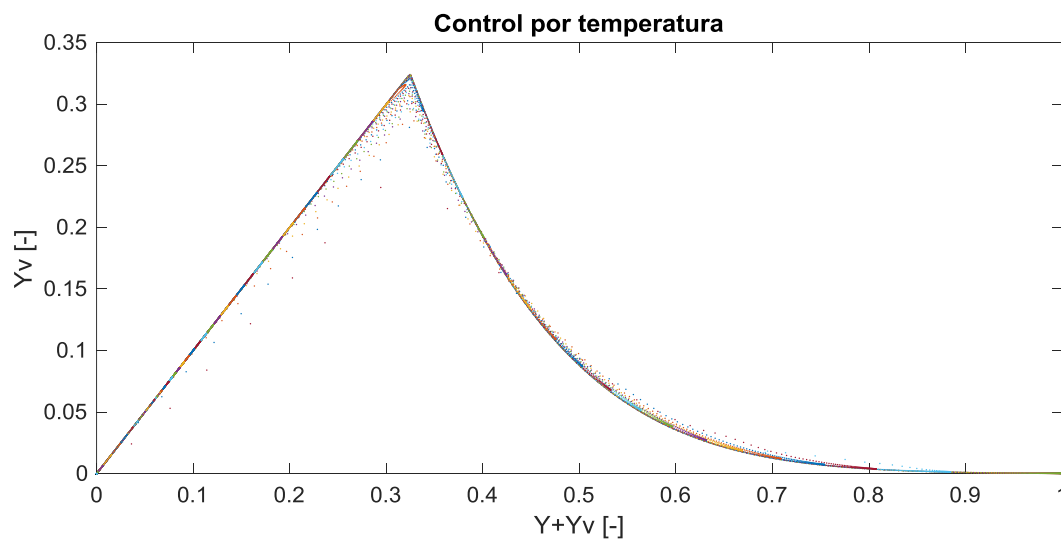


Figura 4.10. Combustible gaseoso según composición. Control por temperatura.

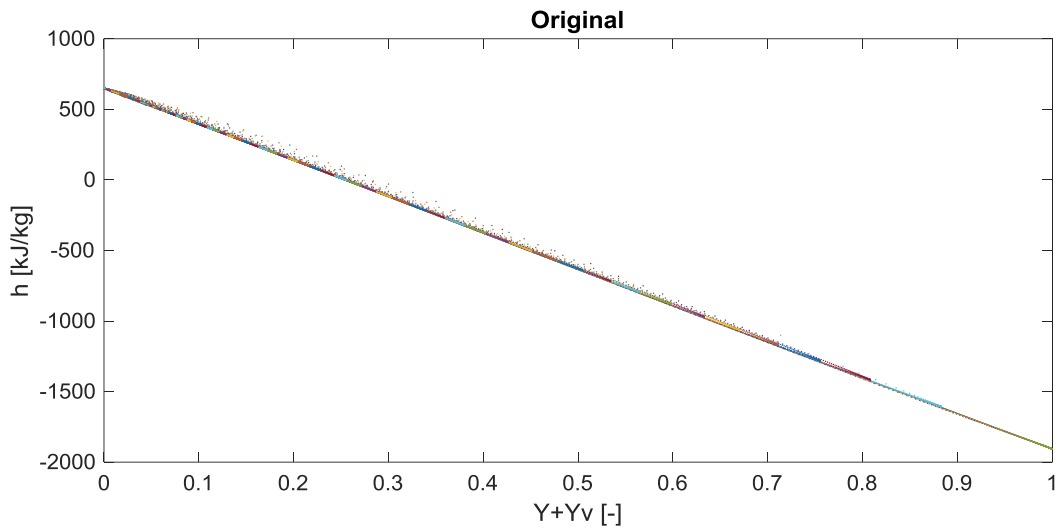


Figura 4.11. Entalpía según composición de la mezcla. Modelo original.

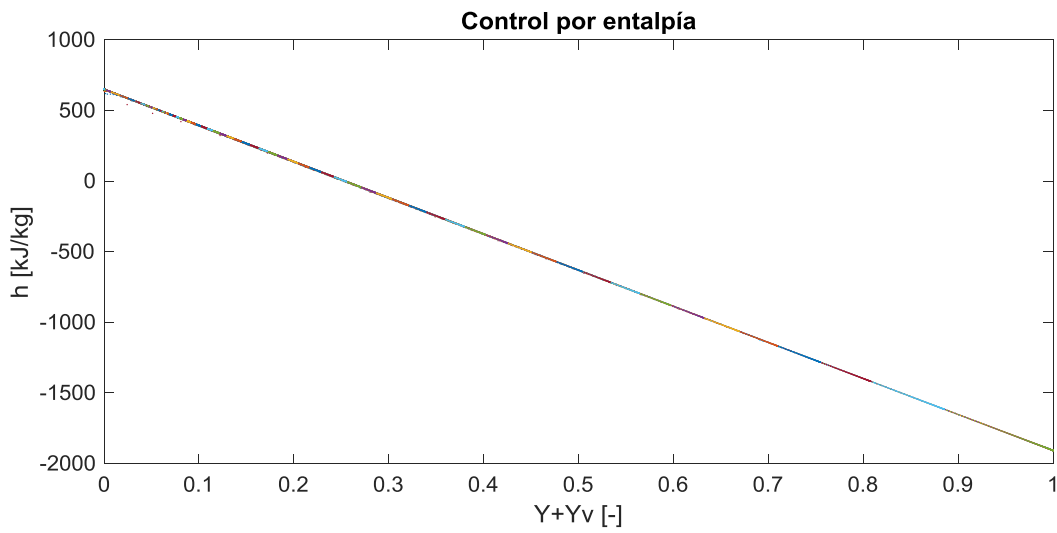


Figura 4.12. Entalpía según composición de la mezcla. Control por entalpía.

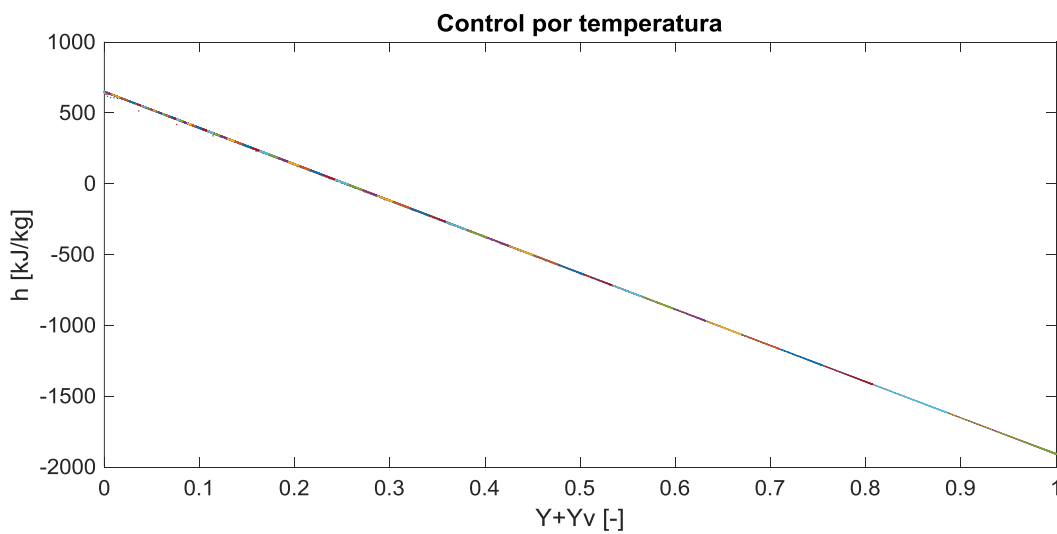


Figura 4.13. Entalpía según composición de la mezcla. Control por temperatura.

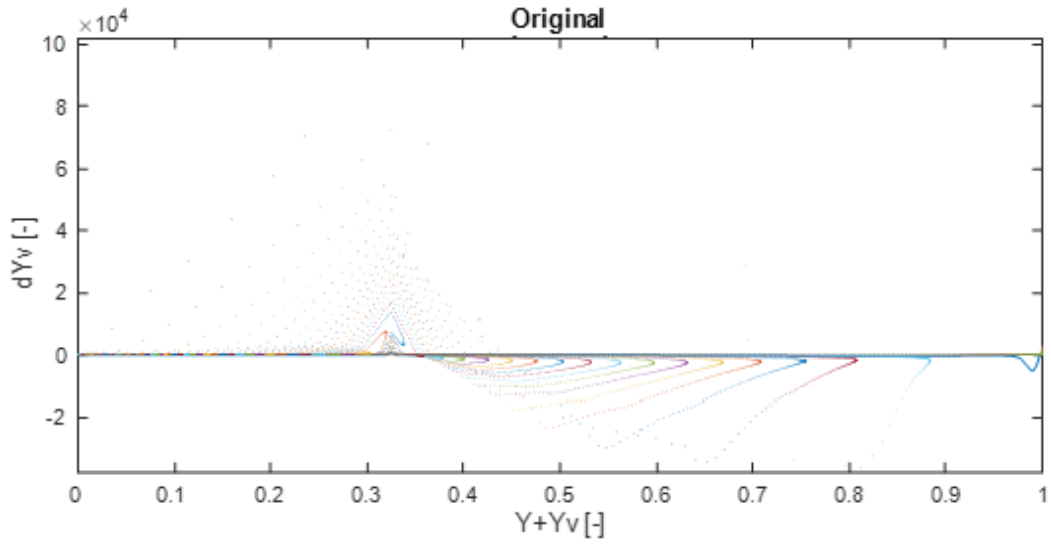


Figura 4.14. Tasa de evaporación/coalescencia según composición. Modelo original.

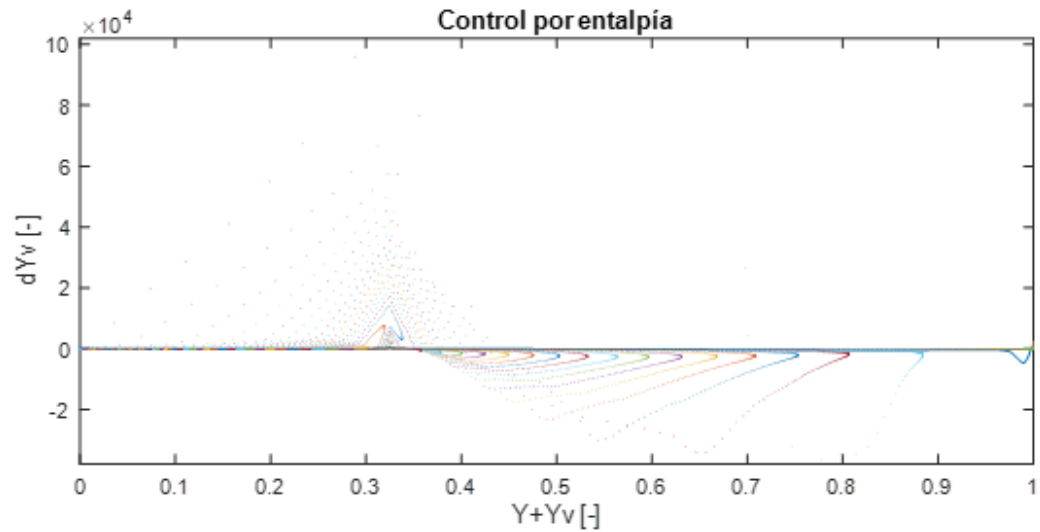


Figura 4.15. Evaporación/coalescencia según composición. Control por entalpía.

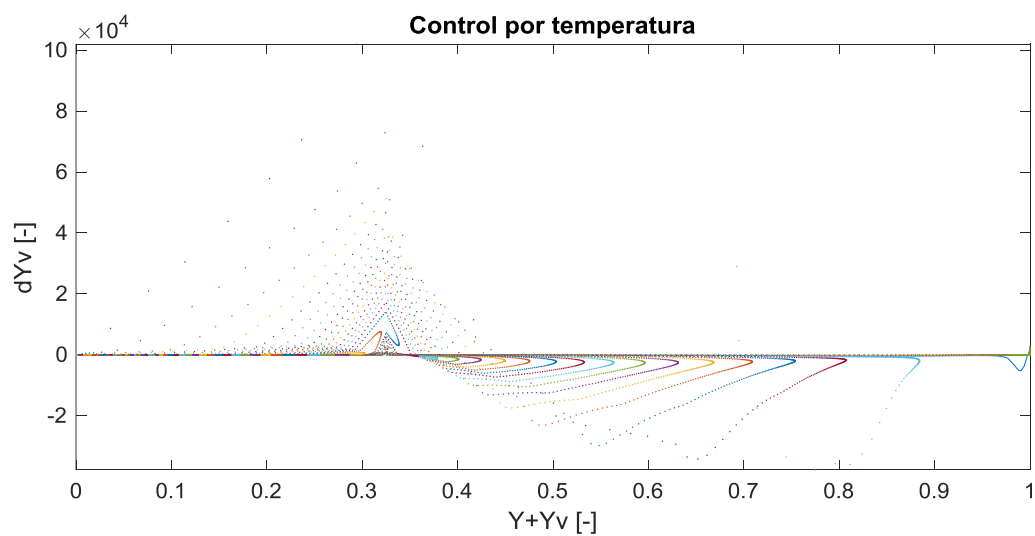


Figura 4.16. Evaporación/coalescencia según composición. Control por temperatura.

La Figura 4.17 y la Figura 4.18 muestran el error relativo entálpico entre la entalpía resultante de la Ecuación 3.7 y la entalpía de la Ecuación 3.61. En ellas sí se observa una diferencia importante entre los distintos modelos. Comenzando por el modelo original, a priori se ve que éste mantiene un error relativo de entalpía bajo, sin embargo, en la zona dónde se produce la bifurcación de la temperatura para bajas temperaturas, se tienen valores muy altos para el mismo. En los dos nuevos modelos, si bien aparecen unos picos cerca del punto de inflexión común en el resto de variables, el error de entalpía para estos picos es ínfimo, alrededor del 0.12% para el método de cálculo de la temperatura controlado por la entalpía y del 0.06% para el método controlado por la propia temperatura. Por tanto, el error presentado por el sistema iterativo controlado por la temperatura es un poco menor que el error propio del método iterativo controlado por la entalpía.

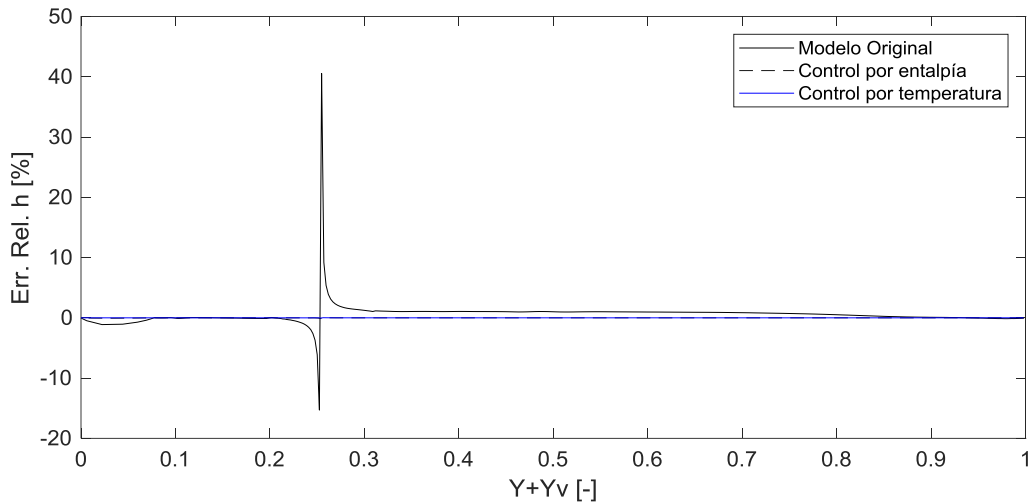


Figura 4.17. Error relativo de entalpía para los tres modelos.

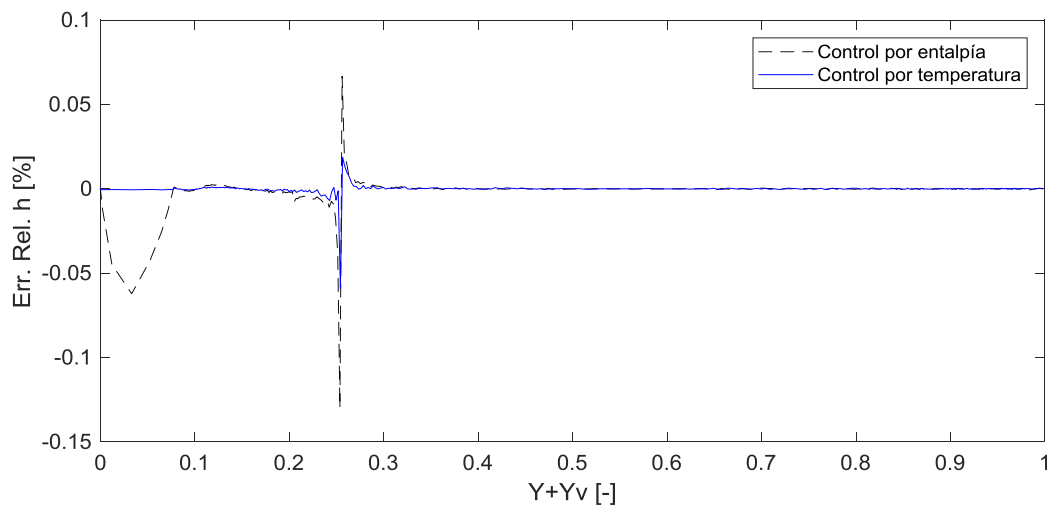


Figura 4.18. Error relativo de entalpía para los modelos controlados por entalpía y temperatura.

Pasando a las representaciones espaciales, aunque la Figura 4.19 represente la distribución de temperatura del chorro de combustible para el caso cuyo modelo termodinámico es controlado por la temperatura también da una idea general de la situación del propio chorro en el espacio, pues en los tres modelos se tienen resultados muy parecidos. En esta imagen se aprecia cómo el valor de la temperatura se distribuye por niveles según la cantidad de combustible existente en la mezcla, cuanto más cerca de la tobera de inyección y del eje longitudinal (dónde se halla gran parte del combustible líquido) menor es la temperatura. Además, el frente del chorro de combustible tiene una forma parecida a la cabeza de un champiñón, este suceso se debe a las fuerzas aerodinámicas de resistencia que aparecen por el movimiento relativo entre el combustible y el gas del interior del cilindro.

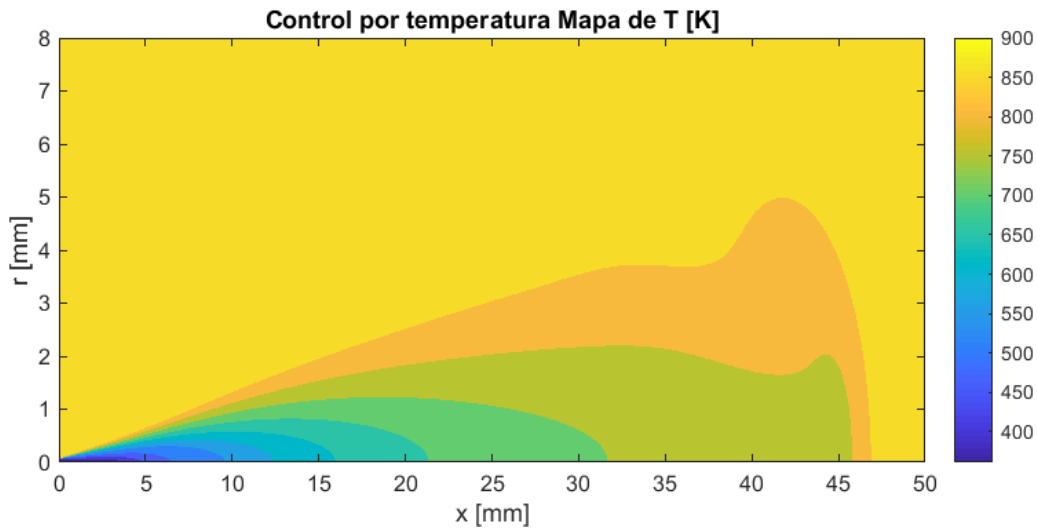


Figura 4.19. Mapa de temperatura para el modelo controlado por la temperatura.

Hacer zoom en una zona más cercana a la tobera (Figura 4.20, Figura 4.21 y Figura 4.22) permite comparar las diferencias existentes entre los tres modelos, y salta a la vista otro detalle: para el modelo original se hace visible un escalonamiento de la temperatura en las zonas próximas al inyector debido al truncamiento implementado, mientras que en los dos nuevos modelos existe continuidad.

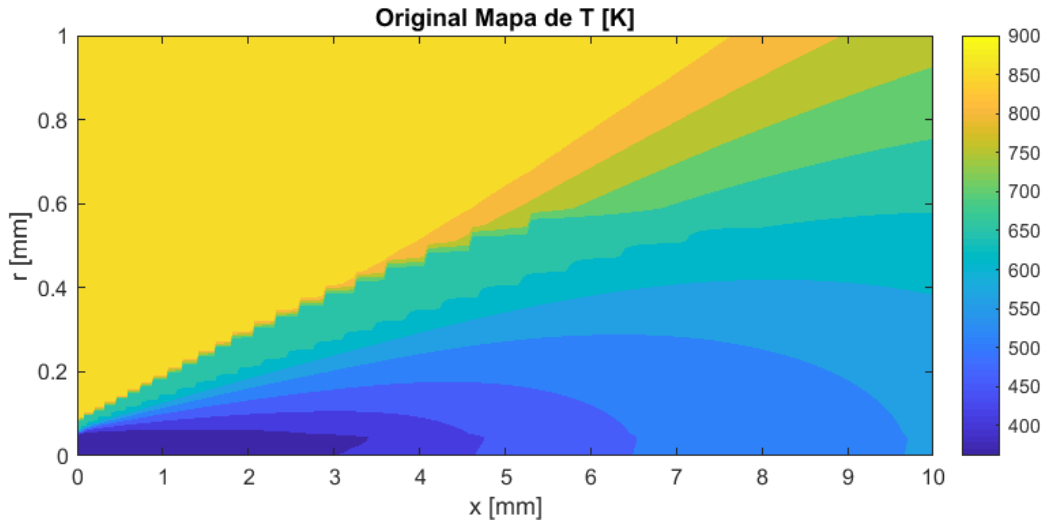


Figura 4.20. Mapa de temperatura para el modelo original.

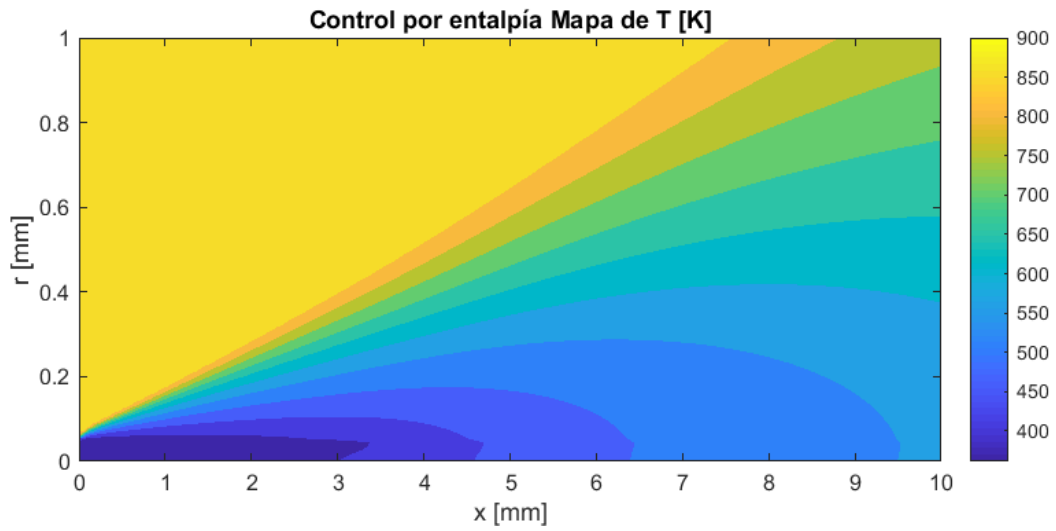


Figura 4.21. Mapa de temperatura para el modelo controlado por entalpía.

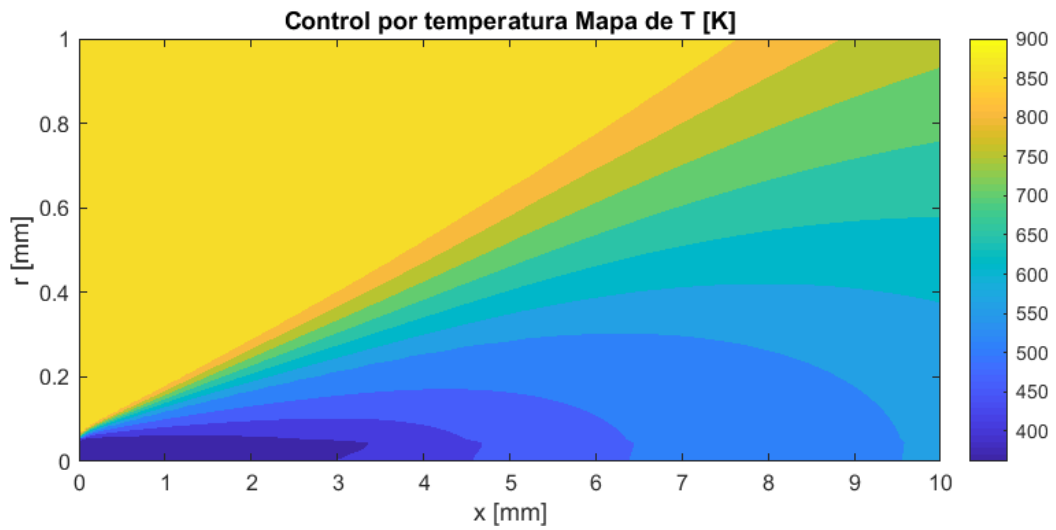


Figura 4.22. Mapa de temperatura para el modelo controlado por temperatura.

Así mismo, la representación de la fracción másica de combustible, tanto líquido como gaseoso (de la Figura 4.23 a la Figura 4.28), hace mucho más visual y fácil de entender su distribución espacial: el combustible líquido se concentra en la zona cercana al inyector entorno al eje longitudinal, para luego dar paso al combustible en estado gaseoso, que conforme se aleja del inyector aumenta hasta un máximo para volver a caer. Al comparar los tres modelos, se puede ver en la Figura 4.29 que la profundidad del chorro de combustible (líquido y gaseoso) es muy parecida, los modelos en los que el método numérico para calcular la temperatura está controlado por la propia temperatura (original y nuevo) alcanzan mayor profundidad; lo que queda corroborado también por la Tabla 4.3, que recoge las longitudes líquidas de los modelos. El hecho de tener una longitud líquida algo mayor es un punto positivo para los modelos controlados por la temperatura, pues se sabe del proyecto anterior que la longitud líquida obtenida se quedaba ligeramente inferior a la que decían los datos experimentales (parámetro difícil de medir y con un error considerable en su medida). El ejemplo original y el nuevo prototipo de cálculo cuyo modelo termodinámico está controlado por la temperatura presentarían un alcance del chorro idéntico, sin embargo, el chorro del modelo original supera en alcance de longitud líquida al nuevo por 3 centésimas de milímetro, cómo muestra la siguiente tabla:

Long. Líquida Original	Long. Liq. Ctrol. Entalpía	Long. Liq. Ctrl. Temp.
9.81 mm	9.66 mm	9.78 mm

Tabla 4.3. Longitud líquida. Original, control por entalpía y control por temperatura.

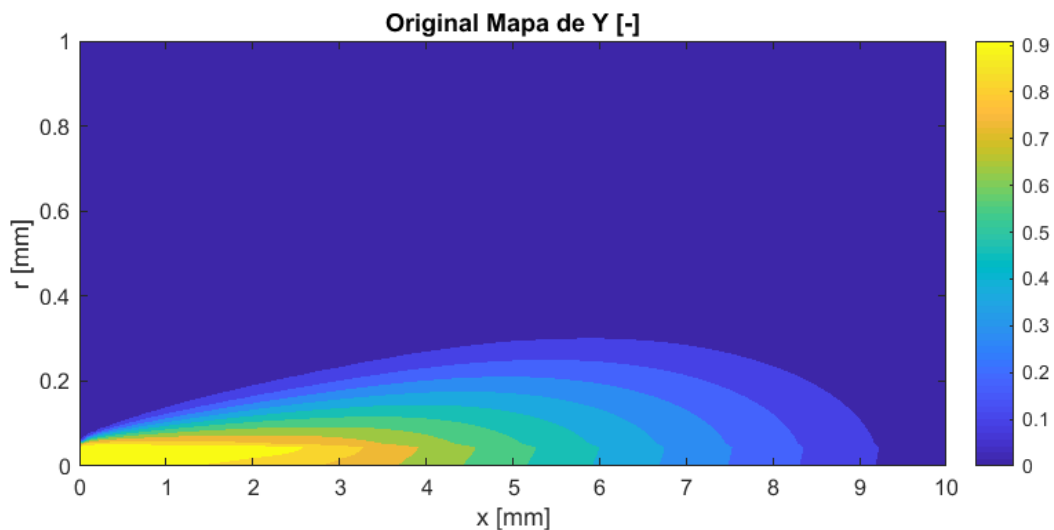


Figura 4.23. Mapa de combustible líquido para el modelo original.



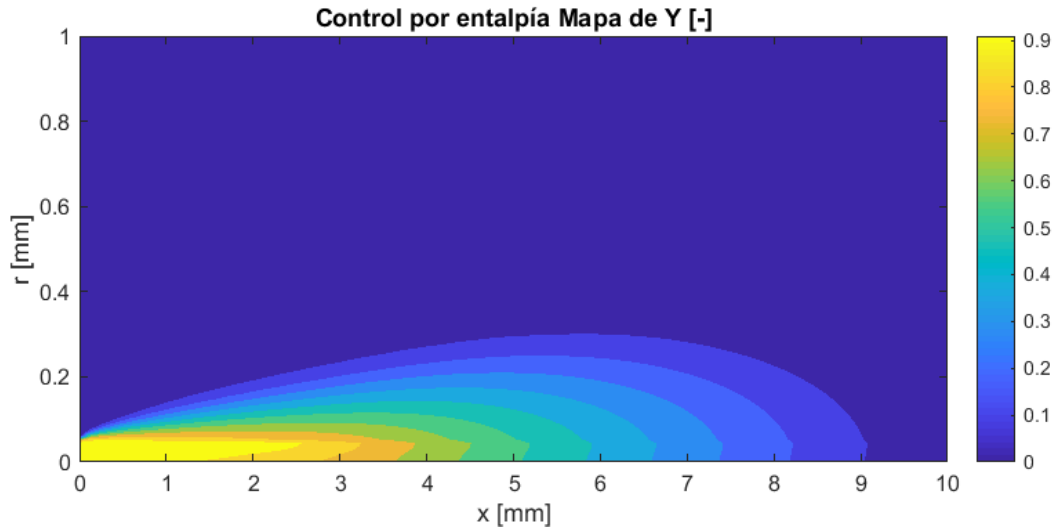


Figura 4.24, Mapa de combustible líquido para el modelo controlado por entalpía.

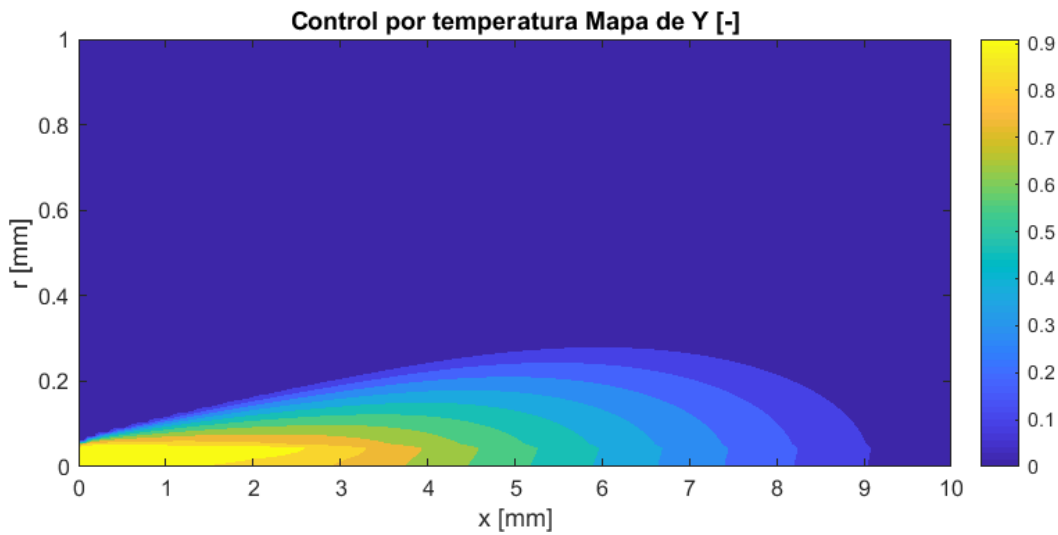


Figura 4.25. Mapa de combustible líquido para el modelo controlado por temperatura.

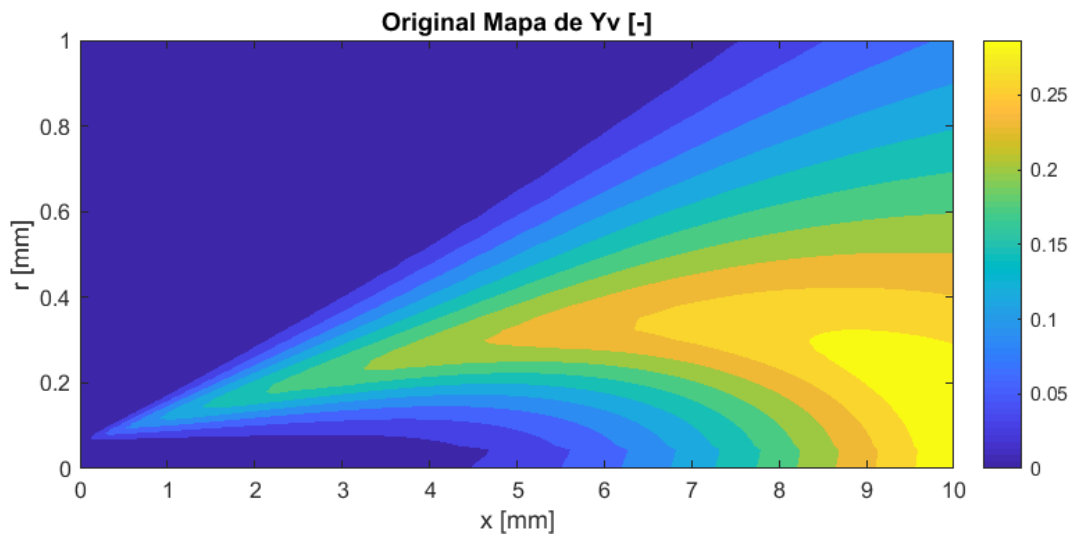


Figura 4.26. Mapa de combustible gaseoso para el modelo original.

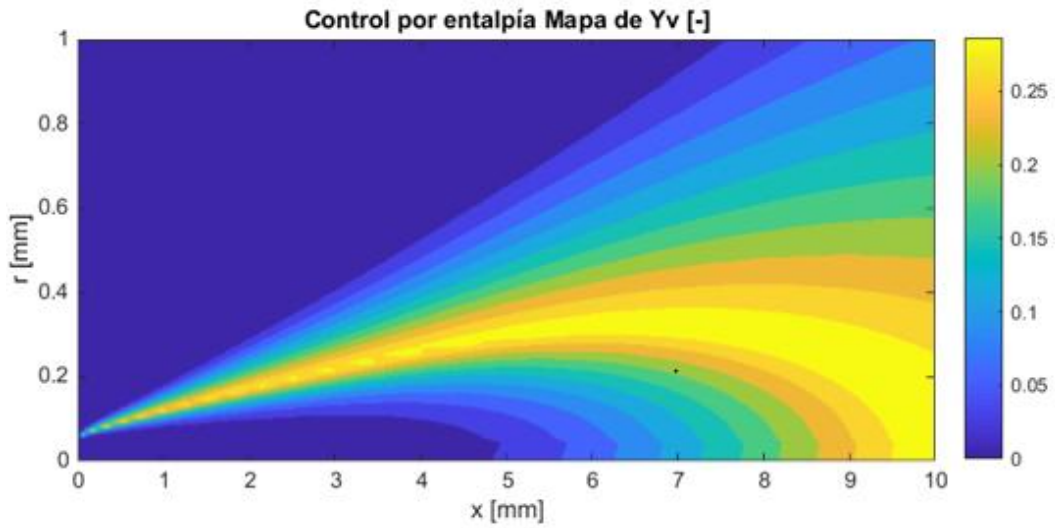


Figura 4.27. Mapa de combustible gaseoso para el modelo controlado por entalpía.

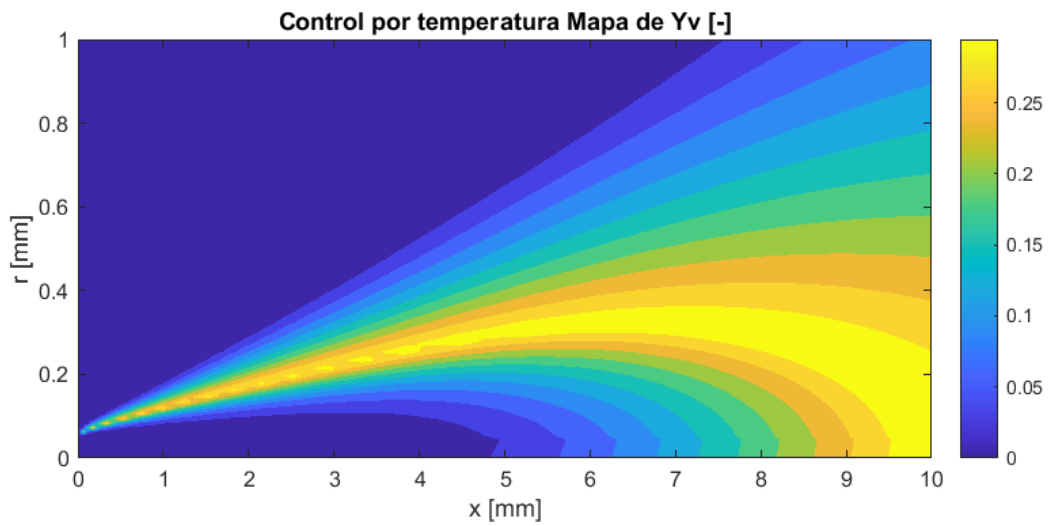


Figura 4.28. Mapa de combustible gaseoso para el modelo controlado por temperatura.

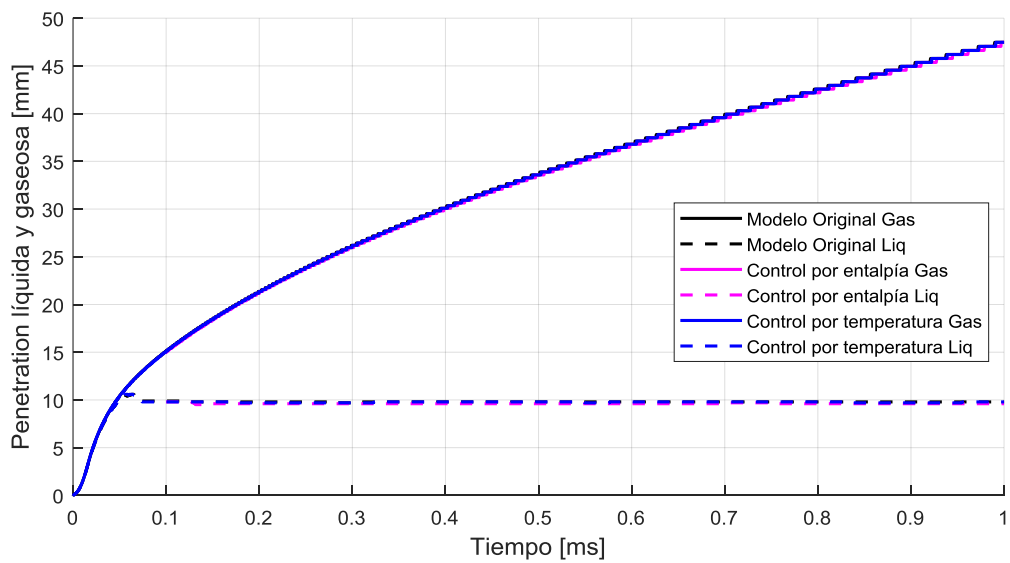


Figura 4.29. Penetración del chorro: original, control por entalpía y por temperatura.

Los resultados obtenidos con los tres modelos son muy similares, pero el modelo original queda descartado por las pequeñas discontinuidades debidas al truncamiento implementado y el error tan alto que presenta en el cálculo de la entalpía. Entre los nuevos modelos, queda decidir cuál es el mejor. En cuanto a los nuevos modelos se refiere, el modelo cuyo método para calcular la temperatura está controlado por la temperatura presenta un menor error relativo de entalpía mientras que el modelo cuyo método para calcular la temperatura es controlado por la entalpía posee una mayor longitud líquida, por lo que están igualados.

Para decidir, se debe estudiar también el tiempo de cálculo y el coste computacional. En ese aspecto, la siguiente tabla recoge el tiempo de cálculo necesario por cada modelo para llegar a 1ms del proceso de formación de chorro diésel:

Modelo Original	Control por temperatura	Control por entalpía
30150s	21000s	25300s

*Tabla 4.4. Tiempo de cálculo para 1ms.*

Los dos nuevos modelos reducen el tiempo de cálculo con respecto al original considerablemente, pero el modelo cuyo método es controlado por la temperatura es más rápido que el modelo con el método controlado por la entalpía, éste ahorra un 30% de tiempo de cálculo mientras que el controlador por la entalpía recorta un 16%. Por tanto, teniendo esto en cuenta y los puntos a favor antes mencionados, el modelo de cálculo es aquel con el modelo termodinámico controlado por temperatura.

#### Comparativa del modelo con la nueva ecuación de estado del combustible líquido con el modelo original

En este ejemplo, el caso calculado ha sido ejecutado con un nuevo modelo que mantiene los cambios introducidos por el nuevo modelo termodinámico del combustible para calcular la temperatura a partir de la entalpía, por tanto su modelo termodinámico está controlado por la temperatura y salva las discontinuidades presentadas por el modelo original. Pero también presenta una modificación que afecta a la ecuación de estado del combustible en estado líquido: se adopta la ecuación de estado que proporciona el NIST para el n-dodecano y también se tabulan los valores de la densidad y sus derivadas respecto de la temperatura y la presión con el objetivo de reducir el coste computacional. Esta tabulación se ha realizado teniendo en cuenta dos variables, la temperatura y la presión del combustible.

Como se puede apreciar en las figuras que van de la Figura 4.30 a la Figura 4.39, además de en la Tabla 4.5, apenas hay diferencias con respecto a los resultados del modelo anterior, por lo que se puede dar como válido este modelo. Ni si quiera la penetración del combustible gaseoso varía, su evolución a lo largo del tiempo prácticamente se superpone

con la otra; e incluso el valor de la longitud líquida apenas varía. La diferencia más notable se da en el error relativo de entalpía: los niveles de error son ínfimos, al igual que antes, salvo que, en el punto de inflexión anterior comentado, este error crece ligeramente hasta un valor del 0.9%; un error más que aceptable.

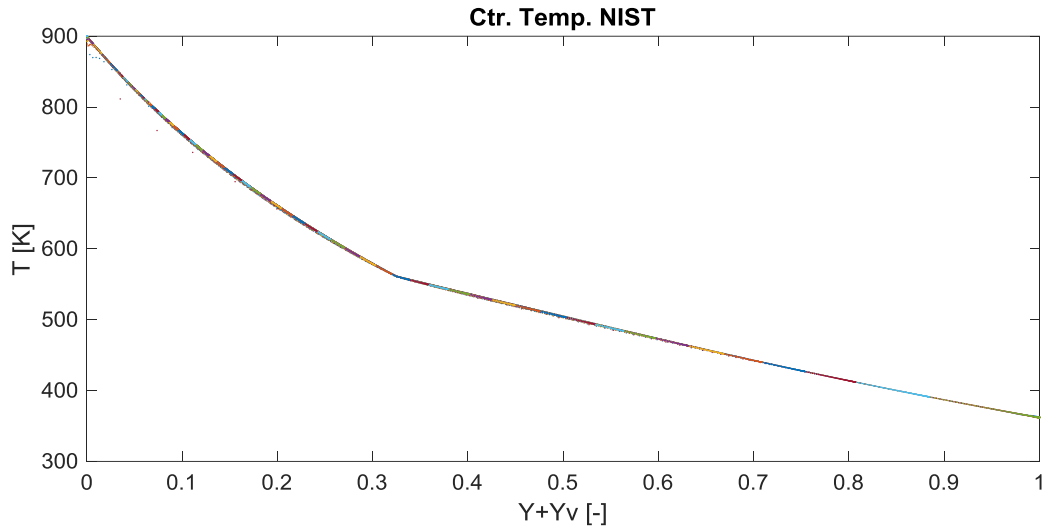


Figura 4.30. Temperatura según composición. Control por temperatura NIST.

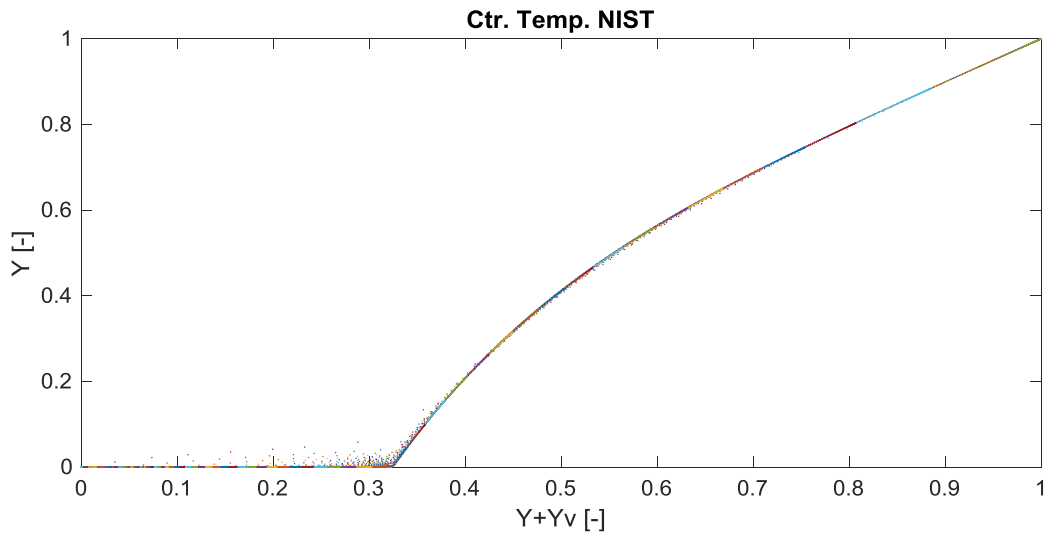


Figura 4.31. Combustible líquido según composición. Control por temperatura NIST.

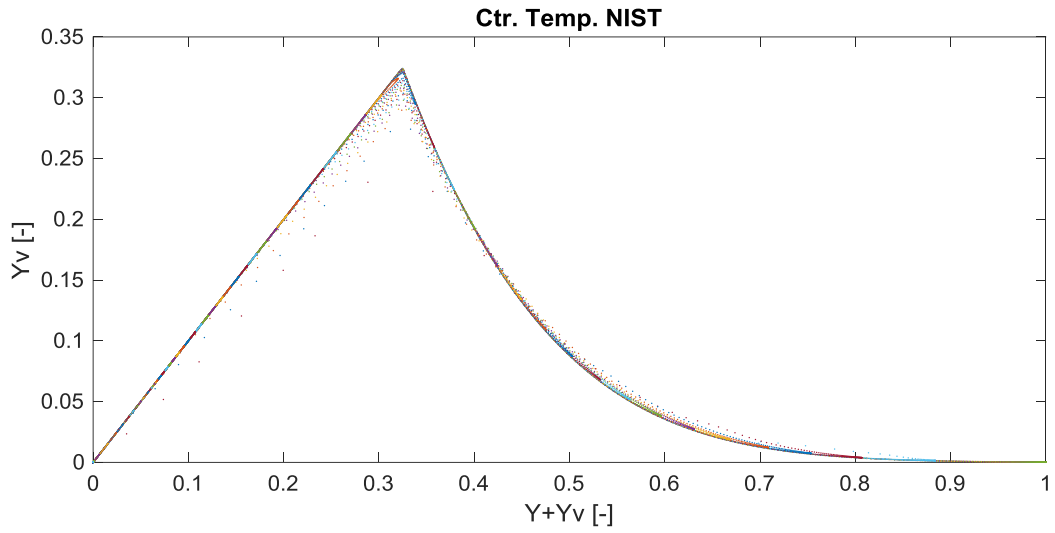


Figura 4.32. Combustible líquido según composición. Control por temperatura NIST.

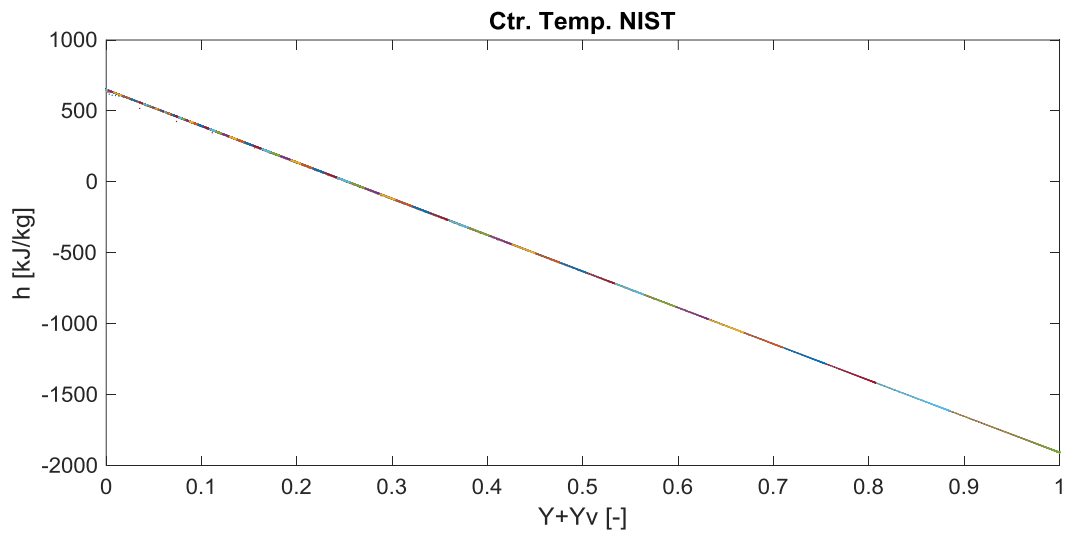


Figura 4.33. Entalpía según composición. Control por temperatura NIST.

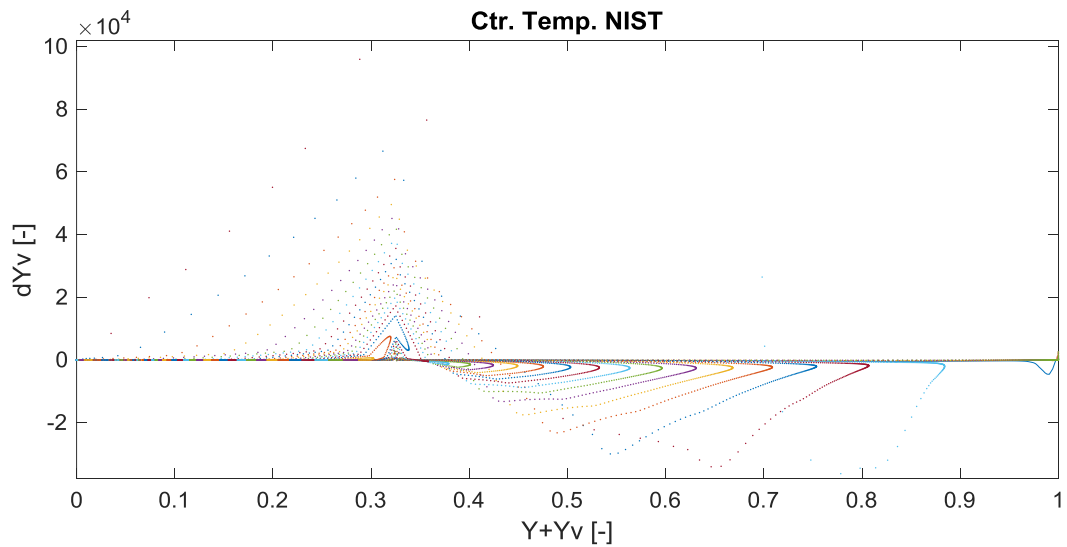


Figura 4.34. Evaporación/coalescencia según composición. Control temperatura NIST.

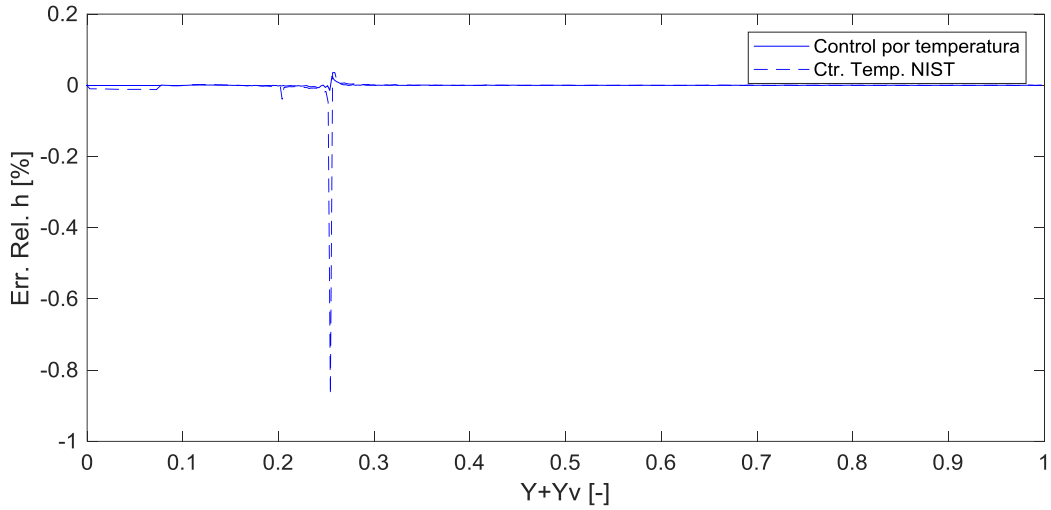


Figura 4.35. Error relativo de entalpía para los modelos controlados por temperatura.

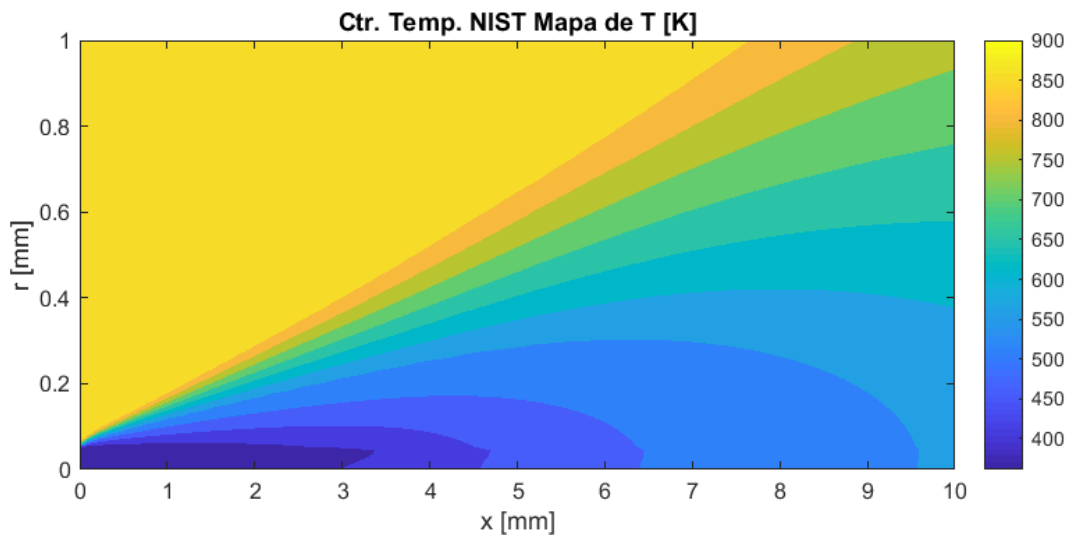


Figura 4.36. Mapa de temperatura. Control por temperatura NIST.

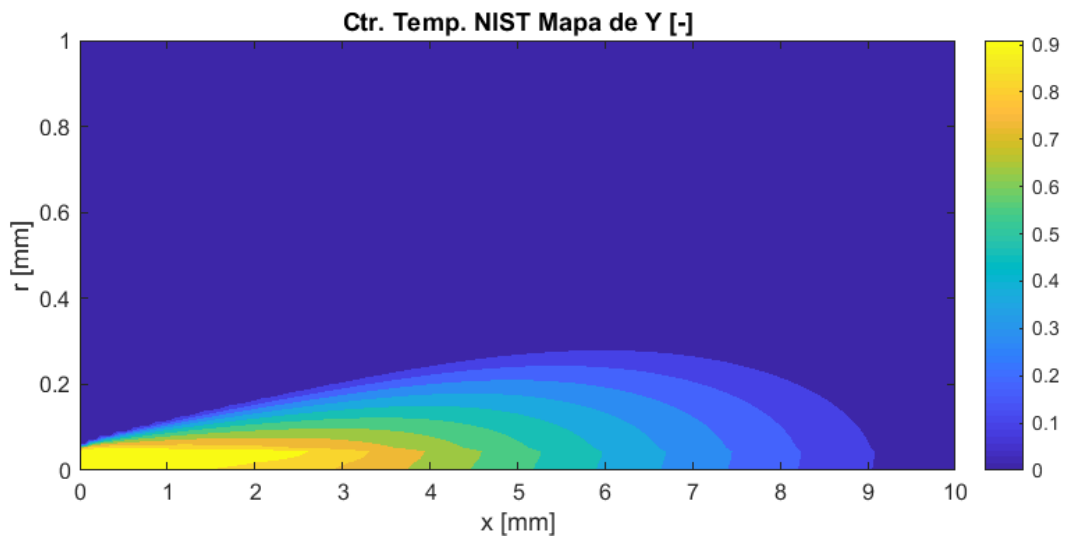


Figura 4.37. Mapa de combustible líquido. Control por temperatura NIST.

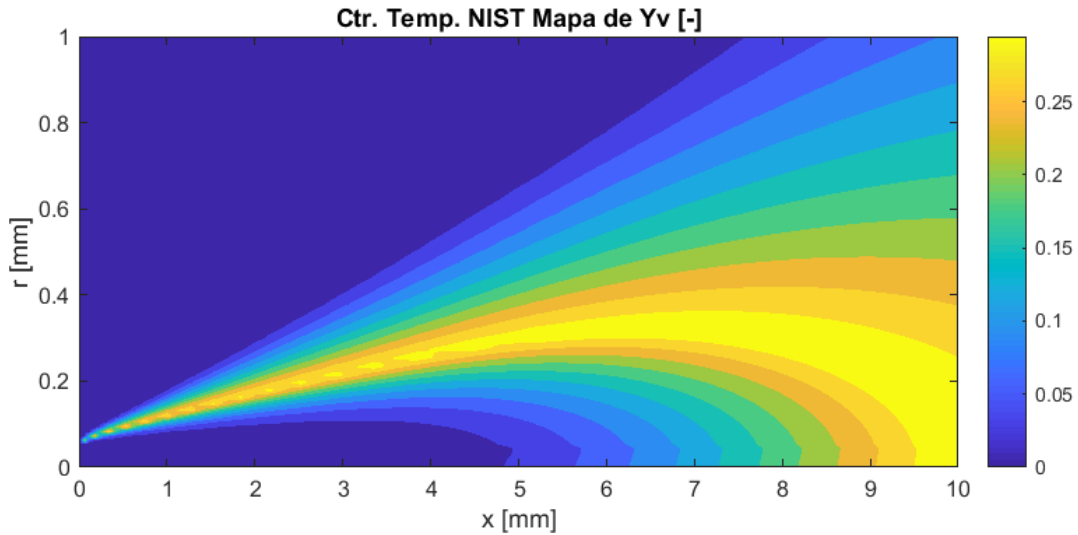


Figura 4.38. Mapa de combustible gaseoso. Control por temperatura NIST.

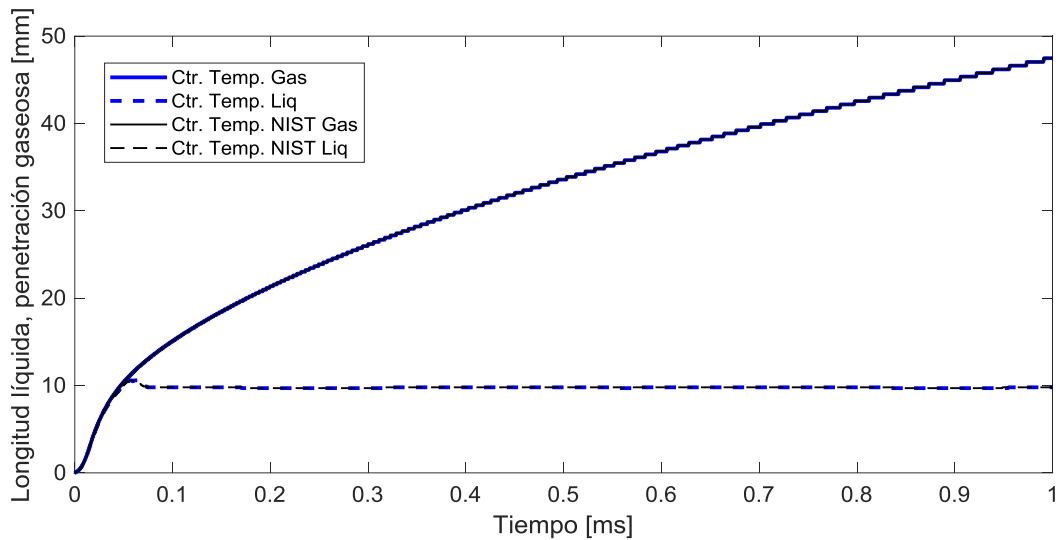


Figura 4.39. Penetración del chorro para los modelos controlados por temperatura.

Long. Liq. Mod. Original	Long. Liq. Ctr. Temp.	Long. Liq. Ctr. Temp. NIST
9.81 mm	9.78 mm	9.78 mm

Tabla 4.5. Longitud líquida. Original y modelos controlados por temperatura.

Por otro lado, un aspecto extremadamente positivo de este modelo es el reducido coste computacional y su velocidad de cálculo. Si el modelo que sólo incluía el cambio del modelo termodinámico para calcular la temperatura de la mezcla tarda unos 21000s al ejecutar el caso de estudio y llegar a 1ms del proceso de inyección, reduciendo el tiempo de cálculo un 30% respecto al modelo original; este nuevo modelo tan sólo tarda 9100s en llegar al mismo punto del proceso de inyección, por lo que tarda menos de la mitad

del tiempo de ejecución del último modelo y reduce el tiempo de ejecución con respecto al modelo original en un 70%:

Modelo Original	Control por temperatura	Ctr. Temp. NIST
30150s	21000s	9100s

Tabla 4.6. Tiempo de cálculo para 1ms de tiempo físico, segundo cambio en el modelo.

Por tanto, este último modelo sería el elegido para llevar a cabo nuevos estudios, pues apenas cambia la solución con respecto al modelo anterior pero si reduce drásticamente el tiempo necesario de cálculo.

### 4.3.2 Resultados LES

Insistiendo de nuevo en lo que se ha comentado antes, debido a la metodología que se ha llevado a cabo durante este proyecto para así asegurar que cada cambio introducido en el modelo funcionara correctamente, el enorme tiempo de cálculo que requieren los cálculos LES por su elevado coste computacional y la falta de horas disponibles en súper computadores tan sólo se ha podido ejecutar el caso de estudio con el modelo en el que únicamente ha sido modificada la forma de calcular las temperatura a partir de la entalpía. Por tanto, en este sub-apartado se compararán los resultados del caso calculado como RANS y del calculado como LES, así como también los resultados LES con los del caso experimental proporcionado por la ECN.

En el caso del caso LES hay que señalar que, al contrario que para cálculos RANS, no se modela la turbulencia (salvo para escalas muy pequeñas) por lo que los resultados obtenidos a cada instante no son homogéneos dentro del volumen del control en ningún caso, incluso si el chorro de combustible estuviera completamente estable y se tuvieran fenómenos de turbulencia periódicos. Esto es causado por la irregularidad, cualidad inherente a la turbulencia. El desarrollo de un fenómeno turbulento es completamente dependiente de las condiciones iniciales y también un suceso transitorio, no es estable y cualquier perturbación aleatoria que aparezca puede originar importantes cambios en su evolución. No obstante, también es un suceso estadístico y se puede predecir la probabilidad de que un evento concreto suceda de cierta manera. Es por ello que el caso como LES ha sido ejecutado y analizado durante un periodo de inyección de 2ms, de esta manera, y una vez se “estabiliza” el chorro de combustible (en torno a 1ms), se realiza el promedio a lo largo del tiempo de las variables que se quieren analizar. Así se consiguen datos estadísticos de la evolución del chorro de combustible y se tienen datos que aclaren qué sucede con mayor probabilidad.



### Comparativa resultados RANS frente a resultados LES

Una forma de ver si los resultados proporcionados por los cálculos LES son válidos o no es, una vez el chorro de combustible se encuentra “estable” dentro del cilindro y tras recopilar suficientes muestras temporales como para promediar, comparar la temperatura y las fracciones másicas de combustible (tanto líquido como sólido) según la composición misma de la mezcla. Atendiendo a la representación de estas variables para el ejemplo calculado en RANS (Figura 4.4, Figura 4.7 y Figura 4.10) y las mismas representaciones propias del caso calculado en LES (Figura 4.40, Figura 4.41 y Figura 4.43), insistiendo en que son una representación promedio de sus valores, se puede comprobar que los resultados son muy parecidos, en ambos casos se siguen las mismas tendencias.

No obstante, los cálculos LES presentan una mayor dispersión en los resultados que el RANS no. La razón de esto reside en la forma de resolver la turbulencia, pues como ya se ha mencionado, para los cálculos RANS la turbulencia se modela en todas las escalas mientras que para los cálculos LES tan sólo se modela en escalas de turbulencia de tamaño inferior a las propias celdas de la malla; y al resolver la turbulencia de forma directa a través de las ecuaciones, se tiene una solución diferente a cada instante por su carácter transitorio, haciendo necesario un promedio de los resultados para obtener la solución más probable del fenómeno estudiado.

Si se atiende a la Figura 4.42 y a la Figura 4.44, representaciones instantáneas de la fracción másica de combustible líquido y gaseoso según la composición de la mezcla respectivamente, es fácil apreciar que para estas representaciones instantáneas el punto de inflexión está más claramente marcado que en las representaciones promediadas de las variables. En este caso, la curva descrita por las variables se parece más a las de las simulaciones RANS, aunque la dispersión de los resultados LES sigue siendo ligeramente mayor; claro que, por la forma de resolver la turbulencia, los resultados oscilan en cada instante de tiempo, de ahí que al realizar el promedio temporal las curvas de las variables se suavicen y el punto de inflexión no quede tan marcado en los gráficos.

Al haber resuelto de esta manera las ecuaciones de un problema turbulento, se tiene una solución más próxima a la realidad, y de ahí que se tengan unas tendencias generales (que coinciden con las descritas en el apartado 4.3.1 de este capítulo y serían la solución estacionaria del problema) alrededor de las cuales oscilan la solución total del problema (compuesta por la estacionaria y la turbulenta). En este caso, el ejemplo calculado como LES no presenta un punto de inflexión tan marcado para la composición de la mezcla que afecte significativamente al resto de propiedades del fluido; más bien aparece un zona cercana a este valor de inflexión (fracción másica de combustible total en torno a 0.3) en la composición de la mezcla en la cuál se da el cambio de tendencia a seguir para las diferentes variables analizadas.

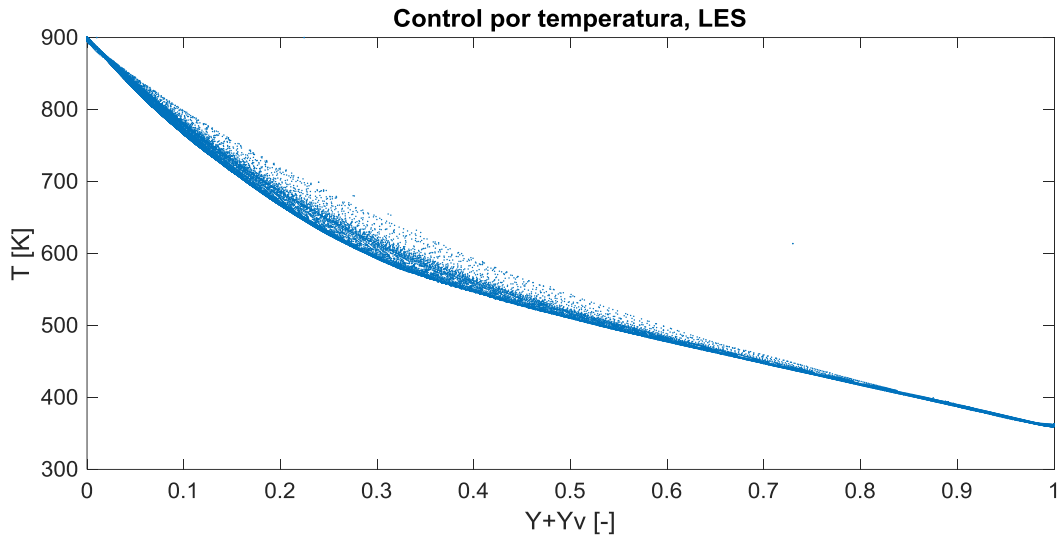


Figura 4.40. Temperatura según composición. Modelo LES. Promedio.

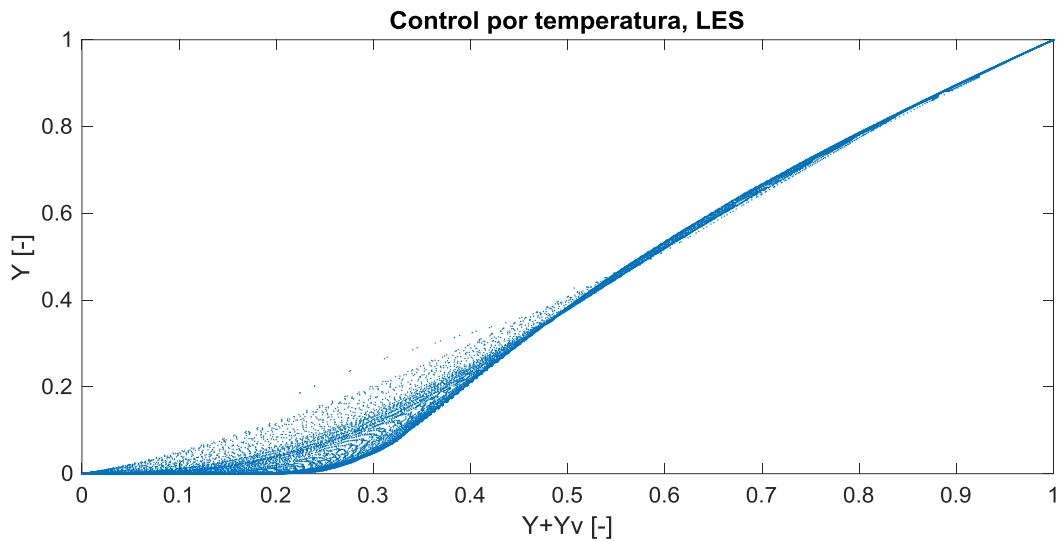


Figura 4.41. Combustible líquido según composición. Modelo LES. Promedio.

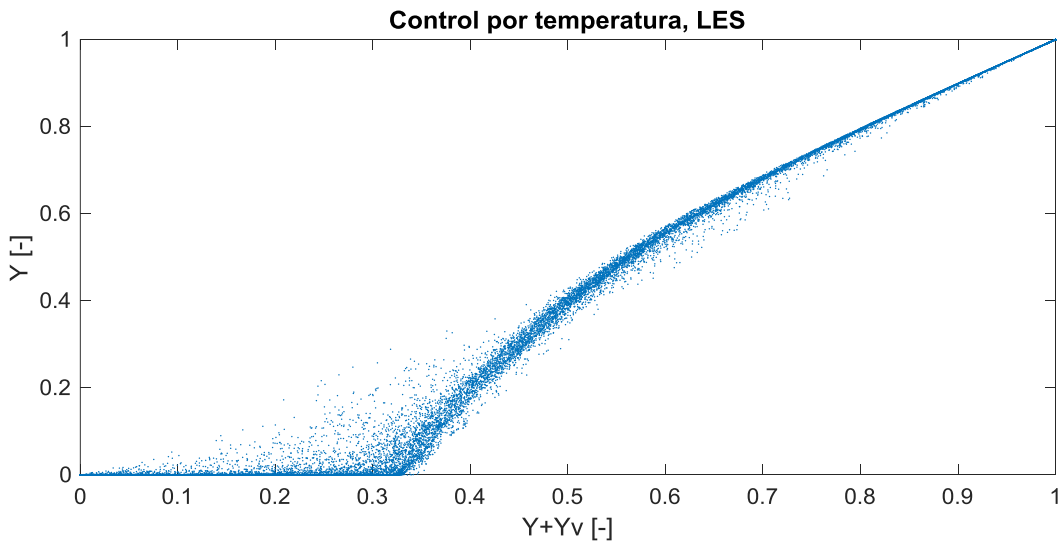


Figura 4.42. Combustible líquido según composición. Modelo LES. Instante 1.9 ms.

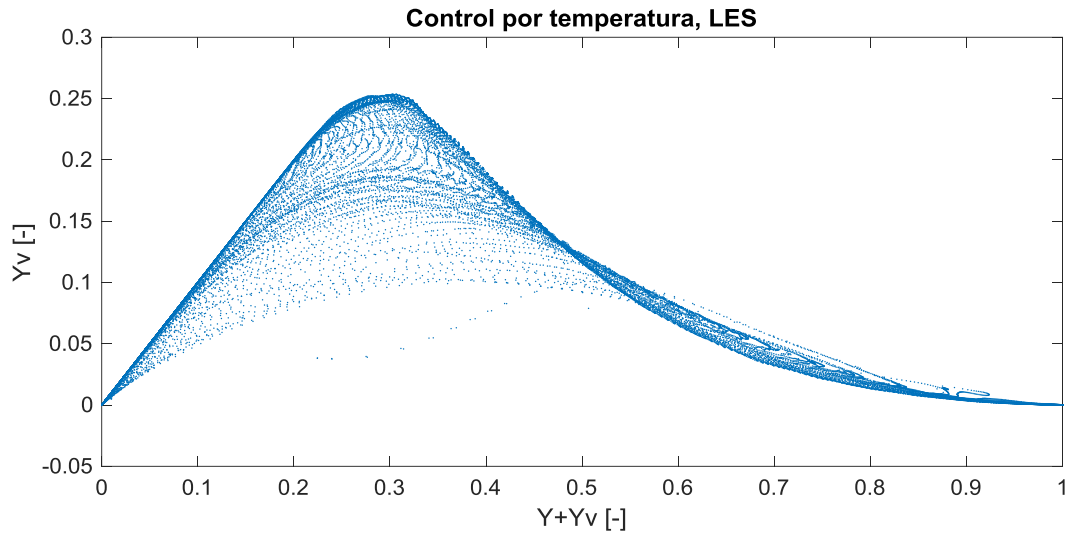


Figura 4.43. Combustible gaseoso según composición. Modelo LES.

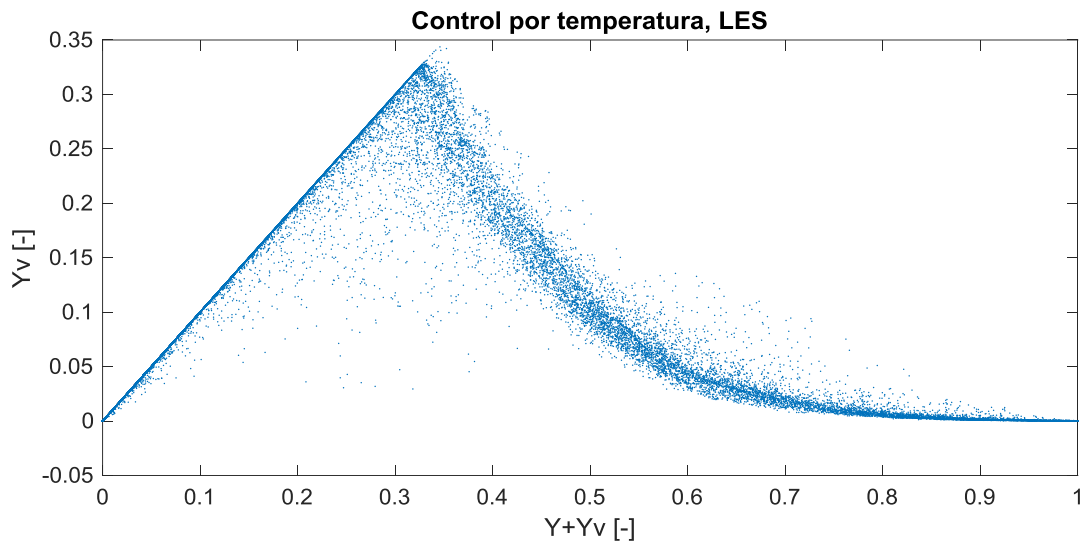


Figura 4.44. Combustible gaseoso según composición. Modelo LES. Instante 1.9 ms.

En lo referente a la penetración y longitud líquida del chorro (Figura 4.45 y Tabla 4.7), se aprecia que la evolución temporal de la penetración es muy parecida tanto en el estudio RANS como LES, sin embargo, sí se aparecen diferencias claras para la longitud líquida. A diferencia del caso RANS, la longitud líquida del chorro oscila todo el tiempo para el estudio LES y adquiere mayores valores (y más parecidos a los del caso real, como se verá más adelante). Estas diferencias se deben a que el caso RANS modela por completo la turbulencia mientras que el caso LES sólo las escalas más pequeñas, resolviendo de esta manera las ecuaciones de Navier-Stokes de una forma más directa, además de que usa un modelo de turbulencia diferente.

Long. Liq. Modelo Original	Long. Liq. Ctr. T. RANS	Long. Liq. Ctr. T LES
9.81 mm	9.78 mm	11.60 mm

Tabla 4.7. Longitud líquida. Original, modelo RANS y modelo LES.

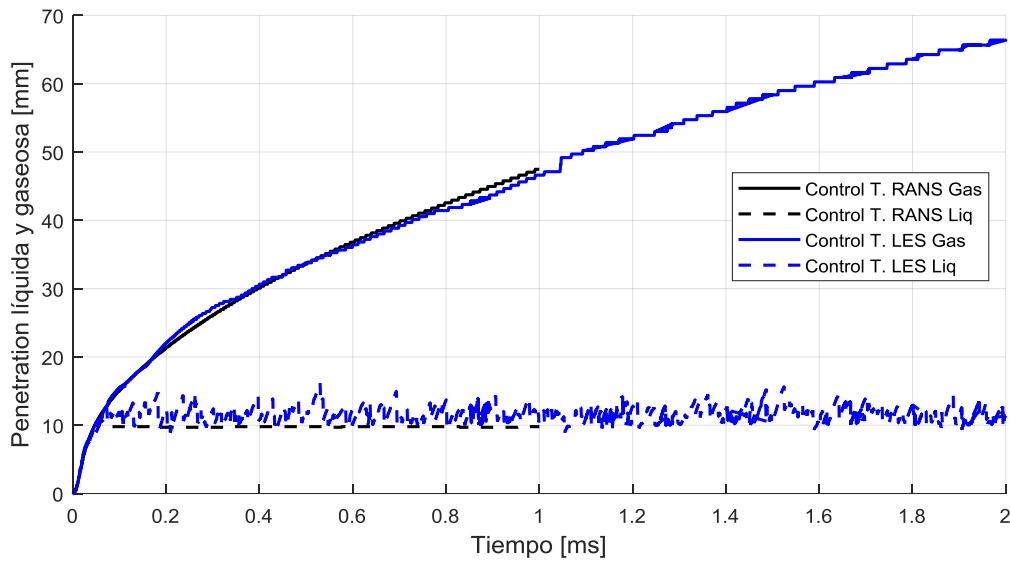


Figura 4.45. Penetración del chorro. Modelo LES.

Si se representa el mapa de temperatura de un plano longitudinal, Figura 4.46, se puede crear una idea de la evolución del chorro de combustible y compararla con la imagen formada por la Figura 4.19. Existen parecidos entre ambos modelos: al alejarse del inyector y del eje axial, la temperatura de la mezcla crece, pues el combustible frío se va mezclando con el gas del interior del cilindro, calentando y evaporando; también se distingue como una cabeza de champiñón, salvando las distancias, en el frente del chorro al ser este frenado por las fuerzas de resistencia aerodinámicas. No obstante, en este ejemplo calculado mediante LES, se observa que la evolución del chorro de combustible no es tan pulcra como sugería la figura, ni mucho menos ideal. La turbulencia ocasiona volutas y torbellinos que le confieren al chorro un aspecto un tanto irregular.

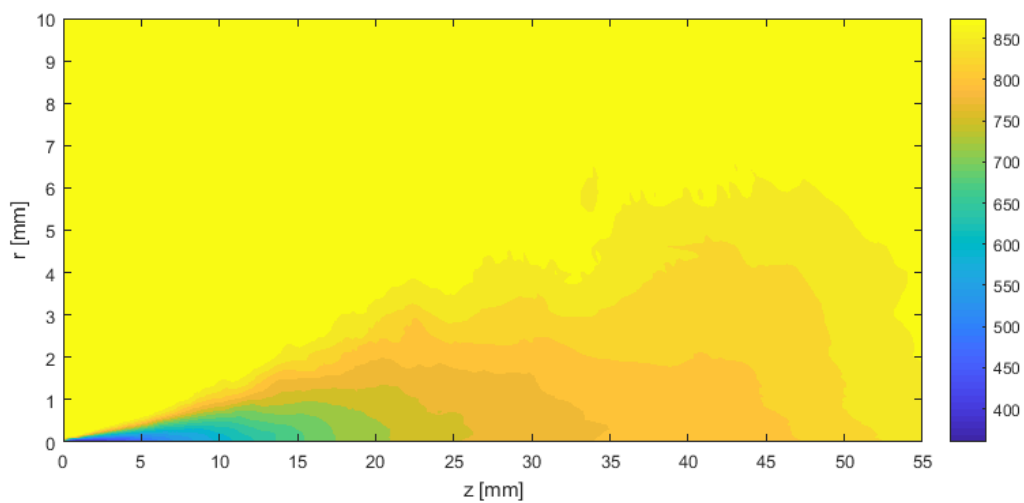


Figura 4.46. Mapa de temperatura [K]. Modelo LES.

Comparativa resultados LES frente a resultados experimentales

Si bien los resultados anteriores, al compararlos con la solución proporcionada por el modelo de cálculo en RANS, guardan similitudes y tendencias entre ellos, como un indicador de que el modelo desarrollado para LES es válido, falta corroborar esto con los resultados experimentales. En primer lugar se analizará la profundidad del chorro de combustible frente al tiempo y la longitud líquida, continuando con una comparativa normalizada entre los contornos del chorro de combustible medido experimentalmente y el calculado computacional mente, y por último se estudiará la evolución normalizada de la velocidad del chorro y de la fracción másica de combustible en estado gaseoso a través del eje longitudinal y en dirección radia para dos planos transversales.

La Figura 4.47 muestra la evolución temporal de la penetración gaseosa del chorro diésel tanto para el caso experimental como el modelo para LES. Se observa que no son idénticas y que, durante la primera parte de la formación del chorro, el modelo LES alcanza valores ligeramente superiores a los experimentales; pero también se aprecia que a partir de que comienza a estabilizarse el chorro, ambos resultados se hacen prácticamente coincidentes y el modelo computacional permanece dentro de los estrechos márgenes de incertidumbre de los datos experimentales. Por otro lado, con la Tabla 4.8 es posible comparar las longitudes líquidas del modelo original para RANS, del caso experimental y del modelo LES. Aquí se ve claramente la evolución de la solución aportada por los modelos de cálculo computacional. En primera instancia, con el modelo original para cálculos RANS, se partía de una solución para la cuál la longitud líquida del chorro es de 9.81 mm, mientras que con el nuevo modelo de cálculo LES la longitud líquida del chorro diésel crece hasta las 11.6 mm, más próxima al rango de tamaños proporcionados por los datos experimentales. El caso RANS dista de este caso por debajo del rango mientras que el modelo LES lo hace por arriba, aunque es cierto que la diferencia con este rango de valores es ligeramente menor. Estas diferencias pueden deberse a varios motivos: el primero es que las medidas experimentales, aun siendo realizadas con rigurosidad, no dejan de ser inexactas y presentar cierto error, por la complejidad que supone tomarlas, y de ahí que no se tenga un valor muy concreto en lugar de un rango de valores y éste pueda variar entorno a un 10%; el segundo motivo es el ajuste de los coeficientes utilizados en los modelos, para aproximar más la solución a la realidad debería realizarse un estudio de los coeficientes integrantes de los modelos que afecten en mayor medida a los parámetros que más disten de los datos experimentales; y el último es la forma de realizar el promedio ya que el modelo LES sí tiene en cuenta las fluctuaciones turbulentas y se necesita promediar muchos instantes del fenómeno transitorio, por lo que puede dar soluciones para diferentes instantes que sí se encuentren dentro de este rango de valores experimentales.

Long. Liq. Modelo Original	Long. Liq. Experimental	Long. Liq. Ctr. T LES
9.81mm	10.7-11 mm	11.60 mm

*Tabla 4.8. Longitud líquida. Original, experimental, LES.*

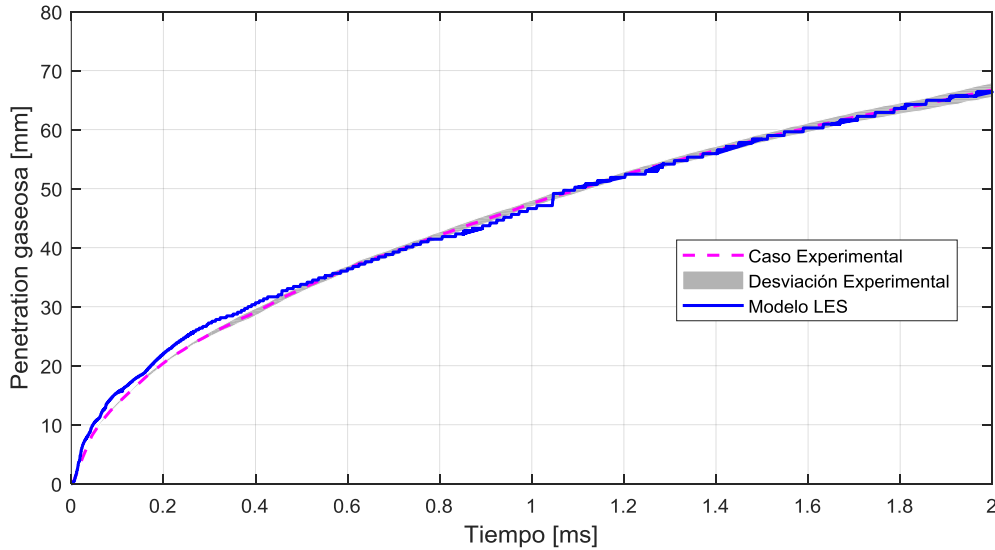


Figura 4.47. Penetración del chorro para el caso experimental y el modelo LES.

Al comparar el contorno de los chorros se debe indicar que por motivos técnicos, en la prueba experimental no se pudo determinar el contorno del combustible gaseoso en su totalidad, por lo que la Figura 4.48 se ajusta a la franja en la que sí es posible realizar esta comparación. También se debe señalar que se ha considerado como límite para la presencia de combustible gaseoso un valor de la fracción másica de combustible en este estado igual a 0.001. Al observar la imagen, es fácil apreciar ambos contornos y ver cómo se entrelazan. Lógicamente, no coinciden con exactitud las trazas que delimitan su contorno por los errores de medición del caso experimental y porque la solución computacional no dejar de ser una aproximación; pero sí queda claro que la solución computacional se ajusta muy bien al contorno real del chorro de combustible.

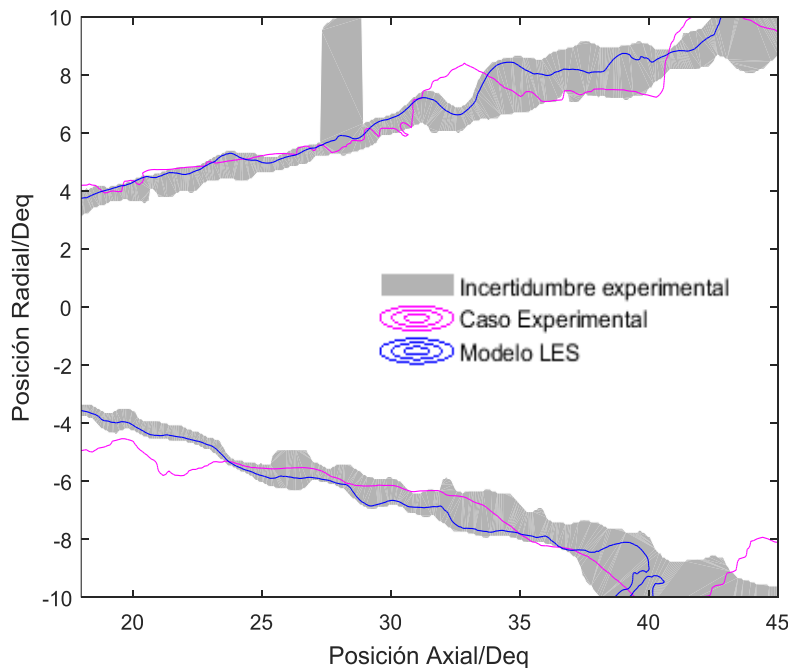


Figura 4.48. Contorno del chorro para el caso experimental y el modelo LES.

Tras esta comparación, le llega el turno al estudio de la evolución de la velocidad en el eje axial y en dirección radial, además de la evolución del combustible en estado gaseoso para ambas direcciones.

En lo referente a la velocidad, la Figura 4.49 y la Figura 4.50 representan la velocidad normalizada de las partículas del eje de simetría del cilindro en esta misma dirección y también el promedio (según su posición radial) de esta velocidad (partido de su valor en el eje axial) para dos planos transversales al eje de simetría, situados a 25 y 45 mm de la tobera de inyección respectivamente.

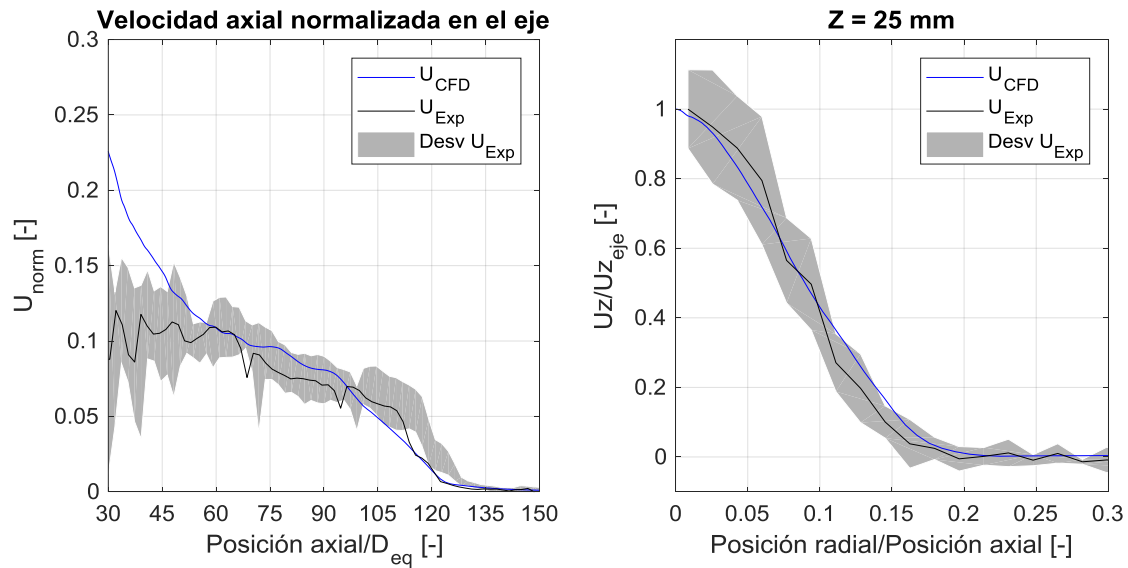


Figura 4.49. Velocidad normalizada. Eje longitudinal y plano normal a 25 mm.

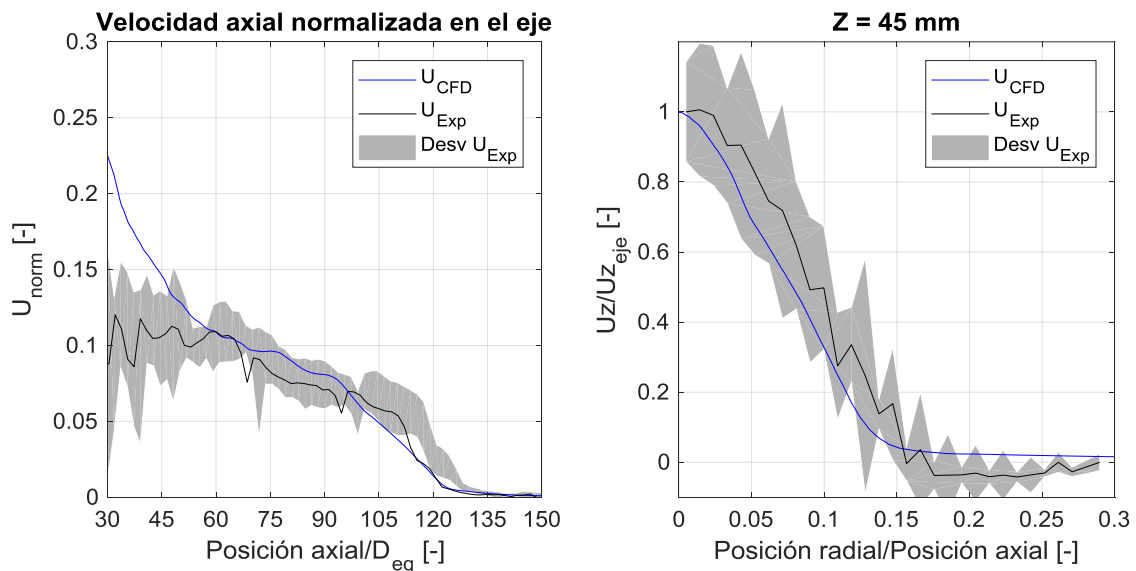


Figura 4.50. Velocidad normalizada. Eje longitudinal y plano normal a 45 mm.

La predicción de la velocidad en el eje se acerca mucho a la proporcionada por los datos experimentales, y la mayor parte de su recorrido a lo largo del eje se encuentra dentro del rango de incertidumbre del caso experimental. Sin embargo, existe una

diferencia bastante notable en la zona más próxima al inyector, en esta parte del dominio el modelo computacional sobrestima el valor de la velocidad. Por otro lado, al fijarse en la evolución de la velocidad axial en el radio, la curva dada por los cálculos LES se ajusta en los dos planos estudiados muy bien. Aunque la diferencia no es grande, sí es cierto que el ajuste es mejor en el plano situado a 25 mm del inyector que en el situado a 45. Esto se puede explicar con la Figura 4.51, que representan la distribución espacial del valor de la velocidad en estos planos normales al eje. En el corte transversal a 25 mm de la tobera se aprecia una distribución de velocidades alrededor del eje axial mucho más concéntrica, regular y de mayor valor que en el plano situado a 45 mm, dónde la velocidad se frena y su distribución se vuelve más irregular ante la aparición de mayores fuerzas de resistencia aerodinámica y dónde los efectos de la turbulencia son más notables, por lo que la dispersión de los valores también crece.

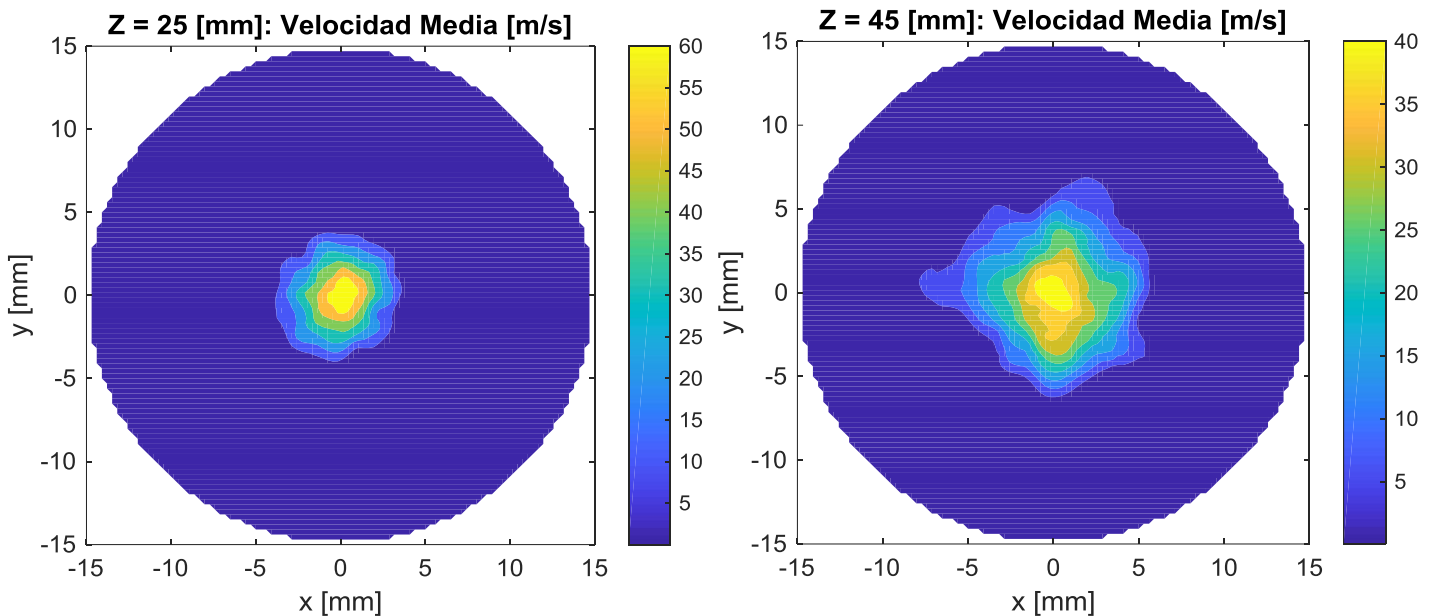


Figura 4.51. Contornos de velocidad para el modelo LES.

La Figura 4.52 y la Figura 4.53, en este caso, recrean la evolución de la fracción másica de combustible gaseoso a lo largo del eje longitudinal y también en dirección radial para los mismos planos que antes, uno situado a 25 y otro a 45 mm de la boca del inyector. La representación del valor de la fracción másica de combustible en estado gaseoso es su promedio en los planos de corte transversales y está dividido de su valor en el eje.

Se observa claramente que la predicción dada por el modelo de cálculo computacional tiene un pequeño error prácticamente constante a lo largo del eje, por el cual subestima ligeramente la cantidad de combustible gaseoso de la mezcla en el eje longitudinal del cilindro. En cambio, y al igual que pasa con la velocidad, la distribución del combustible gaseoso en dirección radial para los planos normales al eje se ajusta mejor que a lo largo del mismo eje. Para el plano más próximo a la tobera se tiene que el modelo sobrestima



muy ligeramente la presencia de combustible gaseoso mientras que en el plano más alejado la subestima. La Figura 4.54 muestra la distribución espacial para los planos anteriores de la fracción de combustible gaseoso, y se ve cómo apenas hay diferencias entre ambos planos. Estas diferencias, como ya se ha mencionado con anterioridad, pueden deberse tanto a errores de medida, a un ajuste mejorable de algún parámetro de los modelos o la propia forma de realizar el promediado de los resultados LES. La distribución en los dos planos es muy parecida: mantienen los mismos niveles de concentración de combustible, la distribución de combustible es bastante concéntrica y regular alrededor del eje axial; de ahí que apenas oscile su valor a lo largo de la dirección radial.

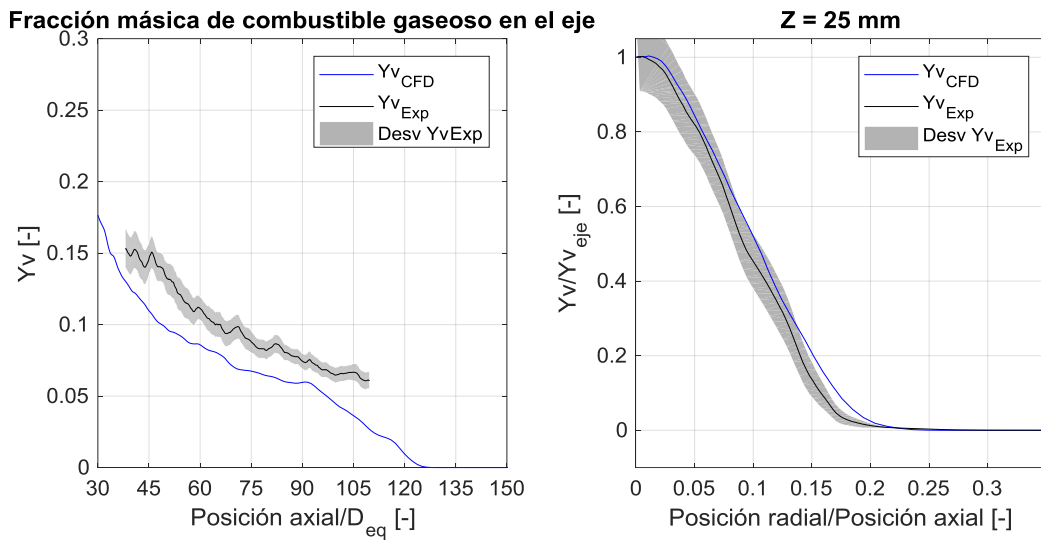


Figura 4.52. Combustible gaseoso. Eje longitudinal y plano normal a 25 mm.

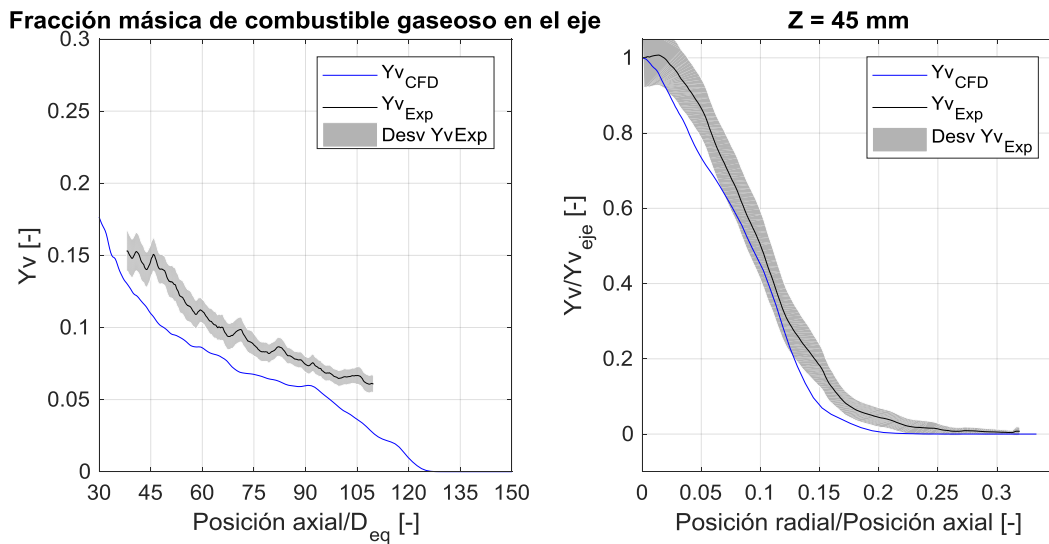


Figura 4.53. Combustible gaseoso. Eje longitudinal y plano normal a 45 mm.

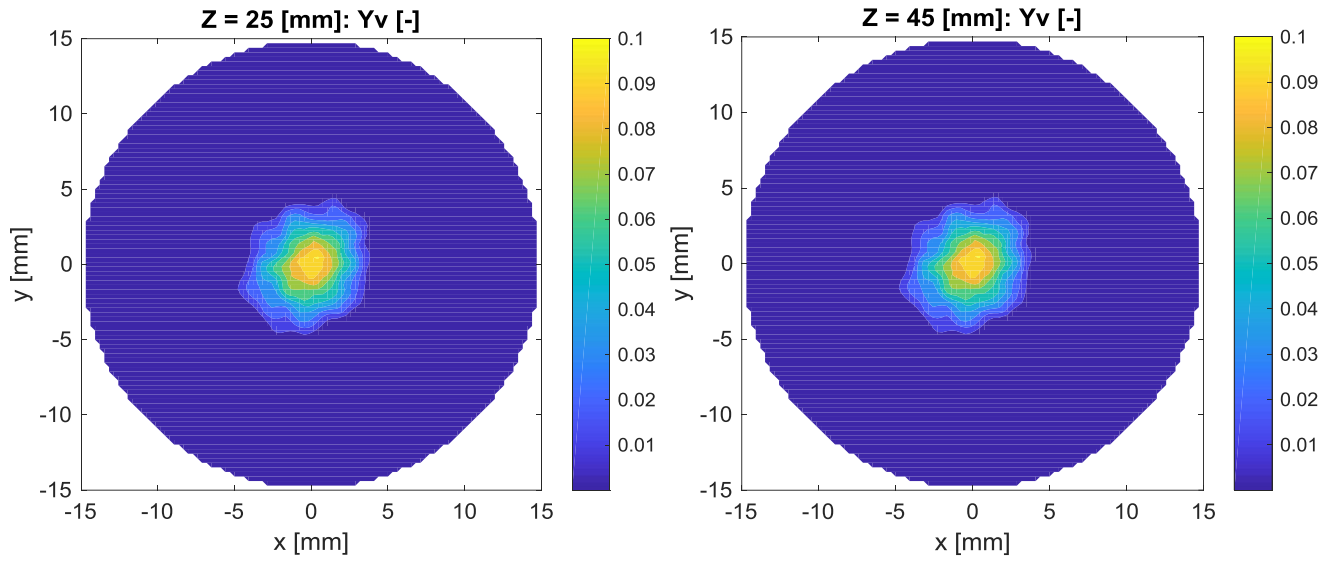


Figura 4.54. Contornos de combustible gaseoso para el modelo LES.

## 5 Conclusiones y futuros trabajos

En esta sección se establecen diferentes conclusiones para los diferentes análisis llevados a cabo en el apartado anterior, por lo que se atenderá a la comparativa inicial entre los tres modelos termodinámicos para el cálculo de la temperatura de la mezcla, la modificación de la ecuación de estado del combustible líquido, el salto del modelo de cálculo RANS al modelo computacional LES y la veracidad de los resultados aportados por el modelo LES frente a los resultados experimentales.

### RANS: Cálculo de la temperatura

De esta comparativa se pueden extraer varias conclusiones:

- Dado que el modelo original de cálculo había sido validado en un proyecto anterior y los dos nuevos modelos (tanto el que controla el cálculo de la temperatura con la entalpía como el que lo hace con la propia temperatura) siguen las mismas tendencias del primero, se puede considerar que estos modelos son también válidos.
- Si bien la profundidad del chorro y la longitud líquida que proporcionan los nuevos modelos son muy parecidos a las del antiguo (incluso reducen muy ligeramente la longitud líquida), estos dos modelos dan solución al problema de continuidad que presentaba el original en la zona cercana a la tobera, debido a altos gradientes, y, lógicamente, también reducen el error.
- Estos modelos también permiten dar cuenta de un punto de inflexión para una composición de la mezcla concreta a partir del cuál cambian las tendencias de varias propiedades del chorro de combustible diésel, como la temperatura. Este valor se establece para una presencia de combustible en la mezcla entorno al 30%.
- Por último, tabular el valor de diferentes variables del proceso de cálculo de la temperatura agiliza de forma importante el cálculo total del caso. Aunque hay que señalar que el modelo de cálculo controlado por la temperatura es más rápido que el modelo controlado por la entalpía: el primer emplea un 30% de tiempo menos que el modelo original, el segundo un 16%.

### RANS: Cambio en la ley de estado del combustible líquido

A partir de cambiar la ecuación de estado del combustible en estado líquido y la forma de calcular determinadas variables según esta ley, aquí se deben destacar dos ideas:

- En primer lugar se adoptó la ecuación de estado para el n-dodecano proporcionada por el NIST en lugar la ecuación de estado del modelo original con el fin de incrementar la longitud líquida y aproximarla más a la que muestran los datos experimentales. Sin embargo, este parámetro apenas permaneció intacto. Por ello, ambas ecuaciones deben ser igual de aptas para el caso de estudio.
- En segundo lugar, la implementación del nuevo proceso de cálculo para las variables dependientes de la ecuación de estado, que hace uso de datos tabulados e interpolaciones, permitió acelerar en gran medida el proceso de cálculo y disminuir el tiempo necesario de ejecución del caso. Al menos en RANS, este incremento de velocidad supuso emplear un 50% menos de tiempo con respecto al modelo modificado sólo para el cálculo de la temperatura y un 70% menos de tiempo con respecto al modelo original. Por tanto, esta forma de cálculo es una buena vía para hacer práctico este tipo de análisis.

### LES: Salto desde el modelo RANS

Aunque las ecuaciones que gobiernan un problema a estudiar son las mismas tanto si se prepara el caso para cálculos RANS como si se prepara para LES, la forma de resolver la turbulencia es la misma: en los casos RANS se modelan todas las escalas de turbulencia mientras que en los casos LES sólo se modelan las escalas cuyo tamaño es menor al de la resolución de la malla. Es por esto que los resultados RANS y LES presentan similitudes, pues tienen puntos en común como el modelo termodinámico del combustible o su ecuación de estado, y al mismo tiempo presentan claras diferencias, debidas a los fenómenos turbulentos.

Por ejemplo, tanto el caso calculado en RANS como el calculado en LES las variables del chorro de combustible siguen las mismas tendencias según la composición de la mezcla. Ahora bien, en el caso ejecutado en LES existe mayor dispersión de valores para una misma composición de la mezcla debido al carácter irregular y estadístico de la turbulencia.

En cuanto a la profundidad del chorro de combustible y la longitud líquida del mismo, sus valores se ven aumentados. El hecho de no modelar gran parte de la turbulencia y resolver las ecuaciones de Navier-Stokes de forma más directa ayuda a que la solución se aproxime más a los resultados experimentales. No obstante, debería realizarse un estudio para llevar a cabo un mejor ajuste de los coeficientes de los modelos utilizados ya que aún la longitud líquida proporcionada por el modelo LES dista un poco del rango de valores ofrecidos por los datos experimentales.

- Por tanto, en comparación con el caso RANS, el ejemplo calculado mediante LES proporciona una solución que se acerca más a la realidad y permite entender mejor el fenómeno estudiado.

#### LES: Comparativa con la muestra experimental

El ejemplo calculado en LES, al igual que el RANS, aporta una solución próxima a la ofrecida por los resultados prácticos. Sin embargo esta solución es más próxima a la realidad, y lo demuestra proporcionando datos casi idénticos a los experimentales, entre ellos la silueta del chorro de combustible. En definitiva, se puede concluir que:

- Por el gran parecido de diferentes parámetros del chorro de combustible como la profundidad, la longitud líquida, el contorno y la evolución de la velocidad o de la presencia de combustible en la dirección del eje axial y en dirección radial, se puede calificar como válido el modelo de formación del chorro de combustible para cálculos LES.
- Además, al ser un modelo de cálculo computacional, permite extraer mayor cantidad de información que de los datos experimentales, pues se pueden evaluar muchas propiedades del fenómeno que de forma experimental no.

#### Trabajos futuros

Según lo visto en este proyecto, los futuros trabajos deberían ser enfocados a implementar en LES el método de cálculo por tabulación de las propiedades del combustible líquido vinculadas a su ecuación de estado, pues como se ha visto para el modelo RANS junto con el método de cálculo de temperatura por tabulación, reducía un 70% el tiempo empleado en ejecutar un caso.

Por otro lado también sería aconsejable revisar ciertas propiedades de los sub-modelos internos del modelo global, de forma que algunos parámetros del chorro de combustible de la solución computacional se acerquen más a los valores de los datos experimentales, como la longitud líquida o la evolución de la fracción másica de combustible gaseoso.



## 6 Bibliografía

La siguiente lista recoge todos los documentos o sitios web que han sido de utilidad para la realización de este proyecto:

- D. Correrías Jiménez, Estudio Teórico-Experimental del Chorro Libre Diesel Isotermo, Universidad Politécnica de Valencia, Valencia, 1998.
- Desantes Fernández, José María/Payri González, Francisco, Motores de combustión interna alternativos, Universidad Politécnica de Valencia, Valencia, 2011.
- Liñan Martínez, A., Mecánica de Fluidos, Escuela Técnica Superior de Ingenieros Aeronáuticos, Madrid, 2005.
- A. Pandal Blanco, Implementation and Development of an Eulerian Spray Model for CFD Simulations and of Diesel Sprays, Universidad Politécnica de Valencia, 2016.
- A. Crespo. Mecánica de Fluidos, Paraninfo SA, 2006.
- Ivor Horton, Beginning C++: the complete language, Birmingham: Wrox Press, 1998.
- “Engine Combustion Network”, [En línea]. Disponible: <http://www.sandia.gov/ecn/cvdata/dsearch/dataDef.php>.
- “Centro de Motores Térmicos”, [En línea]. Disponible: <http://www.cmt.upv.es/ECN03.aspx>.
- I. Gil García, Modelado CFD de la evolución de un chorro Diesel en condiciones de motor, Universidad Politécnica de Valencia, 2016.
- Vallet A. and Borghi R., Modélisation Eulerienne de l’atomisation d’un jet liquide, C.R. Acad. Sci, Paris, 1999.
- Jasak H, Error Analysis and Estimation for the Finite Volume Method with Applications to Fluid Flows, Doctoral Thesis, Imperial College, 1996.
- Guía del usuario de OpenFOAM, [En línea]. Disponible: <https://www.openfoam.com/documentation/user-guide/>.
- Guía el programador de OpenFOAM, [En línea]. Disponible: <https://www.openfoam.com/documentation/>.
- Thomson G, H., Brobst K. E. and Hankinson R. W. “An Improved correlation Compressed Liquids and Liquid Mixtures”. AIChE Journal, Vol. 28 nº 4, 1979.
- Sitio web del “NIST”: <https://webbook.nist.gov/chemistry/>.
- Rowlinson John Shipley, Liquids and liquid mixtures, Butterworths, London, 1969.





Documento 2:  
Pliego de condiciones



## I Objeto del pliego de condiciones

El pliego de condiciones es el documento que marca las directrices y normas para llevar a cabo el proyecto deseado, pues determina los derechos, obligaciones y responsabilidades de las partes participantes del mismo, además del planteamiento del trabajo y del modo en el que ha de resolverse cualquier incidente que pueda aparecer. Por tanto, este documento ha de establecer claramente las condiciones en las que debe sobrellevar el proyecto en materia relacionada con aspectos legales, condiciones generales y detalles técnicos.

Es de obligado cumplimiento, y debe tenerse siempre presente para cualquier decisión. En caso de contradicción entre lo estipulado por la memoria y por el este documento, prevalecerá la memoria si la incompatibilidad se corresponde a un aspecto técnico, mientras que tendrá mayor validez el pliego de condiciones para cualquier otro aspecto.

## II Condiciones generales

En este apartado se determina el trabajo asignado a cada participante del estudio, al igual que las relaciones entre dichas partes, sus derechos y sus obligaciones. Para la evolución de este trabajo se deberían establecer tres partes principales: el alumno o ingeniero principal del proyecto, el tutor y el director del proyecto. Dado que la mayor carga de trabajo cae sobre el alumno o ingeniero principal, este documento se centra en dicho ingeniero, sin dejar de mencionar brevemente las responsabilidades de su tutor y el director del proyecto.

En este caso, el tutor del proyecto es alguien muy familiarizado con el mismo, y está encargado del seguimiento de la evolución del proyecto, ayudando y guiando al alumno a la hora de resolver los problemas de carácter técnico que plantea el proyecto. También puede orientar al alumno en relación con los ensayos que se deberían llevar a cabo y los parámetros a estudiar.

En el caso del director del proyecto, éste sirve de guía para marcar la línea de investigación a seguir y ofrece consulta ante cuestiones técnicas de gran relevancia para el desarrollo del trabajo.

### II. I Condiciones generales facultativas

Estas condiciones hacen referencia al contexto y situaciones del proyecto desarrollado por el ingeniero principal. Define su pauta de trabajo:

- El ingeniero debe seguir una línea de acción definida por el tutor del proyecto, junto con el director. Si se considera que se ha llegado al final de esta línea de acción antes de la finalización del proyecto, deberá consultar con el tutor otras nuevas.
- El ingeniero debe informar semanalmente de la evolución del proyecto al tutor y al director, además de cuándo se tenga resultados relevantes pertinentes a los estudios definidos en la estrategia o la línea de acción decidida. De esta manera, la organización del proyecto y la definición de los objetivos puede abordarse desde una posición más dinámica.
- El ingeniero también debe atender y tratar de resolver pequeñas consultas relacionadas con la línea de investigación del tutor o el director aunque no aparezcan explícitamente en la estrategia definida siempre que mantenga relación con la línea de investigación del proyecto.
- Debe asegurarse, junto con el tutor, de que la información usada en el proyecto es y tiene un origen veraz y fiable.
- Pondrá empeño en alcanzar de forma rigurosa las metas fijadas y extraer tantas conclusiones como sea posible en el estudio de los resultados obtenidos durante la evolución del proyecto.

### III Condiciones particulares

Esta sección explica las condiciones del hardware y el software empleando para la realización del proyecto

#### III. I Condiciones de hardware

Para la realización de este proyecto final de máster se han utilizado varios equipos:

- Un ordenador de sobre-mesa de gran capacidad para la implementación del código, tanto RANS como LES, además de para ejecutar los cálculos RANS y hacer pruebas de compilación de los casos LES. Su sistema operativo es Ubuntu, cuenta con una gran capacidad de almacenamiento en los discos duros y en la memoria RAM (64 GB) y también posee 12 potentes procesadores Intel Xenon de doble núcleo.
- El supercomputador RIGEL de la Universidad Politécnica de Valencia para realizar ciertas pruebas de los casos LES.

- El supercomputador de la Universidad de Málaga, Picasso, para la ejecución de los casos calculados como LES, debido a la gran capacidad y tiempo de cálculo que necesitan, pues para el cálculo completo de un caso se tardaban semanas.
- Un ordenador personal para el tratamiento de los resultados y la redacción del presente documento. Este ordenador cuenta con el sistema operativo Windows 7, cuatro discos duros de 250GB cada uno, una memoria RAM de 8GB y un procesador Intel Core i7.

### III. II Condiciones de software

#### Software para la simulación del chorro de combustible

El modelo desarrollado para la formación de chorros de combustible diésel evaporativo ha sido implementado en OpenFOAM. Éste es un software libre que hace uso del lenguaje C++, un lenguaje orientado a objetos. Su código es libre bajo la licencia GNU, General Public License. La fundación OpenFOAM mantiene y distribuye de forma gratuita este software, y los usuarios son libres de hacer modificaciones en el código que ellos adquieren, de añadir nuevos modelos o rutinas y compartirlos.

La licencia GNU presenta dos características distintivas: todo software que haga uso de código con una licencia GNU hereda este mismo tipo de licencia, y cualquier software con licencia GNU que sea distribuido, debe distribuirse junto con su código fuente.

#### Sistema operativo

OpenFOAM fue escrito para utilizarse con otros sistemas operativos libres, como lo es Linux. Por este motivo, los equipos utilizados para la implementación del código y la ejecución de los casos de cálculo computacional utilizan Ubuntu, un sistema operativo diseñado por Linux. Su código también es abierto y cuenta con licencia GNU.

#### Software de post-procesado de datos

La herramienta usada para el tratamiento de los resultados procedentes de las simulaciones computacionales es Matlab. Es un software de lenguaje de alto nivel dotado de un entorno propicio para el cálculo numérico, la visualización de y tratamiento de resultados y la programación.

Este software, a diferencia de OpenFOAM, sí está protegido por una licencia comercial, por lo que es ilegal su copia y su distribución. Dado que este proyecto no tienen ningún fin lucrativo y se trata de un trabajo de fin de máster, se empleó una licencia de estudiante proporcionada por Matlab para la Universidad Politécnica de Valencia.



# Documento 3: Presupuesto





# I Objeto del presupuesto y consideraciones

Este documento está pensado para hacer visible los gastos que conlleva el desarrollo de este trabajo final de máster. Recoge los costes propios de la amortización de los equipos, de los materiales y de los recursos humanos empleados, de forma que se puedan ver los costes parciales de cada uno y los totales del proyecto.

Dado que este proyecto resulta ser un trabajo de investigación y su objetivo no consiste en la obtención de un producto material, las especificaciones del presupuesto se centran en el trabajo de una sola fase informática: el desarrollo y aplicación del modelo CFD avanzado para chorros de combustible.

## II Desglose del presupuesto

El desarrollo y la mejora un modelo CFD avanzado para chorros de combustible no sólo consiste en la elaboración e implementación del código del modelo para el cálculo computacional, también se debe comprobar la validez del mismo. De ahí que el presupuesto se pueda desglosar no sólo en el coste de mano de obra, el del equipo informático y licencias empleados y de materiales fungibles sino también en los gastos asociados a la simulación de los casos, pues ha sido necesario utilizar supercomputadores externos para la ejecución de las simulaciones LES.

### II. I Coste de la mano de obra

El personal implicado en la elaboración del trabajo y responsable del resultado final el proyecto son, como se ha explicado en el pliego de condiciones, está compuesto por el director y el tutor del proyecto, junto con el alumno o ingeniero principal sobre el que recae la mayor carga de trabajo. El director y el tutor del proyecto son los encargados de marcar las directrices del trabajo, a la vez que el tutor también es el encargado de orientar al ingeniero principal y de su supervisión.

En cuanto a tareas más activas y visibles para el desarrollo del modelo y la obtención de resultados, se tiene que el ingeniero principal ha modificado el código del modelo original, introduciendo cambios y elaborando rutinas externas al código de dicho modelo que facilitaran y agilizaran el proceso de cálculo. También ha ejecutado los casos de estudio RANS y pruebas de los casos LES. Sin embargo, el tutor del proyecto se ha

encargado de adaptar las pruebas de cálculo LES a la arquitectura de los sistemas operativos propios de los supercomputadores para poder ejecutarlas.

El coste de los ingenieros director y tutor del proyecto se ha calculado de acuerdo con las bases de cotización según la categoría profesional para el año 2018, año de la realización del proyecto. Éstas se pueden consultar en el siguiente sitio web: <http://www.seg-social.es/wps/portal/wss/internet/Trabajadores/CotizacionRecaudacionTrabajadores/10721/10957/583/576#576>.

Según esto, el sueldo base mínimo al mes para un ingeniero titulado, como lo son los tres componentes del equipo, es de 1199.10€ y el sueldo base máximo de 3803.70€. Suponiendo que tanto el director como el tutor cobren el 90% del máximo estipulado, éstos tendrían un sueldo bruto al mes de 3422.70€. Partiendo de este dato, el cálculo del coste de las horas de trabajo de los ingenieros debe tener en cuenta contribuciones a la seguridad social, al desempleo, a formación profesional y a un fondo de garantía salarial. La siguiente tabla recoge el coste total de un ingeniero para la empresa:

Salario bruto mensual	3422.70 €
Salario anual (14 pagas)	47917.80 €
Desempleo (5.5%)	2635.48 €
Fondo de garantía salarial (0.2%)	95.84 €
Formación profesional (0.6%)	287.51 €
Seguridad social (23.6%)	11308.60 €
Coste anual para la empresa	62245.23 €

*Tabla II. 1. Coste anual de un ingeniero para la empresa.*

Si se tiene en cuenta que el pago total se ha de dividir en 12 meses de 22 días laborables y jornada laboral completa de 8 horas cada día, el coste del director y el tutor sale a razón de 29.47€/h. Para el ingeniero principal del proyecto, se supone un salario igual al 50% del salario del tutor y el director, por lo que su coste sería de 14.74€/h.

La duración del proyecto ha sido de 4 meses y medio. Durante este periodo, el ingeniero principal ha estado trabajando casi por completo en el trabajo final de máster, mientras que el tutor y director han dedicado un número de horas menor al proyecto. Esta tabla recoge las horas dedicadas por cada uno y el coste asociado:

Concepto	Horas de trabajo	Coste por hora	Coste
Ingeniero principal	700	14.74 €/h	10318 €
Tutor del proyecto	100	29.47 €/h	2947 €
Director del proyecto	10	29.47 €/h	294.70 €
Coste global			13559.70 €

*Tabla II. 2. Coste global de los ingenieros implicados en el proyecto.*

## II. II Amortización del equipo informático y licencias

Se debe incluir este apartado por dos razones: el coste del equipo informático empleado para la realización del trabajo y la obsolescencia a la que están sometidos, pues con el paso del tiempo las prestaciones de los equipos decrecen y en el mercado aparecen nuevas herramientas con mejores características.

En este aspecto se deben tener en cuenta principalmente el ordenador de sobremesa y el ordenador personal descritos en el pliego de condiciones. Así mismo, se deben incluir las licencias del software utilizado, OpenFOAM y Matlab. No obstante, OpenFOAM es un software de código libre y no conlleva gasto para el usuario, por lo que sólo se tendrá en cuenta la licencia de Matlab:

Concepto	Coste total	Vida útil	Periodo amortizado	Coste de amortización
Ordenador de sobre-mesa	3000€	3 años	4.5 meses	375 €
Ordenador personal	1300€	6 años	4.5 meses	187.50 €
Licencia de Matlab	2000€	1 año	4.5 meses	750 €
Coste global				1312.5 €

*Tabla II. 3. Coste por amortización de los equipos y las licencias.*

## II. III Costes asociados a la simulación de los casos

El coste asociado a la simulación de los casos de cálculo no es más que el precio de usar las estaciones de cálculo RIGEL y PICASSO:

Estación de cálculo	Horas de uso	Coste por hora de procesador	Coste
RIGEL	7000	0.01 €/cpu h	70.00 €
PICASSO	200000	0.02 €/cpu h	4000.00 €
Coste global			4070.00 €

*Tabla II. 4. Coste asociado a la simulación de los casos de estudio.*

## II. IV Costes de materiales fungibles

Material fungible es aquél consumido durante la realización del proyecto y que no presenta costes de amortización. Ejemplo de este tipo de materiales son folios, bolígrafos, el CD a entregar para la defensa del trabajo, etc. La tabla de debajo recoge brevemente los gastos asociados a estos materiales.

Concepto	Unidades	Coste unitario	Coste total
Material de oficina	-	-	140 €
Material de trabajo	-	-	30 €
Presentación del proyecto	1	50 €/ud	50 €
Coste global			220 €

Tabla II. 5. Coste de materiales fungibles.

### III Resumen del presupuesto

Presupuesto Total del Proyecto	
Concepto	Importe
Coste de la mano de obra	13559.70 €
Coste de la amortización de los equipos y las licencias	1312.50 €
Coste de la simulación de los casos	4070.00 €
Coste de materiales fungibles	220 €
Subtotal	19162.20 €
Costes Indirectos (5% del subtotal)	958.11 €
Beneficio Empresarial (6% del subtotal y costes indirectos)	1207.22 €
IVA 21% (salvo amortización y costes indirectos)	4001.95 €
Total	25329.48 €

Tabla II. 6. Resumen del presupuesto

En definitiva, el precio total de este proyecto alcanza la cifra de #VEINTICINCO MIL TRESCIENTOS VEINTINUEVE EUROS CON CUARENTA Y DOS CÉNTIMOS DE EURO#.

# Documento 4:

## Anexos



# I Tutorial de OpenFOAM

Esta sección no es más que un tutorial para que futuros usuarios del modelo puedan configurar y lanzar el caso de estudio de formación de chorros de combustible que precisen. Las siguientes dos secciones de este documento de anexos se corresponde con los programas necesarios para, por un lado preparar los datos necesarios requeridos por los nuevos modelos y sistemas de cálculo introducidos, y por otro, llevar a cabo el tratamiento o post-procesado de los resultados proporcionados por el modelo y así ser capaces de realizar un análisis de los mismos.

Más concretamente, en este tutorial se configura el caso ejecutado como RANS para el análisis realizado en este trabajo: el caso de un chorro de combustible diésel inerte y evaporativo. Los archivos y datos requeridos para el lanzamiento del caso, a parte del mismo ejecutable que queda listo tras su compilación, están recogidos en una sola carpeta, la carpeta inicial o principal del caso de ejecución. Esta carpeta se divide a su vez en otras tres: la carpeta “0” contiene las condiciones iniciales del volumen de control, la carpeta “constant” contiene la información necesaria para la construcción de la malla y la configuración de diferentes modelos como el de turbulencia, y la carpeta “system” posee archivos necesarios para la configuración del cálculo del caso. Además, esta carpeta general también presenta tres archivos relacionados con las condiciones de contorno del caso.

## I. I Condiciones de contorno e iniciales

Las condiciones de contorno del caso configurado están controladas por tres archivos incluidos en la carpeta inicial o principal del caso. Estos archivos se denominan “tiempo-velocidad-instantanea\_SprayA”, “k\_SprayA\_coSigma” y “eps\_SprayA\_coSigma”; y representan las condiciones de contorno de velocidad, energía cinética turbulenta y disipación turbulenta respectivamente.

Por otro lado, las condiciones iniciales para el volumen de control están recogidas en la carpeta “0”, dentro de la carpeta principal del caso. Dentro la carpeta de condiciones iniciales encontramos archivos con diferentes variables: dispersión térmica (archivo “alphan”), disipación turbulenta (archivo “eps”), energía cinética turbulenta (archivo “k”), viscosidad (archivo “mut”), presión (archivo “p”), densidad de superficie entre fases (archivo “Sigma”), temperatura (archivo “T”), velocidad (archivo “U”), fracción másica de combustible líquido (archivo “Y”), fracción másica de combustible gaseoso (archivo “Yv”) y la desviación de la fracción másica de combustible gaseoso a lo largo del tiempo (archivo “Zv”). Los ficheros más interesantes son los correspondientes a la presión, temperatura, velocidad, energía cinética turbulenta y disipación turbulenta.

Las variables más importantes son la presión y la temperatura, ya que establecen en el interior del cilindro las condiciones de presión, temperatura y, consecuentemente, densidad a partir de las dos anterior. En sus correspondientes ficheros se introduce el valor de la presión y temperatura del gas ambiental del interior del cilindro.

Las variables de velocidad, energía cinética turbulenta y disipación turbulenta quedan determinadas por los tres archivos exteriores a la carpeta “0”, por los archivos correspondientes a estas variables del interior de la carpeta “0” deben llamar a los ficheros externos: “k\_SprayA\_coSigma”, “eps\_SprayA\_coSigma” y “tiempo-velocidad-instantanea\_SprayA” para obtener el valor de estas variables.

## I. II Malla y configuración de los modelos

La configuración de la malla y de varios modelos importantes para la evolución queda determinada por varios archivos que se encuentran en el interior de la carpeta “constant”. Así mismo, esta carpeta también contiene los datos tabulados de los que hacen uso tanto el nuevo modelo termodinámico para el cálculo de la temperatura a partir de la entalpía como la nueva ecuación de estado del combustible líquido. Esta carpeta antes contaba con archivos necesarios para las tablas JANAF, ahora eliminadas tras las modificaciones aplicadas.

### Configuración de la malla

Los archivos necesarios para la configuración y generación de la malla están en el interior de la carpeta “polyMesh”. El fichero “blockMeshDict” es aquél que contiene la información necesaria para generarla así como el archivo que lee el ejecutable encargado de construirla.

La Figura I.1 es un ejemplo de configuración de la malla. En la primera línea aparece el comando “convertToMeters” seguido de “0.001” para establecer como las unidades de la malla los milímetros. La extensión de la malla se define estableciendo vértices que definen los bloques de la malla: en este caso aparecen nueve vértices y dos bloques. Cada bloque queda definido por un conjunto de vértices, pero en su definición también debe aparecer el número de celdas que ha de tener en cada dirección y la progresión de crecimiento en dichas direcciones.

Así, el bloque número 1 queda definido por los vértices del 0 al 5, cuenta con 550 celdas dirección axial y 5 en dirección radial (se trata de una cuña). El último vector que aparece para este bloque, y escrito entre paréntesis, se corresponde con la variación del tamaño de las celdas: en la dirección axial crecerán mientras que en la dirección radial el tamaño de estas celdas no variará.



```

/*-----*/
|=====|
| \ \ \ / | F i e l d . | OpenFOAM: The Open Source CFD Toolbox
| \ \ \ / | O p e r a t i o n . | Version: 1.6
| \ \ \ / | A n d . | Web: http://www.openfoam.org
| \ \ \ / | M a n i p u l a t i o n . |
/*-----*/
FoamFile
{
    version      2.0;
    format       ascii;
    class        dictionary;
    object       blockMeshDict;
}
// ***** //

convertToMeters 0.001;

//La longitud axial de la malla ahora son 108 mm y radial 25 mm
//y el patch fuel (que hace las veces de inyector) será de 0.0447 mm de radio

vertices
(
    (0 0 0) //vert=0
    (108 0 0) //vert=1
    (0 0.044529903 -0.003895862) //vert=2
    (108 0.044529903 -0.003895862) //vert=3
    (108 0.044529903 0.003895862) //vert=4
    (0 0.044529903 0.003895862) //vert=5
    (0 24.904867452 -2.178893569) //vert=6
    (108 24.904867452 -2.178893569) //vert=7
    (108 24.904867452 2.178893569) //vert=8
    (0 24.904867452 2.178893569) //vert=9
);

blocks
(
    hex (2 3 4 5 0 1 1 0) (550 1 5) simpleGrading (102.4185404 1 1)
    hex (5 4 3 2 9 8 7 6) (550 1 100) simpleGrading (102.4185404 1 136.1162146) //6 7 8 9 2 3 4 5
)

```

Figura I.1. Ejemplo 1 de archivo “blockMeshDict”.

Tras definir las dimensiones, límites y bloques de la malla, queda definir los tipos de pared de la propia malla, para así poder establecer las condiciones de contorno apropiadas a cada una, como muestra la Figura I.2. Cada zona se especifica con los vértices que la conforman y se le proporciona un nombre distintivo. En las zonas donde la malla es abierta, es decir, dónde no hay pared (zonas de contacto entre bloques, por ejemplo), se han de modificar las condiciones de contorno de temperatura y de presión del contorno (archivos de la carpeta “0”). En este caso, la presión debería definirse como gradiente nulo (“zeroGradient”) y debería establecerse un valor constante y equivalente al de la presión en el interior de la cámara. En lo referente a la velocidad, también debe plantearse como gradiente nulo para este tipo de contorno.

Como es lógico, si se introduce alguna modificación en el archivo que origina la malla, debe aplicarse de nuevo el ejecutable “BlockMesh” para que los cambios introducidos se lleven a cambio y se pueda lanzar el caso con la nueva malla rectificada.

```

edges
(
);
patches
(
  patch fuel
  (
    (2 0 0 5)
  )
  wall wall1
  (
    (6 2 5 9)
  )
  wall wall2
  (
    (3 1 1 4)
    (7 3 4 8)
    (6 7 8 9)
  )
  wedge side1
  (
    (2 3 1 0)
    (6 7 3 2)
  )
  wedge side2
  (
    (5 4 1 0)
    (9 8 4 5)
  )
  empty axis
  (
    (0 1 1 0)
  )
);
mergePatchPairs
(
);

```

Figura I.2. Ejemplo 2 de archivo “blockMeshDict”.

### Configuración de los modelos

Las propiedades del modelo termo-físico del combustible, el modelo de transporte y el modelo de turbulencia quedan recogidas en los archivos “thermophysicalProperties”, “transportProperties” y “turbulenceProperties” respectivamente. Y las propiedades relacionadas con el modelo de turbulencia High Density Ratio K-Epsilon quedan completas con el fichero “RASProperties”, que establece varias constantes.

En lo referente al modelo  $\Sigma - Y$ , sus parámetros y coeficientes quedan definidos en el archivo “sigmayConstants”. Entre los parámetros controlados por este archivo se hallan las temperaturas del gas del interior del cilindro y del combustible, las especies de la mezcla o los coeficientes modificables de la ecuación  $\Sigma$  (C1 y C2), además de compresibilidad, tensión superficial, temperatura y presión crítica, etc. Este archivo también contiene la temperatura del combustible y del gas del cilindro, por lo que si se cambian en las condiciones de contorno, aquí también debe realizarse dicho cambio.

El modelo termodinámico para el cálculo de la temperatura de la mezcla a partir de la entalpía de la misma hace uso de datos tabulados según la propia temperatura, obteniendo así la descomposición de la entalpía total de la mezcla según sus componentes: gas ambiental, combustibles en estado líquido y combustibles en estado gaseoso; como dicta la Ecuación 3.61. De esta manera puede compararse ambos valores de entalpía para la mezcla y determinar si continuar o no con el proceso iterativo para el cálculo de la temperatura. Esta tabla proporciona la entalpía de cada uno de los componentes de la mezcla en función de la temperatura a la que se encuentran, y el archivo que la contiene se denomina “Table\_h\_cp”. Más adelante se mostrará el código del programa que genera dicha tabla.

Por su parte, el nuevo modelo para la ecuación de estado del combustible líquido también se sirve de otras tres tablas para calcular la densidad y sus derivadas en función de la presión y la temperatura. Los archivos que contienen estas tres tablas son “m\_rho”, “m\_drhodP” y “m\_drhodT”; y contienen las tablas de la densidad, la derivada de la densidad respecto de la presión y la derivada de la densidad respecto de la temperatura, respectivamente.

### I. III Configuración del cálculo

La carpeta “system” contiene ficheros que permiten definir ciertos parámetros determinantes en la ejecución del proceso de cálculo, como el salto temporal mínimo o máximo entre cada iteración. Los archivos que controlan estos parámetros son: “controlDict”, “decomposeParDict”, “fvSchemes”, “fvSolution” y “sampleDict”.

El fichero “controlDict” establece el valor de determinados factores de control de cálculo, como son el tiempo de inicio de cálculo (si empezar de nuevo o partir de un punto de control ya guardado), la duración del cálculo, el paso temporal de cada iteración o el paso temporal que determina cada cuánto guardar los resultados obtenidos. También permite ajustar el número de Courant, lo que afecta al paso temporal para cada iteración, permitiendo acelerar o ralentizar el proceso de cálculo según las necesidades. Aumentar el número de Courant máximo supone incrementar también el paso temporal y acelerar el proceso de cálculo a costa de favorecer la aparición de inestabilidades en el cálculo, mientras que la reducción de este número acarrea un paso temporal más bajo y mayor coste temporal, pero permite hacer el proceso de cálculo más estable cuando la convergencia es difícil de alcanzar.

El fichero “decomposeParDict” permite descomponer el proceso de cálculo en tantos procesadores como se quiera y como el ordenador permita. Esta descomposición se puede realizar mediante tres métodos, lo más común es elegir el método simple. Con este método, es el mismo programa OpenFOAM el que distribuye la malla y el dominio del problema en los diferentes procesadores.

Los ficheros “fvSchemes” y “fvSolution” contienen información relacionada con la forma de resolver los sistemas de ecuaciones y las variables que contienen. Por ejemplo, el archivo “fvSchemes” se usa para determinar el método numérico (Gauss, Euler...) por el cuál resolver cada ecuación o sistema de ecuaciones definido en el modelo, mientras que el fichero “fvSolution” es utilizado para controlar parámetros de la resolución de estos sistemas, como la tolerancia de la solución o el número de iteraciones máximos para los métodos numéricos aplicados en su resolución.

Para finalizar, aparece el fichero “sampleDict”. Este archivo es de gran importancia para el tratamiento de la solución propuesta por el modelo, pues en este fichero se definen las variables que se quieren estudiar así como los ejes, los planos o el volumen dónde se quieren estudiar dichas variables.

## I. IV Ejecución del caso

Una vez se tiene listo el “set-up” o la configuración del caso, es momento de ejecutar el caso que se quiere estudiar. Para ello, se debe inicializar OpenFOAM en la máquina que se vaya a ejecutar el modelo y crear la malla a través del comando “blockMesh”. Si ya se tiene generada la malla y se está relanzando el caso, no es necesario volver a generarla. Si se quiere descomponer el lanzamiento del caso en varios procesadores, previamente se debe haber configurado el fichero “decomposeParDict” para ello, y antes de llamar al ejecutable del modelo, se debe introducir el comando “decomposePar”. Es posible que lleve un poco de tiempo la ejecución de este comando, dependerá del tamaño del dominio y del número de procesadores a usar.

Ahora sí, es el turno del lanzamiento del caso. Este puede hacerse de dos maneras posibles: la primera, lanzando todo el caso al mismo procesador, y la segunda, dividiendo su ejecución en varios procesadores. Para ejecutar el caso en un solo procesador basta con introducir el comando:

$$\text{Nombre\_del\_solver} > \text{log} \ \&$$

Al introducir “> log &” se lleva la ejecución del caso a un segundo plano, con escribir el nombre del ejecutable es suficiente para hacer funcionar el modelo. Si se pretende lanzar el caso en varios procesadores, el comando que se debe utilizar es el siguiente:

$$\text{mpirun} \ -np \ n \ \text{Nombre\_del\_solver} \ -parallel > \text{log} \ 2 > \ \&$$

Para poder ver la evolución del caso, si éste se estuviera ejecutando en segundo plano, se puede emplear el comando “tail -f log”. El archivo que se genera al lanzar el caso y que registra su evolución se denomina “log”, y el comando que permite visualizar el archivo es “tail”, pues muestra el final del archivo. La opción “-f” muestra las diez últimas líneas del fichero, y se va actualizando conforme se va sobre-escribiendo el archivo log.

Si en lugar de vez las diez últimas líneas se quisieran ver las cincuenta últimas, sería suficiente escribir “-50” en lugar de “-f”. Otra opción para monitorizar el caso en ejecución es leer el fichero “mysummary”. También es posible escribir nuevos ficheros para monitorizar cualquier variable que se quiera o incluso añadirlas a los ficheros ya existentes.

En caso de tener que cortar la ejecución del caso, debe aplicarse el comando “top”, identificar los procesos correspondientes al caso en ejecución y usar un segundo comando “kill” junto con el identificador del proceso. De esta manera, se interrumpe el cálculo.

## I. V Post-procesado del caso

El post-procesado del caso se realiza una vez ha finalizado la ejecución del mismo. Antes de comenzar con el tratamiento de los resultados, si el cálculo del problema a estudiar se ha dividido en varios procesadores, estos datos se deben reconstruir mediante el comando “reconstructPar”. Al dividir el cálculo en varios núcleos, aparece una carpeta por cada uno de estos procesadores, y guardan las variables deseadas del dominio del problema que maneja cada procesador. Por tanto, al ejecutar el comando “reconstructPar”, se unifican los ficheros de estas variables para todo el dominio y aparece una carpeta por cada instante de tiempo en el que han sido guardadas dichas variables.

Así mismo, un comando importante para el tratamiento de los resultados, como se ha mencionado anteriormente, es el fichero “sampleDict”, que permite obtener el valor de determinadas variables para unos sectores concretos del dominio (ejes, planos...). Para configurar el fichero “sampleDict” se ha de establecer el tipo de interpolación y el formato en el que se desean los archivos de salida con las variables.

Otro aspecto importante dentro de este archivo es la elección de los sectores dónde se quiere saber el valor de las variables de estudio, como un eje o un plano. Como se aprecia en la Figura I.3, “axis” y “surfaces” son las referencias para distinguir entre ejes o planos, y éstos han de ser nombrados y definidos como se muestra en dicha imagen.

En último lugar, dentro de este fichero, se deben definir las variables que se pretenden estudiar mediante su correspondiente nombre. Así, una vez se ejecute el comando “sample”, se crearan varios archivos distinguiendo entre los sectores del volumen de control y las variables escogidos para el estudio. Al usar solamente el comando “simple”, aparecerán archivos con las variables por cada muestra de tiempo en la que se han guardado datos durante el caso. Ahora bien, se pueden obtener los datos para un instante de tiempo constante añadiendo “-times” seguido del instante o instantes deseados a la orden “sample”.

```

interpolationScheme cellPoint;
//interpolationScheme cell;

setFormat      raw;
surfaceFormat  raw;
sets
(
  axis
  {
    type      uniform;
    axis      x;
    start     ( 0.0 1e-8 0.00 );
    end       ( 0.108 1e-8 0.00 );
    nPoints   300;
  }

  rad-0.001069
  {
    type      uniform;
    axis      y;
    start     (0.001069 0.0 0.0 );
    end       (0.001069 0.015 0.0);
    nPoints   300;
  }

  rad-0.002139
  {
    type      uniform;
    axis      y;
    start     (0.002139 0.0 0.0 );
    end       (0.002139 0.015 0.0);
    nPoints   300;
  }

surfaces (
  axisymm2Dplane000
  {
    interpolate      true;
    type plane;
    basePoint (0 0 0);
    normalVector (0 0 1);
  }

);

```

Figura I.3. Ejemplo del fichero “sampleDict”.

Además de lo anterior, también se generan dos ficheros de texto: “log” y “mysummary”, que registran información sobre el cálculo y son útiles para monitorizar su evolución, en especial el fichero “log”. Por su parte, el fichero “mysummary” resulta de gran utilidad en el tratamiento de los resultados, pues permite calcular parámetros geométricos del chorro como la penetración o la longitud líquida.

## II Programas para agilizar el proceso de cálculo

Cómo ya se ha explicado en las secciones 3.3.3 y 3.3.4 del capítulo 3 de la memoria, en el modelo original se realizaron dos cambios significativos con el fin de acelerar el proceso de cálculo del modelo y conseguir una solución más fiel a la realidad. Estos cambios se encuentran en el modelo termodinámico del combustible, dónde se calcula la temperatura de la mezcla a partir de su entalpía mediante un proceso iterativo, y en la ecuación de estado para el combustible líquido, utilizada para determinar la densidad del combustible líquido (y también de la mezcla total) a partir de la presión y la temperatura a la que se halla y también las derivadas de la densidad respecto de estas dos variables.

En ambos modelos, los cambios introducidos en los modelos implicaban un cambio en el método de cálculo de las variables que se veían afectadas, y estos cambios tienen un denominador común: la tabulación de valores. Tanto en el cálculo de la temperatura de la mezcla como en el cálculo de la densidad del combustible líquido se recurre a la lectura e interpolación de valores en tablas. Estas tablas han sido construidas con ayuda de otro software, MATLAB, y tienen un código fuera del implementado para OpenFOAM. En este apartado del documento de anexos se explica el código que permite generar estas tablas, importantes en la reducción del tiempo de cálculo del modelo.

### II. I Modelo termodinámico para calcular la temperatura

Para generar la tabla utilizada en esta parte del código se necesitan dos “scripts”. La tabla que se genera contiene el valor de entalpía y su derivada respecto de la temperatura (equivalente al calor específico a presión constante) de las especies del gas en el interior del cilindro y del combustible utilizado (distinguiendo en este último caso si se encuentra en estado líquido o gaseoso) en función de la temperatura a la que se encuentren. De esta manera, el código de OpenFOAM puede leer estos valores según la temperatura, calcular la entalpía total de la mezcla y reajustar la temperatura de la mezcla.

Aunque los dos programas utilizados en MATLAB para generar esta tabla son considerados dos funciones, una de ellas es la principal y otra la auxiliar. La función o programa principal, *therm\_par\_table.m*, identifica las especies del gas ambiental y el combustible empleados en el caso de estudio y lee los ficheros continentes de los coeficientes JANAF (necesarios para el cálculo de la entalpía de acuerdo con los polinomio de la NASA tanto para las especies del gas ambiental como para el combustible) y de las propiedades exclusivas de los combustibles. También prepara los datos el cálculo de los pesos moleculares de dichas especies y del combustible (pues el cálculo de estos pesos moleculares lo realiza la función auxiliar), calcula los valores de las entalpías y su derivada respecto a la temperatura y escribe las tablas requeridas para el caso de estudio. Así pues,

aquí debajo aparece el código de la función principal, con comentarios que van explicando qué se hace en cada parte:

```
function [ Table_h, Table_dh ] = therm_par_table( especies_fuel, ...
    ruta_THERM, ruta_props )
% Se tienen 3 entradas:
%
% *Entrada "especies_fuel". Consiste en un listado de especies halladas en
% el aire y de combustibles utilizados. Se puede formular de 2 maneras:
%
% -Opción 1: Un fichero txt con la siguiente estructura:
% Fila 1: número de especies en el aire, n
% De la fila 2 a la n+1: lista de los nombres de las n especies
% De la fila n+2 hasta el final: lista de los nombre de los combustible
%
% -Opción 2: un vector tipo celda del estilo:
% {n, 'especie 1', ... , 'especie n', 'fuel 1', ... , 'fuel m'}
%
% *Entrada "ruta_THERM": es la dirección del archivo "THERM.DAT"
%
% *Entrada "ruta_props": es la dirección del archivo
% "props_DICOM_DB_LV.xlsx". Este fichero contiene propiedades de los
% combustibles a utilizar, como la temperatura crítica o el factor
% acéntrico.
%
% VÉASE que los nombres del fichero txt coincidan con los del archivo
% THERM.DAT, pues de aquí se leen todos los coeficientes requeridos para
% generar la tabla que proporciona lo siguientes parámetros:
%
% Temperatura
% Entalpía de las especies en estado gaseoso, h
% Derivada de la entalpía de las especies respecto a T, dh
% Entalpía del combustible en estado gaseoso (vapor), hvf
% Entalpía del combustible en estado líquido, hlf
% Derivada de la entalpía del combustible en estado gaseoso respecto T, dhv
% Derivada de la entalpía del combustible en estado líquido respecto T, dhl
%

flog = fopen('Summary.txt','w');
time_style = 'dd/mm/yyyy HH:MM:SS.FFF';
fprintf(flog, '%s Inicio de la función\n', datestr(now, time_style));

%% Datos de entrada
% Identificación de especies y combustibles
if iscell(especies_fuel)
    n = cell2mat(especies_fuel(1)); % Lectura del número de especies
    especies = especies_fuel(2:n+1); % Lectura del nombre de las especies
    combustibles = especies_fuel(n+2:end); % Lectura del combustible
    m = length(combustibles); % Número de combustibles

    fprintf(flog, '%s Lectura de los datos de ', datestr(now, time_style));
    fprintf(flog, 'entrada como vector de celdas \n');
else
    fid = fopen(especies_fuel, 'r'); % Lectura del archivo txt, del número
    % de especies, de especies y del combustible

    if fid == -1
        fprintf(flog, '%s Datos de entrada no válidos\n', datestr(now, ...
            time_style));
        fclose(flog);
        return
    end
end
```



```

end

linea = fgetl(fid);
n = str2num(linea);
linea = fgetl(fid);
i=1;
while linea ~= -1
    especies(i) = cellstr(linea);
    i=i+1;
    linea = fgetl(fid);
end
combustibles = especies(n+1:end);
especies = especies(1:n);
m = length(combustibles);    % Número de combustibles
fid = fclose(fid);

fprintf(flog, '%s Lectura de los datos de ', datestr(now, time_style));
fprintf(flog, 'entrada desde el archivo %s\n', especies_fuel);
end

% Lectura de propiedad de los combustibles (Tc y omega)
[num, txt, raw]=xlsread(ruta_props);
% Descompone la lectura del archivo en una matriz numérica, otra de texto y
% una conjunta. La matriz de texto tiene 2 filas más que la numérica.
list_fuel = txt(2:end-1,1);    % Lista de combustibles
list_omegaf = num(:,5);    % Lista de omegaf
list_Tc = num(:,6);    % Lista de omegaf
lf = length(list_Tc);    % Longitud de las listas
clear num txt raw;

Tc_comb = [];    % Inicilizamos vectores
omega_comb = [];

for i = 1 : m
    word1 = cell2mat(combustibles(i));
    for j = 1 : lf
        word2 = cell2mat(list_fuel(j));
        if length(word1) == length(word2) & word1 == word2
            Tc_comb(i) = list_Tc(j);    % Temperatura crítica [K]
            w_comb(i) = list_omegaf(j);    % Omega f [-]
        end
    end
end
end

if prod(Tc_comb) == 0
    fprintf(flog, '%s No coinciden el nombre de', datestr(now, time_style));
    fprintf(flog, ' los combustibles del archivo %s con', especies_fuel);
    fprintf(flog, ' los del archivo %s\n', ruta_props);
    return;
end
end

```

Hasta aquí llega la identificación de las especies y el combustible además de la lectura de las propiedades exclusivas de los combustibles (temperatura crítica y factor acéntrico). Ahora, se hará un paréntesis para explicar la función auxiliar, *calcPM.m*, pues se utiliza para calcular el peso molecular de las especies y los combustibles aparece en la siguiente parte del código de la función principal. La extensión de esta función auxiliar es mucho más pequeña que la de la principal:

```

function [ PM ] = calcPM( list_esp, list_num )
% Calcula el peso molecular de la especie seleccionada. Debe introducirse
% mediante dos entradas.
%
% -list_esp es un vector tipo "cell", donde cada "cell" contiene un
% elemento de los que componen una molécula del compuesto.
% Ejemplo: {'C','H','Br'}      CH3CBr3
%
% -list_num es un vector numérico que contiene (según el orden de aparición
% de los elementos en el vector "list_esp") el número de átomos de cada
% elemento del compuesto.
% Ejemplo: [2, 3, 3,]        CH3CBr3

%% Definición de pesos atómicos en [g/mol]

% Elementos, id añadiendo conforme se quiera
elementos = {'H','C','N','O','AL','Br','Cl','F'};

% Pesos atómicos [g/mol]
PMele = [1.0079, 12.011, 14.00674, 15.9994, 26.9815, 79.904, 35.453,...
         18.9984];

%% Cálculo PM
n = length(list_esp);
m = length(PMele);
PM=0;

for j=1:n
    esp=list_esp{j};
    lesp=length(esp); % Longitud del elemento

    for i=1:m
        if lesp == length(elementos{i}) & esp == elementos{i}

            PM = PM+list_num(j)*PMele(i);
            break;

        end
    end
end
end

```

Cómo se observa, el código de esta función es corto y nada complejo. Pero existe una pega, deben incluirse manualmente en el código el nombre y el peso atómico de los elementos que componen las especies del gas ambiental y del combustible, ya que éste código fue escrito y adaptado para el caso que se ha analizado en este proyecto.

En la siguiente parte del código de la función principal se leen los coeficientes JANAF, que aparecen en el fichero *THERM.DAT*, de las especies y de los combustibles y también se calcula el peso molecular de los mismos, con ayuda de la función auxiliar *calcPM.m*:

```

%% Cálculo de PM y extracción de coeficientes
% Lectura del archivo con los coeficientes de todas las especies
format longE;
fid = fopen('THERM.DAT');
linea = fgetl(fid);
nosp = find(linea == ' ',1);
fsp = '%s%d%s%d%s%d%s%d';

CFc = zeros(m,15); % Inicializamos vector de coeficientes. Combustible
CFe = zeros(n,15); % Inicializamos matriz de coeficientes. Especies
PM_esp = []; % Inicializamos vector pesos moleculares. Especies
PM_comb = []; % Inicializamos vector pesos moleculares. Combustibles

% Bucle de extracción de coeficientes. Si la línea de lectura comienza con
% alguna de las especies, calculamos los pesos moleculares y extraeremos
% los coeficientes.

while isempty(linea)==1 | linea ~= -1

    bandera = 0;
    % Especies
    for i = 1:n
        busca = cell2mat(especies(i)); % Nombre de especie
        lbusca = length(busca); % Longitud nombre especie
        if length(linea)<2
            break;
        elseif busca == linea(1:lbusca) & (nosp-1)==lbusca & ...
            sum(CFe(i,:)) == 0

            % Peso Molecular
            lPM = textscan(linea(25:44),fsp); % Elementos en el compuesto
            clear list_elem list_num;

% Máximo de 4 elementos por compuesto según THERM.DAT
% Aquí, de acuerdo el formato especificado para "textscan" comprobamos que
% las posiciones pares del vector que devuelve no estén vacías y no sean 0.
% En ese caso, quiere decir que tenemos cierto número de átomos (posiciones
% pares del vector) de un mismo elemento (posiciones impares)
            for k = 2:2:8
                if isempty(lPM{k})==0 & lPM{k}~=0
                    list_elem(k/2) = lPM{k-1}; % Forma de cell
                    list_num(k/2) = double(lPM{k}); % Forma numérica
                end
            end
            PM_esp(i) = calcPM(list_elem,list_num); % Cálculo PM

% Si no calcula el PM, se registra y se para la función.
            if PM_esp(i) == 0;
                fprintf(flog,'%s Peso molecular d',datestr(now,time_style));
                fprintf(flog,'e %s nulo. Revisar elementos calcPM.m\n',busca);
                fclose(flog);
                return;
            end

% Si calcula el PM, también se registra.
            fprintf(flog,'%s Peso molecular de ', datestr(now,time_style));
            fprintf(flog,'%s calculado\n',busca);

% Coeficientes
            c = 1:5;

```

```

for j = 1:3
    linea = fgetl(fid);
    CFe(i,c) = [str2double(linea(1:15))...
               str2double(linea(16:30)) str2double(linea(31:45))...
               str2double(linea(46:60)) str2double(linea(61:75))];
    c = c+5;
end

% Queda registrada la entra de coeficientes.
fprintf(flog, '%s Coeficientes de ', datestr(now,time_style));
fprintf(flog, '%s leídos\n', busca);
bandera = 1;
break;
end
end

% Combustibles
for i = 1:m
    busca = cell2mat(combustibles(i)); % Nombre del combustible
    lbusca = length(busca);           % Longitud nombre combustible
    if length(linea)<2 | bandera == 1
        break;
    elseif busca == linea(1:lbusca) & (nosp-1)==lbusca & ...
        sum(CFc(i,:)) == 0

        lPM = textscan(linea(25:44), fsp); % Elementos en el compuesto
        clear list_elem list_num;

% Aquí, de acuerdo el formato especificado para "textscan" comprobamos que
% las posiciones pares del vector que devuelve no estén vacías y no sean 0.
% En ese caso, quiere decir que tenemos cierto número de átomos (posiciones
% pares del vector) de un mismo elemento (posiciones impares)
        for k = 2:2:8
            if isempty(lPM{k})==0 & lPM{k}~=0
                list_elem(k/2) = lPM{k-1};           % Forma de cell
                list_num(k/2) = double(lPM{k});       % Forma numérica
            end
        end
        PM_comb(i) = calcPM(list_elem,list_num); % Cálculo PM

% Si no calcula el PM, se registra y se para la función.
        if PM_comb(i) == 0;
            fprintf(flog, '%s Peso molecular d', datestr(now,time_style));
            fprintf(flog, 'e %s nulo. Revisar elementos calcPM.m\n', busca);
            fclose(flog);
            return;
        end

        fprintf(flog, '%s Peso molecular de ', datestr(now,time_style));
        fprintf(flog, '%s calculado\n', busca);

        c = 1:5;
        for j=1:3
            linea = fgetl(fid);
            CFc(i,c) = [str2double(linea(1:15))...
                       str2double(linea(16:30)) str2double(linea(31:45))...
                       str2double(linea(46:60)) str2double(linea(61:75))];
            c = c+5;
        end
        fprintf(flog, '%s Coeficientes de', datestr(now,time_style));

```

```

        fprintf(flog, ' %s leídos\n', busca);
        break;
    end
end

% Salto de renglón
linea = fgetl(fid);
nosp = find(linea == ' ',1);
end

% En caso de que se obtengan los coeficientes, se para la función y también
% se registra.
for i = 1:n
    if sum(CFe(i,:)) == 0
        fprintf(flog, '%s Revisar nombre de ', datestr(now,time_style));
        fprintf(flog, '%s en %s\n', especies{i}, ruta_THERM);
        fclose(flog);
        return;
    end
end

for j = 1:m
    if sum(CFc(j,:)) == 0
        fprintf(flog, '%s Revisar nombre de ', datestr(now,time_style));
        fprintf(flog, '%s en %s\n', combustibles{j}, ruta_THERM);
        fclose(flog);
        return;
    end
end

fprintf(flog, '%s Extracción de coeficientes ', datestr(now,time_style));
fprintf(flog, 'completada\n');

```

Tras la lectura de todos los coeficientes y el cálculo de los pesos moleculares de las especies y del combustible del caso de estudio, se procede al cálculo de los parámetros termodinámicos, es decir, la entalpía y su derivada:

```

%% Cálculo de parámetros termodinámicos y generación de archivos
%T = 298:0.5:5000; % T=Vector de temperaturas de 300 a 5000K. delT=0.5
T = 200:0.01:1200;
limit = find(T>1000,1)-1; % Posición de cambio de ecuación
T1 = T(1:limit); T2 = T(limit+1:end);

R = 8.3144621; % Constante de los gases ideales [J/mol K]
Table_h = zeros(length(T),1+n+2*m); % Inicilizamos las tablas: entalpias
Table_dh = zeros(length(T),1+n+2*m); % (1) y derivadas (2).
Table_h_cp = zeros(length(T),1+n*2+4*m);

Table_h(:,1) = T;
Table_dh(:,1) = T;
Table_h_cp(:,1) = T;

```

```

h = zeros(1,length(T));      % Inicializamos entalpías, calores latentes y
dh = h; Lvf = h; dLv = h;   % derivadas
for i = 1 : n
    %Calculamos h y dh para las especies teniendo en cuenta el cambio de
    %los valores de los coeficientes a partir de 1000K
    h(1:limit) = (CFe(i,8)*T1+CFe(i,9)/2*T1.^2+CFe(i,10)/3*T1.^3+CFe(i,11)...
        /4*T1.^4+CFe(i,12)/5*T1.^5+CFe(i,13))*R*1000/PM_esp(i);

    dh(1:limit) = (CFe(i,8)+CFe(i,9)*T1+CFe(i,10)*T1.^2+CFe(i,11)*T1.^3+...
        CFe(i,12)*T1.^4)*R*1000/PM_esp(i);

    h(limit+1:end) = (CFe(i,1)*T2+CFe(i,2)/2*T2.^2+CFe(i,3)/3*T2.^3+...
        CFe(i,4)/4*T2.^4+CFe(i,5)/5*T2.^5+CFe(i,6))*R*1000/PM_esp(i);

    dh(limit+1:end) = (CFe(i,1)+CFe(i,2)*T2+CFe(i,3)*T2.^2+CFe(i,4)...
        *T2.^3+CFe(i,5)*T2.^4)*R*1000/PM_esp(i);

    Table_h(:,i+1) = h;      % Construimos las tablas
    Table_dh(:,i+1) = dh;
    Table_h_cp(:,2*i:2*i+1) = [h' dh'];
end

% Cálculo de h y dh del combustible.
for i = 1 : m
    h(1:limit) = (CFc(i,8)*T1+CFc(i,9)/2*T1.^2+CFc(i,10)/3*T1.^3+CFc...
        (i,11)/4*T1.^4+CFc(i,12)/5*T1.^5+CFc(i,13))*R*1000/PM_comb(i);

    dh(1:limit) = (CFc(i,8)+CFc(i,9)*T1+CFc(i,10)*T1.^2+CFc(i,11)*T1.^3+...
        CFc(i,12)*T1.^4)*R*1000/PM_comb(i);

    h(limit+1:end) = (CFc(i,1)*T2+CFc(i,2)/2*T2.^2+CFc(i,3)/3*T2.^3+...
        CFc(i,4)/4*T2.^4+CFc(i,5)/5*T2.^5+CFc(i,6))*R*1000/PM_comb(i);

    dh(limit+1:end) = (CFc(i,1)+CFc(i,2)*T2+CFc(i,3)*T2.^2+CFc(i,4)*T2.^...
        3+CFc(i,5)*T2.^4)*R*1000/PM_comb(i);

    % Cálculo de h1 y dh1 del combustible. A partir de la Tc del combustible
    % no puede haber fase líquida,
    limit1 = find(T > Tc_comb(i),1)-1;
    T3 = T(1:limit1);

    Lvf(1:limit1) = (7.08*(1-T3/Tc_comb(i)).^(0.354)+10.95*w_comb(i)*(1-...
        T3/Tc_comb(i)).^(0.456))*R*Tc_comb(i)*1000/PM_comb(i);

    dLv(1:limit1) = (-7.08*0.354/Tc_comb(i)*(1-T3/Tc_comb(i)).^(-0.646)-...
        10.95*w_comb(i)*0.456/Tc_comb(i)*(1-T3/Tc_comb(i)).^(-0.544))*R*...
        Tc_comb(i)*1000/PM_comb(i);

    Lvf(limit1+1:end) = 0;
    dLv(limit1+1:end) = 0;

    h1f = h-Lvf;
    dh1f = dh-dLv;

    Table_h(:,n+1+i:n+2+i) = [h',h1f'];      % Finalizamos las tablas
    Table_dh(:,n+1+i:n+2+i) = [dh',dh1f'];
    Table_h_cp(:,2*n+(i-1)*4+2:2*n+(i-1)*4+5) = [h',h1f',dh',dh1f'];
end

```

```
fprintf(flog, '%s Cálculo de parámetros termodinám', datestr(now, time_style));
fprintf(flog, 'micos completado\n');
```

Tras el cálculo de las variables termodinámicas haciendo uso de los polinomios de la NASA y los coeficientes JANAF, el código continúa con la generación de la tabla utilizada por el código de OpenFOAM y guarda los datos, finalizando así el código.

```
cabecera1 = ['T[K]' blanks(16)];
cabecera2 = ['T[K]' blanks(16)];
cabecera3 = ['T[K]' blanks(16)];
for i = 1 : n
    word = cell2mat(especies(i));
    % Se ajusta la cabecera al espacio que ocuparan las listas numéricas.
    cabecera1 = [cabecera1 ' h_' word '[J/kg]' blanks(12-length(word))];
    cabecera2 = [cabecera2 ' dh_' word '[J/kg·K]' blanks(9-length(word))];
    cabecera3 = [cabecera3 ' h_' word '[J/kg]' blanks(12-length(word)) ...
        ' dh_' word '[J/kg·K]' blanks(9-length(word))];
end

for i = 1 : m
    word = cell2mat(combustibles(i));
    cabecera1 = [cabecera1 ' h_' word '[J/kg]' blanks(12-length(...
word)) ' h1_' word '[J/kg]' blanks(11-length(word))];
    cabecera2 = [cabecera2 ' dh_' word '[J/kg·K]' blanks(9-length(...
word)) ' dh1_' word '[J/kg·K]' blanks(8-length(word))];
    cabecera3 = [cabecera3 ' h_' word '[J/kg]' blanks(12-length(...
word)) ' h1_' word '[J/kg]' blanks(11-length(word)) ' dh_' word ...
'[J/kg·K]' blanks(9-length(word)) ' dh1_' word '[J/kg·K]' ...
blanks(8-length(word))];
end

fiw = fopen('Table_h.txt', 'wt');
fow = fopen('Table_dh.txt', 'wt');
fuw = fopen('Table_h_cp.txt', 'wt');

fprintf(fiw, '%s\n', cabecera1);
fprintf(fow, '%s\n', cabecera2);

% En las 3 primeras líneas indicamos el número de filas, el número de
% columnas y el ancho de las filas de cara a la lectura en openfoam
fprintf(fuw, '%d\n%d\n%d\n', length(T), n*2+m*4+1, (n*2+m*4+1)*21);
fprintf(fuw, '%s\n', cabecera3);

m = size(Table_h);

for i = 1:m(1)
    fprintf(fiw, '%+.13e ', Table_h(i,:)); fprintf(fiw, '\n');
    fprintf(fow, '%+.13e ', Table_dh(i,:)); fprintf(fow, '\n');
    fprintf(fuw, '%+.13e ', Table_h_cp(i,1:end-1));
    fprintf(fuw, '%+.13e ', Table_h_cp(i,end));
    fprintf(fuw, '\n');
end
```





```

    % Lectura del fichero y registro en la matriz de densidades
    m = fscanf(fid,fl,[13,inf]);
    matriz{ind} = m';
    fid = fclose(fid);
end

% Se determina el tamaño de la matriz de densidades. Al tratarse del
% combustible líquido, esta matriz sólo recogerá temperaturas que no
% superen la temperatura crítica. En este caso, al tratarse del dodecano,
% la temperatura crítica es 658 K.
TNIST = matriz{1}(:,1);
fil = find(TNIST > 658, 1); % Filas de la matriz de densidades
TNIST = TNIST(1:fil);
col = length(p);          % Columnas de la matriz de densidades

for i = 1:fil
    for j = 1:col
        mt(j,i) = matriz{j}(i,3);
    end
end

m_rho = mt'; % Filas: gradiente T;  Columnas: gradiente P
[fil, col] = size(m_rho);

```

Esta primera parte del código lee el valor de la densidad del combustible líquido para las posibles situaciones que se pueden tener durante el proceso de inyección y las guarda en una matriz de forma ordenada según los valores de presión y temperatura. Tras esto, el código pasa a calcular la derivada de la densidad respecto de la temperatura y la presión:

```

% Derivada respecto a T
% Las derivadas son numéricas y se obtienen por diferencias centrales salvo
% para los valores extremos de los datos.
m_drhodT = zeros(fil,col);
for j = 1:col
    m_drhodT(1,j) = (m_rho(2,j)-m_rho(1,j))/(TNIST(2)-TNIST(1));
    m_drhodT(end,j) = (m_rho(end,j)-m_rho(end-1,j))/...
        (TNIST(end)-TNIST(end-1));
    m_drhodT(2:end-1,j) = (m_rho(3:end,j)-m_rho(1:end-2,j))./...
        (TNIST(3:end)-TNIST(1:end-2));
end

% Derivada respecto a p
% Las derivadas son numéricas y se obtienen por diferencias centrales salvo
% para los valores extremos de los datos.
m_drhodP = zeros(fil,col);
p = p*1e+5;
for i = 1:fil
    m_drhodP(i,1) = (m_rho(i,2)-m_rho(i,1))/(p(2)-p(1));
    m_drhodP(i,end) = (m_rho(i,end)-m_rho(i,end-1))/...
        (p(end)-p(end-1));
    m_drhodP(i,2:end-1) = (m_rho(i,3:end)-m_rho(i,1:end-2))./...
        (p(3:end)-p(1:end-2));
end

```

Por último, una vez tiene las tres matrices, densidad y derivadas, las guarda y escribe las tablas en tres ficheros de texto.

```
%Añadimos una columna T y una fila P a las matrices
save m_rho.mat m_rho;
save m_drhodT.mat m_drhodT;
save m_drhodP.mat m_drhodP;

m_rho = [TNIST, m_rho];
m_rho = [0 p; m_rho];
m_drhodT = [TNIST, m_drhodT];
m_drhodT = [0 p; m_drhodT];
m_drhodP = [TNIST, m_drhodP];
m_drhodP = [0 p; m_drhodP];

[fil, col] = size(m_rho);

fid = fopen('m_rho.txt','wt');
fod = fopen('m_drhodT.txt','wt');
fud = fopen('m_drhodP.txt','wt');

fs = [];
for i = 1 : col;
    fs = [fs '%+7e '];
end
fs = [fs '\n'];

fprintf(fid, '%d\n', fil);
fprintf(fid, '%d\n', col);

for i = 1:fil
    fprintf(fid, fs, m_rho(i, :));
    fprintf(fod, fs, m_drhodT(i, :));
    fprintf(fud, fs, m_drhodP(i, :));
end
```

Al igual que en el caso anterior, basta con archivar los ficheros con las tres matrices en la carpeta “constant” del caso de estudio. Una vez hecho esto, el código de OpenFOAM es capaz de reconocer estos ficheros, leerlos y aplicar los valores que presentan las tablas para continuar con la ejecución del modelo.

### III Programa de post-procesado

Para el post-procesado de los datos obtenidos tras el cálculo del caso se realiza utilizando MATLAB y se lleva a cabo ejecutando un solo programa principal denominado *Resultados.m*. Este script está compartimentado en otros para facilitar su lectura y entendimiento. Aun así, en esta parte del código total se definen rutas para acceder a todos los archivos, parámetros comunes para la obtención de gráficos y otras variables de apoyo, además de explicar qué gráficos se obtendrán ya que contiene muchos comentarios:

```

%% Post-procesado de todo.
% Este script se lanzará para post-procesar los resultados obtenidos en los
% modelos CFD de inyección y para compararlos con los datos experimentales.
% Se llamarán diferentes scripts para simplificar el código.
% Distinguiremos entre estudios en planos normales al eje longitudinal y
% planos que incluyen a este eje.

% Gráficos de los planos normales:
% - U/U(axis) vs R/Xaxial
% - Yv/Yv(axis) vs R/Xaxial
% - Contornos de colores para Yv, Umean

% Gráficos de los planos coincidentes con el eje axial:
% - U/U_0 vs X/Deq en el eje
% - Yv vs X/Deq en el eje
% - Penetración Líquida y Gaseosa vs Tiempo
% - Contornos de Líquido y Vapor (R/Deq vs X/Deq)
% - Yv, Y, T en el eje longitudinal
% - Contornos de colores para Yv, Y, T

clear all; clc; close all;

% Diámetros equivalentes
deq677 = 0.084*sqrt(713.13/22.8);
deq678 = 0.088*sqrt(713.13/22.8);
deqcf = 0.0894*sqrt(713.13/23.27);

% Carpeta CFD
carpeta_cfd1 = 'E:/TFM/LES/surfaces/0.0019/';
carpeta_cfd2 = 'E:/TFM/LES/sets/0.0019/';

% Carpeta Experimental
carpeta_exp1 = 'E:/TFM/PC Jaime/EVAP_DATA/PIV/PIV_SA900I_ECN2/';
carpeta_exp2 = 'E:/TFM/PC Jaime/EVAP_DATA/MixFrac/';

% Parámetros para los gráficos
xLimit = [0,1];
stotal = get(0, 'ScreenSize');
tamano = stotal.*[0.9 0.9 0.65 0.5];
Proporcion = 5.77/11.02;

```

Es fácil apreciar que este código no es extenso y es sencillo de dividir en partes. Hasta aquí, lo único que ha hecho ha sido definir rutas y variables auxiliares. Después se diferencia claramente cuatro sectores diferentes para la representación de los resultados y parámetros geométricos del chorro. Estas cuatro secciones se distinguir por:

- Planos normales al eje longitudinal. En este apartado se ejecuta el programa de apoyo *eje\_normales.m*, dónde se seleccionan dos planos normales al eje longitudinal del cilindro y el mismo eje para estudiar diferentes variables: representación radial y axial de la fracción másica de combustible gaseoso y los contornos en los planos de estas dos variables.
- Plots a puntos para planos longitudinales. Aquí se leen variables como la temperatura, la fracción másica de combustible, la velocidad o la tasa de evaporación de combustible en un plano que contiene al eje longitudinal. Se representan las variables anteriores frente a la composición de la mezcla. Llama al “script” de apoyo *plot\_a\_puntos.m*.
- Penetración. Hace uso del programa *penetracion.m* para hacer el estudio temporal de la penetración del chorro gaseoso de combustible y la longitud líquida del mismo.
- Contornos. Ésta es la última sección del programa principal y consigue representar, cómo ya indica el propio nombre de la sección, los contornos de la temperatura de la mezcla y la fracción másica de combustible en un plano continente del eje longitudinal. Para ello hace uso del programa homónimo *contornos.m*.

```
%% Planos normales al eje longitudinal

% Elección de los planos normales
nplanos = 2;      % Número de planos
dep = 20;        % Distancia entre planos
plano1 = 25;     % Primer plano. Situado a 25 mm del punto de inyección

% Planos Normales y Eje Axial
eje_normales

%% Plots a puntos para planos longitudinales
plot_a_puntos

%% Penetración
penetracion

%% Contornos
carpeta_cfd1 = 'E:/TFM/LES/surfaces/0.0019/';
carpeta_exp = [carpeta_exp2 '900K22.8kg150MPa_'];
contornos
```

### III. I Eje axial y planos normales

Si se revisa el código del programa *eje\_normales.m*, se tiene que éste es el “script” más complejo y largo de los cuatro programas auxiliares para el tratamiento de los resultados obtenidos, ya que ha de leer varias variables de diferentes sectores del dominio (eje axial y dos planos perpendiculares al mismo), tratarlas, leer datos del caso experimental y compararlos en gráficamente.

En la primera parte del código se cargan las variables sobre el eje axial del cilindro, de los planos normales al eje:

```
%% eje_normales

%% Carga de datos del eje CFD

eje1 = 'axis_U_UMean.xy';
%[z Ux Uy Uz Ux_mean Uy_mean Uz_mean]
eje2 =
'axis_Y_Yv_Ybar_rho_rholiq_rhogas_d32_Sigma_h_TMean_p_YMean_YvMean_Zv_ZvMean.
xy';
%[z Y Yv Ybar rho rholiq rhogas d32 Sigma h T p Ymean YvMean Zv ZvMean]
eje3 = 'axis_UPrime2Mean.xy';

% U Axis
Uaxis = dlmread([carpeta_cfd2 eje1], '');
Z_axis = Uaxis(:,1)*1000;
U_axis = Uaxis(:,7);

% Amalgama Axis
Amalgama_axis = dlmread([carpeta_cfd2 eje2], '');
Yv_axis = Amalgama_axis(:,14);
Y_axis = Amalgama_axis(:,13);
T_axis = Amalgama_axis(:,11);
YY_axis = Yv_axis + Y_axis;

clear Uaxis Amalgama_axis;

%% Carga de datos de los planos normales CFD

for i = 1:nplanos
    Um = dlmread([carpeta_cfd1 'UMean_rad' num2str(plano1 + (i-1)*dep)...
        '_2Dplane.raw'], '', 2, 0);
    % Um = [X Y Z Ux Uy Uz]
    U{i} = abs([Um(:,4) Um(:,5) Um(:,6)...
        (Um(:,4).^2 + Um(:,5).^2 + Um(:,6).^2).^(1/2)]);
    % U{i} = [Ux Uy Uz |U|]

    Yvm = dlmread([carpeta_cfd1 'YvMean_rad' num2str(plano1 + (i-1)*dep)...
        '_2Dplane.raw'], '', 2, 0);
    % Yvm = [X Y Z Yv]
    Yv{i} = abs([Yvm(:,4)]);
```

```

% Yv{i} = [Yv]

radio{i} = round([Um(:,1) Um(:,2) (Um(:,1).^2 + Um(:,2).^2).^(1/2)]*...
    1000000)/1000;
% Redondeo a micrómetros por motivos de hacer los plots
end

% Dominio para los contourf
X = linspace(min(radio{1}(:,1)),max(radio{1}(:,1)),100);
Y = linspace(min(radio{1}(:,2)),max(radio{1}(:,2)),100);
[xmap, ymap] = meshgrid(X,Y);

clear Um Yvm X Y;

```

Tras esto se pasa a la lectura de los datos experimentales, para el eje y los planos normales a éste, y el tratamiento de los mismos para poder representarlos y compararlos con los resultados obtenidos por el modelo.

```

%% Datos experimentales en el eje
% U en el eje
load([carpeta_exp1 'inside_04us_avg_axcorr_unc.mat']);

Ze_axis = data_m(1,18).zcoord_m*1000;
re_axis = abs(data_m(1,18).rcoord_m'*1000);

umat = data_m(1,18).zvelmat_m_s;
umat_std = data_m(1,18).zvelmat_m_s_unc;

% Obtener fila del eje
[val, fil] = min(re_axis);
Ue_axis = umat(fil,:);
UePOS_axis = Ue_axis + umat_std(fil,:);
UeNEG_axis = Ue_axis - umat_std(fil,:);

% Yv en el eje
Zexp_axis = csvread(sprintf([carpeta_exp2 'xv.csv']));
rexp_axis = csvread(sprintf([carpeta_exp2 'yv.csv']));

% Fracción másica de combustible gaseoso
A = csvread(sprintf([carpeta_exp2 '900K22.8kg150MPa_mean/'...
    '900K22.8kg150MPa_mean.csv']));

% Desviación
B = csvread(sprintf([carpeta_exp2 '900K22.8kg150MPa_meanunc/'...
    '900K22.8kg150MPa_meanunc.csv']));

% Obtener fila del eje (r=0)
fila = find(rexp_axis==0);
Yve_axis = A(fila,:);
YvePOS_axis = Yve_axis + B(fila,:);
YveNEG_axis = Yve_axis - B(fila,:);

```

```

%% Datos experimentales en planos normales.

% U en planos normales
Ze_norm = data_m(1,18).zcoord_m*1000;
re_norm = abs(data_m(1,18).rcoord_m'*1000);

umat = data_m(1,18).zvelmat_m_s;
umat_std = data_m(1,18).zvelmat_m_s_unc;

% Ordenar Matrices
j = 1;
while j < length(re_norm)
    [val, pos] = min(re_norm(j:end));
    pos = pos+j-1;
    re_norm([j,pos]) = re_norm([pos,j]);
    umat([j,pos], :) = umat([pos,j], :);
    umat_std([j,pos], :) = umat_std([pos,j], :);
    j = j+1;
end

% Fin de ordenar matrices

% Obtener columnas de los planos
for i = 1:nplanos
    z = planol+(i-1)*dep;
    [s,p] = min(abs(Ze_norm-z));
    zp(i) = p;
end

Ue_norm = umat(:, zp);

UePOS_norm = Ue_norm + umat_std(:, zp);
UeNEG_norm = Ue_norm - umat_std(:, zp);

% Yv en planos normales
Zexp_norm = csvread(sprintf([carpeta_exp2 'xv.csv']));

rexp_norm = csvread(sprintf([carpeta_exp2 'yv.csv']));
rexp_norm = round(abs(rexp_norm)*1000)/1000;

% Fracción másica de combustible gaseoso
A = csvread(sprintf([carpeta_exp2 '900K22.8kg150MPa_mean/'...
    '900K22.8kg150MPa_mean.csv']));

% Desviación
B = csvread(sprintf([carpeta_exp2 '900K22.8kg150MPa_meanunc/'...
    '900K22.8kg150MPa_meanunc.csv']));

% Ordenar Matrices
j = 1;
while j < length(rexp_norm)
    [val, pos] = min(rexp_norm(j:end));
    pos = pos+j-1;
    rexp_norm([j,pos]) = rexp_norm([pos,j]);
    A([j,pos], :) = A([pos,j], :);
    B([j,pos], :) = B([pos,j], :);
    j = j+1;
end

```

```

% Fin de ordenar matrices

% Obtener columna de los planos
for i = 1:nplanos
    z = plano1+(i-1)*dep;
    [s,p] = min(abs(Zexp_norm-z));
    zp(i) = p;
end

Yve_norm = A(:, zp);

YvePOS_norm = Yve_norm + B(:, zp);
YveNEG_norm = Yve_norm - B(:, zp);

% Corregimos los puntos dobles
rexp_norm = [rexp_norm(1), rexp_norm(3:2:end-2), rexp_norm(end-1:end)];
Yve_norm = [Yve_norm(1, :); (Yve_norm(3:2:end-2, :) + ...
    Yve_norm(2:2:end-3, :))/2; Yve_norm(end-1:end, :)];
YvePOS_norm = [YvePOS_norm(1, :); (YvePOS_norm(3:2:end-2, :) + ...
    YvePOS_norm(2:2:end-3, :))/2; YvePOS_norm(end-1:end, :)];
YveNEG_norm = [YvePOS_norm(1, :); (YveNEG_norm(3:2:end-2, :) + ...
    YveNEG_norm(2:2:end-3, :))/2; YveNEG_norm(end-1:end, :)];

```

El post-proceso de los datos computacionales es mucho más sencillo que el de los datos experimentales:

```

%% Tratamiento de datos CFD de los planos normales al eje axial

for i = 1 : nplanos
    Umap_norm{i} = griddata(radio{i}(:,1), radio{i}(:,2), U{i}(:,3), xmap, ymap);
    Yvmap_norm{i} = griddata(radio{i}(:,1), radio{i}(:,2), Yv{1}, xmap, ymap);
end

% Distinción de radios y sus características
% U
for i = 1:nplanos
    aux = sort(radio{i}(:,3));
    j = 1;
    filas = length(aux);
    matrizU_Geom{i} = [];
    % [Radio, Repeticiones, Media Geom, Varianza Geom, Desviación Geom]
    old_pos = 0;
    while j <= filas
        valor = aux(j);
        pos = max(find(aux == valor));
        matrizU_Geom{i} = [matrizU_Geom{i}; valor, pos-old_pos];
        old_pos = pos;
        j = pos + 1;
    end
    [nradios{i}, ~] = size(matrizU_Geom{i});
end
% En este primer bucle se rellenan las columnas "Radio" y "Repeticiones" de
% "matrizU_Geom"

% En el segundo bucle se rellenan las columnas "Media Geom", "Varianza
% Geom" y "Desviación Geom"

```



```

for j = 1:nplanos
    for i = 1:nradios{j}
        pos = find(matrizU_Geom{j}(i) == radio{j}(:,3));
        matrizU_Geom{j}(i,3) = sum(U{j}(pos,3))/matrizU_Geom{j}(i,2);
    end
end

% Yv
% Para
for j = 1:nplanos
    matrizYv_Geom{j} = matrizU_Geom{j}(:,1:2);
    % [Radio, Repeticiones, Media, Varianza, Desviación]
    for i = 1:nradios{j}
        pos = find(matrizYv_Geom{j}(i) == radio{j}(:,3));
        matrizYv_Geom{j}(i,3) = sum(Yv{j}(pos))/matrizYv_Geom{j}(i,2);
    end
end
end

```

Ya sólo queda representar gráficamente la velocidad y la fracción másica de combustible axialmente (tanto para el caso computacional como el experimental), así como radialmente (para todos los planos escogidos), y el contorno de ambas variables para los resultados proporcionados por el modelo computacional.

```

%% Gráficos
% U: Valores en el eje y según distancia radial por planos

for i = 1 : nplanos
    figure('Position',[250 150 900 400]);

    subplot(1,2,1)
    XS = [Ze_axis, fliplr(Ze_axis)]/deq677;
    YS = [UeNEG_axis, fliplr(UePOS_axis)]/590;    %/Uz (1)

    ER1 = fill(XS,YS,[0.7 0.7 0.7],'EdgeColor','w');
    hold on;

    p1 = plot(Z_axis/deq678, U_axis/540, 'b');    %/Uz (1)
    hold on;
    p2 = plot(Ze_axis/deq678,Ue_axis/590, 'k');    %/U1 (1)

    grid on;
    axis([30 150 0 0.3])
    xlabel('Posición axial/D_{eq} [-]');
    ylabel('U_{norm} [-]');
    title('Velocidad axial normalizada en el eje');
    legend([p1,p2,ER1], 'U_{CFD}', 'U_{Exp}', 'Desv U_{Exp}');
    set(gca, 'Xtick', 30:15:150);
    set(gca, 'Ytick', 0:0.05:0.3);
    set(gca, 'FontName', 'Arial', 'FontSize', 12)

    subplot(1,2,2)
    XS = [re_norm, fliplr(re_norm)]/(plano1 + (i-1)*dep);
    YS = [UeNEG_norm(:,i); flipud(UePOS_norm(:,i))]/Ue_norm(1,i);

```

```

ER1 = fill(XS,YS,[0.7 0.7 0.7], 'EdgeColor', 'w');
hold on;

p1 = plot(matrizU_Geom{i}(:,1)/(plano1 + (i-1)*dep), ...
    matrizU_Geom{i}(:,3)/matrizU_Geom{i}(1,3), 'b');
hold on;
p2 = plot(re_norm/(plano1 + (i-1)*dep),Ue_norm(:,i)/Ue_norm(1,i), 'k');

grid on;
axis([0 0.3 -0.1 1.2])
xlabel('Posición radial/Posición axial [-]');
ylabel('Uz/Uz_{eje} [-]');
title(['Z = ' num2str(plano1 + (i-1)*dep) ' mm']);

legend([p1,p2,ER1], 'U_{CFD}', 'U_{Exp}', 'Desv U_{Exp}')
set(gca, 'Xtick', 0:0.05:0.3);
set(gca, 'Ytick', 0:0.2:1.5);
set(gca, 'FontName', 'Arial', 'FontSize', 12)
end

% Yv: Valores en el eje y según la distancia radial por planos
for i = 1 : nplanos
    figure('Position',[250 150 900 400]);

    subplot(1,2,1)

    XS = [Zexp_axis, fliplr(Zexp_axis)]/deq677;
    YS = [YveNEG_axis, fliplr(YvePOS_axis)];    %/Uz(1)

    ER1 = fill(XS,YS,[0.7 0.7 0.7], 'EdgeColor', 'w');
    hold on;
    p1 = plot(Z_axis/deqcf, Yv_axis, 'b');
    hold on;
    p2 = plot(Zexp_axis/deq677,Yve_axis, 'k');

    grid on;
    axis([30 150 0 0.3])

    xlabel('Posición axial/D_{eq} [-]');
    ylabel('Yv [-]');
    %ylabel('Yv/Yv_{eje} [-]');
    title('Fracción málica de combustible gaseoso en el eje');
    legend([p1, p2, ER1], 'Yv_{CFD}', 'Yv_{Exp}', 'Desv Yv{Exp}');
    set(gca, 'Xtick', 30:15:150);
    set(gca, 'Ytick', 0:0.05:0.3);
    set(gca, 'FontName', 'Arial', 'FontSize', 12)

    subplot(1,2,2)

    XS = [rexp_norm, fliplr(rexp_norm)]/(plano1 + (i-1)*dep);
    YS = [YveNEG_norm(:,i); flipud(YvePOS_norm(:,i))]/Yve_norm(1,i);

    ER1 = fill(XS,YS,[0.7 0.7 0.7], 'EdgeColor', 'w');
    hold on;

```

```

p1 = plot(matrizYv_Geom{i}(:,1)/(plano1 + (i-1)*dep),...
          matrizYv_Geom{i}(:,3)/matrizYv_Geom{i}(1,3), 'b');
hold on;
p2 = plot(rexp_norm/(plano1+(i-1)*dep),Yve_norm(:,i)/Yve_norm(1,i), 'k');

grid on;
axis([0 0.35 -0.1 1.05])
xlabel('Posición radial/Posición axial [-]');
ylabel('Yv/Yv_{eje} [-]');
title(['Z = ' num2str(plano1 + (i-1)*dep) ' mm']);

legend([p1, p2, ER1], 'Yv_{CFD}', 'Yv_{Exp}', 'Desv Yv_{Exp}')
set(gca, 'Xtick', 0:0.05:0.3);
set(gca, 'Ytick', 0:0.2:1);
set(gca, 'FontName', 'Arial', 'FontSize', 12)
end

% Y e Yv en el eje
figure('Position', [250 150 500 400]);
p1 = plot(Z_axis/deqcfid, Yv_axis, 'b');
hold on;
p2 = plot(Z_axis/deqcfid, Y_axis, 'b--');

grid on;
axis([0 0.15 0 1])
xlabel('Z/D_{eq} [-]');
ylabel('Yv_{eje} [-], Y_{eje} [-]');
title('Fracción másica de combustible en el eje');
legend([p1,p2], 'Yv_{CFD}', 'Y_{CFD}');
set(gca, 'Xtick', 0:0.03:0.15);
set(gca, 'Ytick', 0:0.1:1);
set(gca, 'FontName', 'Arial', 'FontSize', 12)

% T, Y, Yv, h vs Y+Yv
xLimit = [0,1];
figure('Position', stotal.*[1 1 0.95 0.95*Proporcion]);
p1 = plot(YY_axis, T_axis, 'k');
xlim(xLimit);
ax1 = gca;
set(ax1, 'YColor', 'k');
pos1 = get(ax1, 'Position');
pos3 = pos1 + pos1(1)*[-1/2 0 1/2 0];
ax2 = axes('Position', pos1, 'Color', 'none', 'XAxisLocation', ...
          'bottom', 'YAxisLocation', 'right', 'XLim', xLimit, 'XColor', 'k', ...
          'YColor', 'b', 'NextPlot', 'add');
p2 = plot(ax2, YY_axis, Y_axis, 'b--');
p3 = plot(ax2, YY_axis, Yv_axis, 'b');
xlabel(ax1, 'Y+Yv [-]');
ylabel(ax1, 'T [K]');
ylabel(ax2, 'Y, Yv [-]');
legend([p1,p2,p3], 'Temperature', 'Y', 'Yv', 'Location', 'North')
title('Modelo CFD: Algunas variables en el eje longitudinal');

% Contornos

for i = 1 : nplanos
    % U
    figure('Position', [250 150 500 400]);
    contourf(xmap, ymap, Umap_norm{i}, 'Linecolor', 'none');
    xlabel('x [mm]');

```

```

ylabel('y [mm]');
title(['Z = ' num2str(plano1+(i-1)*dep) ' [mm]: Velocidad Media [m/s]']);
colorbar;
set(gca, 'FontName', 'Arial', 'FontSize', 12)

%Yv
figure('Position',[250 150 500 400]);
contourf(xmap, ymap, Yvmap_norm{i}, 'Linecolor', 'none');
xlabel('x [mm]');
ylabel('y [mm]');
title(['Z = ' num2str(plano1 + (i-1)*dep) ' [mm]: Yv [-]']);
colorbar;
set(gca, 'FontName', 'Arial', 'FontSize', 12)
end

```

Con esto se llega al final del código del programa *eje\_normales.m*. Ejemplos de representaciones de esta parte del código son la Figura 4.41, la Figura 4.49, la Figura 4.17 y la Figura 4.51 del capítulo 4.

### III. II Planos continentes del eje axial

El post-proceso y la representación de las variables situadas en planos que confluyen con el eje axial del cilindro requieren el programa más sencillo de las cuatro situaciones, *plots\_a\_puntos.m*. Aquí sólo pueden compararse datos obtenidos para el caso calculado como RANS con los datos del caso calculado como LES. Las gráficas consisten en la representación de determinadas variables, como la temperatura o el error relativo de entalpía específica de la mezcla, frente a la composición global de la mezcla:

```

% Plano Axial, Plots a puntos

% Temperatura
path = [carpeta_cfd1 'TMean_axisym2DplaneAxial.raw'];
plano_T = dlmread(path, '', 2, 0);
pT = plano_T(:, 4);
pT = (pT <= 900) .* pT + (pT > 900) * 900;

% Yv
path = [carpeta_cfd1 'YvMean_axisym2DplaneAxial.raw'];
plano_Yv = dlmread(path, '', 2, 0);
pYv = plano_Yv(:, 4);

% Y
path = [carpeta_cfd1 'YMean_axisym2DplaneAxial.raw'];
plano_Y = dlmread(path, '', 2, 0);
pY = plano_Y(:, 4);

posiciones = find(pY >= 5e-4);
longitudes = plano_Y(posiciones, 3);
LL = max(longitudes);

```

```

% h
path = [carpeta_cfd1 'h_axisym2DplaneAxial.raw'];
plano_h = dlmread(path, ',', 2, 0);
ph = plano_h(:, 4);
pht = zeros(length(ph), 1);
m_hcp = readhcp('../Table_h_cp.txt');
for i = 1 : length(ph)
    pht(i) = hcalc(m_hcp, pT(i), pY(i), pYv(i))/1000; %[kJ/kg]
end

pYY = pYv + pY;

% Plots
figure('Position', tamaño);
plot(pYY, pT, '.', 'MarkerSize', 0.5)
xlabel('Y+Yv [-]');
ylabel('T [K]');
xlim([0 1]);
title('Control por temperatura, LES')
set(gca, 'FontName', 'Arial', 'FontSize', 12)

figure('Position', tamaño);
plot(pYY, pYv, '.', 'MarkerSize', 0.5)
xlabel('Y+Yv [-]');
ylabel('Yv [-]');
xlim([0 1]);
title('Control por temperatura, LES')
set(gca, 'FontName', 'Arial', 'FontSize', 12)

figure('Position', tamaño);
plot(pYY, pY, '.', 'MarkerSize', 0.5)
xlabel('Y+Yv [-]');
ylabel('Y [-]');
xlim([0 1]);
title('Control por temperatura, LES')
set(gca, 'FontName', 'Arial', 'FontSize', 12)

figure('Position', tamaño);
plot(pYY, (ph-pht) ./ ph * 100, 'k');
xlabel('Y+Yv [-]');
ylabel('Err. Rel. h [%] [-]');
xlim([0 1]);
title('Control por temperatura, LES')
set(gca, 'FontName', 'Arial', 'FontSize', 12)

```

Ejemplo de este código son la Figura 4.4 y la Figura 4.18 del capítulo 4.

### III. III Penetración y longitud líquida

En comparación con el programa *eje\_normales.m*, el programa utilizado para el cálculo de la penetración del chorro de combustible y la longitud líquida, *penetracion.m*, es bastante simple y más corto. No obstante requiere de dos pequeñas funciones extra:

*selecorvalordecampo.m* y *findcell.m*. Estas dos funciones son necesarias para la correcta lectura de los datos experimentales y su posterior tratamiento.

El programa principal, en primer lugar realiza la lectura de los casos RANS y LES a partir del fichero “mysummary.txt” de cada caso para compararlos:

```
%% Penetración

%% Penetración CFD
% LES
pnt = dlmread('mysummary.txt','\t', 1, 0);
ta = pnt(:,1)*1000;
LLa = pnt(:,6)*1000;
Sa = pnt(:,8)*1000;

% RANS
pnt = dlmread('../T_control/mysummary.txt','\t', 1, 0);
tb = pnt(:,1)*1000;
LLb = pnt(:,6)*1000;
Sb = pnt(:,8)*1000;

figure('Position',tamano);
plot(tb,Sb,'b','LineWidth',2)
hold on;
plot(tb,LLb,'b--','LineWidth',2)
hold on;
plot(ta,Sa,'k','LineWidth',1.5)
hold on;
plot(ta,LLa,'k--','LineWidth',1.5)
hold on;

xlabel('Tiempo [ms]');
ylabel('Longitud líquida, penetración gaseosa [mm]');
legend('RANS Gas','RANS Liq','LES Gas','LES Liq')
set(gca,'FontName','Arial','FontSize', 12)
```

Después de estas gráficas, el código continúa con la lectura de los datos experimentales (dónde hace uso de la función *selecorvalordecampo.m*) y su tratamiento:

```
%% Penetración experimental
load('../Data_SprayA_Michele/PenetrationInert_Std.mat');

A = selecorvalordecampo(RES,'AmbTemp',900);
B = selecorvalordecampo(A,'Density',22.8);

C1 = selecorvalordecampo(B,'Prail_bar',1500);
C1 = C1(1,1);

te1 = C1.TimeM2/1000 - 0.295;
Se1 = C1.PenetrationM2*1.02;
```

```

repl = max(C1.Rep);
conf1 = 1.96*C1.SpVapPenetration/sqrt(repl);

S1D = Se1 - conf1;
S1U = Se1 + conf1;

% Experimental LL
load('../Data_SprayA_Michele/dataDBIECN2.mat');

A1 = selecporvalordecampo(RES, 'AmbTemp', 900);
B1 = selecporvalordecampo(A1, 'Density', 22.8);

C2 = selecporvalordecampo(B1, 'Prail_bar', 1500);
C2 = C2(1,1);

LL1 = C2.LLextAvg;
Std1 = C2.LLextStd;

x = 1;
y = LL1;
E = Std1;

```

Por último, pasa a comparar los resultados del modelo computacional con los datos ofrecidos por el caso experimental:

```

%% plot the result
f1=figure;

set(f1, 'Position', [10 40 500 750], 'Color', 'w'); %Vertical

subplot(2,1,1)

Xe1 = [te1; flipud(te1)];
Ye1 = [S1D, fliplr(S1U)];

ER1 = fill(Xe1, Ye1, [0.7 0.7 0.7], 'EdgeColor', 'w');
hold on
E1 = plot(te1, Se1, 'm--', 'LineWidth', 1.5);
hold on
C1 = plot(ta, Sa, 'b-', 'LineWidth', 1.5);
hold off
xlabel('Time [ms]', 'FontName', 'Arial', 'FontSize', 12)
ylabel('Vapor Penetration [mm]', 'FontName', 'Arial', 'FontSize', 12)
grid 'on'

l1=legend([E1 C1], 'Experimental', 'Modelo CFD', 'LOCATION', 'SouthEast');

h=findobj(gcf, 'type', 'axes', 'tag', 'legend');
Pos=get(h, 'position');
Pos(1)=0.9*Pos(1); % Move left
Pos(2)=1.01*Pos(2);
Pos(3)=1.25*Pos(3); % Double the length
set(h, 'position', Pos) % Implement it
set(gca, 'Xtick', 0:0.5:2.5, 'XtickLabel', strcat('0|', ...
    sprintf('%0.1f|', 0.5:0.5:2.5)));

```

```

axis([0 2.5 0 80])
set(gca, 'FontName', 'Arial', 'FontSize', 12)

subplot(2,1,2)

E1 = errorbar(x,y,E, 'ko', 'MarkerSize',8, 'LineWidth',1.5);
hold on
D1 = plot(x,Y, 'bo', 'MarkerSize',8, 'LineWidth',1.5);
hold off
xlabel ('Modelo', 'FontName', 'Arial', 'FontSize', 12)
ylabel ('Liquid Length [mm]', 'FontName', 'Arial', 'FontSize', 12)
grid 'on'
l2=legend([E1 D1], 'Experimental', 'CFD Model',
'LOCATION', 'NorthEast');
p11 = get(l1, 'Position');
p12 = get(l2, 'Position');
set(l2, 'position', [p11(1) p12(2) p11(3) p12(4)])

set(gca, 'FontName', 'Arial', 'FontSize', 12)
set(f1, 'PaperPositionMode', 'auto');

name = sprintf('Parametric/Penet_LL_P');

```

Por otra parte, las funciones auxiliares *selecorvalordecampo.m* y *findcell.m* son dos funciones cortas y simples que sirven para seleccionar campos concretos de súper-estructuras, como es el caso de la estructura dónde se almacenan los datos experimentales que permiten calcular la penetración y longitud líquida del chorro. El código de la primera función auxiliar, que hace uso de la segunda función, es el siguiente:

```

function S=selecorvalordecampo(P,campo,valores)
%S=selecorvalordecampo(P,campo,valores)
%S=selecorvalordecampo(P,campo,valor)
%
%selecciona en una superestructura
%las elementos coincidentes con el valor o valores
%para un determinado campo
%valores puede ser un unico valor (string o numero)
%o un cell con los distintos valores (strings o numeros)
% tambien admite vector de numeros

%si valores es sencillo o vector de datos, lo hago cell
if iscell(valores)==0
    if isnumeric(valores)
        valores = num2cell(valores);
    else
        valores={valores};
    end
end

%extraigo el vector del campo
Vcampo={P.(campo)};

%selecciono

```



```

jjj=[];
L=length(valores);
for ival=1:L
    valor=valores{ival};

    j=findcell(Vcampo,valor);
    jjj=[jjj; j];
end

S=P(jjj);

```

Y el código de la función auxiliar *findcell.m* lee como sigue:

```

function j=findcell(V,valor,nocasesensitive)
%j=findcell(V,valor)
%j=findcell(V,valor,'nocasesensitive')
%
%como find pero para cell (numeros o letras)

switch nargin
    case 2
        casesensitive=1;
    case 3
        casesensitive=0;
    otherwise
        error('Numero incorrecto de argumentos')
end

L=length(V);
j=[];

if isnumeric(valor) %para numeros
    for i=1:L
        pala=V{i};
        if isnumeric(pala)
            if pala==valor
                j=[j;i];
            end
        end
    end
end

else %para caracteres
    for i=1:L
        pala=V{i};
        if isnumeric(pala)
            pala=num2str(pala);
        end
        if casesensitive
            if strcmp(pala,valor)
                j=[j;i];
            end
        else
            if strcmpi(pala,valor)
                j=[j;i];
            end
        end
    end
end
end

```

end

Con esto finaliza el código de esta parte del post-proceso, y ejemplos de las gráficas que se tienen son la Figura 4.45 y la Figura 4.47 del capítulo 4.

### III. IV Contornos de variables

Esta parte del código sirve para determinar y comparar los contornos del chorro de combustible gaseoso del caso experimental y del modelo computacional LES, además de para graficar los contornos de temperatura, fracción másica de combustible o velocidad. Hace uso del programa auxiliar *contornos.m*, el cuál no tiene muchas líneas de código. Como el resto de programas auxiliares, primero ha de leer y tratar las variables de la solución aportada por el modelo y la solución del caso experimental:

```
%% Contornos

%% Sólo Contorno: YvMean e YMean
[m_YvMean] = leer_contorno([carpeta_cfd1
'YvMean_xzp_2Dplane.raw'], [3,1,4],500,201);
[m_YvMean] = leer_contorno([carpeta_cfd1
'YMean_axisym2DplaneAxial.raw']...
, [3,2,4],700,501);
m_z = m_YvMean(:, :, 1);
m_r = m_YvMean(:, :, 2);
m_YvMean = m_YvMean(:, :, 3);

[fil, col] = size(m_z);

for i = 1 : col
    m_YvMean(:, i) = m_YvMean(:, i) / m_YvMean(round(fil/2), i);
end

m_YvMean = (m_YvMean > 1) * 1 + (m_YvMean <= 1) .* m_YvMean;

z = m_z(1, :);
r = m_r(:, 1)';

% Experimental
Yve = csvread(sprintf([carpeta_exp
'mean/900K22.8kg150MPa_mean.csv']));
re = csvread(sprintf([carpeta_exp 'mean/yv.csv']));
ze = csvread(sprintf([carpeta_exp 'mean/xv.csv']));

[mz, mr] = meshgrid(ze, re);
```

Después de leer y tratar las variables, procede a construir las gráficas:

```

%figure('Position',stotal.*[1 1 0.95 0.95*Proporcion]);
%subplot(1,2,1)
figure;
p1 = contour(m_z,m_r,m_YvMean,[10,0.001],'LineColor','b');
hold on;
p2 = contour(mz,mr,Yve,[10,0.001],'LineColor','m');
axis([18 45 -10 10])
legend('Caso LES','Caso Experimental')
xlabel('Posición Axial/Deq');
ylabel('Posición Radial/Deq');

%% Contourf
% T
[m_T] = leer_contorno([carpeta_cfd1
'TMean_axisym2DplaneAxial.raw'],...
[3,2,4],700, 501);

% Contourf
m_T(:,:,3) = (m_T(:,:,3)<=900).*m_T(:,:,3) + (m_T(:,:,3)>900)*900;
figure('Position',tamano);
contourf(m_z, m_r, m_T(:,:,3),20,'Linecolor','none');
xlabel('z [mm]');
ylabel('r [mm]');
title('Modelo CFD: Mapa de Temperatura [K]');
colorbar;
axis([0 55 0 10]);
clear m_T;

% U
[m_U] = leer_contorno([carpeta_cfd1
'UMean_axisym2DplaneAxial.raw'],...
[3,2,6],700,501);

% Contourf
figure('Position',tamano);
contourf(m_z,m_r,m_U(:,:,3),50,'Linecolor','none');
xlabel('z [mm]');
ylabel('r [mm]');
title('Modelo CFD: Mapa de Velocidad Axial [m/s]');
colorbar;
axis([0 55 0 10]);
clear m_U;

% Contourf
figure('Position',tamano);
contourf(m_z,m_r,m_YvMean,50,'Linecolor','none');
xlabel('z [mm]');
ylabel('r [mm]');
title('Modelo CFD: Mapa de Yv [-]');
colorbar;
axis([18 55 0 10]);

```

Con esto finaliza todo el código de post-proceso. Ejemplos de esta última parte son la Figura 4.23 y la Figura 4.46 del capítulo 4.

