



**Politecnico
di Torino**

Politecnico di Torino

Corso di Laurea Magistrale in
Ingegneria Chimica e dei Processi Sostenibili

A.a. 2020/2021

Sessione di Laurea Ottobre 2021

Studio computazionale di rottura di gocce in emulsione

Relatori:

Prof. Antonio Buffo

Prof. Marco Vanni

Prof. Daniele Marchisio

Prof. Gianluca Boccardo

Candidato:

Francesco De Roma

Indice

1	Introduzione	1
1.1	Stato dell'arte	1
1.2	Obiettivo del lavoro di tesi	6
1.3	Struttura del lavoro	7
2	Teoria e metodi	9
2.1	Equazioni fondamentali	9
2.1.1	Ipotesi	9
2.1.2	Equazione di continuità	10
2.1.3	Equazione di Navier – Stokes	10
2.1.4	Campo di moto e <i>mixing index</i>	11
2.1.5	Numeri adimensionali	14
2.2	Metodo dei volumi finiti	15
2.2.1	Metodi di discretizzazione spaziale	17
2.2.2	Metodi di discretizzazione temporale	20
2.2.3	Algoritmi per l'accoppiamento pressione – velocità	20
2.3	Metodi <i>Volume of Fluid</i>	22
2.3.1	Descrizione matematica	23
2.3.2	Tensione interfacciale	24
2.3.3	Vantaggi e svantaggi	26
3	Dettagli numerici	29
3.1	Software CFD – OpenFOAM	29
3.1.1	Struttura di una simulazione in OpenFOAM	30
3.1.2	Algoritmo di avvezione FCT – MULES	32
3.1.3	Solutore <code>interFoam</code>	33
3.1.4	Accoppiamento pressione – velocità: PISO	36
3.2	Dominio Computazionale	37
3.2.1	Geometria	37
3.2.2	Griglia di calcolo	40
3.3	Condizioni al contorno	42
3.3.1	Impostazione dei profili di velocità e <code>codedFixedValue</code>	42
3.3.2	Impostazione pressione	44
3.3.3	Impostazione del campo α	45
3.4	Studio sui numeri adimensionali	45
3.5	Condizioni operative e simulazioni effettuate	46
3.6	Tattamento delle correnti parassite	48
3.6.1	Tracciamento del centro di massa	49

4	Risultati	51
4.1	Simulazioni bidimensionali	51
4.1.1	Rottura a rapporti di viscosità elevati	51
4.1.2	Rottura a rapporti di viscosità bassi	55
4.1.3	Deformazione delle gocce	61
4.1.4	Dipendenza dal <i>mixing index</i> e campi di moto rotazionali . . .	65
4.2	Simulazioni tridimensionali	67
5	Conclusioni	69
	Appendici	71
A	Equazione di trasporto per α	73
B	Function object <code>strainRate</code>	75
C	Imposizione condizioni al contorno tramite <code>codedFixedValue</code>	79
	Lista dei simboli	83
	Bibliografia	87
	Ringraziamenti	91

Elenco delle figure

1.1	Effetto del rapporto di viscosità sulle condizioni critiche di rottura, da Grace [9]	3
1.2	Morfologia della goccia. (a) Goccia inizialmente sferica. (b) Goccia deformata sotto l'effetto di un campo di moto esterno [6]	3
1.3	Variazione del numero di capillarità critico rispetto al rapporto delle viscosità e al tipo di campo di moto. Linea continua: teoria delle piccole deformazioni; linea tratteggiata: teoria del corpo snello; cerchi verdi: $\alpha = 0.2$; triangoli verdi: $\alpha = 0.4$; rombi verdi: $\alpha = 0.6$; triangoli rovesciati verdi: $\alpha = 0.8$; stelle verdi: $\alpha = 1.0$; quadrati gialli: risultati numerici ottenuti da Rallison [15]. Figura da Suhas [6]	4
1.4	Variazione del numero di capillarità critico rispetto al rapporto delle viscosità per moto di taglio puro, da De Bruijn [16]	6
2.1	Casi particolari al variare del campo di moto	13
2.2	Campi di moto analizzati in questo studio	14
2.3	Rappresentazione di una cella tridimensionale utilizzata in un codice CFD con metodo dei volumi finiti[22]	16
2.4	Celle adiacenti per un caso monodimensionale, le lettere in maiuscolo rappresentano i centri delle celle, quelle in minuscolo le facce.[23] . . .	17
2.5	Confronto tra UDS e CDS per un flusso convettivo, relativo ad un problema presente in [23], il valore analitico (<i>'true' value</i>) corrisponde in questo caso ad una legge di potenza	19
2.6	Algoritmo SIMPLE da [22]	22
2.7	Modello di un'interfaccia fluido – fluido attraverso metodo VOF: (a) è l'interfaccia reale, (b) è la frazione volumica calcolata con il metodo VOF e (c) è la ricostruzione lineare dell'interfaccia [23]	23
2.8	Modellizzazione dell'interfaccia tra due fluidi secondo il metodo CSF, da Brackbill <i>et al.</i> [26]	26
3.1	Panoramica della struttura generale del software OpenFOAM [31] . .	30
3.2	Struttura minima di una cartella <i>case</i> in OpenFOAM [31]	31
3.3	Valori assunti dal limitatore dell'algoritmo MULES, λ_M , pari a uno vicino all'interfaccia e a zero nelle zone lontane da questa [27]	33
3.4	Dominio utilizzato per le simulazioni bidimensionali	38
3.5	Dominio utilizzato per le simulazioni tridimensionali	39
3.6	Imposizione della goccia al centro del dominio come condizione iniziale	40
3.7	Griglia di calcolo utilizzata nel caso bidimensionale	41
3.8	Distacco di materiale dalla goccia dovuto ad una rappresentazione troppo grossolana dell'interfaccia	42

3.9	<i>Contour plot</i> del <i>mixing index</i> utilizzato per confermare la correttezza del campo di moto per la condizione $MI=0.5$	43
3.10	Valori assunti da Re e Ca al variare di μ^I e σ	47
4.1	Confronto tra i dati ottenuti tramite simulazione VOF, rappresentati dai cerchi pieni, e i dati sperimentali riportati da Bentley e Leal [1], rappresentati dai cerchi vuoti	53
4.2	Indicatore di fase e vettore velocità per una goccia a deformazione stazionaria simulata in condizioni: $\lambda = 4$, moto di taglio puro, $Ca = 0.43$. Il <i>contour</i> in bianco che rappresenta la superficie della goccia è ottenuto ad $\alpha = 0.5$	54
4.3	Indicatore di fase e vettore velocità per una goccia a deformazione stazionaria simulata in condizioni: $\lambda = 4$, moto puramente elongazionale, $Ca = 0.11$. Il <i>contour</i> in bianco che rappresenta la superficie della goccia è ottenuto ad $\alpha = 0.5$	55
4.4	Morfologia della goccia. (a) Goccia inizialmente sferica. (b) Goccia deformata sotto l'effetto di un campo di moto esterno [6]	55
4.5	Confronto tra i dati ottenuti tramite simulazione VOF, rappresentati dai cerchi pieni, e i dati sperimentali riportati da Bentley e Leal [1], rappresentati dai cerchi vuoti	57
4.6	Indicatore di fase e vettore velocità per una goccia simulata alle condizioni di rottura: $\lambda = 0.01$, moto di taglio puro, $Ca = 0.58$. Il <i>contour</i> in bianco che rappresenta la superficie della goccia è ottenuto ad $\alpha = 0.5$	58
4.7	Indicatore di fase e vettore velocità per una goccia simulata alle condizioni di rottura: $\lambda = 0.01$, moto puramente elongazionale, $Ca = 0.2$. Il <i>contour</i> in bianco che rappresenta la superficie della goccia è ottenuto ad $\alpha = 0.5$	58
4.8	Indicatore di fase e vettore velocità per una goccia a deformazione stazionaria simulata in condizioni: $\lambda = 0.01$, moto di taglio puro, $Ca = 0.57$. Il <i>contour</i> in bianco che rappresenta la superficie della goccia è ottenuto ad $\alpha = 0.5$	60
4.9	Indicatore di fase e vettore velocità per una goccia a deformazione stazionaria simulata in condizioni: $\lambda = 0.01$, moto puramente elongazionale, $Ca = 0.19$. Il <i>contour</i> in bianco che rappresenta la superficie della goccia è ottenuto ad $\alpha = 0.5$	60
4.10	Confronto tra gocce deformate per $\lambda = 1$ e $Ca=0.1$ al variare del <i>mixing index</i>	61
4.11	Deformazioni D delle gocce al variare delle condizioni analizzate . . .	63
4.12	Deformazioni L/a delle gocce al variare delle condizioni analizzate . .	64
4.13	Risultati delle simulazioni VOF per campi di moto con componente elongazionale maggiore o uguale alla componente rotazionale	65
4.14	Risultati delle simulazioni VOF, sono inclusi i valori ottenuti per $MI=0.45$	67

Elenco delle tabelle

3.1	Condizioni al contorno per le simulazioni bidimensionali	44
3.2	Condizioni al contorno per le simulazioni tridimensionali	45
3.3	Proprietà dei fluidi utilizzate in tutte le simulazioni.	47
3.4	Condizioni operative simulate in questo lavoro	48
4.1	Valori di numero di capillarità critico individuati per λ pari a 4	52
4.2	Valori di numero di capillarità critico individuati per λ pari a 1	52
4.3	Valori ottenuti da Rallison [15] e confronto con i valori ottenuti mediante le simulazioni VOF per $\lambda = 1$	53
4.4	Valori di numero di capillarità critico individuati per λ pari a 0.1 . . .	56
4.5	Valori di numero di capillarità critico individuati per λ pari a 0.01 . . .	56
4.6	Differenza tra i valori critici del numero di capillarità tra condizioni di campo di moto simulate differenti	66
4.7	Risultati ottenuti dalle simulazioni VOF tridimensionali con griglia grossolana per $MI = 0.5$, confrontati con i valori provenienti dalle relazioni sperimentali di De Bruijn [16].	67

Capitolo 1

Introduzione

Il comportamento di una goccia dispersa in un fluido con cui è immiscibile è stato a lungo studiato, in quanto di interesse sia accademico che pratico.

In generale, nelle applicazioni pratiche la rottura di gocce per ottenere una dispersione più fine è utilizzata per formare emulsioni o per aumentare l'area superficiale tra le due fasi, aumentando l'entità dello scambio di materia o di energia. Esempi di queste applicazioni sono le dispersioni di agenti antistatici, dispersioni di coloranti concentrati o la combinazione di polimeri immiscibili a formare strutture bifasiche con particolari proprietà.[1] Inoltre, la distribuzione delle dimensioni delle gocce, risultante dagli eventi di rottura, determina le proprietà meccaniche e la reologia di questi sistemi dispersi. La rottura delle gocce è importante quindi per applicazioni ingegneristiche, ma anche in ambito biomedico, partendo dalla formazione di spray ed aerosol fino alla produzione di sistemi per la somministrazione di farmaci.[2], [3]

Lo studio di rottura e deformazione di gocce è di grande importanza anche per l'industria alimentare. Per quanto riguarda quest'ultima, le emulsioni sono sia utilizzate come intermedi che come prodotti finiti, basti pensare a determinate salse, come la maionese, che ricadono nella definizione di HIPE o *high internal phase emulsion*. Queste sono emulsioni che presentano un elevato volume di fase dispersa al loro interno, nelle quali la distribuzione delle dimensioni delle gocce gioca un ruolo fondamentale in proprietà come la struttura, la sensazione in bocca e il colore.[4]

1.1 Stato dell'arte

Da un punto di vista teorico, la soluzione al problema di deformazione e rottura della goccia è complessa: è necessario risolvere il campo di moto all'interno ed all'esterno della goccia, utilizzando delle condizioni al contorno sulla sua superficie. Questo è ulteriormente complicato dalla forma della goccia, che non è nota a priori e deve essere ricavata come parte della soluzione.[1]

Il primo ad interessarsi della deformazione e rottura di gocce di fluidi newtoniani è stato Taylor, il cui studio iniziale [5] è stato effettuato per due tipologie di campi di moto: il moto di taglio puro ed il moto puramente irrotazionale, detto anche moto elongazionale o *straining flow*.

Dallo studio di Taylor ci sono stati numerosi passi avanti, ma non è ancora stata raggiunta una comprensione completa del fenomeno, e ad oggi da esperimenti e risultati numerici vengono osservati nuovi meccanismi che causano la rottura delle gocce, per i quali devono essere ancora sviluppati e validati adeguati modelli.[6]

Una prima distinzione sui meccanismi di rottura agenti viene fatta in base al numero di Reynolds del sistema. Per numeri di Reynolds bassi la rottura è dovuta agli sforzi viscosi nel fluido esterno alla goccia e dipende dal numero di capillarità, dalla forma iniziale della goccia e dal rapporto delle viscosità λ . Aumentando il numero di Reynolds le forze inerziali diventano responsabili della rottura, che dipende dal rapporto delle viscosità, dal numero di Weber, dal numero di Ohnesorge, ma anche dalla forma della particella che va incontro a rottura. In caso di flusso turbolento il fenomeno è ulteriormente complicato e dipende anche dalla storia del campo di moto. Nel caso di flussi semplici, *e.g.* in assenza di turbolenza, i meccanismi di rottura sono ormai noti e compresi e le ricerche si concentrano principalmente sull'identificazione delle condizioni critiche per la rottura.[6]

Ponendo maggiore attenzione sulla rottura dovuta a meccanismi viscosi, dal precedentemente citato studio di Taylor iniziano a delinearci alcuni degli aspetti qualitativi più importanti nei processi di deformazione e rottura:[1]

1. Le gocce sferiche si deformano in sferoidi prolatti, il cui asse maggiore è inizialmente allineato con la direzione principale dello sforzo.
2. Quando la viscosità della goccia è piccola rispetto a quella del fluido che la circonda, lo *shear rate* necessario per ottenere la rottura è piuttosto elevato, la goccia raggiunge stati stazionari con forma molto deformata e estremità "appuntite".
3. Quando il rapporto di viscosità λ è grande, il comportamento della goccia è differente a seconda del tipo di campo di moto. Per il moto elongazionale, la rottura avviene a valori di *strain rate* piuttosto bassi, mentre per il moto di taglio la rottura necessita di valori sempre più elevati di *shear rate*, e la rottura risulta impossibile oltre un valore critico del rapporto di viscosità.

A causa delle differenze riscontrate al variare del rapporto di viscosità si può suddividere la trattazione in tre parti, per valori di λ alti, bassi ed intermedi.

Per rapporti di viscosità elevati, con $\lambda \sim 1$ o superiore, si parla di teoria delle piccole deformazioni. Secondo questa, la goccia in partenza sferica presenta delle deformazioni contenute al crescere del numero di capillarità Ca , quindi al crescere degli sforzi di tipo viscoso, fino al raggiungimento di un valore critico, oltre il quale non esiste una forma stazionaria della goccia e si verifica rottura.

Il primo lavoro teorico di Taylor nel 1932 [7] e il seguente sperimentale del 1934 [5] si concentrano su bassi valori di *shear rate*. I risultati sperimentali risultano in linea con quanto predetto dalla trattazione teorica, che descrive bene la deformazione, ma non fornisce un valore critico per la rottura.

Tra i lavori successivi dal punto di vista teorico, Barthes-Biesel e Acrivos nel 1973 [8] si concentrano sull'individuazione delle condizioni di rottura nel moto di taglio puro; il loro lavoro sovrastima la deformazione della goccia, ma fornisce dei valori di *shear rate* critico che sono in accordo con i lavori sperimentali [6]. Tra questi ultimi, Grace nel 1971 [9] analizza il numero di capillarità critico al variare della viscosità per moto di taglio puro, ottenendo il grafico in figura 1.1. Questo mostra l'influenza di λ sulle condizioni critiche, ovvero come il numero di capillarità critico diminuisca all'aumentare del rapporto di viscosità fino a raggiungere un minimo in un intorno di $\lambda = 1$ per poi aumentare nuovamente con un gradiente molto marcato. Si giunge quindi ad una condizione limite, individuata come $\lambda = 3.5$, oltre la quale

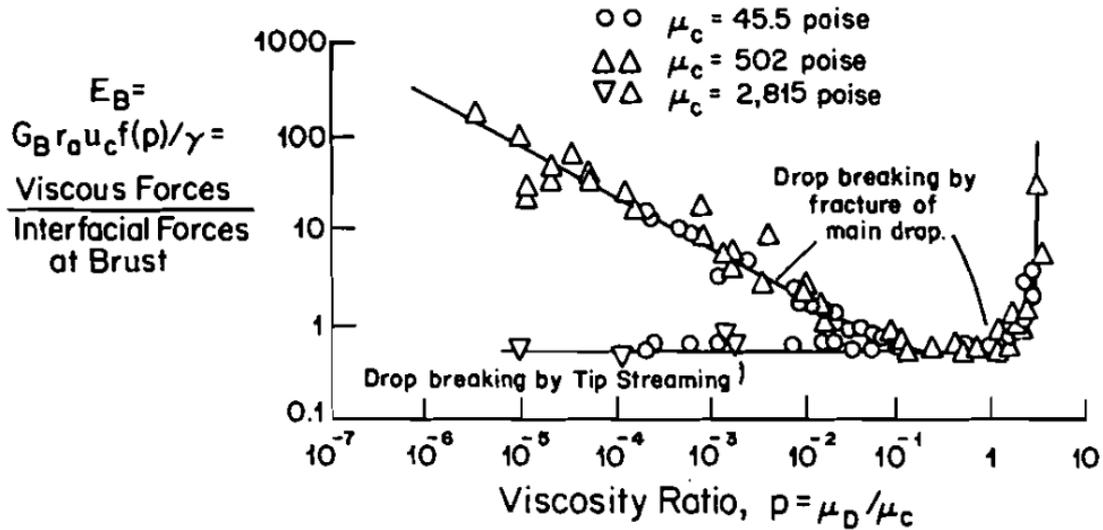


Figura 1.1: Effetto del rapporto di viscosità sulle condizioni critiche di rottura, da Grace [9]

non è possibile ottenere rottura di una goccia in un campo di moto di taglio puro aumentando lo sforzo di taglio.

Per valori molto bassi del rapporto di viscosità si fa riferimento alla teoria del corpo snello (*slender-body theory*): in questo caso la deformazione a cui va incontro la goccia è molto accentuata e questa assume una forma molto elongata, che corrisponde ad una condizione $L \gg B$, con riferimento alla figura 1.2. Anche in questo

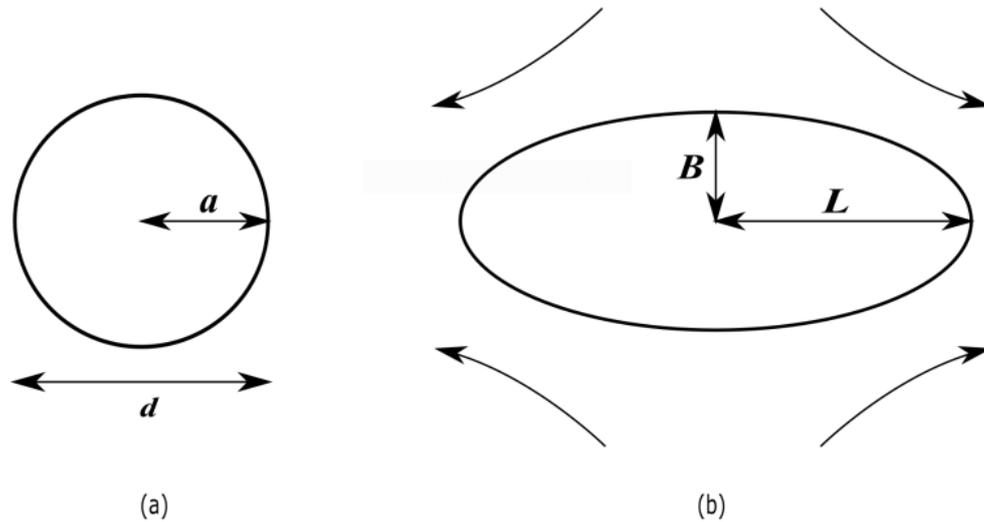


Figura 1.2: Morfologia della goccia. (a) Goccia inizialmente sferica. (b) Goccia deformata sotto l'effetto di un campo di moto esterno [6]

caso il problema venne affrontato per primo da Taylor [10], con uno studio teorico di una particella in moto puramente elongazionale, la cui presenza viene rappresentata distribuendo delle singolarità lungo l'asse della particella. Successivamente Buckmaster [11], [12] e Acrivos e Lo [13] continuarono gli studi teorici, affinando il lavoro di Taylor e investigando moto puramente elongazionale di fluidi ad elevata viscosità che contengono particelle inviscide ed a bassa viscosità. Hinch e Acrivos

[14] risolsero il problema delle gocce a bassa viscosità nel caso di moto di taglio puro.

Per quanto riguarda la rottura, questa avviene poiché l'aumento dello *shear rate* porta ad una diminuzione del raggio della sezione trasversale della goccia, fino a che la forza derivante dalla tensione superficiale non riesce più a equilibrare gli sforzi viscosi esterni. La diminuzione di sezione porta ad un aumento di L , figura 1.2, ed ad una conseguente diminuzione di pressione al centro della particella, che a sua volta porta ad un'ulteriore diminuzione del raggio della sezione trasversale. Poiché questo processo non può continuare indefinitamente, al raggiungimento di un valore di *shear rate* sufficientemente elevato la goccia si rompe.[6]

Nel caso di valori intermedi del rapporto di viscosità c'è una transizione non netta tra il comportamento riscontrato secondo la teoria del corpo snello e quello spiegato dalla teoria delle piccole deformazioni.

Il lavoro di Bentley e Leal [1] studia un ampio intervallo di valori di λ , e tiene in considerazione anche condizioni per il campo di moto intermedie tra il moto di taglio puro ed il moto puramente elongazionale. Per quanto riguarda la rottura delle gocce i risultati dello studio sono riassunti in figura 1.3. La caratterizzazione del campo

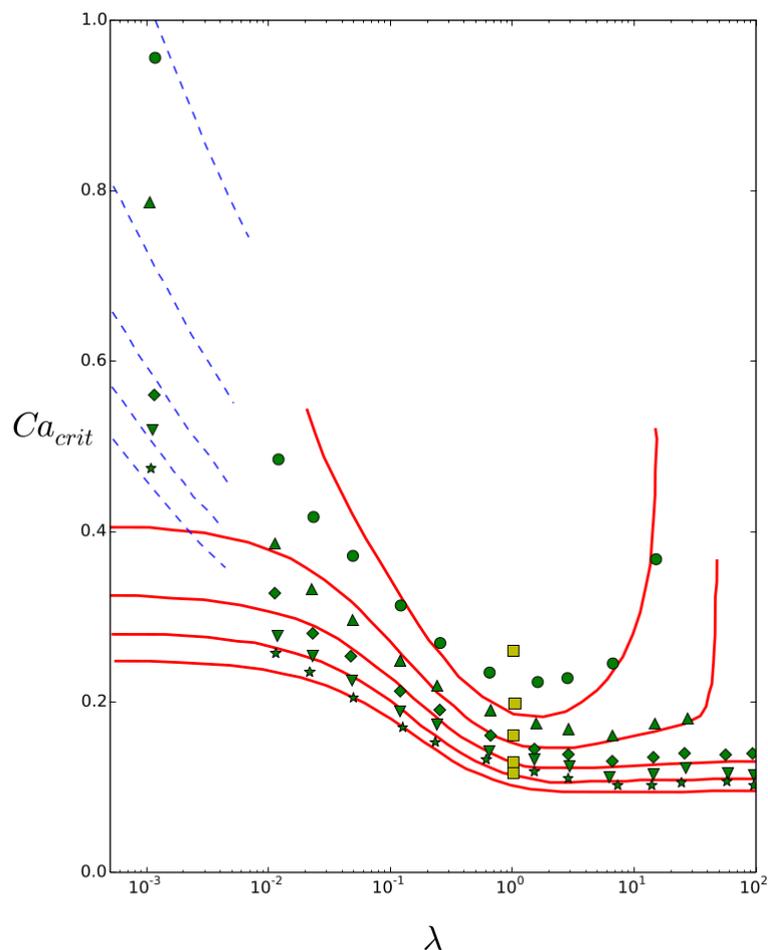


Figura 1.3: Variazione del numero di capillarità critico rispetto al rapporto delle viscosità e al tipo di campo di moto. Linea continua: teoria delle piccole deformazioni; linea tratteggiata: teoria del corpo snello; cerchi verdi: $\alpha = 0.2$; triangoli verdi: $\alpha = 0.4$; rombi verdi: $\alpha = 0.6$; triangoli rovesciati verdi: $\alpha = 0.8$; stelle verdi: $\alpha = 1.0$; quadrati gialli: risultati numerici ottenuti da Rallison [15]. Figura da Suhas [6]

di moto avviene attraverso il parametro α , che ha valore nullo nel caso di moto di taglio puro e valore unitario per il moto puramente elongazionale. Come si nota dalla figura 1.3, la dipendenza del valore del numero di capillarità critico è differente per valori differenti di α : quando il campo di moto ha una buona componente rotazionale, ed è quindi più simile a un moto di taglio, è presente il minimo già visto sperimentalmente da Grace [9]; diminuendo la componente rotazionale del campo di moto, tendendo verso un moto elongazionale, il minimo risulta sempre meno accentuato, fino a scomparire lasciando il posto ad un *plateau*. In figura 1.3 sono anche presenti i risultati numerici ottenuti da Rallison [15], in cui i vari campi di moto vengono analizzati per un valore di rapporto di viscosità pari a 1.

In generale si può riassumere quanto ritrovato da Bentley e Leal nei seguenti punti [6]:

- Per $\lambda < 0.02$ le estremità della particella diventano appuntite, più alto è il numero di capillarità maggiore è la deformazione elongazionale della particella. Per ottenere rottura a questi valori del rapporto di viscosità sono necessari deformazioni molto elevate e valori del numero di capillarità molto alti.
- Per $\lambda > 0.02$ le estremità della particella rimangono arrotondate fino alla rottura. Al crescere del rapporto di viscosità diminuiscono il numero di capillarità critico ed il valore della massima deformazione stabile.
- Per $\lambda > 3$ la parte centrale della particella presenta forma cilindrica. Esiste un valore del rapporto di viscosità oltre il quale la rottura diviene impossibile e la deformazione non aumenta all'aumentare del numero di capillarità.

De Bruijn [16] ripete gli esperimenti per individuare il numero di capillarità critico in condizioni di moto di taglio puro, ottenendo i risultati riportati in figura 1.4, in linea con gli esperimenti effettuati da Grace. Dai dati ottenuti De Bruijn [16] ottiene una correlazione con *fitting* a cinque parametri che esprime il numero di capillarità critico in funzione del rapporto di viscosità per valori del rapporto di viscosità elevato:

$$\log \text{Ca}_{\text{crit}} = -0.506 + 0.0994 \log \lambda + 0.124 \log \lambda^2 + \frac{-0.115}{\log \lambda - \log \lambda_{\text{max}}}, \quad (1.1)$$

con $\lambda_{\text{max}} = 4.08$, ed una correlazione con *fitting* a quattro parametri che descrive cosa accade per bassi valori del rapporto di viscosità:

$$\log \text{Ca}_{\text{crit}} = -1.560 - 0.733 \log \lambda + \frac{-1.135}{\log \lambda - \log \lambda_{\text{max}}}, \quad (1.2)$$

con $\lambda_{\text{max}} = 9.27$.

Questi esperimenti confermano come la curva del numero di capillarità tenda ad infinito quando il rapporto di viscosità ha un valore superiore a quattro: non è possibile ottenere rottura in moto di taglio puro oltre questo valore limite.

La trattazione fin qui riportata fa riferimento allo studio di campi di moto stazionari, fluidi newtoniani e regime fluidodinamico in campo laminare con numeri di Reynolds estremamente bassi, ovvero le condizioni esplorate in questo lavoro. Tuttavia gli studi della rottura di gocce spaziano molto, prendendo in considerazione regimi fluidodinamici differenti e campi di moto non stazionari, ma anche l'influenza della presenza di tensioattivi. Per una trattazione più estesa si rimanda ai lavori di Rallison [17], De Bruijn [16], Stone [18] e Suhas [6].

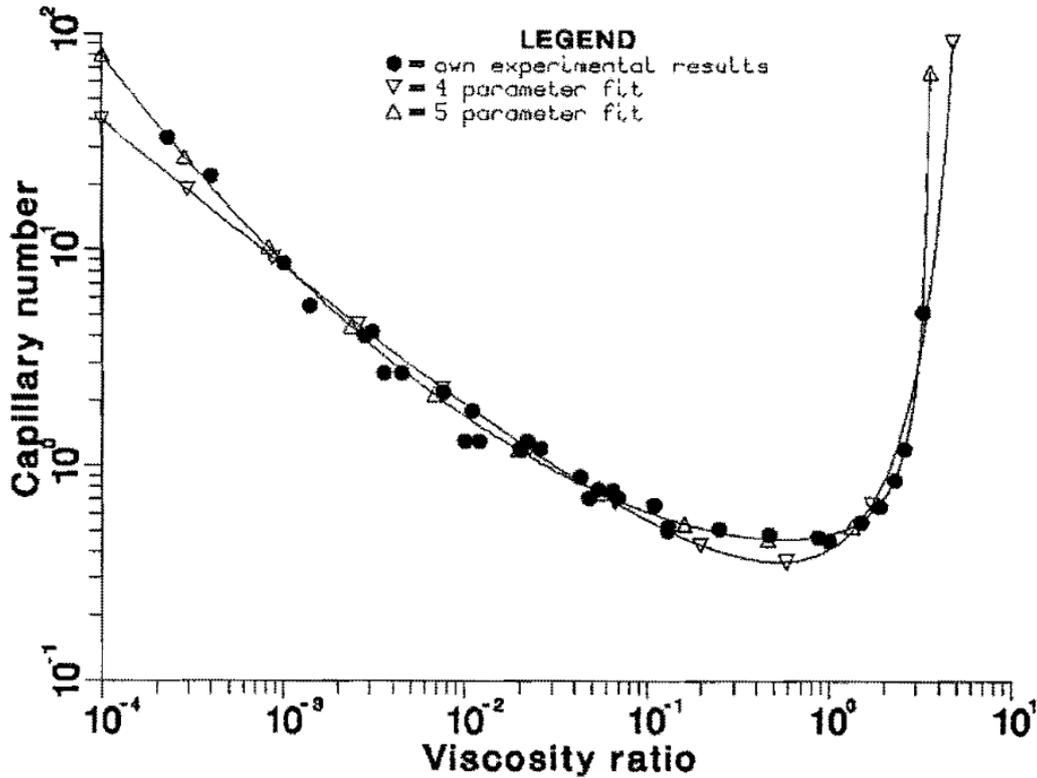


Figura 1.4: Variazione del numero di capillarità critico rispetto al rapporto delle viscosità per moto di taglio puro, da De Bruijn [16]

1.2 Obiettivo del lavoro di tesi

Come già affermato nella sezione precedente, la maggior parte dei nuovi studi sulla rottura delle gocce in campi di moto semplici è focalizzata sulla ricerca delle condizioni critiche. L'ottenimento di valori accurati del numero di capillarità critico e, per quanto possibile, di relazioni analitiche ricavate da questi valori è utile in quanto permette lo sviluppo di migliori modelli predittivi nel campo delle emulsioni.

Nei sistemi dispersi la distribuzione delle dimensioni delle gocce deve essere accuratamente controllata, in modo da ottenere un prodotto finale che rispetti le caratteristiche desiderate. Per fare questo, un approccio sperimentale utilizza varie prove sulle condizioni operative dei macchinari utilizzati, in modo da individuare il miglior set di impostazioni da utilizzare per il prodotto desiderato.

Un approccio differente punta sull'utilizzo della fluidodinamica computazionale, o CFD, per la simulazione di fluidi dispersi. A questo scopo, l'utilizzo di modelli euleriani bi-fluido è la scelta che permette simulazioni con costi computazionali accettabili per i domini delle applicazioni industriali. I molti tentativi effettuati non hanno però mostrato una capacità predittiva soddisfacente.[6] Poiché la distribuzione delle dimensioni delle gocce è dipendente dai processi di rottura e coalescenza, una maggiore comprensione di questi fenomeni darebbe la possibilità di implementarli in maniera più aderente alla realtà, rendendo le simulazioni CFD uno strumento utile nell'ottimizzazione delle condizioni operative utilizzate in ambito industriale.

Il presente lavoro si pone nel contesto descritto, partendo dall'idea di individuare le condizioni critiche tramite CFD. In particolare, i dati da letteratura presentati

in sezione 1.1 sono confrontati con quelli ricavati da simulazioni CFD che utilizzano il metodo VOF (*volume of fluid*), in cui vengono monitorate le condizioni di una singola goccia immersa in un secondo fluido. L'obiettivo è quindi la verifica delle prestazioni del metodo VOF riguardo al problema di rottura della goccia, in modo da ricavare i valori del numero di capillarità critico al variare del rapporto di viscosità e del tipo di campo di moto.

1.3 Struttura del lavoro

Il lavoro è suddiviso in quattro capitoli, il primo, appena presentato, introduce l'importanza del problema studiato e i risultati ottenuti dagli studi precedenti presenti in letteratura. Il secondo capitolo spiega l'impostazione teorica del problema e della sua risoluzione attraverso i metodi di fluidodinamica computazionale. Il terzo capitolo descrive i dettagli del software utilizzato e come avviene l'impostazione del problema, soffermandosi sugli aspetti numerici principali e accennando ad alcuni problemi che sorgono inerenti al metodo computazionale utilizzato. Il quarto capitolo presenta i risultati ottenuti, confrontandoli con quelli già presenti in letteratura. Infine sono presentate le conclusioni.

Capitolo 2

Teoria e metodi

In questo capitolo si espongono i concetti teorici relativi al problema della modellizzazione fluidodinamica di una goccia e alla sua risoluzione. Inizialmente, nella sezione 2.1, si analizzano le equazioni che descrivono la fisica del sistema e i parametri d'interesse risultanti da queste. Successivamente, nella sezione 2.2, è presentata una panoramica teorica sui metodi numerici di discretizzazione e risoluzione delle equazioni di un problema di fluidodinamica computazionale. Infine, nella sezione 2.3, è esposto il metodo di tracciamento dell'interfaccia utilizzato in questo lavoro, di centrale importanza per la simulazione di problemi multifase in cui l'interfaccia è una superficie libera (condizione *free boundary*), analizzandone anche i punti di forza e le debolezze.

2.1 Equazioni fondamentali

Le equazioni fondamentali presentate sono derivanti dallo studio dei fenomeni di trasporto e dalle relative equazioni di bilancio alle varie proprietà, alle quali vengono applicate le ipotesi semplificative tipiche del sistema in esame.

2.1.1 Ipotesi

Il sistema multifase simulato prevede una goccia di fluido dispersa in un secondo fluido, e presenta le seguenti ipotesi:

- è valida l'ipotesi del continuo in quanto lo spazio disponibile al moto del fluido è molto maggiore rispetto al cammino libero medio delle molecole;
- entrambi i fluidi sono incomprimibili e con densità costante;
- entrambi i fluidi sono newtoniani, la loro viscosità resta costante nel corso della simulazione;
- viene trascurato l'effetto della gravità, in quanto porterebbe ad un moto della goccia dovuto alla differenza di densità, mentre nel caso in esame è fondamentale che la goccia rimanga centrata nel dominio, in modo da rispettare i vincoli sul campo di moto imposto;
- i due fluidi sono immiscibili, non avviene scambio di materia tra loro;

- la tensione interfacciale è costante;
- il sistema è isoterma;

2.1.2 Equazione di continuità

L'equazione di continuità esprime la conservazione della massa, viene quindi ricavata partendo da un bilancio di massa su un volume di controllo infinitesimo, e descrive la variazione della densità del fluido in punto dello spazio.[19] La sua forma generale in notazione indiciale è:

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho U_i)}{\partial x_i} = 0, \quad (2.1)$$

dove ρ è la densità del fluido (kg m^{-3}) e U_i è la componente i -esima del vettore velocità riferito al fluido (m s^{-1}).

Sotto l'assunzione di fluido incompressibile, poichè la densità è costante, l'equazione assume una forma più semplice:

$$\frac{\partial U_i}{\partial x_i} = 0. \quad (2.2)$$

In questa trattazione è adottato un sistema di riferimento cartesiano, per cui l'equazione 2.2 diventa:

$$\frac{\partial U_x}{\partial x} + \frac{\partial U_y}{\partial y} + \frac{\partial U_z}{\partial z} = 0. \quad (2.3)$$

2.1.3 Equazione di Navier – Stokes

Effettuando un bilancio di quantità di moto su un volume di controllo infinitesimo, che esprime la conservazione della suddetta proprietà, si ricava l'equazione del moto di un fluido [19]:

$$\frac{\partial \rho U_i}{\partial t} = -\frac{\partial}{\partial x_j}(\rho U_i U_j) - \frac{\partial p}{\partial x_i} - \frac{\partial \tau_{ij}}{\partial x_j} + \rho g_i, \quad (2.4)$$

in cui p (Pa) è la pressione nel fluido, ρ (kg m^{-3}) la densità del fluido e τ_{ij} (Pa) la componente in posizione (i, j) nel tensore degli sforzi viscosi.

Sotto le ipotesi di fluido incompressibile e newtoniano, e quindi densità e viscosità costanti, dal bilancio si ottiene l'equazione di Navier – Stokes:

$$\frac{\partial U_i}{\partial t} + U_j \frac{\partial U_i}{\partial x_j} = -\frac{1}{\rho} \frac{\partial p}{\partial x_i} + \nu \frac{\partial^2 U_i}{\partial x_j^2}, \quad (2.5)$$

dove ν ($\text{m}^2 \text{s}^{-1}$) è la viscosità cinematica del fluido. Inoltre nell'equazione 2.5 è assente l'ultimo termine dell'equazione 2.4, poichè la forza di gravità è assunta trascurabile nel sistema studiato e pertanto l'accelerazione di gravità \mathbf{g} (m s^{-2}) viene considerata nulla.

Analizzando gli altri elementi dell'equazione 2.5, al primo membro sono presenti l'accumulo e il flusso relativo al trasporto convettivo di quantità di moto, mentre al secondo membro si trovano le forze di pressione e il flusso legato al trasporto viscoso della quantità di moto.

In un sistema bifasico i due fluidi sono separati dall'interfaccia, considerata secondo il modello della membrana di Young come una superficie liscia e omogenea, che ha spessore nullo. Risulta necessario allora risolvere un'equazione di Navier – Stokes per ogni fluido

$$\begin{aligned}\frac{\partial U_i^{\text{II}}}{\partial t} + U_j^{\text{II}} \frac{\partial U_i^{\text{II}}}{\partial x_j} &= -\frac{1}{\rho^{\text{II}}} \frac{\partial p^{\text{II}}}{\partial x_i} + \nu^{\text{II}} \frac{\partial^2 U_i^{\text{II}}}{\partial x_j^2}, \\ \frac{\partial U_i^{\text{I}}}{\partial t} + U_j^{\text{I}} \frac{\partial U_i^{\text{I}}}{\partial x_j} &= -\frac{1}{\rho^{\text{I}}} \frac{\partial p^{\text{I}}}{\partial x_i} + \nu^{\text{I}} \frac{\partial^2 U_i^{\text{I}}}{\partial x_j^2},\end{aligned}\tag{2.6}$$

in cui gli apici indicano a quale dei due fluidi si riferisce una data proprietà.

Secondo lo stesso modello, in corrispondenza dell'interfaccia agisce una forza per unità di lunghezza, detta tensione interfacciale σ (N m^{-1}), che quantifica la tendenza “a contrarsi” tipica delle superfici fluide: questa permette ai fluidi di assumere una forma che minimizzi il rapporto superficie su volume, e di conseguenza l'energia del sistema, in quanto le molecole in corrispondenza di una discontinuità risultano essere più energetiche [20]. Date le sue caratteristiche, la tensione interfacciale non può direttamente rientrare nell'equazione di Navier – Stokes, in quanto questa deriva da un bilancio su un volume, quindi in problemi bifasici viene introdotta tramite le condizioni al contorno, che derivano da un bilancio di quantità di moto su un elemento infinitesimo di superficie [19]. La condizione in direzione normale all'interfaccia è:

$$n_i^{\text{II}}(p^{\text{II}} - p^{\text{I}}) + n_i^{\text{II}}(\tau_{ij}^{\text{II}} - \tau_{ij}^{\text{I}}) = n_i^{\text{II}} \left(\frac{1}{R_1} + \frac{1}{R_2} \right) \sigma,\tag{2.7}$$

in cui n_i è la componente i -esima del versore normale alla superficie infinitesima, con verso uscente dal fluido II, R_1 ed R_2 sono i raggi di curvatura principali, con segno positivo se i loro centri sono situati all'interno del fluido II.

Mentre le condizioni al contorno nelle direzioni tangenti all'interfaccia sono:

$$\begin{aligned}t_i^\alpha (\tau_{ij}^{\text{I}} - \tau_{ij}^{\text{II}}) &= 0 \\ t_i^\beta (\tau_{ij}^{\text{I}} - \tau_{ij}^{\text{II}}) &= 0\end{aligned},\tag{2.8}$$

in cui t_i^α e t_i^β sono le componenti i -esime dei vettori tangenti alla superficie infinitesima, i quali sono perpendicolari tra loro.

La curvatura può essere espressa in un modo alternativo, come mostrato nella seguente formula [19]:

$$\left(\frac{1}{R_1} + \frac{1}{R_2} \right) = \frac{\partial n_i^{\text{II}}}{\partial x_i}.\tag{2.9}$$

2.1.4 Campo di moto e *mixing index*

Come traspare dalla trattazione precedente, il processo di rottura è fortemente influenzato dal tipo di campo di moto sviluppato all'interno del sistema, che viene analizzato in questa sezione, partendo dalla velocità nel fluido fino a definire il *mixing index*, attraverso il quale è possibile identificare il campo di moto basandosi sulla preponderanza della componente rotazionale o di quella elongazionale.

Considerando un sistema di riferimento cartesiano, è possibile definire il gradiente della velocità in un punto all'interno del fluido tramite il prodotto diadico tra l'operatore ∇ e il vettore velocità \mathbf{U} :

$$\nabla \mathbf{U} = \begin{bmatrix} \frac{\partial U_x}{\partial x} & \frac{\partial U_y}{\partial x} & \frac{\partial U_z}{\partial x} \\ \frac{\partial U_x}{\partial y} & \frac{\partial U_y}{\partial y} & \frac{\partial U_z}{\partial y} \\ \frac{\partial U_x}{\partial z} & \frac{\partial U_y}{\partial z} & \frac{\partial U_z}{\partial z} \end{bmatrix}. \quad (2.10)$$

Il tensore risultante può essere scomposto in una componente simmetrica, detta tensore delle deformazioni $\dot{\boldsymbol{\gamma}}$ e una componente antisimmetrica (o emisimmetrica), detta tensore della vorticità $\boldsymbol{\omega}$.

Il tensore delle deformazioni esprime la velocità di deformazione, tenendo in considerazione, gli scorrimenti a cui va incontro il volume di fluido per effetto del campo di moto, e non le rotazioni; la sua espressione è la seguente:

$$\dot{\boldsymbol{\gamma}} = \frac{1}{2}(\nabla \mathbf{U} + (\nabla \mathbf{U})^T) = \frac{1}{2} \begin{bmatrix} 2\frac{\partial U_x}{\partial x} & \frac{\partial U_y}{\partial x} + \frac{\partial U_x}{\partial y} & \frac{\partial U_z}{\partial x} + \frac{\partial U_x}{\partial z} \\ \frac{\partial U_x}{\partial y} + \frac{\partial U_y}{\partial x} & 2\frac{\partial U_y}{\partial y} & \frac{\partial U_z}{\partial y} + \frac{\partial U_y}{\partial z} \\ \frac{\partial U_x}{\partial z} + \frac{\partial U_z}{\partial x} & \frac{\partial U_y}{\partial z} + \frac{\partial U_z}{\partial y} & 2\frac{\partial U_z}{\partial z} \end{bmatrix}, \quad (2.11)$$

dove l'apice T, indica il tensore trasposto. Dal modulo del tensore delle deformazioni, si ottiene lo *strain rate*, calcolabile con la seguente formula:

$$\dot{\boldsymbol{\gamma}} = \sqrt{2\dot{\boldsymbol{\gamma}} : \dot{\boldsymbol{\gamma}}} = \sqrt{2\dot{\gamma}_{ij}\dot{\gamma}_{ij}}. \quad (2.12)$$

Nella formula precedente il simbolo “:” indica un prodotto interno tra due tensori, che restituisce uno scalare e per questo in alcuni casi è chiamato prodotto scalare tra due tensori [19].

La componente antisimmetrica del gradiente della velocità esprime la velocità di rotazione, ed è anche chiamata tensore della vorticità:

$$\boldsymbol{\omega} = \frac{1}{2}(\nabla \mathbf{U} - (\nabla \mathbf{U})^T) = \frac{1}{2} \begin{bmatrix} 0 & \frac{\partial U_y}{\partial x} - \frac{\partial U_x}{\partial y} & \frac{\partial U_z}{\partial x} - \frac{\partial U_x}{\partial z} \\ \frac{\partial U_x}{\partial y} - \frac{\partial U_y}{\partial x} & 0 & \frac{\partial U_z}{\partial y} - \frac{\partial U_y}{\partial z} \\ \frac{\partial U_x}{\partial z} - \frac{\partial U_z}{\partial x} & \frac{\partial U_y}{\partial z} - \frac{\partial U_z}{\partial y} & 0 \end{bmatrix} \quad (2.13)$$

Ripetendo quanto già visto per il tensore dello *strain rate*, calcolando il modulo del tensore in equazione 2.13 attraverso il prodotto interno, si ottiene la vorticità $\tilde{\omega}$:

$$\tilde{\omega} = \sqrt{2\boldsymbol{\omega} : \boldsymbol{\omega}} = \sqrt{2\omega_{ij}\omega_{ij}}. \quad (2.14)$$

Quest'ultima è anche ottenibile calcolando il modulo del vettore risultante dall'applicazione del rotore al vettore velocità:

$$\tilde{\omega} = \|\nabla \times \mathbf{U}\|. \quad (2.15)$$

A questo punto è possibile passare alla definizione di mixing index, il cui valore dipende proprio dallo *strain rate* e dalla vorticità:

$$\text{MI} = \frac{\dot{\boldsymbol{\gamma}} + \tilde{\omega}}{\dot{\boldsymbol{\gamma}}} \quad (2.16)$$

Questo parametro permette di distinguere le varie possibilità per il campo di moto, tra le quali identifichiamo tre casi particolari, mostrati in figura 2.1:

- campo di moto puramente rotazionale in figura 2.1a, in cui $\dot{\gamma}$ e quindi la componente elongazionale del campo di moto sono nulli, ed il *mixing index* conseguentemente vale 0;
- moto di taglio puro in figura 2.1b, o *shear flow*, in cui $\dot{\gamma}$ e $\tilde{\omega}$ hanno lo stesso valore, quindi il peso della componente elongazionale equivale quello della componente rotazionale ed il *mixing index* vale 0.5;
- moto puramente elongazionale, figura 2.1c, in cui non c'è componente rotazionale, $\tilde{\omega}$ vale zero e il *mixing index* ha valore unitario.

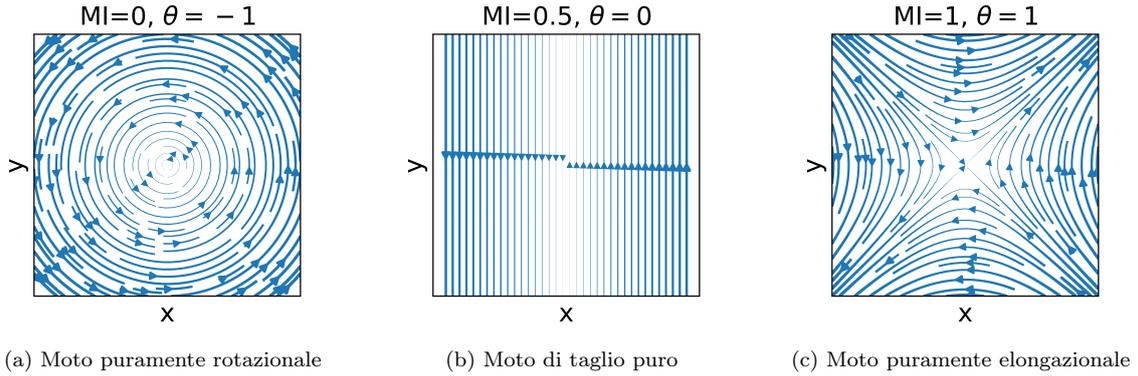


Figura 2.1: Casi particolari al variare del campo di moto

In questa trattazione per impostare il campo di moto viene utilizzato il parametro θ , legato al *mixing index* secondo la seguente equazione:

$$\theta = 2MI - 1, \quad (2.17)$$

che permette una semplice implementazione del tipo di campo di moto partendo dalla definizione del gradiente:

$$\nabla \mathbf{U} = \begin{bmatrix} \frac{\partial U_x}{\partial x} & \frac{\partial U_y}{\partial x} & \frac{\partial U_z}{\partial x} \\ \frac{\partial U_x}{\partial y} & \frac{\partial U_y}{\partial y} & \frac{\partial U_z}{\partial y} \\ \frac{\partial U_x}{\partial z} & \frac{\partial U_y}{\partial z} & \frac{\partial U_z}{\partial z} \end{bmatrix} = G \begin{bmatrix} 0 & 1 & 0 \\ \theta & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad (2.18)$$

dove il parametro G (s^{-1}) è espressione dell'intensità dei gradienti di velocità, e può essere messo in relazione con lo *strain rate* utilizzando l'equazione 2.12, ottenendo:

$$\dot{\gamma} = G(1 + \theta). \quad (2.19)$$

Lo studio presentato in questo lavoro si concentra principalmente su valori di *mixing index* compresi tra 0.5 e 1, come riportato in figura 2.2, quindi su campi di moto in cui la componente rotazionale è in un caso limite nulla, ma comunque sempre minore o uguale a quella elongazionale. Unica eccezione è $MI = 0.45$, studiato per confronto con gli altri valori del *mixing index* in modo da confermare gli andamenti riscontrati.

Questa scelta è dettata dalle condizioni che si presentano principalmente in ambito dell'industria alimentare, all'interno degli emulsificatori.

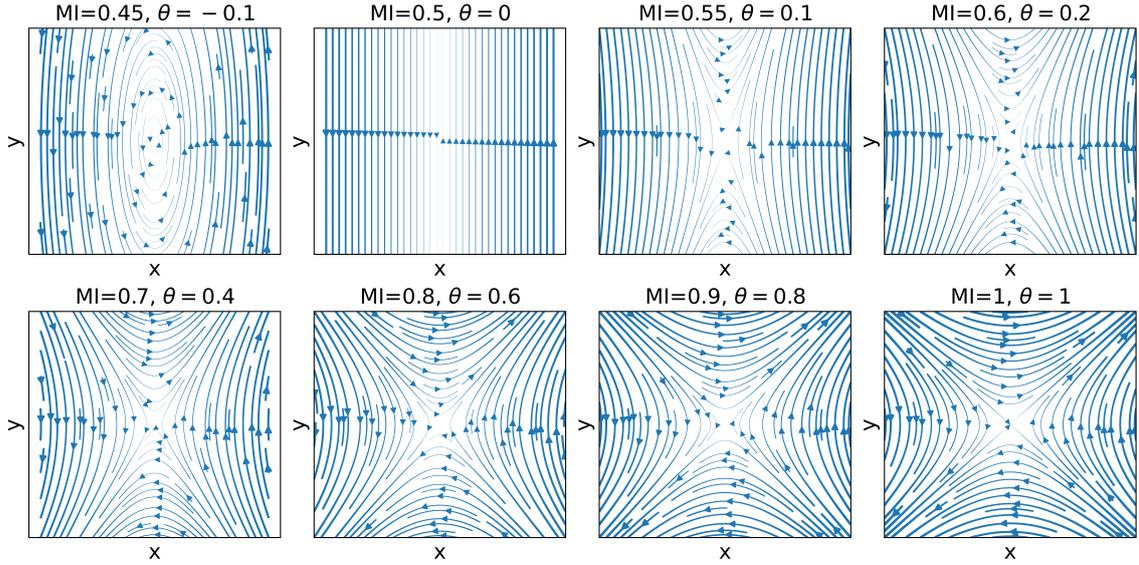


Figura 2.2: Campi di moto analizzati in questo studio

2.1.5 Numeri adimensionali

Dall'adimensionalizzazione delle equazioni che descrivono il problema, in sezione 2.1.3, è possibile ricavare dei gruppi adimensionali caratterizzanti nello studio della rottura di gocce di un fluido II immerso in un fluido I [1], [6], [19]. In questa trattazione, acquisiscono particolare importanza:

$$\text{Re} = \frac{\rho^I U^I a}{\mu^I} = \frac{\rho^I (Ga) a}{\mu^I} = \text{numero di Reynolds}, \quad (2.20)$$

$$\text{We} = \frac{\mu^I (U^I)^2 a}{\sigma} = \frac{\mu^I (Ga)^2 a}{\sigma} = \text{numero di Weber}, \quad (2.21)$$

$$\text{Ca} = \frac{\mu^I U^I}{\sigma} = \frac{\mu^I Ga}{\sigma} = \text{We} \cdot \text{Re}^{-1} = \text{numero di capillarità}, \quad (2.22)$$

$$\lambda = \frac{\mu^{II}}{\mu^I} = \text{rapporto di viscosità}, \quad (2.23)$$

dove a (m) è la dimensione caratteristica, pari al raggio della goccia non deformata, $\mu = \rho\nu$ (Pa·s) è la viscosità dinamica e la U^I è una velocità caratteristica del fluido I.

Questi numeri adimensionali danno idea dell'importanza relativa delle forze presenti nel sistema: il numero di Reynolds mette a rapporto le forze inerziali su quelle viscosi, il numero di Weber indica la relativa importanza delle forze inerziali rispetto a quelle derivanti dalla tensione interfacciale e il numero di capillarità raffronta le forze viscosi rispetto a quelle legate alla tensione superficiale.

Come riportato in letteratura (sezione 1.1), i suddetti numeri sono fondamentali nella descrizione della dinamica del sistema, e permettono di indentificare le condizioni alle quali avviene la rottura.

Le condizioni analizzate in questo lavoro si concentrano nei casi in cui la rottura è causata dalle forze viscosi: i valori dei numeri di Reynolds e Weber sono molto bassi ($\ll 1$) e la rottura è governata principalmente dal numero di capillarità (equazione 2.22) e dal rapporto delle viscosità (equazione 2.23) [6].

Si identifica quindi per le varie condizioni operative, ovvero al variare di λ e del *mixing index*, un numero di capillarità a cui si verifica rottura, chiamato numero di capillarità critico Ca_{crit} ; a questo valore le forze viscosse che tendono a rompere la goccia prevalgono rispetto alle forze legate alla tensione interfacciale, che si oppongono alla rottura.

2.2 Metodo dei volumi finiti

Le equazioni presentate nella sezione 2.1 vengono implementate in un codice di fluidodinamica computazionale, o CFD, per essere risolte numericamente. Per fare ciò è possibile utilizzare differenti metodi numerici, tra i più comuni in uso in ambito CFD ci sono il metodo delle differenze finite, il metodo degli elementi finiti ed il metodo dei volumi finiti. Quest'ultimo è utilizzato in questo lavoro e prevede, partendo dalla generica equazione del trasporto [21]:

$$\frac{\partial(\rho\phi)}{\partial t} + \nabla \cdot (\rho\phi\mathbf{U}) = \nabla \cdot (\Gamma\nabla\phi) + q_\phi, \quad (2.24)$$

la sua integrazione su un volume di controllo Ω

$$\int_{\Omega} \frac{\partial(\rho\phi)}{\partial t} dV + \int_{\Omega} \nabla \cdot (\rho\phi\mathbf{U}) dV = \int_{\Omega} \nabla \cdot (\Gamma\nabla\phi) dV + \int_{\Omega} q_\phi dV, \quad (2.25)$$

dove ρ è la densità del fluido, ϕ è la proprietà trasportata e Γ è il coefficiente di trasporto molecolare.

Successivamente, viene applicato il teorema di Gauss per i termini di trasporto diffusivo e convettivo, che diventano integrali sulla superficie \mathcal{S} che racchiude il volume di controllo Ω :

$$\int_{\Omega} \frac{\partial(\rho\phi)}{\partial t} dV + \int_{\mathcal{S}} \rho\phi\mathbf{U} \cdot \mathbf{n} dS = \int_{\mathcal{S}} \Gamma(\nabla\phi) \cdot \mathbf{n} dS + \int_{\Omega} q_\phi dV, \quad (2.26)$$

dove \mathbf{n} è il versore normale alla superficie dS .

A seconda dei valori di ϕ , Γ e del termine di generazione q_ϕ si riproducono le equazioni precedentemente esaminate in sezione 2.1.

Per il metodo dei volumi finiti, il dominio viene suddiviso tramite una griglia in celle, ognuna delle quali è un volume di controllo su cui viene applicata l'equazione 2.26, permettendo la conservazione della data proprietà a livello del suddetto volume di controllo e di conseguenza su tutto il dominio. Dalla somma di tutte le equazioni su tutte le celle infatti si ottiene l'equazione di conservazione globale[21].

In figura 2.3 è rappresentata una cella tridimensionale utilizzata nel metodo dei volumi finiti: si possono identificare il nodo al centro della cella in esame P e i nodi delle celle adiacenti *east*, *west*, *north*, *south*, *top* e *bottom* (E , W , N , S , T , B). Con la stessa nomenclatura si identificano le facce che separano la cella in esame dalle adiacenti, utilizzando le lettere minuscole e , w , n , s , t e b .

Partendo dalla cella appena descritta si può analizzare la riformulazione dei termini dell'equazione 2.26 a seguito della discretizzazione.

Il termine convettivo rappresenta il flusso di proprietà ϕ trasportato dal moto macroscopico del fluido ed è il secondo termine nell'equazione 2.26. Il prodotto scalare $\mathbf{U} \cdot \mathbf{n}$ è equivalente alla componente della velocità perpendicolare a dS ; il fluido può

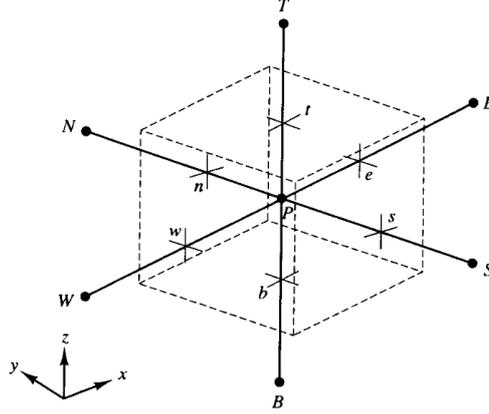


Figura 2.3: Rappresentazione di una cella tridimensionale utilizzata in un codice CFD con metodo dei volumi finiti[22]

entrare o uscire dal volume di controllo attraversando una delle facce in figura 2.3 e di conseguenza l'integrale viene riscritto attraverso lo seguente sommatoria:

$$\int_S \rho \phi \mathbf{U} \cdot \mathbf{n} dS = -\rho [(SU_x \phi)_w - (SU_x \phi)_e + (SU_y \phi)_s - (SU_y \phi)_n + (SU_z \phi)_b - (SU_z \phi)_t], \quad (2.27)$$

dove S è la superficie di ognuna delle sei facce e la ρ viene considerata costante, pertanto raccolta a factor comune al di fuori della sommatoria.

I segni negativi nell'equazione 2.27 derivano dalla convenzione dei segni dei versori normali alle superfici delle facce [23], che permettono di considerare il contributo della convezione negativo se il flusso è entrante nel volume di controllo, anche nel caso in cui la velocità sia concorde con gli assi del sistema di riferimento, quindi di segno positivo.

Il terzo termine nell'equazione 2.26 si riferisce al trasporto diffusivo, dovuto a meccanismi molecolari e dipendente dal gradiente della proprietà trasportata ϕ . È possibile un trattamento analogo a quanto visto per il termine convettivo:

$$\int_S \Gamma (\nabla \phi) \cdot \mathbf{n} dS = - \left[\left(S \Gamma \frac{\partial \phi}{\partial x} \right)_w - \left(S \Gamma \frac{\partial \phi}{\partial x} \right)_e + \left(S \Gamma \frac{\partial \phi}{\partial y} \right)_s - \left(S \Gamma \frac{\partial \phi}{\partial y} \right)_n + \left(S \Gamma \frac{\partial \phi}{\partial z} \right)_b - \left(S \Gamma \frac{\partial \phi}{\partial z} \right)_t \right]. \quad (2.28)$$

L'ultimo termine dell'equazione 2.26 è quello di generazione, che tiene in considerazione i fenomeni che portano alla produzione o alla dissipazione di ϕ . Esempi di questo termine sono le forze di volume, come la forza di gravità, e il gradiente di pressione, entrambi visti nell'equazione di Navier – Stokes [23]. Il termine sorgente viene semplicemente espresso come il prodotto del volume di controllo per il valore medio di q_ϕ all'interno dello stesso volume:

$$\int_\Omega q_\phi dV = \bar{q}_\phi \Omega, \quad (2.29)$$

dove \bar{q}_ϕ è proprio il valore medio della generazione di proprietà per unità di volume all'interno del volume di controllo.

2.2.1 Metodi di discretizzazione spaziale

Il metodo dei volumi finiti si basa sul calcolo del valore delle proprietà del fluido nei centri delle varie celle, che viene tuttavia effettuato a partire dai valori delle proprietà sulle facce. Questi ultimi non sono però noti e per risolvere le equazioni discretizzate è necessario interpolare i valori di U_i , ϕ , Γ e del gradiente di ϕ tra celle adiacenti, in modo da calcolare quanto questi valgano sulla faccia in comune alle due celle.

Qui vengono descritti alcuni metodi per l'interpolazione, per dare un'idea del processo di risoluzione attraverso una trattazione generale.



Figura 2.4: Celle adiacenti per un caso monodimensionale, le lettere in maiuscolo rappresentano i centri delle celle, quelle in minuscolo le facce.[23]

Considerando il sistema monodimensionale mostrato in figura 2.4, l'analisi, che può essere estesa per sistemi bidimensionali e tridimensionali, parte dal metodo CDS (*Central Differencing Scheme*), che adotta un'interpolazione lineare:

$$\phi_e = \phi_E \lambda_e + \phi_P (1 - \lambda_e). \quad (2.30)$$

Il fattore di interpolazione λ_e dipende dalle dimensioni delle celle della griglia in esame:

$$\lambda_e = \frac{x_e - x_P}{x_E - x_P}, \quad (2.31)$$

conseguentemente, nel caso in figura 2.4, poichè la griglia è uniforme, il valore della proprietà ϕ per ognuna delle facce è:

$$\phi_e = \frac{\phi_P + \phi_E}{2}, \quad (2.32a)$$

$$\phi_w = \frac{\phi_W + \phi_P}{2}. \quad (2.32b)$$

È possibile mostrare come l'errore di troncamento per questo metodo è proporzionale al quadrato della dimensione di cella [21].

L'ipotesi di variazione lineare fornisce anche la formula per l'approssimazione del gradiente:

$$\left(\frac{\partial \phi}{\partial x} \right)_e = \frac{\phi_E - \phi_P}{x_E - x_P}, \quad (2.33a)$$

$$\left(\frac{\partial \phi}{\partial x} \right)_w = \frac{\phi_P - \phi_W}{x_P - x_W}. \quad (2.33b)$$

In questo caso l'errore è proporzionale al quadrato della dimensione di cella solo se la griglia è uniforme, e la formula presenta accuratezza del primo ordine nel caso di griglie non uniformi. Nonostante questo, infittendo la griglia l'errore diminuisce in maniera simile a quanto accade per formule approssimanti del secondo ordine, anche per griglie non uniformi.[21]

Una delle maggiori problematiche relative al CDS è l'impossibilità di identificare la direzione del flusso e tenere conto dell'influenza della convezione. Nelle equazioni 2.30, 2.31 l'influenza di ϕ_E e ϕ_P è presente in ogni caso, ma in flussi dominati dalla convezione, ad esempio in direzione da *west* verso *east*, il valore di ϕ_e sulla faccia dovrebbe essere influenzato quasi esclusivamente da valore nel punto P . [22]

Questo limite risulta evidente considerando il numero di Péclet, che esprime il rapporto tra il trasporto di massa convettivo e quello diffusivo:

$$Pe = \frac{\rho U}{\Gamma/\delta x} = \frac{\rho U \delta x}{\Gamma}, \quad (2.34)$$

dove, δx è la lunghezza caratteristica, pari alla dimensione di cella. Quando il contributo della convezione è superiore a quello della diffusione oltre un certo limite, la soluzione fornita dal CDS non è limitata, e i risultati possono non avere significato fisico e divergere [22], [23]. In particolare, per ottenere la limitatezza della soluzione è necessario rispettare il criterio:

$$|Pe| < 2, \quad (2.35)$$

Per ovviare a quanto appena visto, si possono utilizzare gli UDS (*Upwind Differencing Scheme*), tra i quali il più semplice è il *first-order upwind*. Secondo questo approccio la proprietà sulla faccia assume lo stesso valore che ha nel centro cella a monte rispetto alla direzione del flusso. [23]

Per velocità positive (da W verso E), si ha

$$\phi_w = \phi_W, \quad (2.36a)$$

$$\phi_e = \phi_P, \quad (2.36b)$$

mentre per velocità negative (da E verso W)

$$\phi_w = \phi_P, \quad (2.37a)$$

$$\phi_e = \phi_E. \quad (2.37b)$$

I gradienti vengono calcolati anche in questo caso secondo l'equazione 2.33.

Questo schema risulta limitato, ma in alcuni casi può sovrastimare il trasporto di proprietà in direzione del flusso, portando a fenomeni di diffusione numerica. [23]

È possibile notare le differenze ottenute utilizzando il CDS o il *first-order* UDS per un flusso che presenta un termine convettivo importante in figura 2.5: l'UDS sovrastima l'influenza della convezione nel calcolo del valore alla faccia, ma risulta comunque più preciso del CDS.

Per aumentare l'accuratezza è possibile utilizzare lo schema *second-order upwind*, che considera le due celle a monte della faccia in esame. Ipotizzando di voler calcolare il valore della proprietà ϕ sulla faccia e in figura 2.4, si impone che il gradiente tra i centri delle celle W e P sia uguale al gradiente tra il centro della cella P e la faccia in esame, come nella seguente formula:

$$\frac{\phi_e - \phi_P}{x_e - x_P} = \frac{\phi_P - \phi_W}{x_P - x_W}, \quad (2.38)$$

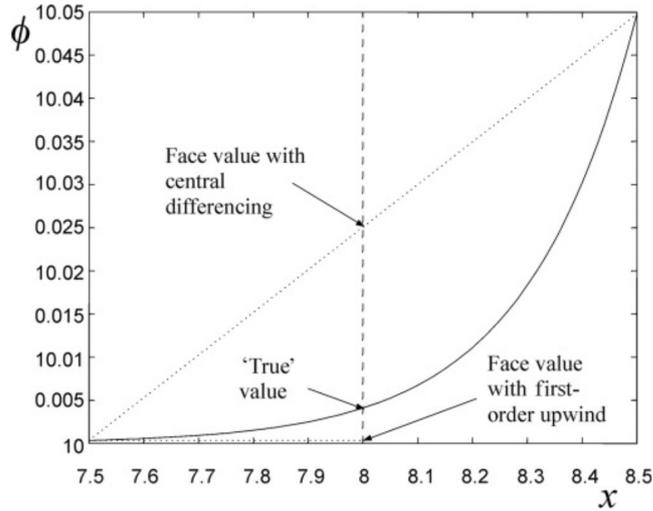


Figura 2.5: Confronto tra UDS e CDS per un flusso convettivo, relativo ad un problema presente in [23], il valore analitico (*'true' value*) corrisponde in questo caso ad una legge di potenza

dalla quale, isolando ϕ_e si ottiene:

$$\phi_e = (\phi_P - \phi_W) \frac{x_e - x_P}{x_P - x_W} + \phi_P. \quad (2.39)$$

Nel caso di griglia uniforme come in figura 2.4, l'equazione 2.39 diventa:

$$\phi_e = \frac{3}{2}\phi_P - \frac{1}{2}\phi_W. \quad (2.40)$$

Il principale svantaggio di questo metodo è la non limitatezza, motivo per cui sono stati sviluppati dei metodi del secondo ordine limitati, tra i quali viene riportato lo schema di Van Leer a titolo d'esempio[23].

Assumendo velocità positiva (da *west* verso *east*), lo schema di Van Leer valuta la seguente disuguaglianza:

$$|\phi_E - 2\phi_P + \phi_W| \geq |\phi_E - \phi_W|. \quad (2.41)$$

Se questa risulta verificata, quindi la derivata seconda di ϕ è 'piccola', viene implementata una formula *upwind* del secondo ordine:

$$\phi_e = \phi_P + \frac{(\phi_E - \phi_P)(\phi_P - \phi_W)}{\phi_E - \phi_W}, \quad (2.42)$$

altrimenti, è utilizzato uno schema *upwind* del primo ordine:

$$\phi_e = \phi_P. \quad (2.43)$$

Nel tempo sono stati sviluppati altri metodi, anche di ordine superiore al secondo, che risultano più accurati a seconda del numero di Péclet o del tipo di flusso simulato, come il QUICK (*Quadratic Upstream Interpolation for Convective Kinetics*) e il MUSCL (*Monotone Upstream-centered Scheme for Conservation Laws*), che ha maggiore accuratezza rispetto agli schemi del secondo ordine, in particolar modo per flussi con componente rotazionale importante.

2.2.2 Metodi di discretizzazione temporale

In questo lavoro si considera un problema transitorio, in cui la derivata temporale dell'equazione 2.24, relativa al termine di accumulo, deve essere discretizzata, in modo da permettere la risoluzione di un problema che varia nel tempo. Si illustra quindi il metodo di Eulero implicito, incondizionatamente stabile, applicato all'equazione generica del trasporto con schema di discretizzazione per le derivate spaziali CDS, che risulta in:

$$\phi_i^{n+1} = \phi_i^n + \left[-U \frac{\phi_{i+1}^{n+1} - \phi_{i-1}^{n+1}}{2\Delta x} + \frac{\Gamma}{\rho} \frac{\phi_{i+1}^{n+1} - 2\phi_i^{n+1} + \phi_{i-1}^{n+1}}{(\Delta x)^2} \right] \Delta t. \quad (2.44)$$

Vengono introdotti i seguenti parametri adimensionali:

$$c = \frac{U\Delta t}{\Delta x}, \quad (2.45)$$

chiamato numero di Courant, e

$$d = \frac{\Gamma\Delta t}{\rho(\Delta x)^2}, \quad (2.46)$$

che è un indice del tempo impiegato da un disturbo per essere trasmesso tramite meccanismo diffusivo per una distanza Δx [21].

Il numero di Courant, che è un parametro chiave in fluidodinamica computazionale, è collegato al tempo per trasmettere l'informazione ad una distanza Δx per meccanismo convettivo e segue un criterio per assicurare la stabilità dei metodi espliciti in problemi dominati dalla convezione, proposto per la prima volta da Courant e Friedrichs nel 1920 ed ancora in uso[21].

L'equazione 2.44 può essere riscritta in questo modo:

$$(1 + 2d)\phi_i^{n+1} + \left(\frac{c}{2} - d\right)\phi_{i+1}^{n+1} + \left(-\frac{c}{2} - d\right)\phi_{i-1}^{n+1} = \phi_i^n. \quad (2.47)$$

Questa discretizzazione porta all'ottenimento di un sistema di equazioni algebriche, che può essere risolto numericamente.

2.2.3 Algoritmi per l'accoppiamento pressione – velocità

La convezione della proprietà ϕ dipende dal campo di velocità, quindi dal modulo e l'orientazione locale del vettore velocità come da equazione 2.24. Questo campo vettoriale non è quasi mai noto a priori, quindi la sua determinazione è parte integrante del processo di soluzione delle equazioni precedentemente analizzate[22].

L'equazione che regola il trasporto di quantità di moto è l'equazione 2.4, ma la velocità \mathbf{U} deve anche rispettare l'equazione 2.1, o equazione di continuità. Questo sistema di equazioni presenta alcune difficoltà aggiuntive nella sua risoluzione:

- la presenza di termini non lineari nell'equazione di trasporto della quantità di moto;
- l'interconnessione tra le equazioni, data dalla presenza della velocità in ognuna di esse;

- il trattamento del gradiente di pressione, che è presente nelle equazioni per la quantità di moto, ma non può essere ricavato direttamente da un'equazione per la pressione.

Nella maggior parte dei casi il gradiente di pressione non è noto, e il problema risulta aggirabile solo nel caso di fluido comprimibile: si utilizza l'equazione di continuità come equazione di trasporto della densità, la quale viene utilizzata per il calcolo della pressione tramite un'equazione di stato adatta al problema studiato[22].

Per fluidi incomprimibili bisogna adottare delle procedure iterative, tra le quali sono presenti gli algoritmi SIMPLE (*Semi-Implicit Method for Pressure-Linked Equation*) o PISO (*Pressure Implicit with Splitting of Operators*). In particolare negli algoritmi della famiglia SIMPLE, un valore iniziale viene assunto per il campo di pressione, in modo da poter risolvere l'equazione 2.4 per le varie componenti ed ottenere il campo di velocità; poichè il valore di pressione ipotizzato non è quello corretto, le velocità ottenute non rispetteranno la continuità. A questo punto si introducono i fattori correttivi per pressione e velocità e le equazioni di trasporto per queste quantità corrette vengono risolte. Successivamente le altre equazioni di trasporto, per ottenere i valori delle altre proprietà ϕ , vengono risolte. Il processo viene ripetuto fino a che non viene raggiunta la convergenza[23].

Uno schema di flusso che rappresenta più in dettaglio l'algoritmo SIMPLE è presente in figura 2.6.

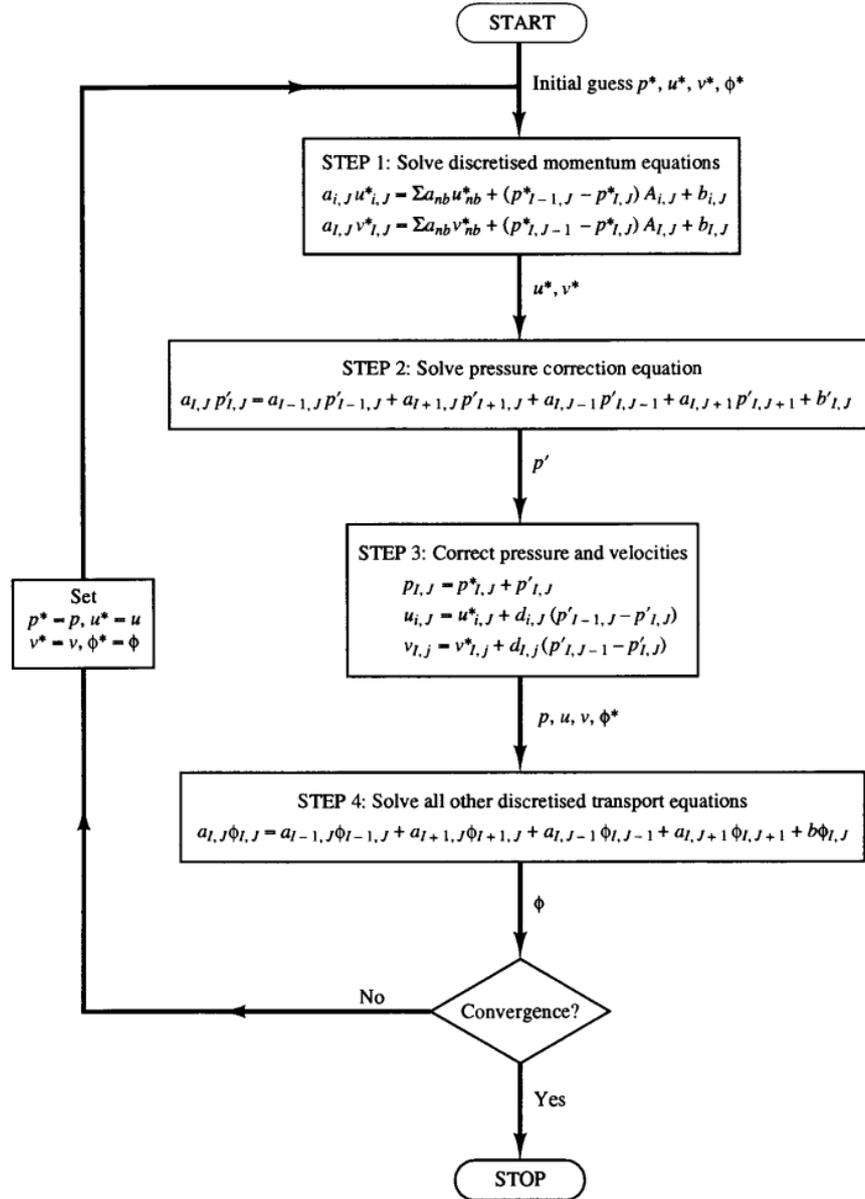


Figura 2.6: Algoritmo SIMPLE da [22]

2.3 Metodi *Volume of Fluid*

I metodi *volume of fluid*, o VOF, sono dei metodi euleriani di tracciamento dell'interfaccia che risolvono l'equazione di Navier–Stokes in entrambi i fluidi [23].

Il metodo VOF è stato inizialmente proposto da Hirt e Nichols [24], e si basa sull'idea di definire le regioni in cui sono presenti i fluidi, piuttosto che definire direttamente l'interfaccia, ottenendo così un risparmio di risorse computazionali.

Questo è possibile grazie all'introduzione di una quantità scalare α , che, per un sistema bifasico, rappresenta la frazione volumica occupata dal fluido II:

$$\alpha = \begin{cases} 1 & \longrightarrow \text{fluido II} \\ 0 < \dots < 1 & \longrightarrow \text{interfaccia} \\ 0 & \longrightarrow \text{fluido I} \end{cases} \quad (2.48)$$

Il valore di α è quindi calcolato e salvato per ogni cella della griglia di calcolo, ottenendo di fatto una rappresentazione discretizzata dei volumi dei due fluidi.

Oltre a fornire informazioni su quale fluido è contenuto in una data cella della griglia di calcolo, il metodo VOF è in grado di descrivere dove si trovano i due fluidi all'interno di una cella posizionata all'interfaccia, ricostruendo pertanto la forma dell'interfaccia, come visibile in figura 2.7. La direzione normale all'interfaccia sarà infatti quella lungo la quale il valore di α varierà più velocemente [24]. L'introduzione della variabile scalare α , anche chiamata indicatore di fase, porta alla necessità di scrivere un'equazione di avvezione aggiuntiva, che permette di descrivere l'evoluzione di questo campo scalare, come sarà descritto nella sezione 2.3.1.

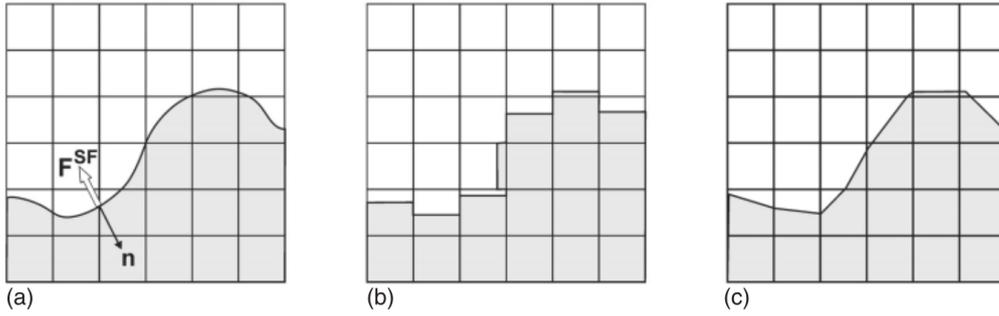


Figura 2.7: Modello di un'interfaccia fluido – fluido attraverso metodo VOF: (a) è l'interfaccia reale, (b) è la frazione volumica calcolata con il metodo VOF e (c) è la ricostruzione lineare dell'interfaccia [23]

2.3.1 Descrizione matematica

Da un punto di vista matematico, il metodo VOF permette la risoluzione di un solo sistema di equazioni, eliminando la necessità di risolvere separatamente per i due fluidi il campo di moto e il trasporto delle altre proprietà. Per ottenere ciò, tutte le proprietà locali del fluido vengono sottoposte ad un'operazione di media pesata rispetto all'indicatore di fase [23]:

$$\phi = \alpha\phi^{\text{II}} + (1 - \alpha)\phi^{\text{I}}, \quad (2.49)$$

dove ϕ è la generica proprietà, mentre il numero in apice indica a quale fluido si riferisce il valore di quella la proprietà.

Sotto l'assunzione di continuità del campo di moto tra le due fasi e introducendo le seguenti proprietà medie:

$$\rho = \langle \alpha\rho^{\text{II}} + (1 - \alpha)\rho^{\text{I}} \rangle, \quad (2.50a)$$

$$\boldsymbol{\tau} = \langle \alpha\boldsymbol{\tau}^{\text{II}} + (1 - \alpha)\boldsymbol{\tau}^{\text{I}} \rangle, \quad (2.50b)$$

è possibile riscrivere l'equazione di continuità e di trasporto della quantità di moto come segue [25]:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{U}) = 0, \quad (2.51)$$

$$\frac{\partial \rho \mathbf{U}}{\partial t} + \nabla \cdot (\rho \mathbf{U} \mathbf{U}) = \nabla \cdot \boldsymbol{\tau} + \rho \mathbf{g} + \mathbf{f}_\sigma, \quad (2.52)$$

dove l'ultimo termine dell'equazione 2.52 rappresenta la forza derivante dalla tensione interfacciale.

L'equazione di trasporto per α è la seguente:

$$\frac{\partial \alpha}{\partial t} + \frac{\partial U_i \alpha}{\partial x_i} = 0. \quad (2.53)$$

Per quest'ultima equazione risulta particolarmente importante l'utilizzo di appropriati metodi di discretizzazione per il termine di avvezione, poichè schemi del primo ordine portano ad un'interfaccia diffusa, che si allontana dal modello membranale di Young, mentre schemi di ordine superiore risultano instabili e presentano oscillazioni numeriche nei risultati [25].

Conseguentemente negli anni sono stati sviluppati vari schemi numerici per trattare al meglio questo problema di avvezione, una panoramica di questi è presente in Gopala *et al* [25].

2.3.2 Tensione interfacciale

Nella sezione 2.1.3 il problema teorico descrive i campi di moto dei due fluidi in maniera separata e ciò che accade all'interfaccia viene modellizzato attraverso la condizione al contorno in equazione 2.7. Il metodo VOF prevede però la risoluzione di una sola equazione di Navier–Stokes e, conseguentemente, il trattamento della condizione al contorno suddetta deve essere espresso in maniera differente, ma lo stesso vale anche per la tensione interfacciale. Quest'ultima infatti non è una forza per unità di volume, ma nell'equazione 2.52 si può notare come la tensione interfacciale venga infatti introdotta nel bilancio di quantità di moto attraverso il termine \mathbf{f}_σ .

Tra i vari modelli esistenti per esprimere la forza della tensione interfacciale, uno dei più utilizzati è il CSF (*Continuum Surface Force*), presentato da Brackbill *et al.* nel 1992 [26].

Il CSF è stato sviluppato per ovviare alle difficoltà nella modellizzazione di interfacce topologicamente complesse, riscontrate nei metodi sviluppati precedentemente. L'articolo di Brackbill *et al.* si concentra sulla modellizzazione della condizione al contorno normale per l'interfaccia tra fluidi inviscidi ($\mu = 0$) e incomprimibili, con coefficiente della tensione superficiale σ costante[26]. Sotto queste ipotesi, l'equazione di Young – Laplace che descrive il salto di pressione in corrispondenza della superficie diventa:

$$p_s = \sigma \kappa, \quad (2.54)$$

dove p_s è detta anche pressione di superficie, e κ è la curvatura dell'interfaccia.

La tensione interfacciale porta allora alla formazione della pressione di superficie, che è di fatto la forza normale per unità di superficie, agente sulla superficie A dell'interfaccia. In un punto \mathbf{x}_s su A , si ha:

$$p_s(\mathbf{x}_s) = |\mathbf{f}_{sa}^{(n)}|, \quad (2.55)$$

dove il pedice sa indica una forza per unità di area agente sulla superficie, e l'apice n indica la sola componente normale della forza totale agente sulla superficie.

Nel caso analizzato, grazie alle assunzioni di fluidi inviscidi e coefficiente della

tensione superficiale costante, vale:

$$\mathbf{f}_{sa} = \mathbf{f}_{sa}^{(n)} + \mathbf{f}_{sa}^{(t)} = \mathbf{f}_{sa}^{(n)}, \quad (2.56)$$

per cui la componente tangenziale $\mathbf{f}_{sa}^{(t)}$ è nulla.

La forza superficiale per unità di area d'interfaccia diventa:

$$\mathbf{f}_{sa} = \sigma \kappa(\mathbf{x}_s) \mathbf{n}(\mathbf{x}_s), \quad (2.57)$$

dove $\mathbf{n}(\mathbf{x}_s)$ è il versore normale all'interfaccia nel punto \mathbf{x}_s .

Per distinguere i due fluidi, si introduce una funzione caratteristica $c(\mathbf{x})$, così fatta:

$$c(\mathbf{x}) = \begin{cases} c_1 & \longrightarrow \text{fluido 1} \\ c_2 & \longrightarrow \text{fluido 2} \\ \langle c \rangle = \frac{c_1 + c_2}{2} & \longrightarrow \text{interfaccia} \end{cases}. \quad (2.58)$$

La funzione caratteristica appena vista è discontinua, ma nel codice viene implementata una "funzione colore" del fluido $\tilde{c}(\mathbf{x})$, che varia in maniera graduale da c_1 a c_2 in una distanza $\mathcal{O}(h)$, dove h è comparabile con la dimensione Δx della cella della griglia computazionale [26].

In questo modo si passa dall'interfaccia come discontinuità (modello della membrana di Young) ad una regione di transizione graduale tra i due fluidi, dove non è più appropriato parlare di salto di pressione dovuto alla tensione superficiale, poichè questa dovrebbe essere considerata agente su tutta la regione di transizione. Questa interpretazione della tensione superficiale come effetto continuo e tridimensionale attraverso l'interfaccia porta ad una riformulazione volumica della forza derivante dalla tensione interfacciale.[26]

Si può considerare quindi una forza per unità di volume \mathbf{f}_{sv} , che restituisce il corretto valore della forza della tensione superficiale per unità di area interfacciale \mathbf{f}_{sa} per $h \rightarrow 0$:

$$\lim_{h \rightarrow 0} \int_{\Delta V} \mathbf{f}_{sv}(\mathbf{x}) d^3x = \int_{\Delta A} \mathbf{f}_{sa}(\mathbf{x}_s) dA, \quad (2.59)$$

dove l'area di integrazione ΔA è una porzione dell'interfaccia racchiusa nel piccolo volume di integrazione ΔV . In particolare, ΔV è costruito in modo che i suoi bordi siano normali alla superficie e il suo spessore h sia più piccolo del raggio di curvatura dell'interfaccia.

La forza \mathbf{f}_{sv} deve inoltre essere nulla al di fuori della regione di interfaccia.

Da questa definizione e dalle proprietà della funzione colore $\tilde{c}(\mathbf{x})$ si arriva all'identificazione della forza volumica derivante dalla tensione superficiale, che, per h finito, è:

$$\mathbf{f}_{sv}(\mathbf{x}) = \sigma \kappa(\mathbf{x}) \frac{\nabla \tilde{c}(\mathbf{x})}{[c]}, \quad (2.60)$$

dove $[c] = c_2 - c_1$, è il salto della funzione colore.

Dalle formule precedenti traspare il ruolo della curvatura nei sistemi in cui è presente un'interfaccia. Per una superficie A , la curvatura in un punto \mathbf{x}_s è calcolata in questo modo:

$$\kappa = -(\nabla \cdot \hat{\mathbf{n}}), \quad (2.61)$$

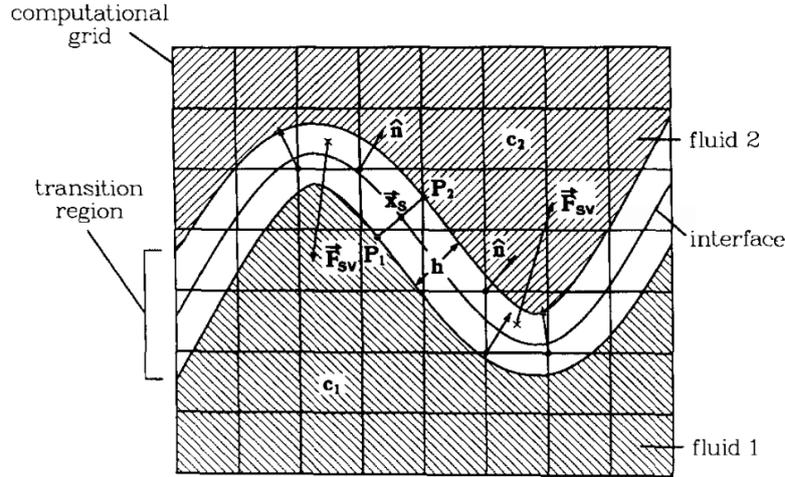


Figura 2.8: Modellizzazione dell'interfaccia tra due fluidi secondo il metodo CSF, da Brackbill *et al.*[26]

dove \mathbf{n} è il versore normale alla superficie, nel punto in cui la curvatura viene calcolata.

Nel modello CSF la superficie è sostituita da un volume, che può essere considerato composto da infinite superfici che presentano valore della funzione colore costante[26]. Le normali a queste superfici sono date dal gradiente della funzione colore stessa nel punto della zona di transizione valutato:

$$\mathbf{n} = \nabla \tilde{c}(\mathbf{x}), \quad (2.62)$$

e, conseguentemente, il versore normale relativo si calcola in questa maniera:

$$\hat{\mathbf{n}} = \frac{\nabla \tilde{c}(\mathbf{x})}{|\nabla \tilde{c}(\mathbf{x})|} \quad (2.63)$$

Segue allora che il termine $\kappa \nabla \tilde{c}$, necessario per il calcolo della forza di volume generata dalla tensione superficiale in equazione 2.60, viene computato nella seguente maniera:

$$\kappa \nabla \tilde{c} = -\mathbf{n}(\nabla \cdot \hat{\mathbf{n}}). \quad (2.64)$$

Si noti come essendo $\nabla \tilde{c}$ diversa da zero solo nella regione di transizione tra i due fluidi, anche la forza \mathbf{f}_{sv} risulterà non nulla solo nella medesima regione.

La scelta della funzione colore è di importanza fondamentale nell'utilizzo del metodo CSF, e per fluidi incompressibili può essere utilizzata la densità dei due fluidi. Ma nel contesto di un metodo VOF è naturale l'utilizzo la funzione indicatore α , in quanto questa è già prevista inerentemente nel modello e ha l'esatta funzione di distinguere i due fluidi.

2.3.3 Vantaggi e svantaggi

Per quanto concerne gli algoritmi ed i metodi di tracciamento dell'interfaccia per le simulazioni multifluido, ci sono almeno quattro punti fondamentali per l'ottenimento di risultati accurati [27].

1. Rappresentazione accurata dell'interfaccia e sua avvezione.

La posizione dell'interfaccia, la sua normale e la sua curvatura sono, nei metodi VOF, note implicitamente, tramite la funzione colore o funzione indicatore di fase α . I metodi VOF in genere tendono a causare una diffusione dell'interfaccia, che è documentato avvenga in seguito all'utilizzo di schemi di avvezione non adatti [25], [27]. Per ovviare a ciò sono stati sviluppati nel tempo dei metodi di ricostruzione geometrica dell'interfaccia, che presentano problemi nell'implementazione in dimensioni superiori e richiedono elevati costi computazionali, e metodi algebrici, solitamente basati su schemi di discretizzazione compressivi.

Per quanto riguarda la difficoltà nella stima della curvatura, dovute alla natura discontinua della funzione indicatore di fase, sono proposti degli algoritmi più performanti, che però presentano ancora errori importanti per valori elevati di curvatura o regioni con griglie di calcolo non sufficientemente fitte.

2. Conservazione della massa.

Per come sono stati derivati e costruiti, i metodi VOF conservano la massa, per questo vengono preferiti in alcuni casi a metodi più accurati nella rappresentazione dell'interfaccia, ma che presentano “perdita di massa”, come i *level set methods*.

3. Correnti parassite o spurie

Quasi tutti i metodi impliciti di tracciamento dell'interfaccia sono soggetti alla generazione, in corrispondenza dell'interfaccia, di un flusso che non ha significato fisico, ma è dovuto esclusivamente al metodo numerico adottato. Queste correnti sono causate dalla curvatura dell'interfaccia non correttamente stimata e da uno sbilanciamento di forze. Numerose strategie sono state adottate per mitigare l'effetto di queste correnti parassite, che sono particolarmente problematiche in flussi capillari, in cui i contributi inerziali non sono dominanti. [27]–[29]

4. Elevati rapporti di viscosità

Simulazioni con elevati rapporti di viscosità portano ad inconsistenze nella formulazione dell'equazione di continuità ed in quella del trasporto della quantità di moto. I risultati ottenuti sono affetti dalla presenza di flussi che non hanno significato fisico e deformazioni artificiali dell'interfaccia [27].

Si ricorda, inoltre, che uno dei vantaggi dei metodi VOF risiede nella necessità di memorizzare un solo valore scalare, quello della funzione α , per cella, a differenza dei metodi a particelle marcatrici (MAC), che richiedono molte risorse computazionali per tenere traccia di tutte le particelle e necessitano dell'aggiunta di nuove particelle marcatrici nel caso in cui l'interfaccia vada incontro ad una deformazione importante [24], [25].

Infine, rispetto alla maggior parte dei metodi che impongono direttamente la condizione al contorno sull'interfaccia e, pertanto, la tracciano esplicitamente, come il *front tracking* e il *line segment*, i metodi VOF permettono di trattare in maniera semplice cambiamenti topologici della superficie complessi come ripiegamenti su se stessa, rottura e coalescenza [24], [25], [30].

Capitolo 3

Dettagli numerici

In questo capitolo vengono analizzati più nel dettaglio gli aspetti relativi alla numerica, partendo dall'implementazione della teoria, vista nel capitolo precedente, all'interno del software utilizzato in sezione 3.1. Successivamente si discute dell'inizializzazione delle simulazioni, dalla fase iniziale di impostazione del dominio, in sezione 3.2, a quella fondamentale di scelta delle condizioni al contorno, in sezione 3.3. Le sezioni 3.4 e 3.5 esplorano le condizioni operative e le proprietà utilizzate per le simulazioni, dando un'idea di come le simulazioni sono state impostate nella pratica. Infine, la sezione 3.6 pone l'attenzione su una delle maggiori problematiche affrontate durante il lavoro, proponendo alcune soluzioni per mitigare l'effetto delle correnti parassite.

3.1 Software CFD – OpenFOAM

Le equazioni descritte nella sezione 2.1, devono essere implementate nella loro forma discretizzata e risolte, come in sezione 2.2, in un codice di fluidodinamica computazionale.

In questo lavoro, la scelta ricade su OpenFOAM (*Open-source Field Operation and Manipulation*, <https://www.openfoam.org>), un *framework* che permette lo sviluppo di applicazioni utilizzando delle funzionalità pacchettizzate, contenute all'interno di librerie scritte in linguaggio C++. In OpenFOAM sono integrate circa 250 applicazioni già scritte, che si distinguono tra *solvers*, ognuno dei quali è progettato per la risoluzione di particolari problemi della meccanica del continuo attraverso il metodo dei volumi finiti, e *utilities*, programmi per la manipolazione dei dati utilizzati per il *pre-processing* ed il *post-processing*. [31]

OpenFOAM presenta molti solutori pronti per l'uso, in campi che variano dalle applicazioni più semplici di CFD ai problemi di combustione, dallo scambio di calore all'elettromagnetismo, ma una delle caratteristiche più interessanti è la possibilità di modificare direttamente il codice, quindi sviluppare ulteriori solutori, partendo sia da zero, che da applicazioni già presenti e testate. Questo è possibile grazie al linguaggio utilizzato e, in particolare, alla programmazione ad oggetti. La creazione di classi permette infatti di traslare dal linguaggio verbale e matematico oggetti fisici ed entità astratte, come i concetti di campi scalari, vettoriali e tensoriali. Il risultato è una migliore gestione del codice, in cui ogni oggetto appartenente ad una data classe (*e.g.* campo di velocità \mathbf{U}) rispetterà le proprietà matematiche e

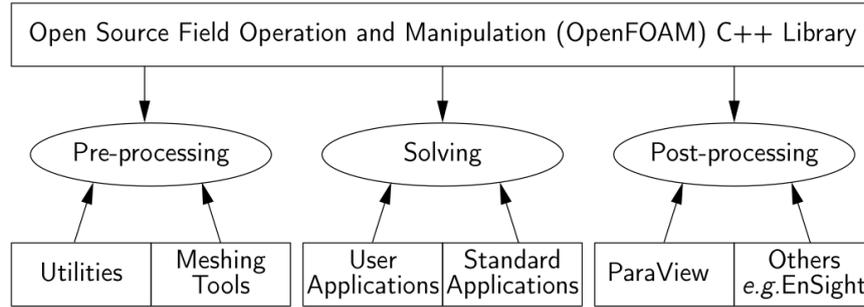


Figura 3.1: Panoramica della struttura generale del software OpenFOAM [31]

le regole di interazione con gli operatori tipiche di quella classe (campo vettoriale), rendendo il codice più coerente e facile da modificare. [31], [32]

La rappresentazione e la sintassi delle equazioni nei *solvers*, inoltre, richiamano molto le equazioni differenziali alle derivate parziali che devono essere risolte, ad esempio:

$$\frac{\partial \rho \mathbf{U}}{\partial t} + \nabla \cdot (\phi \mathbf{U}) - \nabla \cdot (\mu \nabla \mathbf{U}) = -\nabla p \quad (3.1)$$

viene scritta nel codice nella seguente maniera:

```

1 solve
2 (
3   fvm::ddt(rho, U)
4   +fvm::div(phi, U)
5   -fvm::laplacian(mu, U)
6   ==
7   -fvc::grad(p)
8 );

```

Le librerie di OpenFOAM presentano sia la possibilità di generare la griglia di calcolo necessaria per i volumi finiti direttamente tramite le *utilities* già presenti, sia la possibilità di importare *mesh* generate con software di terze parti, purché prodotte in uno dei formati supportati. Nel primo caso è possibile generare una griglia di calcolo a celle esaedriche tramite `blockMesh`, per geometrie non troppo complesse, e `snappyHexMesh`, per griglie di calcolo complesse che necessitino dello *splitting*. [31]

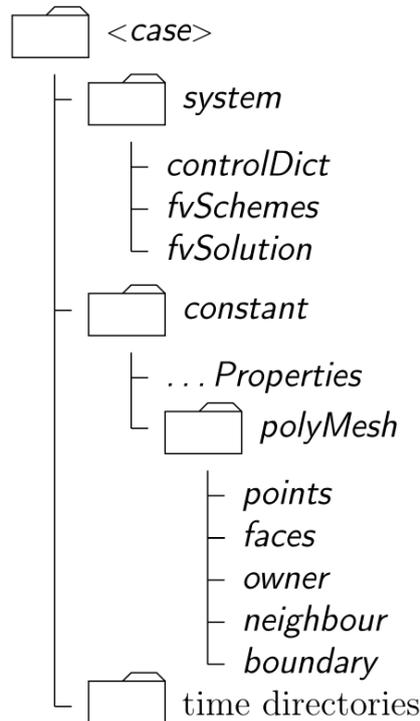
Una varietà di strumenti permette il *post-processing* per le simulazioni effettuate, che può essere avviato direttamente dalla linea di comando attraverso le *utilities* desiderate, nei dizionari di controllo del processo di simulazione o, in alcuni casi, dal *software* di visualizzazione stesso, Paraview, avviabile con il comando `paraFoam`. [31]

3.1.1 Struttura di una simulazione in OpenFOAM

Le singole simulazioni, in ambito OpenFOAM, vengono costruite, inizializzate e avviate attraverso *files* di *input* e di *output* contenuti in una cartella, chiamata “*case*”, la cui struttura è mostrata in figura 3.2.

Analizzando i file minimi necessari per avviare correttamente una simulazione, figura 3.2, si distinguono le seguenti sottocartelle:

- La cartella `constant`, contiene le informazioni sulla griglia di calcolo, o *mesh*, salvate nella cartella `polyMesh`, e le informazioni sulle proprietà del sistema,


 Figura 3.2: Struttura minima di una cartella *case* in OpenFOAM [31]

contenute nei *file* di *input* o dizionari, come `transportProperties` che permette l’inserimento dei valori delle proprietà chimico–fisiche dei fluidi simulati, o `momentumTransport`, che descrive il regime fluidodinamico della simulazione (laminare o turbolento).

- La cartella `system`, che contiene almeno tre file: `controlDict`, che esplicita il *solver* da utilizzare, il tempo di inizio/fine simulazione, il passo temporale e la gestione dell’output; `fvSchemes`, tramite il quale vengono scelti gli schemi di discretizzazione da usare nella risoluzione; `fvSolution`, che contiene i metodi di risoluzione per le equazioni dei sistemi, le tolleranze e gli algoritmi di controllo.
- Le cartelle dei tempi, che contengono i file in cui sono presenti i valori di alcuni campi, *e.g.* velocità e pressione. Questi valori possono essere valori iniziali o condizioni al contorno, oppure valori scritti da OpenFOAM a seguito della risoluzione delle equazioni del problema a quel dato istante di tempo. Si ricorda, inoltre, che in tutti i casi questi campi devono essere sempre inizializzati, anche quando la soluzione non lo richiede, ad esempio per problemi stazionari. I valori iniziali e le condizioni al contorno sono solitamente scritti sui file dei rispettivi campi salvati nella cartella 0, che corrisponde a $t = 0$, valore temporale da cui solitamente si fa iniziare una simulazione.

Nella cartella `system` sono inoltre presenti, nei casi in cui vengono utilizzate, le *utilities* o i relativi dizionari, necessari per il *pre-processing* e il *post-processing*.

3.1.2 Algoritmo di avvezione FCT – MULES

Come enunciato in sezione 2.3.1, l'equazione 2.53, che descrive il trasporto dell'indicatore di fase α , ha visto negli anni lo sviluppo di vari metodi di discretizzazione, con l'obiettivo di trattare in maniera efficiente il termine di avvezione.

Nel caso di OpenFOAM, il metodo utilizzato è chiamato MULES (*Multidimensional Universal Limiter for Explicit Solutions*), ed è un'implementazione della teoria del modello *Flux Corrected Transport*, o FCT, proposto per la prima volta da Boris e Book [33] e perfezionato ed esteso a casi a più dimensioni da Zalesak [34]. Il metodo FCT ha come obiettivo quello di assicurare la limitatezza della soluzione [35].

Viene illustrato un caso generico d'esempio per la comprensione del metodo FCT, partendo dall'equazione seguente:

$$\frac{\partial \phi}{\partial t} + \nabla \cdot \mathbf{F} = 0, \quad (3.2)$$

dove, nell'impostazione dell'equazione nell'ambito del metodo dei volumi finiti, ϕ è la proprietà trasportata e \mathbf{F} è il suo flusso, funzione della stessa ϕ .

Discretizzando l'equazione con l'utilizzo di un metodo temporale esplicito si arriva alla formulazione

$$\frac{\phi_i^{n+1} - \phi_i^n}{\Delta t} \Omega + \sum_f (\mathbf{F}^n \cdot \mathbf{S})_f = 0, \quad (3.3)$$

in cui i è la cella in esame, l'apice n indica il passo temporale, Ω è il volume di una cella, f la faccia della cella in esame e S il vettore area della suddetta faccia. Isolando ϕ_i^{n+1} e scrivendo $F^n = \mathbf{F}^n \cdot \mathbf{S}$, per una griglia monodimensionale si ottiene:

$$\phi_i^{n+1} = \phi_i^n - \frac{\Delta t}{\Omega} \left(F_{i+\frac{1}{2}}^n - F_{i-\frac{1}{2}}^n \right). \quad (3.4)$$

Per assicurare la limitatezza sul calcolo di F , si utilizzano sia un metodo di basso ordine limitato che un metodo di ordine superiore che potrebbe non essere limitato. I passi descritti dal metodo possono essere riassunti nella seguente maniera [35]:

1. Calcolo di F^L , il flusso ottenuto dall'utilizzo del metodo di basso ordine che garantisce la limitatezza.
2. Calcolo di F^H , il flusso ottenuto dall'utilizzo del metodo di alto ordine.
3. Si definisce il flusso anti-diffusivo, $A = F^H - F^L$.
4. Si calcola il flusso corretto $F^C = F^L + \lambda_M A$, dove $0 \leq \lambda_M \leq 1$.
5. Si utilizza il flusso corretto F^C nell'equazione 3.4, per il calcolo della proprietà ϕ al centro cella:

$$\phi_i^{n+1} = \phi_i^n - \frac{\Delta t}{\Omega} \left(F_{i+\frac{1}{2}}^C - F_{i-\frac{1}{2}}^C \right). \quad (3.5)$$

L'idea è che l'utilizzo corretto di valori provenienti da schemi differenti nelle giuste proporzioni possa mitigare i problemi di diffusività numerica, senza restituire l'instabilità riscontrata nell'utilizzo di schemi di ordine superiore [25]. Conseguentemente lo step critico è il numero quattro, poichè la scelta di λ_M , detto limitatore, è quella che permette di ripartire il peso tra i due schemi.

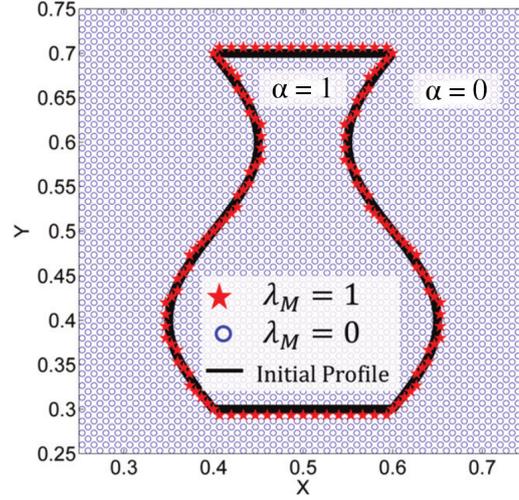


Figura 3.3: Valori assunti dal limitatore dell’algoritmo MULES, λ_M , pari a uno vicino all’interfaccia e a zero nelle zone lontane da questa [27]

3.1.3 Solutore interFoam

Il metodo *volume of fluid* in OpenFOAM è implementato attraverso il solutore `interFoam`, utilizzato per simulazioni in condizioni isoterme di due fluidi immiscibili e incompressibili.[31]

L’implementazione del metodo VOF, descritto in sezione 2.3, può essere effettuata in vari modi, quella presente in OpenFOAM è descritta in maniera estensiva da Ubbink [36] e Rusche [37], mentre una trattazione più recente e aggiornata sul solutore `interFoam` è presente in letteratura ad opera di Márquez [32], [35], Berberović *et al.* [38] e Deshpande *et al.* [27].

È sempre consigliato, tuttavia, verificare differenze presenti nel codice, tramite la documentazione (<https://www.openfoam.org>) e la guida al codice sorgente C++ (<https://cpp.openfoam.org/>), poichè metodi di risoluzione e struttura dei *solver* possono presentare modifiche dipendentemente dalla versione utilizzata.

A partire dagli scritti citati si imposta la trattazione seguente, che porta al metodo VOF secondo la riformulazione di Weller [35], [37]:

$$\left\{ \begin{array}{l} \nabla \cdot \mathbf{U} = 0 \\ \frac{\partial \rho \mathbf{U}}{\partial t} + \nabla \cdot (\rho \mathbf{U} \mathbf{U}) = -\nabla p_{rgh} + \nabla \cdot [\mu(\nabla \mathbf{U} + (\nabla \mathbf{U})^T)] \\ \quad - \mathbf{g} \cdot \mathbf{x} \nabla \rho + \sigma \kappa \nabla \alpha \\ \frac{\partial \alpha}{\partial t} + \nabla \cdot (\alpha \mathbf{U}) + \nabla \cdot [\mathbf{U}_r \alpha (1 - \alpha)] = 0 \end{array} \right. \quad \begin{array}{l} (3.6a) \\ (3.6b) \\ (3.6c) \end{array}$$

dove si ricorda che le proprietà, come la densità ρ , derivano da una media pesata sull’indicatore di fase, come descritto in equazione 2.49.

Analizzando il sistema di equazioni 3.6 appena enunciato, l’equazione 3.6a è semplicemente l’equazione di continuità per fluidi incompressibili, mentre l’equazione 3.6b è il bilancio di quantità di moto già visto, ma con due modifiche degne di nota:

- L'ultimo termine dell'equazione è la forza relativa alla tensione superficiale, espressa secondo il modello CSF (sezione 2.3.2):

$$\mathbf{f}_\sigma = \sigma \kappa \nabla \alpha, \quad (3.7)$$

in cui la funzione colore $\tilde{c}(\mathbf{x})$ utilizzata è proprio la funzione indicatore α .

- L'espressione della pressione che, per semplificare il trattamento delle condizioni al contorno [38], introduce la pressione modificata:

$$p_{rgh} = p - \rho \mathbf{g} \cdot \mathbf{x}, \quad (3.8)$$

dove \mathbf{x} è il vettore posizione. Calcolando il gradiente di pressione si ottiene:

$$-\nabla p = -\nabla p_{rgh} - \mathbf{g} \cdot \mathbf{x} \nabla \rho - \mathbf{g} \rho, \quad (3.9)$$

che porta a

$$-\nabla p + \mathbf{g} \rho = -\nabla p_{rgh} - \mathbf{g} \cdot \mathbf{x} \nabla \rho, \quad (3.10)$$

sostituita per ottenere il bilancio di quantità di moto come in equazione 3.6b.

L'equazione 3.6c, per il trasporto della frazione volumica o indicatore di fase α , è peculiare del metodo VOF – Weller e deriva dai vari tentativi di assicurare la limitatezza e la conservazione della quantità scalare α . A rendere più complesso il raggiungimento di questo obiettivo è la condivisione del campo di velocità tra le due fasi, che rende impossibile individuare direttamente i contributi di queste ultime alla velocità presente nel sistema.[38]

Per ovviare a queste problematiche, il modello parte da un approccio euleriano – euleriano per un fluido bifase, in cui le equazioni per la frazione volumica sono risolte separatamente per le due fasi:

$$\frac{\partial \alpha}{\partial t} + \nabla \cdot (\mathbf{U}^I \alpha) = 0 \quad (3.11a)$$

$$\frac{\partial (1 - \alpha)}{\partial t} + \nabla \cdot [\mathbf{U}^{II} (1 - \alpha)] = 0 \quad (3.11b)$$

In questa maniera, assumendo che i contributi delle velocità all'evoluzione dell'interfaccia siano proporzionali alla frazione volumica dei due fluidi [38], è possibile introdurre la velocità effettiva come media pesata sull'indicatore di fase:

$$\mathbf{U} = \alpha \mathbf{U}^I + (1 - \alpha) \mathbf{U}^{II}, \quad (3.12)$$

e tramite sostituzione in equazione 3.11b è possibile ottenere l'equazione 3.6c, dove $\mathbf{U}_r = \mathbf{U}^I - \mathbf{U}^{II}$ è il vettore della velocità relativa tra le due fasi, che nel modello descritto è la “velocità di compressione”.

Una dimostrazione semplificata, che descrive questo riarrangiamento delle equazioni in modo da ottenere l'equazione 3.6c, è presente in appendice A. Per una trattazione estesa e rigorosa si consiglia di fare riferimento a Marquez [35], dove le equazioni del metodo VOF sono ricavate partendo dalla teoria ASMM (*Algebraic Slip Mixture Model*).

In realtà il valore di \mathbf{U}_r è nullo, poichè come già spiegato, per la teoria VOF, il campo di velocità è unico e condiviso tra le due fasi; conseguentemente il termine non lineare

$$\nabla \cdot [\mathbf{U}_r \alpha(1 - \alpha)] \quad (3.13)$$

potrebbe essere semplificato, ma viene invece incluso come termine di compressione artificiale $\Phi_{f,\alpha}^{\text{compr}}$, descritto in equazione 3.17, in modo da diminuire la diffusione dell'interfaccia e mantenere una rappresentazione accurata della superficie [35], [38]. Si nota, infatti, come questo termine si annulli per α pari a 0 e 1, ovvero al di fuori della zona d'interfaccia: in questi casi viene utilizzato il metodo VOF classico, senza compressione.

Una volta che le equazioni sono impostate si passa alla loro discretizzazione. Il termine di avvezione dell'equazione 3.6c di trasporto per α viene quindi discretizzato secondo la tecnica MULES, e il flusso è espresso come:

$$\sum_f F_f^L + \lambda_M F_f^C, \quad (3.14)$$

dove la sommatoria è su tutte le facce f appartenenti alla cella in esame e il valore di λ_M è nullo all'interno del *bulk* delle due fasi e pari a uno in corrispondenza dell'interfaccia, come si può notare in figura 3.3.

Questo implica l'utilizzo del flusso correttivo F^C solo nelle celle interessate dall'interfaccia, mentre nelle altre si utilizza uno schema *upwind*. Analizzando i due flussi infatti si ha:

$$F^L = \Phi_{f,\alpha}^{\text{UD}}, \quad (3.15)$$

che corrisponde a risolvere l'equazione 2.53 con uno schema di discretizzazione *upwind*, da cui l'apice UD, e

$$F^C = \Phi_{f,\alpha}^{\text{HO}} + \Phi_{f,\alpha}^{\text{compr}} - \Phi_{f,\alpha}^{\text{UD}}, \quad (3.16)$$

in cui il primo termine indica l'utilizzo di uno schema di ordine superiore definito dall'utente e il secondo termine il termine compressivo.

La nomenclatura utilizzata richiama il codice scritto in OpenFOAM, in cui $\Phi_f = \mathbf{U}_f \cdot \mathbf{S}_f$ è la portata volumica e conseguentemente $\Phi_{f,\alpha} = (\alpha \mathbf{U})_f \cdot \mathbf{S}_f$ è la portata di α attraverso la faccia della cella.

Nel dettaglio, $\Phi_{f,\alpha}^{\text{HO}}$ viene definito nel file `fvScheme`, attraverso l'espressione nel listing 3.1, dove `interfaceCompression` indica lo schema di discretizzazione per il termine non lineare, mentre lo schema di ordine superiore, in questo caso, è specificato con la parola chiave `vanLeer`, che fa riferimento all'omonimo schema di discretizzazione, esposto in maniera riassuntiva in sezione 2.2.1.

```
1 div(phi,alpha) Gauss interfaceCompression vanLeer 1;
```

Listing 3.1: Parametri per la discretizzazione dell'equazione dell'indicatore di fase

Il termine compressivo $\Phi_{f,\alpha}^{\text{compr}}$ è presente nel file `interfaceCompression.C` del modello `twoPhaseMixture`, e viene calcolato attraverso l'espressione

```
1 cAlpha_*sign(phi_)*vff*(1 - vff)*nHatf/mesh().magSf(),
```

che può essere riscritto tramite la seguente formula:

$$\Phi_{f,\alpha}^{\text{compr}} = C_\alpha \frac{\text{sgn}(\Phi_f) \alpha_f (1 - \alpha_f)}{|S_f|} \cdot n_f, \quad (3.17)$$

nella quale il termine C_α è una costante scelta dall'utente tramite l'ultimo valore scalare presente nel listing 3.1: un valore più basso di uno riduce il contributo compressivo (al limite di zero lo elimina), portando ad una maggiore diffusione dell'interfaccia, mentre un valore più alto dell'unità aumenta l'effetto compressivo, aumentando l'instabilità per correnti parassite.

Nell'equazione 3.17, n_f è pari al versore normale all'interfaccia moltiplicato scalarmente per il vettore normale alla faccia:

$$n_f = \mathbf{n}_f \cdot \mathbf{S}_f = \frac{(\nabla\alpha)_f}{|(\nabla\alpha)_f + \delta_n|} \cdot \mathbf{S}_f, \quad (3.18)$$

in cui δ_n è un fattore stabilizzatore utilizzato per evitare la divisione per zero. Questo ha espressione:

$$\delta_n = \frac{\epsilon}{\left(\frac{\sum_N \Omega_i}{N}\right)^{\frac{1}{3}}}, \quad (3.19)$$

con $\epsilon = 1 \times 10^{-8}$, e Ω_i pari al volume della cella i -esima.

Il solutore permette un controllo adattivo del passo temporale, che viene calcolato in questa maniera [35]:

$$\Delta t^n = \min \left\{ \frac{Co_{max}}{Co} \Delta t^0, \left(1 + \lambda_1 \frac{Co_{max}}{Co}\right) \Delta t^0, \lambda_2 \Delta t^0, \Delta t_{max} \right\}, \quad (3.20)$$

dove $\lambda_1 = 0.1$ e $\lambda_2 = 1.2$ sono due parametri presenti nel codice OpenFOAM non modificabili dall'utente, che invece definisce Co_{max} e Δt_{max} attraverso il file `controlDict`.

Il valore del numero di Courant, Co è calcolato sulle facce delle celle attraverso la seguente formula:

$$Co = \frac{1}{2} \frac{|\mathbf{U}_f \cdot \mathbf{S}_f|}{\Omega_i} \Delta t. \quad (3.21)$$

In OpenFOAM è inoltre presente la possibilità di specificare un numero di Courant massimo che fa riferimento solo alle celle interessate dall'interfaccia, `maxAlphaCo`, in modo da regolare la risoluzione temporale in base a ciò che accade all'interfaccia, ottimizzando l'utilizzo delle risorse computazionali.

3.1.4 Accoppiamento pressione – velocità: PISO

Come già visto in sezione 2.2.3, nel caso di una simulazione con fluido incomprimibile, è necessario un algoritmo che, tramite una procedura iterativa, permetta di stimare e correggere i campi di pressione e velocità.

Nel caso delle simulazioni non stazionarie, come in questo lavoro, uno degli algoritmi più utilizzati per l'accoppiamento pressione – velocità è il PISO (*Pressure Implicit with Splitting of Operators*). Trattazioni generali sull'algoritmo PISO sono presenti in letteratura [21], [23], in questa sede l'analisi è concentrata sull'implementazione dell'algoritmo in OperFOAM ed in particolare all'interno del solutore `interFoam`.

La procedura di risoluzione si articola in questo modo [35]:

1. Si risolve l'equazione di continuità per la fase α , calcolando il flusso massico. Questo viene ripetuto un numero pari ad `nAlphaSubCycles` di volte, attraverso questi passaggi:

- (a) calcolo del termine compressivo all'interfaccia come in equazione 3.17,
 - (b) l'equazione di trasporto per α (equazione 3.6c) viene risolta in forma discretizzata un numero di volte specificato da `nAlphaCorrectors` tramite la tecnica MULES;
 - (c) Partendo da α il nuovo flusso massico viene calcolato e la densità aggiornata.
2. Se il valore di `momentumPredictor` è impostato su `yes`, l'equazione 3.6b viene risolta in forma discretizzata.
 3. Il ciclo PISO viene risolto un numero di volte specificato attraverso il parametro `nCorrectors`:
 - (a) il flusso proposto alle facce è calcolato tramite il campo di velocità ricavato nel punto 2, e quindi corretto per rispettare l'equazione di continuità;
 - (b) il flusso proposto finale è ottenuto aggiungendo i termini contenenti l'effetto di forza di gravità e tensione interfacciale;
 - (c) l'equazione della pressione viene costruita e risolta per la pressione modificata p_{rgh} un numero di volte pari a `nNonOrthogonalCorrectors`;
 - (d) il flusso proposto finale è corretto attraverso l'effetto del campo di pressione del punto (c), così come lo sono le velocità ai centri di celle;
 - (e) la pressione statica p viene ricostruita a partire da p_{rgh} , secondo l'equazione 3.8.

Nelle simulazioni presentate in questo lavoro, l'opzione `momentumPredictor` è impostata su `no`, conseguentemente il punto 2 della risoluzione verrà saltato, ed il ciclo PISO sarà inizializzato attraverso il campo di velocità calcolato all'iterazione precedente.

Questa scelta è dettata dall'ottenimento di migliori performance del solver per il problema in esame disattivando la risoluzione dell'equazione 3.6b.

3.2 Dominio Computazionale

In OpenFOAM la geometria del sistema che si vuole simulare, così come la griglia di calcolo, possono essere create attraverso software esterni ed essere importate. Nel caso in esame, tuttavia, il sistema da simulare non ha una forma complessa e le funzionalità di generazione della griglia di calcolo fornite dalle librerie di OpenFOAM risultano sufficientemente elaborate, anche se l'implementazione di un sistema di *meshing* più complesso potrebbe contribuire a migliorare l'efficienza delle simulazioni in studi futuri.

3.2.1 Geometria

Poiché l'obiettivo del lavoro è legato alle emulsioni alimentari, il sistema simulato comprende una goccia d'olio (fluido II) immersa in acqua (fluido I).

Sono state analizzate sia geometrie bidimensionali che geometrie tridimensionali, definite tramite il dizionario `blockMeshDict`, che viene utilizzato dall'*utility*

`blockMesh` per creare la geometria e la griglia di calcolo.

Due esempi di domini vuoti sono in figura 3.4, per le simulazioni 2D e in figura 3.5 per le simulazioni in 3D. Si può notare dalle suddette immagini come le facce dei domini vengano contrassegnate secondo una notazione `north`, `south`, `east`, `west`, ma in entrambi i casi le facce perpendicolari all'asse z siano contrassegnate come `frontAndBack`. Questo è dovuto alla possibilità di impostare più facilmente le con-

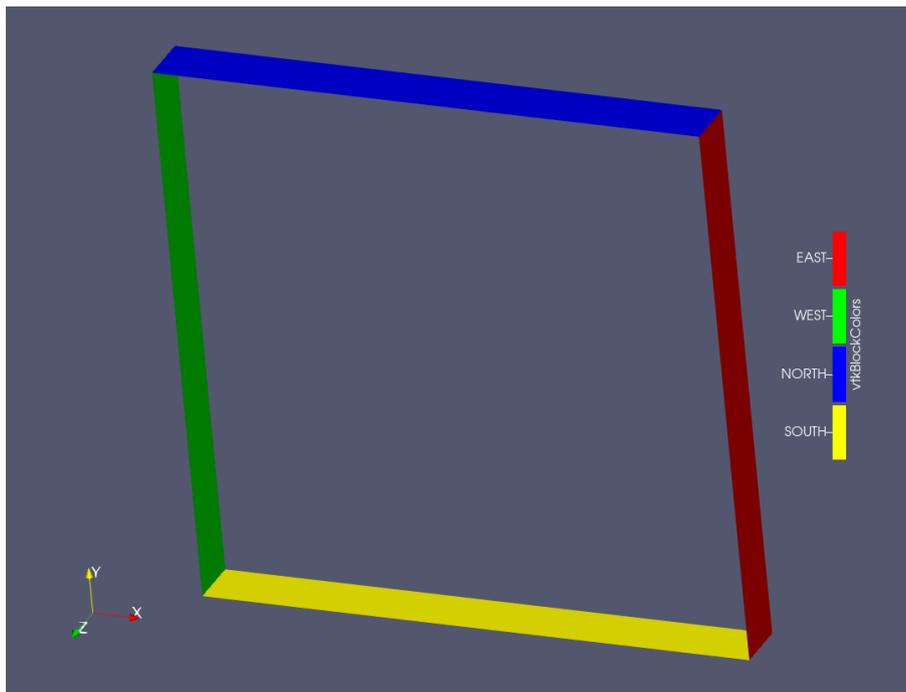


Figura 3.4: Dominio utilizzato per le simulazioni bidimensionali

dizioni al contorno, che, come verrà illustrato nella sezione 3.3, sono le stesse per queste due facce.

L'implementazione di simulazioni bidimensionali su OpenFOAM avviene attraverso un dominio tridimensionale, che presenta una sola cella della griglia nella direzione ortogonale al piano di simulazione, e le cui facce normali alla suddetta direzione sono contrassegnate dalla parola chiave `empty` nel *file* di costruzione del dominio computazionale.

Le dimensioni della *box* che conterrà il sistema sono definite secondo i listing 3.2 e 3.3, in cui si definiscono le coordinate dei vertici del dominio, e sono state scelte e modificate in base al comportamento della goccia e degli effetti sulla dinamica di deformazione e di rottura, ma anche tenendo conto dell'aumento del costo computazionale.

La configurazione finale presenta un dominio cubico (quadrato nel caso bidimensionale) con lato pari a 1.6 centimetri.

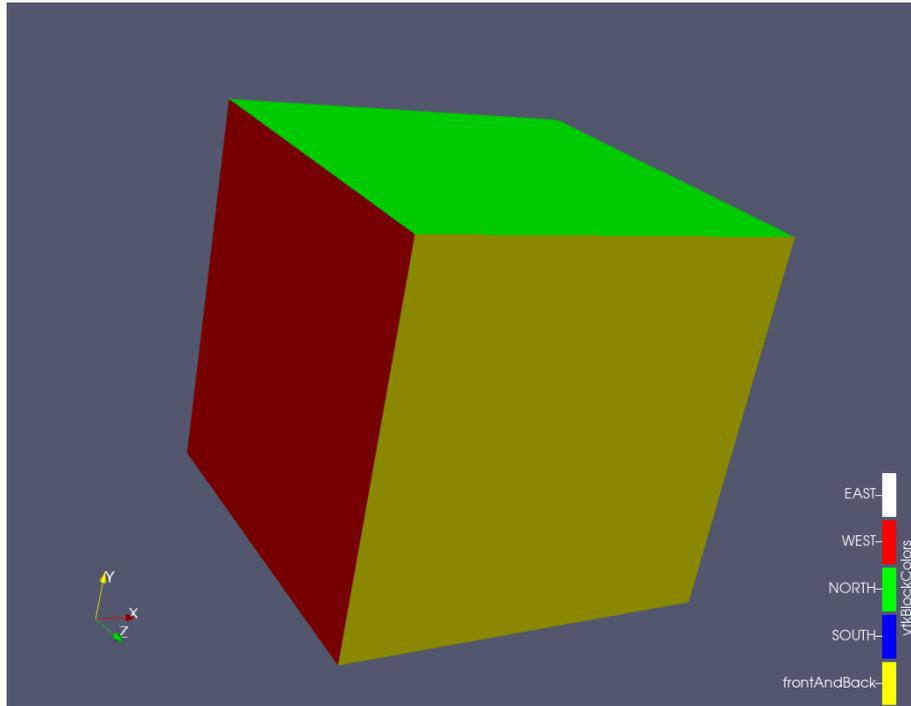


Figura 3.5: Dominio utilizzato per le simulazioni tridimensionali

```

1 convertToMeters 0.01;
2
3 vertices
4 (
5     (-0.8 -0.8 0)
6     (0.8 -0.8 0)
7     (0.8 0.8 0)
8     (-0.8 0.8 0)
9     (-0.8 -0.8 0.1)
10    (0.8 -0.8 0.1)
11    (0.8 0.8 0.1)
12    (-0.8 0.8 0.1)
13 );

```

Listing 3.2: Impostazione delle dimensioni del dominio bidimensionale nel `blockMeshDict`

```

1 convertToMeters 0.01;
2
3 vertices
4 (
5     (-0.8 -0.8 -0.8)
6     (0.8 -0.8 -0.8)
7     (0.8 0.8 -0.8)
8     (-0.8 0.8 -0.8)
9     (-0.8 -0.8 0.8)
10    (0.8 -0.8 0.8)
11    (0.8 0.8 0.8)
12    (-0.8 0.8 0.8)
13 );

```

Listing 3.3: Impostazione delle dimensioni del dominio tridimensionale nel `blockMeshDict`

Il passo successivo è l'imposizione delle due fasi all'interno del dominio computazionale appena creato, che viene specificata attraverso il dizionario `setFieldsDict` attraverso il codice riportato nel listing 3.4, che imposta il dominio completamente costituito da fluido I (`water` nel codice), con una goccia sferica di fluido II al centro, di raggio a pari a due millimetri.

```

1 defaultFieldValues
2 (
3     volScalarFieldValue alpha.water 1
4 );
5
6 regions
7 (
8     sphereToCell
9     {
10        centre ( 0 0 0 );
11        radius 0.002;

```

```

12     fieldValues
13     (
14         volScalarFieldValue alpha.water 0
15     );
16 }
17 );

```

Listing 3.4: Impostazione delle due fasi all'interno del dominio in `setFieldsDict`

Il risultato di questa imposizione è osservabile in figura 3.6, per il dominio bidimensionale e per quello tridimensionale.

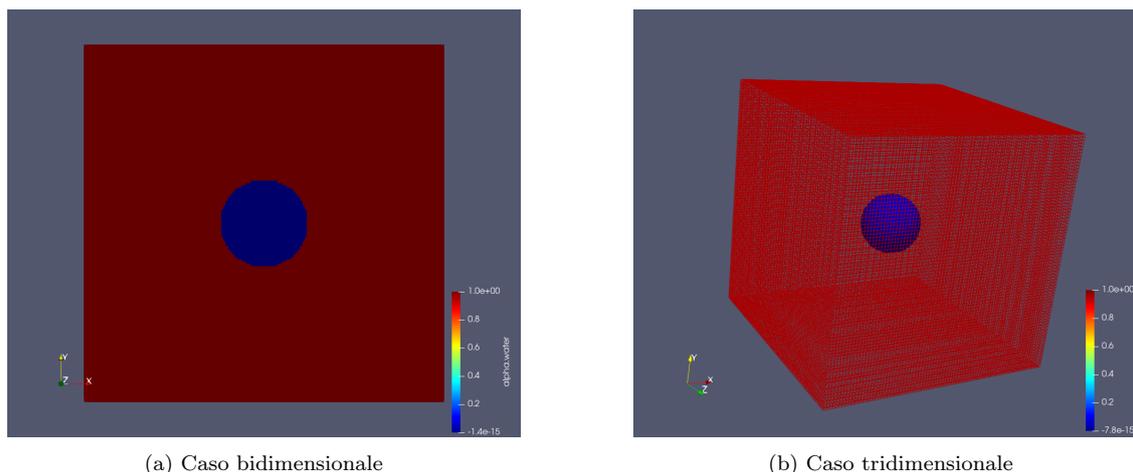


Figura 3.6: Imposizione della goccia al centro del dominio come condizione iniziale

3.2.2 Griglia di calcolo

La griglia di calcolo è anche essa impostata con l'applicazione `blockMesh`, che permette la creazione di griglie semplici, con celle esaedriche ed è adatta a geometrie non molto complesse, congruente col caso studiato. Nelle simulazioni effettuate in questo lavoro si utilizza una griglia uniforme su tutto il dominio, che è inoltre costituito da un solo blocco, a celle cubiche.

```

1 blocks
2 (
3     hex (0 1 2 3 4 5 6 7) (160 160 1) simpleGrading (1 1 1)
4 );

```

Listing 3.5: Definizione griglia di calcolo per simulazioni 2D

La definizione della griglia è presente nei listing 3.5 e 3.6, dove si può notare il numero di celle per lato del dominio, pari a 160 nella configurazione finale, mentre il risultato esemplificativo per il caso bidimensionale è in figura 3.7.

```

1 blocks
2 (
3     hex (0 1 2 3 4 5 6 7) (160 160 160) simpleGrading (1 1 1)
4 );

```

Listing 3.6: Definizione griglia di calcolo per simulazioni 3D

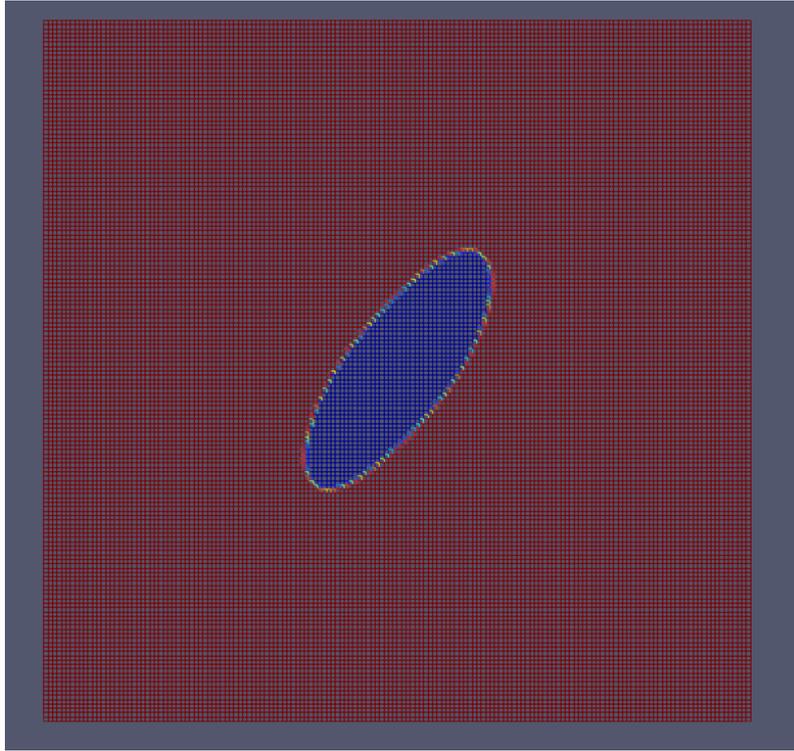


Figura 3.7: Griglia di calcolo utilizzata nel caso bidimensionale

La risoluzione della *mesh* iniziale è stata scelta partendo da un valore di 20 celle per diametro di goccia, tuttavia, fenomeni di diffusione numerica del campo scalare α causavano un distacco di materiale dalla goccia come in figura 3.8, che non trova un riscontro negli esperimenti alle condizioni operative simulate. A quel punto la griglia di calcolo è stata infittita fino al valore, doppio rispetto all'iniziale, di 40 celle per diametro, che ha permesso di ridurre molto i problemi di diffusione numerica, dovuti probabilmente ad una cattiva rappresentazione dell'interfaccia all'aumentare della curvatura.

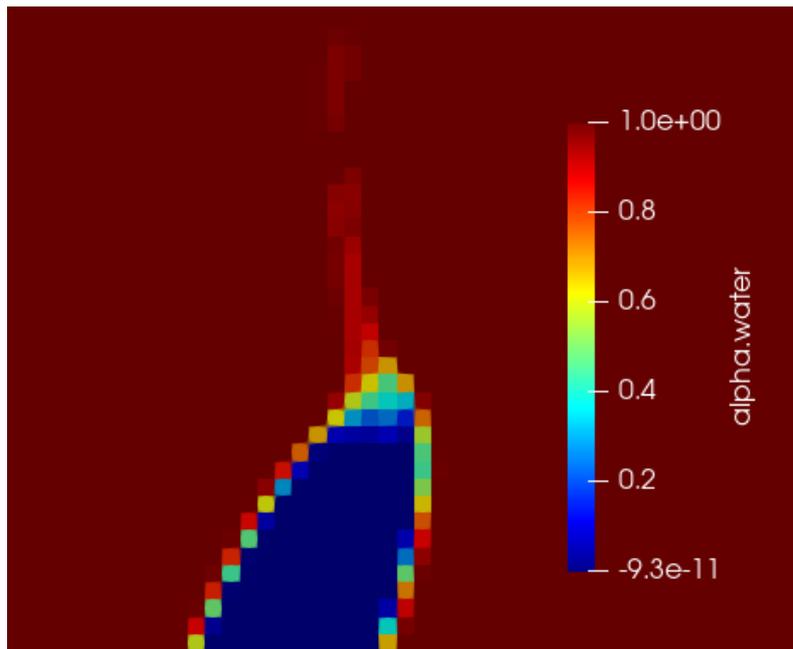


Figura 3.8: Distacco di materiale dalla goccia dovuto ad una rappresentazione troppo grossolana dell'interfaccia

3.3 Condizioni al contorno

In una simulazione di fluidodinamica computazionale le condizioni al contorno giocano un ruolo fondamentale nella corretta descrizione del problema da risolvere.

In OpenFOAM l'imposizione delle condizioni al contorno avviene tramite l'inizializzazione dei campi, effettuata attraverso i file presenti nella cartella del tempo iniziale 0. In questa si ritrovano due file, `U` in cui è possibile inizializzare il campo vettoriale di velocità e `p_rgh`, necessario per definire le condizioni al contorno della pressione modificata, campo scalare definito in equazione 3.8. Le condizioni di bordo per il campo scalare dell'indicatore di fase, o frazione volumica di fluido II α , vengono fornite tramite il file `alpha.water.orig`, ma l'inizializzazione, o imposizione delle condizioni iniziali, avviene tramite il comando `setFields`, che legge quanto scritto nel listing 3.4 generando il campo scalare.

3.3.1 Impostazione dei profili di velocità e `codedFixedValue`

In questo lavoro, l'analisi della deformazione della goccia è effettuata anche al variare del campo di moto, come descritto in sezione 2.1.4. Il parametro del *mixing index* diventa allora chiave per l'impostazione delle condizioni al contorno sulla velocità, poichè il campo vettoriale di velocità è direttamente dipendente dal valore di *mixing index* che si intende simulare.

I sistemi simulati presentano campi di velocità bidimensionali, nei quali il vettore velocità non varia lungo la direzione z , permettendo quindi l'espressione del

gradiente di velocità già visto in equazione 2.18, che viene qui richiamata:

$$\nabla \mathbf{U} = \begin{bmatrix} \frac{\partial U_x}{\partial x} & \frac{\partial U_y}{\partial x} & \frac{\partial U_z}{\partial x} \\ \frac{\partial U_x}{\partial y} & \frac{\partial U_y}{\partial y} & \frac{\partial U_z}{\partial y} \\ \frac{\partial U_x}{\partial z} & \frac{\partial U_y}{\partial z} & \frac{\partial U_z}{\partial z} \end{bmatrix} = G \begin{bmatrix} 0 & 1 & 0 \\ \theta & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad (3.22)$$

dove θ è legato al *mixing index* secondo la relazione in equazione 2.17.

Dall'integrazione delle componenti del gradiente di velocità, è possibile ottenere delle espressioni della velocità

$$\begin{cases} U_x = G \cdot \theta \cdot y & (3.23a) \\ U_y = G \cdot x & (3.23b) \\ U_z = 0 & (3.23c) \end{cases}$$

derivanti dall'imposizione di velocità nulla all'origine degli assi.

Dalle formule appena ricavate si notano le seguenti proprietà:

- entrambe le componenti non nulle sono direttamente proporzionali a G (s^{-1});
- ogni componente non dipende dal valore della coordinata nella direzione della componente stessa (*e.g.* U_x dipende da y e non da x);
- il parametro θ influisce solo sulla componente in direzione x .

Dopo aver ricavato le formule per i profili di velocità, queste sono testate attraverso simulazioni con una sola fase, imponendo un campo di velocità nullo all'interno del dominio e i valori dati dalle equazioni 3.23 ai bordi. Una volta che il campo di moto si è sviluppato, attraverso il *post-processing* su Paraview, dei *contour plot* del mixing index, come quello in figura 3.9, vengono analizzati, per verificare che i profili di velocità ricavati sui bordi portino ad ottenere il campo di moto effettivamente desiderato.

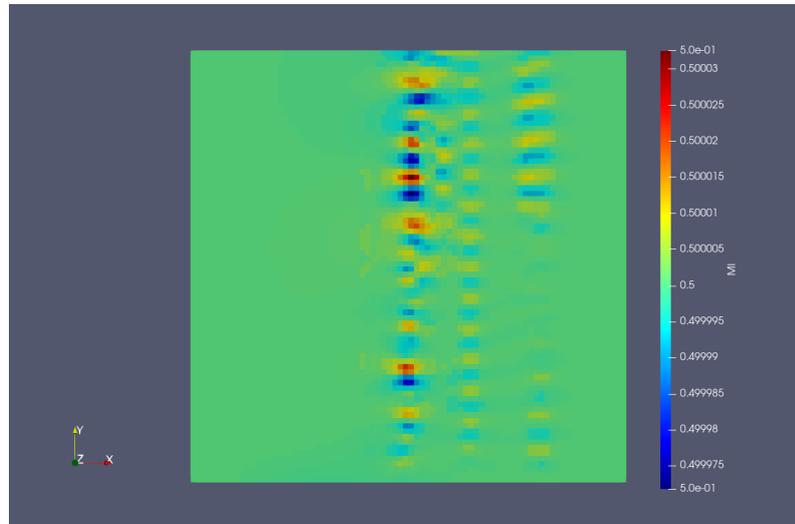


Figura 3.9: *Contour plot* del *mixing index* utilizzato per confermare la correttezza del campo di moto per la condizione $MI=0.5$

La formula utilizzata per computare il *mixing index* è la seguente:

$$MI = \frac{\text{strainRate} + \text{mag}(\text{vorticity})}{\text{strainRate}}, \quad (3.24)$$

già ricavata in equazione 2.16, in cui *vorticity* è un *utility* già presente in OpenFOAM, mentre *strainRate* è una *function object* scritta appositamente per calcolare $\dot{\gamma}$ secondo l'equazione 2.12, che è riportata in appendice B.

Il riscontro positivo del test appena descritto permette di inizializzare correttamente il vettore velocità nel *file* *0/U* con il campo di moto desiderato. Questo viene effettuato in modalità differenti, a seconda del tipo di condizione al contorno:

- Le condizioni iniziali all'interno del dominio sono dichiarate come campo vettoriale non uniforme, assegnando i valori cella per cella. La fattibilità di questo approccio è dovuta alla semplicità del dominio e della mesh utilizzata.
- Le condizioni di bordo alla frontiera del dominio sono dichiarate attraverso *codedFixedValue*, che permette di specificare una funzione del tempo e dello spazio. Per ovviare a problemi di tipo numerico, come verrà spiegato in sezione 3.6.1, questa funzione verrà accoppiata con il tracciamento del centro di massa, in modo da rendere il campo di moto correttamente impostato rispetto alla posizione della goccia, coerentemente con la condizione al contorno di velocità nulla al centro del dominio, utilizzata per ricavare le equazioni 3.23.
- La bidimensionalità del campo di moto, porta ad un trattamento differente per le facce *frontAndBack* descritte attraverso una condizione di gradiente nullo nelle simulazioni tridimensionali.

3.3.2 Impostazione pressione

Il problema simulato impone delle condizioni di pressione identiche su tutte le facce del dominio, la scelta ricade sulla condizione al contorno *fixedFluxPressure* pari a zero, che viene utilizzata quando la condizione è in generale di gradiente nullo ma nelle equazioni sono presenti forze di volume come quella legata alla tensione superficiale nel metodo VOF [31].

Bordo del dominio	Velocità	Pressione
north	<i>codedFixedValue</i>	<i>fixedMean</i>
south	<i>codedFixedValue</i>	<i>fixedFluxPressure</i>
east	<i>codedFixedValue</i>	<i>fixedFluxPressure</i>
west	<i>codedFixedValue</i>	<i>fixedFluxPressure</i>
<i>frontAndBack</i>	<i>empty</i>	<i>empty</i>

Tabella 3.1: Condizioni al contorno per le simulazioni bidimensionali

La scelta precedente porta a non inizializzare la pressione direttamente in nessun punto del dominio, ma l'imposizione di un valore di riferimento è necessaria per effettuare una simulazione CFD. Al contempo l'impostazione sullo stesso bordo di una condizione a valori fissi sia sulla pressione che sulla velocità è deleteria, in quanto una delle due proprietà deve essere lasciata libera di modificare il proprio valore a

seconda dei valori assunti dall'altra.

Dopo vari tentativi, la configurazione che ha permesso di ottenere i risultati corretti, relativamente al campo di moto desiderato, prevede l'utilizzo della condizione `fixedMean` sulla faccia `north`. Questa condizione permette di fissare il valore medio della pressione sulla faccia, di fatto permettendo fluttuazioni nel valore di pressione sulle facce delle singole celle facenti parte di quel bordo del dominio. In questo modo l'inizializzazione è attuabile in maniera corretta ed il campo di moto che si sviluppa è coerente con il mixing index scelto.

Le condizioni di bordo fin ora descritte sono riassunte in tabella 3.1 per le simulazioni 2D e in tabella 3.2 per le simulazioni 3D.

Bordo del dominio	Velocità	Pressione
<code>north</code>	<code>codedFixedValue</code>	<code>fixedMean</code>
<code>south</code>	<code>codedFixedValue</code>	<code>fixedFluxPressure</code>
<code>east</code>	<code>codedFixedValue</code>	<code>fixedFluxPressure</code>
<code>west</code>	<code>codedFixedValue</code>	<code>fixedFluxPressure</code>
<code>frontAndBack</code>	<code>zeroGradient</code>	<code>fixedFluxPressure</code>

Tabella 3.2: Condizioni al contorno per le simulazioni tridimensionali

3.3.3 Impostazione del campo α

Il campo scalare della funzione indicatore presenta la condizione di bordo di gradiente nullo per tutte le simulazioni. Poichè l'obiettivo è verificare a quali condizioni operative avviene la rottura della goccia, risulta di fondamentale importanza che la goccia non arrivi ad interagire con i bordi del dominio prima che la rottura sia accertata o smentita. Questo poichè la condizione di gradiente nullo, dipendentemente dal flusso di α presente al bordo durante l'interazione, può portare ad una fuoriuscita forzata o addirittura ad un ingresso di fluido II dai bordi del dominio stesso.

Come precedentemente illustrato, le condizioni iniziali per il campo α all'interno del dominio sono invece imposte tramite l'*utility* `setFields`.

L'inizializzazione più utilizzata corrisponde ad una goccia sferica al centro del dominio, tuttavia, in alcuni casi l'imposizione di un campo di moto non perturbato derivante dal valore di Ca che si vuole simulare porta ad una rottura prematura della goccia, ovvero a numeri di Ca_{crit} più bassi di quanto accada negli esperimenti. Per ovviare a questo problema, a seguito di una rottura, la simulazione viene re-inizializzata utilizzando il campo della funzione indicatore, ed anche il campo di velocità interno, della simulazione al valore di Ca immediatamente più basso che presentava il raggiungimento di una deformazione stazionaria.

3.4 Studio sui numeri adimensionali

In letteratura sono presenti numerosi studi sulla rottura di gocce disperse in fluidi con cui sono immiscibili. Una delle distinzioni maggiori per quanto riguarda le condizioni di rottura delle gocce è il regime fluidodinamico in cui si trova il sistema [6].

Come già visto in sezione 2.1.5, il sistema può essere descritto più efficacemente attraverso i numeri adimensionali che con l'utilizzo delle singole proprietà dei due fluidi. Allora, ad esempio, i singoli valori delle viscosità dei fluidi possono essere impostati in differenti maniere, a patto di mantenere il corretto valore del loro rapporto λ , che risulta più utile nella descrizione del sistema.

Lo studio della rottura è governato da λ e Ca solo nei casi in cui le forze inerziali sono trascurabili, quindi a bassi valori del numero di Reynolds, in caso contrario anche altri numeri come il rapporto di densità e il numero di Weber sono fondamentali nell'analisi del problema. Per ottenere dei risultati significativi nello studio qui presentato bisogna quindi garantire i corretti valori di Re .

I numeri adimensionali e le proprietà dei fluidi sono gestiti in questo modo nel corso delle simulazioni:

- I valori di densità sono tenuti costanti per i due fluidi in tutte le simulazioni, così come è tenuto costante il valore di viscosità cinematica della fase continua (fluido I). La viscosità cinematica della fase dispersa ν^{II} è invece variabile e impostata in modo da ottenere il valore di λ , rapporto delle viscosità dinamiche, desiderato per la simulazione.
- Il valore del numero di capillarità Ca viene fatto variare attraverso il parametro G , già descritto nelle sezioni 2.1.4 e 2.1.5.

Poichè il parametro G influenza anche il numero di Reynolds, la necessità di simulare Ca elevati in alcuni casi potrebbe portare a non rispettare la condizione $Re \ll 1$. Per ovviare a ciò il numero di capillarità viene gestito abbassando il valore di σ , in modo da poter simulare Ca elevati senza dover aumentare il numero di Reynolds eccessivamente.

Il concetto appena espresso è illustrato in figura 3.10, dove si nota come l'utilizzo di un valore di tensione interfacciale minore (figura 3.10a) permetta di ottenere valori di Reynolds più bassi a parità di numero di capillarità. Nei grafici è utilizzato il valore di Re pari a 1×10^{-2} come esempio di valore di soglia per escludere fenomeni inerziali.

3.5 Condizioni operative e simulazioni effettuate

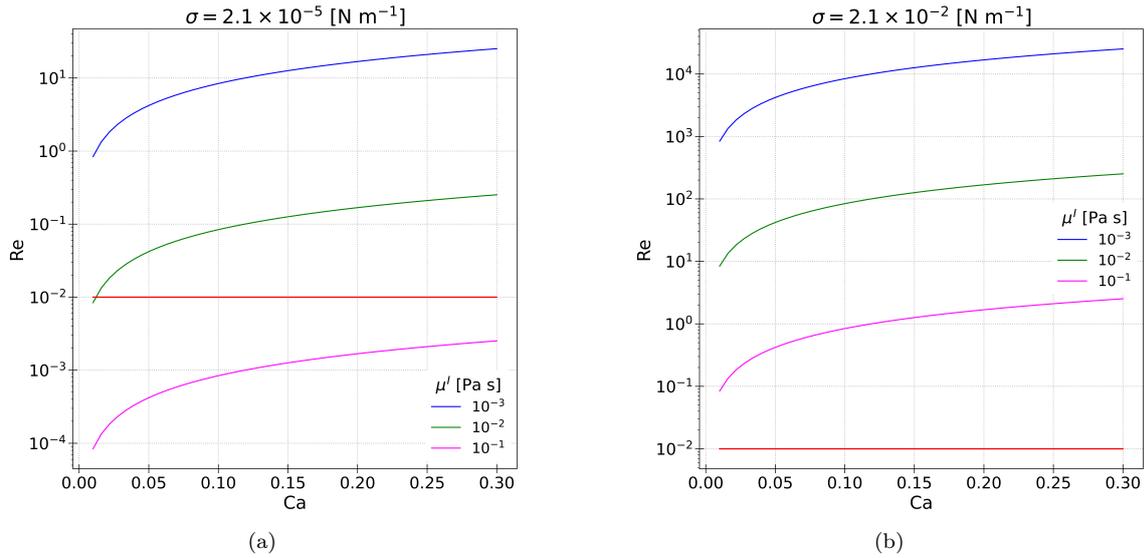
In questa sezione vengono riassunte le condizioni operative e il flusso di lavoro utilizzati nelle simulazioni. Lo studio ha come obiettivo quello di individuare il valore di Ca_{crit} , primo valore del numero di capillarità a cui avviene la rottura, al variare di λ e del tipo di campo di moto imposto.

Come evidenziato in sezione 2.1.4, vengono analizzati al massimo otto valori differenti di *mixing index*, corrispondenti a differenti campi di moto, dall'elongazionale puro al moto di taglio puro. Il valori di λ analizzati variano invece da 1×10^{-2} a 4, che è il valore di λ limite oltre il quale non è possibile rottura sotto le condizioni di moto di taglio puro [6], [9].

Il flusso di lavoro per l'impostazione delle simulazioni prevede i seguenti stadi:

1. Si imposta il valore di ν^{II} in `constant/transportPropries`, dipendentemente dal valore di λ da simulare:

$$\nu^{II} = \frac{\lambda \mu^I}{\rho^{II}} = \frac{\lambda \nu^I \rho^I}{\rho^{II}}. \quad (3.25)$$


 Figura 3.10: Valori assunti da Re e Ca al variare di μ^I e σ

2. Vengono lette dal nome della cartella di simulazione i valori di *mixing index* e del numero di capillarità Ca, in modo da calcolare il valore di θ tramite l'equazione 2.17 e il valore del parametro G :

$$G = \frac{Ca\sigma}{\mu^{II}a} = \frac{Ca\sigma}{\nu^{II}\rho^{II}a}. \quad (3.26)$$

3. Con i valori appena ricavati si aggiornano le condizioni di bordo sulla velocità e si imposta il valore iniziale per il campo di moto all'interno del dominio.
4. Si genera il dominio e la griglia di calcolo tramite il comando `blockMesh`.
5. Si impongono le due fasi all'interno del dominio tramite il comando `setFields`.
6. La simulazione può essere avviata tramite il comando `interFoam`, che richiama appunto il solutore.

In tabella 3.3 sono riassunte le proprietà dei fluidi che risultano costanti in tutte le simulazioni, tramite il file di configurazione `constant/transportProperties`.

Proprietà	Valore
ρ^I (kg m ³)	1000
ν^I (kg m ³)	1×10^{-04}
ρ^{II} (kg m ³)	917
σ (N m ⁻¹)	2.1×10^{-05}

Tabella 3.3: Proprietà dei fluidi utilizzate in tutte le simulazioni.

I valori di λ analizzati, insieme ai campi di moto indagati, sono invece riassunti in tabella 3.4 per entrambi i casi bidimensionale e tridimensionale. Le simulazioni 2D risultano molto meno onerose dal punto di vista computazionale, conseguentemente è stato possibile analizzare un numero più ampio di condizioni, mentre la

Tipologia		Condizioni operative							
		Mixing Index							
		0.45	0.5	0.55	0.6	0.7	0.8	0.9	1
2D	λ	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
		0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
		1	1	1	1	1	1	1	1
		4	4	4	4	4	4	4	4
		Mixing Index							
		0.5							
3D	λ	0.1							
		1							
		4							

Tabella 3.4: Condizioni operative simulate in questo lavoro

quantità di simulazioni tridimensionali effettuate sono limitate a causa delle risorse computazionali disponibili.

In particolare, nel caso bidimensionale, è stata analizzata anche la condizione $MI = 0.45$, con l'intento di convalidare il comportamento delle curve a *mixing index* costante, come si vedrà più in dettaglio nel capitolo 4.

3.6 Trattamento delle correnti parassite

Come già illustrato in sezione 2.3.3, uno dei problemi principali dei metodi VOF è legato alle correnti parassite, dette anche correnti spurie.

La problematica sorge a causa della difficoltà nella stima della curvatura in una griglia di calcolo discretizzata. Le correnti spurie vengono a generarsi a causa dei metodi utilizzati per una corretta avvezione dell'indicatore di fase e dell'interfaccia, che puntano ad evitare la diffusione numerica del campo scalare α .

Nel corso del tempo numerosi metodi per la riduzione delle correnti spurie sono stati proposti, che tengono in considerazione un trattamento diverso del termine di avvezione, un metodo di calcolo differente per la curvatura o, un passo temporale limitato.

I test utilizzati per la quantificazione delle correnti parassite consistono principalmente nell'inizializzazione di una goccia in un campo di moto nullo; di questa viene misurato lo spostamento all'interno del dominio, l'intensità delle velocità artificiali che si generano in corrispondenza dell'interfaccia e, in alcuni casi l'energia cinetica della goccia. [27]–[29], [39], [40]

In letteratura sono presenti vari approcci, che fanno riferimento al *framework* OpenFOAM, per la riduzione delle correnti spurie:

- In Deshpande *et al.* è proposto un criterio temporale, basato su un concetto simile al criterio CFL enunciato in sezione 2.2.2; il Δt_{\max} risulta dipendente dalle proprietà fisiche del sistema ρ, μ, σ [27].
- In Mukherjee *et al.* e in Inguva *et al.* i codici utilizzati implementano differenti calcoli per migliorare la risoluzione dell'interfaccia, in modo da risentire meno

dell'effetto della discretizzazione e ridurre la generazione di velocità artificiali [29], [39].

- Roenby *et al.* hanno proposto un nuovo algoritmo alternativo al MULES, chiamato isoAdvect, con l'obiettivo di mantenere un'interfaccia meno diffusa in seguito al processo di avvezione [40].

In questo lavoro la presenza di correnti spurie è un problema a causa dello sbilanciamento delle forze agenti sull'interfaccia, che portano ad un movimento della goccia. Spostamenti della goccia sono indesiderati per due motivazioni:

1. la necessità di evitare che la goccia venga a contatto con i bordi del dominio prima di aver appurato o scartato la rottura della goccia stessa, in quanto l'interazione con i bordi andrebbe a modificare l'evoluzione del campo scalare α , invalidano la simulazione;
2. la costruzione del campo di moto a partire dal *mixing index* prevede la presenza della goccia al centro del dominio: un suo eventuale spostamento farebbe cadere questa ipotesi e renderebbe non riproducibili i risultati ottenuti.

Poichè la maggior parte degli approcci elencati necessita di una modifica al codice di `interFoam` o utilizza dei solutori non presenti nella distribuzione ufficiale, il metodo impiegato per ridurre l'intensità delle correnti parassite si basa sul seguente criterio temporale [27]:

$$\Delta t \leq \max(10\tau_\mu, 0.1\tau_\rho), \quad (3.27)$$

dove

$$\tau_\mu = \frac{\mu^I \Delta x}{\sigma}, \quad (3.28a)$$

$$\tau_\rho = \sqrt{\frac{\rho^I \Delta x^3}{\sigma}}. \quad (3.28b)$$

3.6.1 Tracciamento del centro di massa

I metodi appena visti sono utili per la riduzione delle correnti parassite, ma non ne permettono la completa eliminazione, che è intrinsecamente presente nelle simulazioni a causa della necessità di rappresentare su una griglia discreta una superficie curva continua. Questo, nelle simulazioni, si traduce in uno spostamento della goccia, che, seppur contenuto, contravviene all'ipotesi con cui è calcolato il campo di moto.

Per ovviare a questo problema, viene implementato il tracciamento del centro di massa della goccia durante la simulazione, che permette di aggiornare le condizioni di bordo per la velocità. In questo modo i profili di velocità vengono ricalcolati in modo che il centro di massa si trovi nel punto in cui il campo di moto teorico non disturbato (quindi calcolato in assenza della goccia) abbia velocità nulla.

Da un punto di vista matematico, introducendo $(x_{C_{oM}}, y_{C_{oM}}, z_{C_{oM}})$ coordinate del centro di massa in cui la velocità deve essere nulla, le componenti del vettore velocità possono essere così calcolate:

$$\begin{cases} U_x = G \cdot \theta \cdot (y - y_{C_{oM}}) & (3.29a) \\ U_y = G \cdot (x - x_{C_{oM}}) & (3.29b) \\ U_z = 0 & (3.29c) \end{cases}$$

L'implementazione nel file `O/U` nella cartella di simulazione è effettuata tramite la condizione `codedFixedValue`, già accennata in sezione 3.3.1, che permette di leggere le coordinate delle facce delle celle ai bordi del dominio e calcolare i valori da imporre ad ogni passo temporale, in base alla posizione della goccia. In appendice C è riportato un esempio di implementazione di tracciamento del centro di massa e imposizione del profilo di velocità.

Capitolo 4

Risultati

In questa sezione vengono presentati i risultati riguardanti deformazione e rottura delle gocce. In sezione 4.1 sono esposti la maggior parte dei risultati, ottenuti tramite simulazioni in due dimensioni, e vengono analizzate le differenze con i dati sperimentali e le simulazioni numeriche presenti in letteratura. In sezione 4.2 sono presenti i risultati provenienti da alcune simulazioni in tre dimensioni, che vengono confrontati con i dati della letteratura, ponendo attenzione sui problemi derivanti dal dover ridurre i domini simulati a causa dei costi computazionali elevati.

4.1 Simulazioni bidimensionali

In questa sezione verranno presentati i risultati ottenuti dalle simulazioni bidimensionali, suddivisi secondo il valore del rapporto delle viscosità. Le simulazioni 2D permettono di limitare l'utilizzo di risorse computazionali, rendendo accettabile lo studio su un set di condizioni operative variabili ampio come quello presentato in questo lavoro.

I risultati qui presentati sono stati ottenuti imponendo il campo di moto indisturbato del fluido I sulla goccia di forma sferica al tempo iniziale, come descritto nel capitolo 3, e l'analisi è stata condotta con un incremento del numero di capillarità tra una simulazione e la successiva pari a 0.05.

4.1.1 Rottura a rapporti di viscosità elevati

A valori del rapporto di viscosità elevati è valida la teoria delle piccole deformazioni, secondo la quale la goccia non va incontro a grandi elongazioni prima della rottura.

In questo lavoro, le condizioni operative esaminate corrispondenti a questa teoria sono $\lambda = 4$ e $\lambda = 1$. I valori di Ca_{crit} ricavati per questi due casi al variare del tipo di campo di moto sono esposti in tabella 4.1 per il primo caso e in tabella 4.2 per il secondo caso.

Dai risultati ottenuti si nota come all'aumentare del *mixing index*, quindi passando da moto di puro taglio a moto puramente elongazionale, il valore del numero di capillarità critico diminuisca. Questo risultato è in linea con quanto riportato in letteratura [1], [5]. La stessa aderenza non si ritrova nel valore critico per il *mixing index* pari a 0.5 in condizioni di rapporto di viscosità massimo tra quelli studiati: Dubbelboer *et al.* [4] riportano $\lambda = 4$ come valore limite oltre il quale non è possibile ottenere rottura in moto di puro taglio. Il valore ottenuto di $Ca_{\text{crit}} = 0.44$

Mixing Index	Ca _{crit}
0.5	0.44
0.55	0.28
0.6	0.23
0.7	0.18
0.8	0.16
0.9	0.14
1	0.125

Tabella 4.1: Valori di numero di capillarità critico individuati per λ pari a 4

non è in linea con i valori sperimentalmente presenti in letteratura, che approssimandosi al valore limite di rapporto di viscosità presentano numeri di capillarità critici nell'intorno o anche superiori alla decina.

Mixing Index	Ca _{crit}
0.5	0.395
0.55	0.29
0.6	0.245
0.7	0.205
0.8	0.18
0.9	0.16
1	0.14

Tabella 4.2: Valori di numero di capillarità critico individuati per λ pari a 1

Per quanto riguarda $\lambda = 1$, il valore critico ottenuto per il moto di taglio puro è confrontabile con la relazione sperimentale ottenuta da De Bruijn, già mostrata in equazione 1.1, e qui riproposta:

$$\log \text{Ca}_{\text{crit}} = -0.506 + 0.0994 \log \lambda + 0.124 \log \lambda^2 + \frac{-0.115}{\log \lambda - \log \lambda_{\text{max}}}, \quad (4.1)$$

che presenta un valore critico di 0.48, con una differenza percentuale del 21.5% rispetto a quanto presente in tabella 4.2.

I risultati di simulazioni numeriche riportati da Rallison [15] per le stesse condizioni operative si presentano simili a quelli ottenuti in questo lavoro per bassi valori del *mixing index*, ovvero per campi di moto a componente rotazionale maggiore. All'aumentare del *mixing index* le differenze diventano più importanti, come si nota in tabella 4.3.

I valori risultanti dalle simulazioni a *mixing index* maggiori di 0.5 non possono essere direttamente confrontati con valori sperimentali, ma da un confronto qualitativo, presentato in figura 4.1, si può notare come i valori ottenuti dalle simulazioni presentino dei numeri di capillarità critici leggermente più elevati rispetto a quelli ottenuti sperimentalmente da Bentley e Leal [1], ma comunque rispettino gli stessi andamenti sia al variare del rapporto di viscosità che del *mixing index*.

Il *post-processing* delle simulazioni permette di notare alcune peculiarità tipiche della teoria delle piccole deformazioni: in figura 4.2 si può notare come raggiunta la

Mixing Index	Ca_{crit} (Rallison, 1981)	Differenza percentuale
0.5	0.422	6.8%
0.6	0.259	5.7%
0.7	0.198	3.4%
0.8	0.161	10.6%
0.9	0.129	19.4%
1	0.116	17.1%

Tabella 4.3: Valori ottenuti da Rallison [15] e confronto con i valori ottenuti mediante le simulazioni VOF per $\lambda = 1$

deformazione stazionaria la parte centrale della goccia tenda ad una forma cilindrica, come riportato da Suhas [6]. La figura 4.2 mostra inoltre come il campo interno della

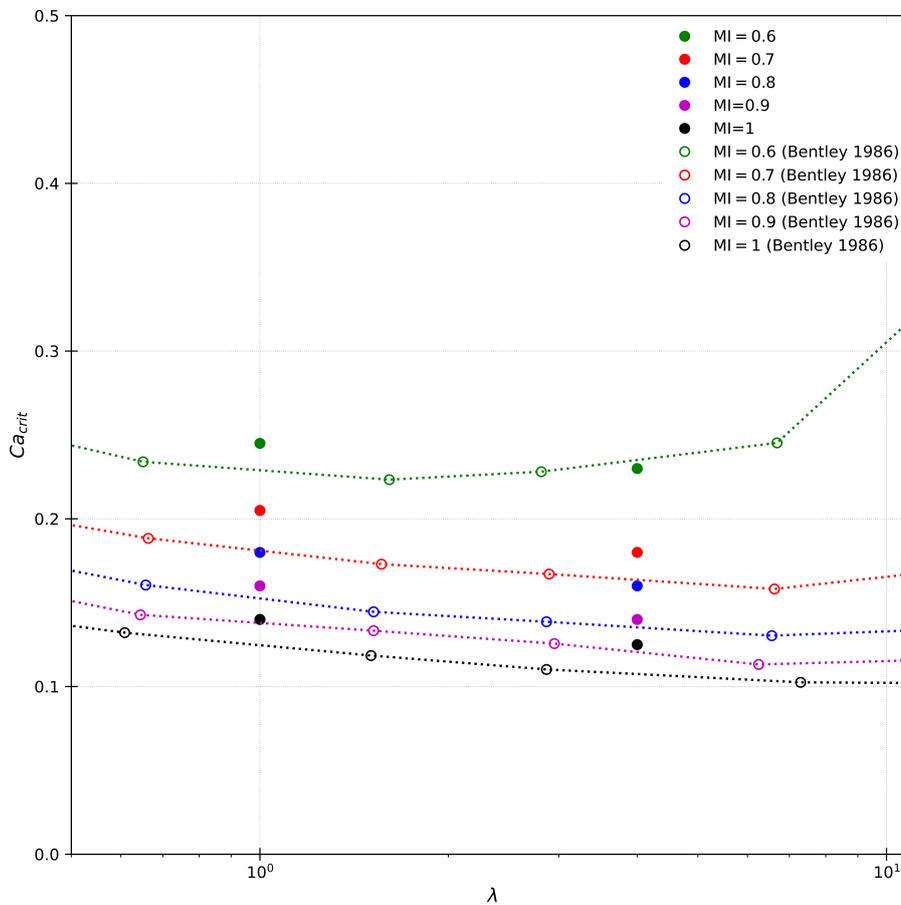


Figura 4.1: Confronto tra i dati ottenuti tramite simulazione VOF, rappresentati dai cerchi pieni, e i dati sperimentali riportati da Bentley e Leal [1], rappresentati dai cerchi vuoti

goccia allo stato stazionario diventi di tipo rotazionale in caso di moto di taglio puro, diversamente da quanto accade nel caso di moto elongazionale, in figura 4.3. In quest'ultimo caso, inoltre, alle estremità della goccia deformata sono visibili dei filetti di fluido fuoriuscenti dal corpo principale, mentre sperimentalmente per valori elevati del rapporto di viscosità le gocce immerse in un campo di moto irrotazionale mantengono delle estremità arrotondate [1], [6]. Questo fenomeno non è riportato in letteratura nelle condizioni studiate ed è da attribuirsi esclusivamente alla diffusione

numerica, negli esperimenti qui condotti è stato possibile ridurlo infitendo la griglia computazionale.

La fuoriuscita di materiale dalle estremità della goccia porta ad una diminuzione della frazione volumica di fluido II presente all'interno del dominio, influenzando di conseguenza i valori del numero di capillarità critico individuati nei vari casi, probabilmente contribuendo ad una leggera sovrastima per i campi di moto a componente elongazionale più marcata. Il fenomeno risulta infatti più importante all'aumentare del *mixing index* e al diminuire del rapporto di viscosità, e dai risultati traspare una riduzione massima della frazione volumica della goccia di 0.39% per $\lambda = 4$ e di 1.59% per $\lambda = 1$, entrambe ottenute per simulazioni che non presentavano rottura della goccia e ad un tempo massimo di simulazione di 1500 s.

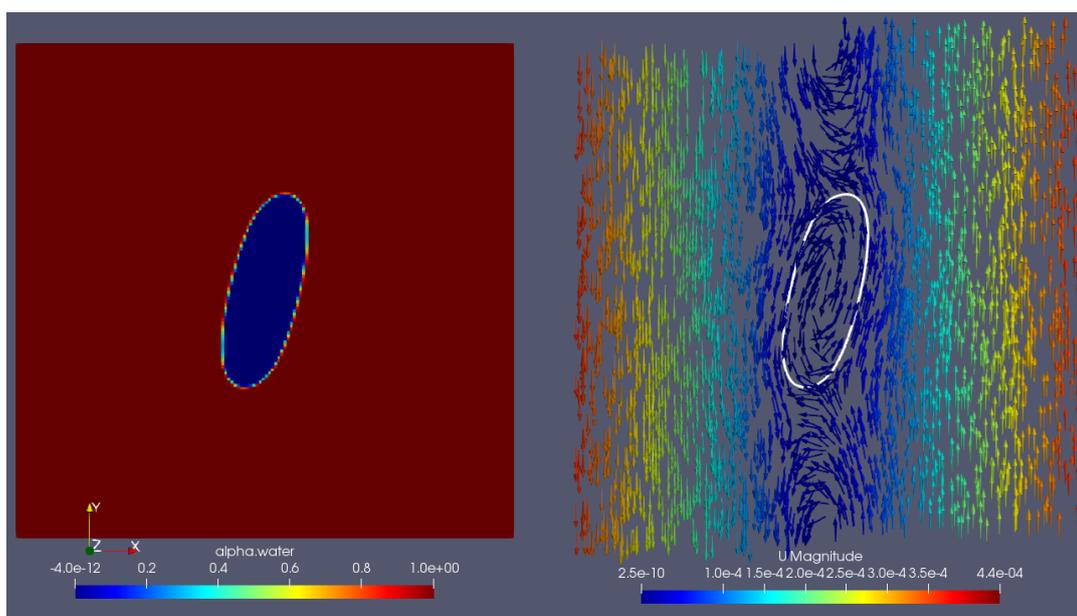


Figura 4.2: Indicatore di fase e vettore velocità per una goccia a deformazione stazionaria simulata in condizioni: $\lambda = 4$, moto di taglio puro, $Ca = 0.43$. Il *contour* in bianco che rappresenta la superficie della goccia è ottenuto ad $\alpha = 0.5$.

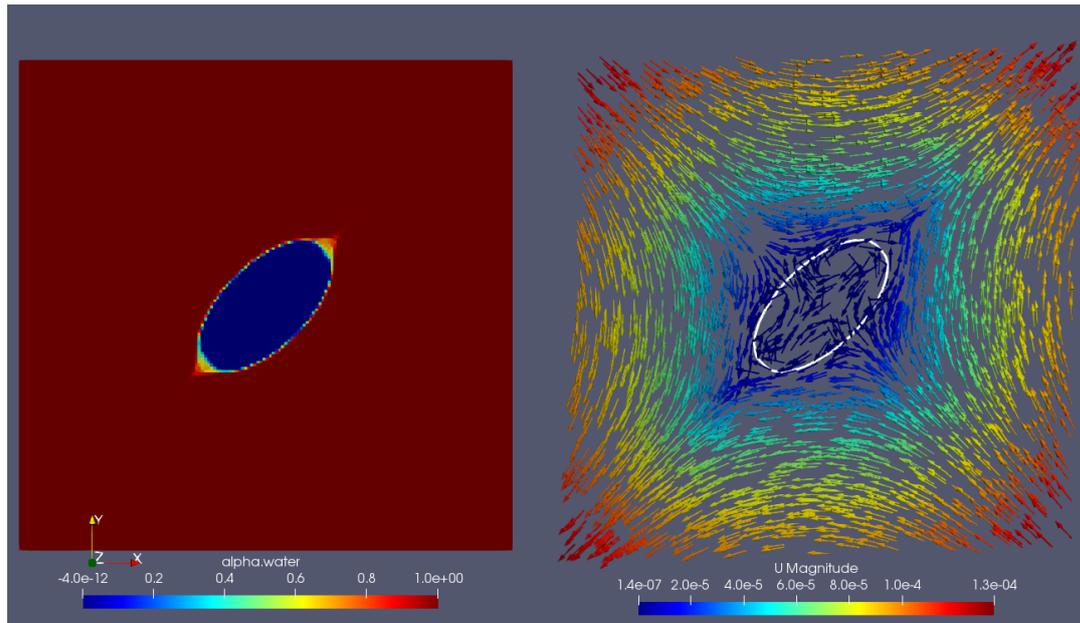


Figura 4.3: Indicatore di fase e vettore velocità per una goccia a deformazione stazionaria simulata in condizioni: $\lambda = 4$, moto puramente elongazionale, $Ca = 0.11$. Il *contour* in bianco che rappresenta la superficie della goccia è ottenuto ad $\alpha = 0.5$

4.1.2 Rottura a rapporti di viscosità bassi

A bassi valori del rapporto di viscosità le gocce si comportano secondo la teoria del corpo snello (*slender body*). Questa prevede il raggiungimento di forme molto elongate, in cui la deformazione sia molto accentuata, ovvero, in riferimento alla figura 4.4, in cui $L \gg B$.

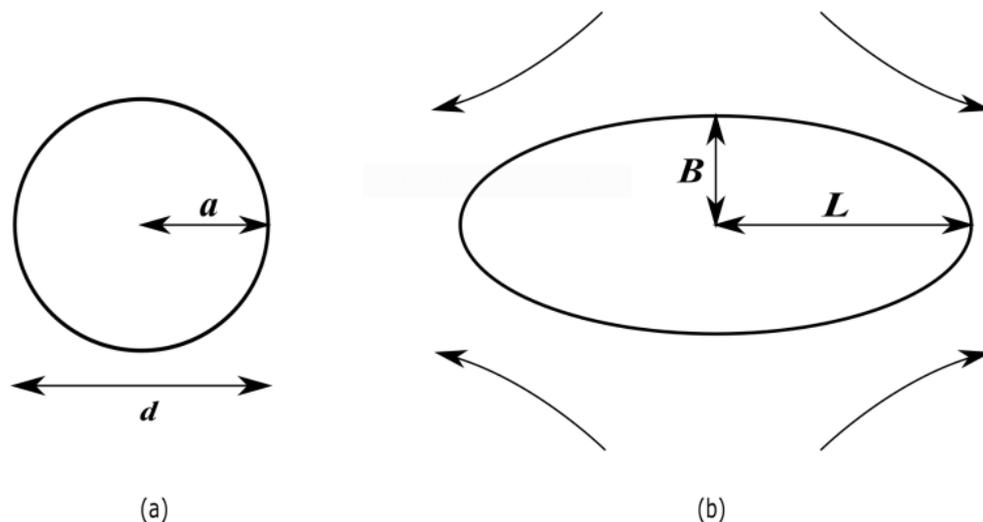


Figura 4.4: Morfologia della goccia. (a) Goccia inizialmente sferica. (b) Goccia deformata sotto l'effetto di un campo di moto esterno [6]

In questo lavoro vengono studiati per questo caso i valori di $\lambda = 0.1$ e $\lambda = 0.01$, presentati rispettivamente in tabella 4.4 e 4.5

Dalle tabelle appena mostrate un primo confronto può essere effettuato con i valori del numero di capillarità critica ottenuti tramite la correlazione sperimentale

Mixing Index	Ca_{crit}
0.5	0.465
0.55	0.375
0.6	0.295
0.7	0.25
0.8	0.225
0.9	0.2
1	0.18

Tabella 4.4: Valori di numero di capillarità critico individuati per λ pari a 0.1

Mixing Index	Ca_{crit}
0.5	0.58
0.55	0.47
0.6	0.345
0.7	0.275
0.8	0.25
0.9	0.225
1	0.2

Tabella 4.5: Valori di numero di capillarità critico individuati per λ pari a 0.01

di De Bruijn [16] per bassi valori del rapporto di viscosità, già mostrata in equazione 1.2, e qui riproposta:

$$\log Ca_{\text{crit}} = -1.560 - 0.733 \log \lambda + \frac{-1.135}{\log \lambda - \log \lambda_{\text{max}}}. \quad (4.2)$$

Da questa è possibile ottenere, per un campo di moto di taglio puro con $\lambda = 0.1$, un valore di $Ca_{\text{crit}} = 0.56$, che presenta una differenza percentuale pari al 20.4% rispetto al valore ottenuto dalle simulazioni VOF, in cui $Ca_{\text{crit}} = 0.465$.

L'errore aumenta considerevolmente per il valore del rapporto di viscosità più basso analizzato in questo studio: nel caso di $\lambda = 0.01$ il valore ottenuto dalla correlazione sperimentale è pari a 1,94, quasi quattro volte superiore a quanto riportato in tabella 4.5 per *mixing index* pari a 0.5.

Queste differenze si ritrovano anche nel confronto con i risultati riportati da Bentley e Leal [1], che sono rappresentati in figura 4.5 per valori del *mixing index* superiori a 0.5. In questo caso si nota un buon accordo tra le simulazioni e i risultati sperimentali per $\lambda = 0.1$, in special modo per campi di moto con componente elongazionale più importante, con le differenze maggiori presenti per $MI = 0.6$. Lo stesso non si può affermare per i valori di $\lambda = 0.01$: questi sono effettivamente maggiori rispetto a quelli ottenuti per $\lambda = 0.1$, ma i valori ottenuti dalle simulazioni VOF sono molto minori rispetto a quelli ottenuti attraverso un apparato sperimentale. Mentre viene quindi confermato come al diminuire del rapporto di viscosità siano richiesti, a parità delle altre variabili, sforzi viscosi maggiori per rompere una goccia, viene meno l'affidabilità delle simulazioni per valori così bassi del rapporto di viscosità.

Questi risultati possono essere spiegati attraverso l'interazione della goccia con i bordi del dominio, analizzando ciò che avviene in figura 4.6 e 4.7.

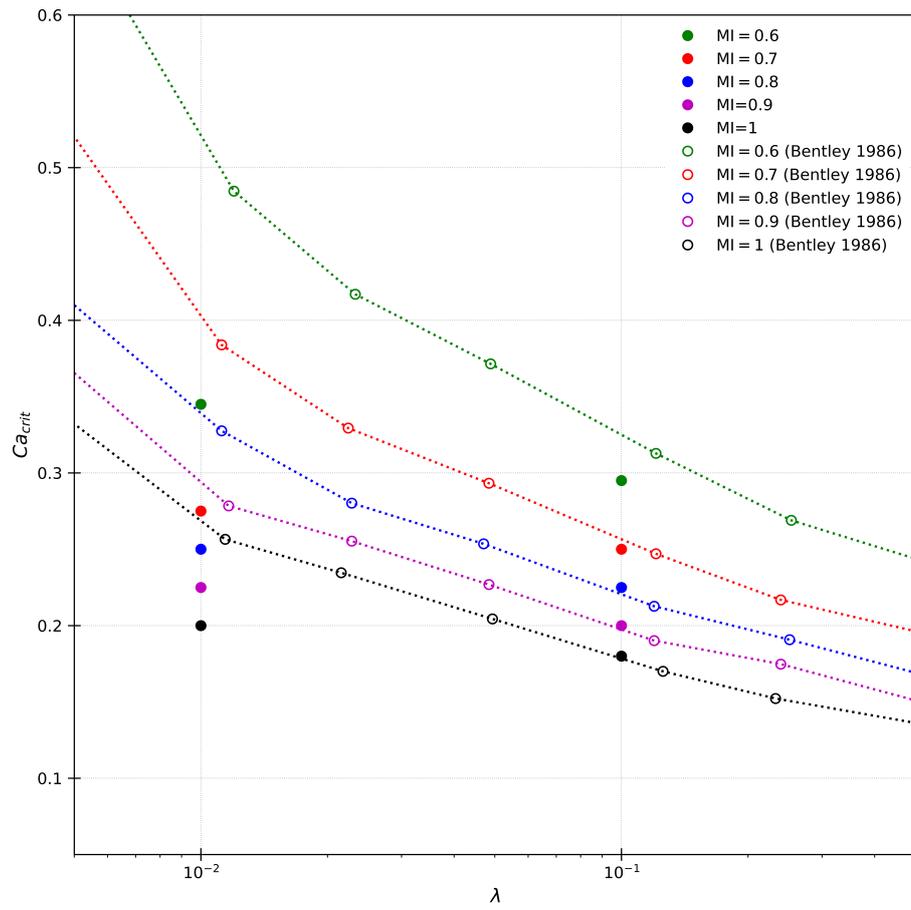


Figura 4.5: Confronto tra i dati ottenuti tramite simulazione VOF, rappresentati dai cerchi pieni, e i dati sperimentali riportati da Bentley e Leal [1], rappresentati dai cerchi vuoti

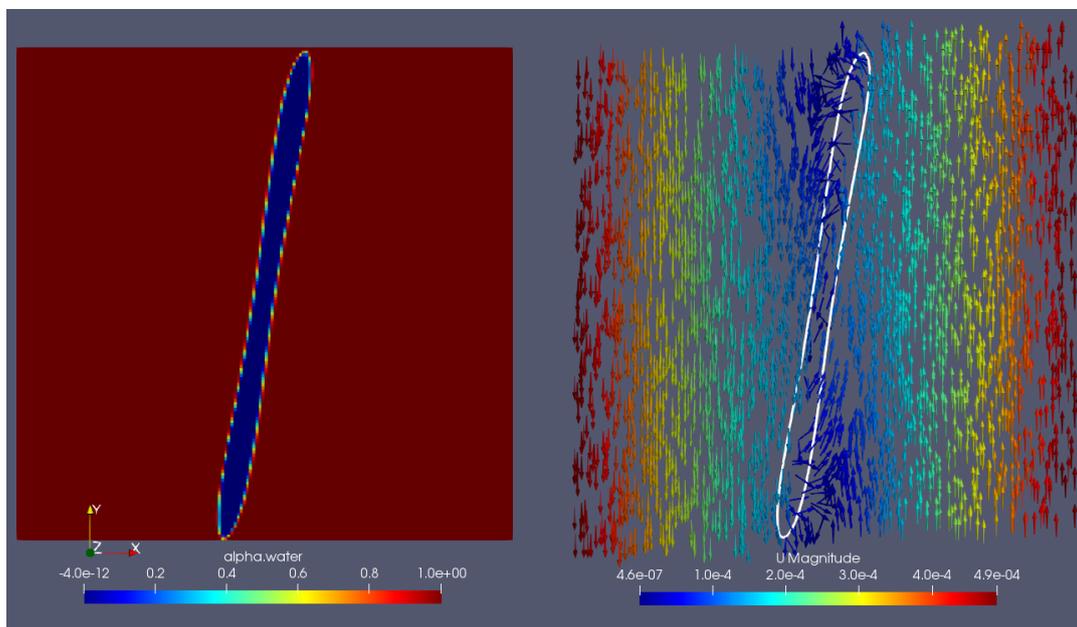


Figura 4.6: Indicatore di fase e vettore velocità per una goccia simulata alle condizioni di rottura: $\lambda = 0.01$, moto di taglio puro, $Ca = 0.58$. Il *contour* in bianco che rappresenta la superficie della goccia è ottenuto ad $\alpha = 0.5$.

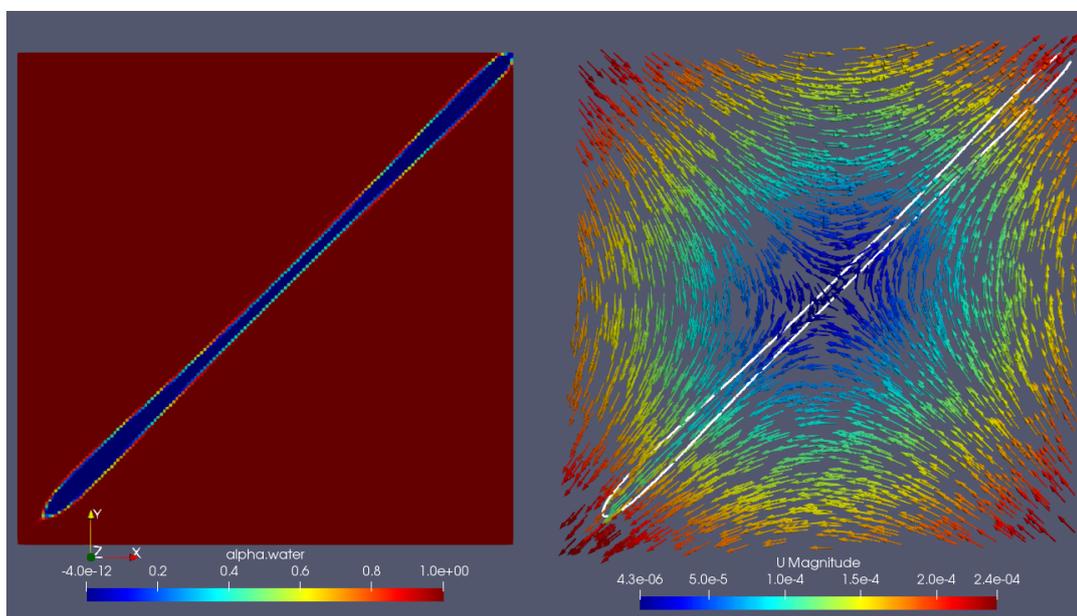


Figura 4.7: Indicatore di fase e vettore velocità per una goccia simulata alle condizioni di rottura: $\lambda = 0.01$, moto puramente elongazionale, $Ca = 0.2$. Il *contour* in bianco che rappresenta la superficie della goccia è ottenuto ad $\alpha = 0.5$.

La distanza dei bordi del dominio può influire sulla dinamica della rottura, come riportato da Li *et al.* [30], e il contatto tra la superficie della goccia ed i bordi del dominio porta ad un “effetto trazione”: l'impostazione della condizione al bordo `zeroGradient` per il campo scalare dell'indicatore di fase α aumenta infatti la fuoriuscita della goccia dal dominio come se questa venisse tirata nelle zone in cui è avvenuto il contatto. Le immagini 4.6 e 4.7 mostrano i due casi limite di *mixing index* analizzato, ma sono rappresentativi della dinamica di tutte le rotture per $\lambda = 0.01$: l'avvicinamento ed il contatto della superficie della goccia avvengono prima della completa separazione del corpo iniziale delle goccia in due o più particelle.

Un'ovvia soluzione al problema sarebbe l'ingrandimento del dominio, in modo da permettere l'elongazione della goccia senza andare incontro a interazioni della stessa con i bordi del dominio computazionale. Tuttavia mentre un approccio del genere è stato utilizzato da Bentley e Leal, il cui apparato sperimentale permetteva alla goccia un'elongazione di 25 volte il raggio iniziale senza andare incontro a deviazioni dalla linearità dei campi di moto [1], da un punto di vista numerico un aumento di dimensioni del dominio di questo tipo potrebbe essere svantaggioso a causa del grande aumento del tempo computazionale necessario per terminare le simulazioni.

Nelle simulazioni che raggiungono uno stazionario, ovvero quando non avviene rottura, si verifica nuovamente il distacco di materia dal corpo centrale della goccia, come illustrato nelle figure 4.8 e 4.9. Mentre nei casi esposti in sezione 4.1.1 non era presente distacco di materia nelle simulazioni in moto di taglio puro, per valori di λ bassi, il fenomeno si riscontra anche a $MI = 0.5$, anche se in questi casi la diminuzione della frazione volumica di fluido II risulta molto contenuta per bassi valori di *mixing index*. I valori ottenuti confermano la dipendenza di questo fenomeno di diffusione numerica dal tipo di campo di moto e dal rapporto di densità, come esposto in sezione 4.1.1. In particolare la diminuzione massima di frazione volumica si ha per $MI = 1$ ed è pari a 5.4% per $\lambda = 0.1$ e a 8.6% per $\lambda = 0.01$.

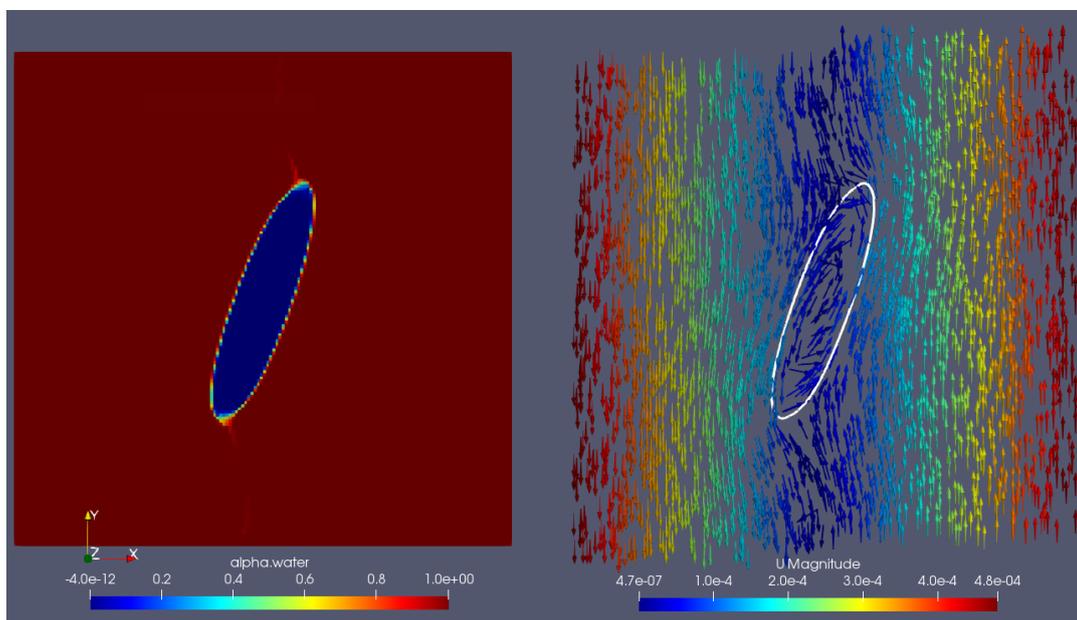


Figura 4.8: Indicatore di fase e vettore velocità per una goccia a deformazione stazionaria simulata in condizioni: $\lambda = 0.01$, moto di taglio puro, $Ca = 0.57$. Il *contour* in bianco che rappresenta la superficie della goccia è ottenuto ad $\alpha = 0.5$.

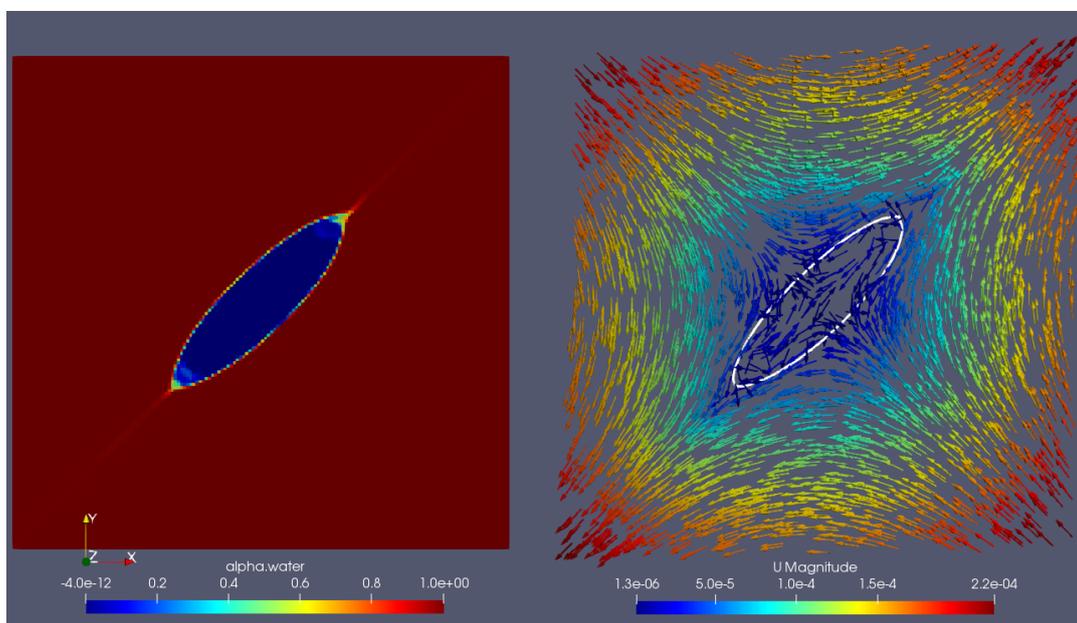


Figura 4.9: Indicatore di fase e vettore velocità per una goccia a deformazione stazionaria simulata in condizioni: $\lambda = 0.01$, moto puramente elongazionale, $Ca = 0.19$. Il *contour* in bianco che rappresenta la superficie della goccia è ottenuto ad $\alpha = 0.5$

4.1.3 Deformazione delle gocce

Dalle simulazioni effettuate è possibile ricavare anche informazioni sulla deformazione a cui la goccia va incontro nelle varie condizioni analizzate. Queste informazioni sono utili in quanto l'entità dei processi di scambio di materia ed energia in un sistema disperso è direttamente proporzionale all'area superficiale tra le fasi. Quest'area dipende ovviamente dalla forma della goccia, che varia a seconda delle condizioni del campo di moto e del rapporto delle viscosità λ .

La deformazione di una goccia venne espressa per la prima volta da Taylor [5] con la seguente formula:

$$D = \frac{L - B}{L + B}, \quad (4.3)$$

dove L e B sono gli assi di una goccia deformata, come in figura 4.4. Tuttavia per gocce molto deformate, cioè per L/B elevato, l'equazione 4.3 restituisce un valore di D che varia poco al crescere della deformazione, raggiungendo un valore asintotico unitario [1], [6]. Per ovviare a tale problema è possibile utilizzare la seguente equazione:

$$\frac{L}{a} = \left(\frac{1 + D}{1 - D} \right)^{\frac{2}{3}}, \quad (4.4)$$

che ipotizza una goccia che assuma la forma di un ellissoide prolato in seguito alla deformazione, in cui a è il raggio della goccia non deformata e D è calcolato tramite l'equazione 4.3 [6].

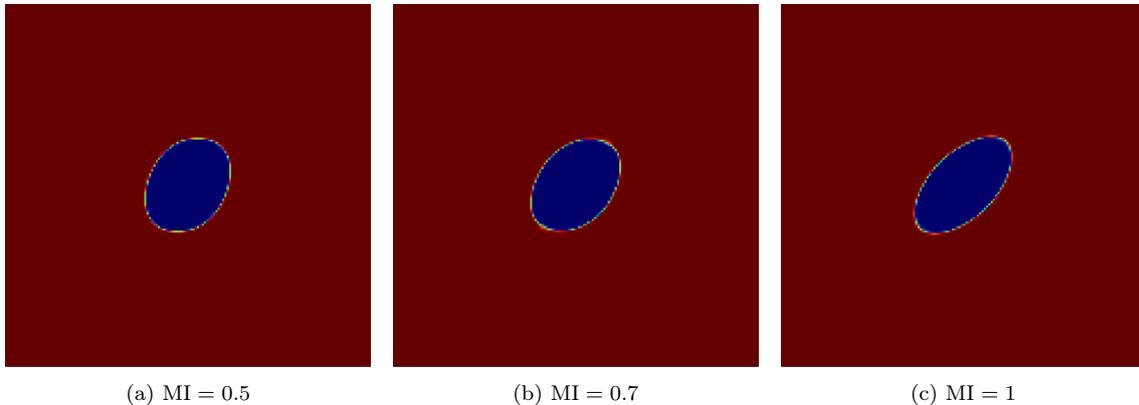


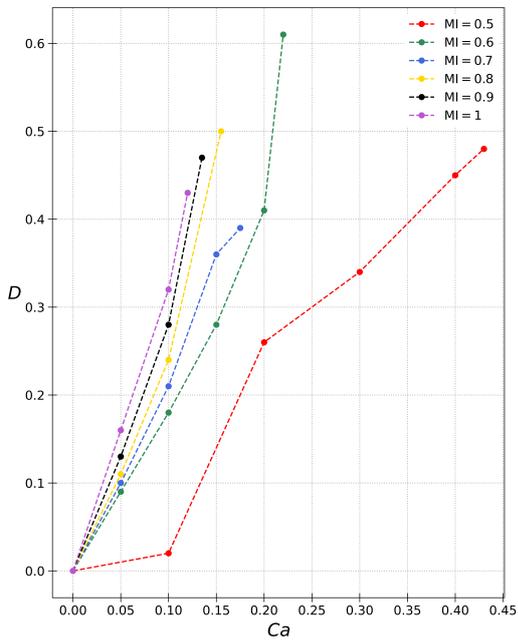
Figura 4.10: Confronto tra gocce deformate per $\lambda = 1$ e $Ca=0.1$ al variare del mixing index

Nelle figure 4.11 e 4.12 sono mostrate le deformazioni calcolate su alcune delle condizioni operative simulate, rispettivamente utilizzando D calcolato secondo l'equazione 4.3 e L/a secondo l'equazione 4.4. I valori di L e B sono valutati tramite analisi delle immagini fornite dal software di *post-processing* Paraview.

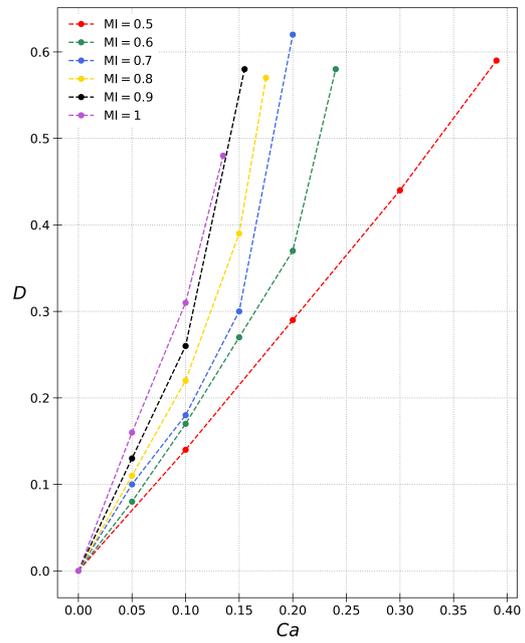
Dal confronto delle due figure si nota come le differenze ad alte deformazioni risultino meno pronunciate utilizzando il parametro D (figura 4.11), rispetto a quelle ottenute con il parametro L/a (figura 4.12), in particolar modo per valori del rapporto di viscosità bassi. Risultano invece invariati gli andamenti, per cui, in tutti i casi, a parità del numero di capillarità si ha una deformazione maggiore al crescere del *mixing index* o, equivalentemente, per ottenere la stessa deformazione è necessario un numero di capillarità via via maggiore passando da moto puramente elongazionale

a moto di taglio puro. In figura 4.10 sono rappresentate le deformazioni di gocce al medesimo valore del rapporto di viscosità e del numero di capillarità, in modo da mostrare gli andamenti appena descritti al variare del tipo di campo di moto.

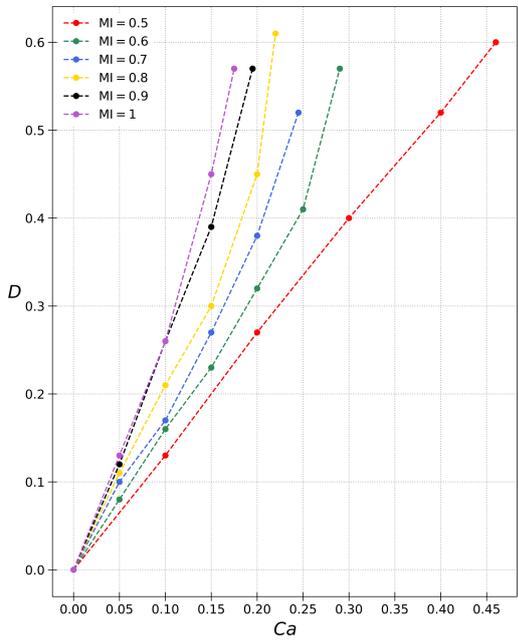
Analizzando il comportamento al variare del rapporto di viscosità si nota come al diminuire di λ la deformazione massima prima della rottura aumenti, seguendo le teorie precedentemente enunciate. Un'eccezione si presenta per $\lambda = 0.01$, in cui solo per bassi valori del *mixing index* è presente una deformazione molto pronunciata, che si può notare solo adottando il parametro L/a per descriverla, in figura 4.12d. Questo comportamento è in linea con quanto già visto in sezione 4.1.2, per cui risultati ottenuti dalle simulazioni con valori di λ molto bassi non sono sufficientemente in accordo con i risultati sperimentali presenti in letteratura.



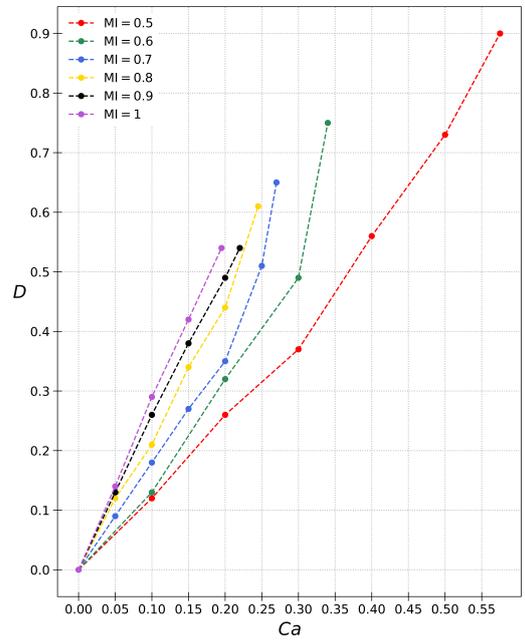
(a) Deformazione D della goccia al variare di *mixing index* e Ca per $\lambda = 4$



(b) Deformazione D della goccia al variare di *mixing index* e Ca per $\lambda = 1$



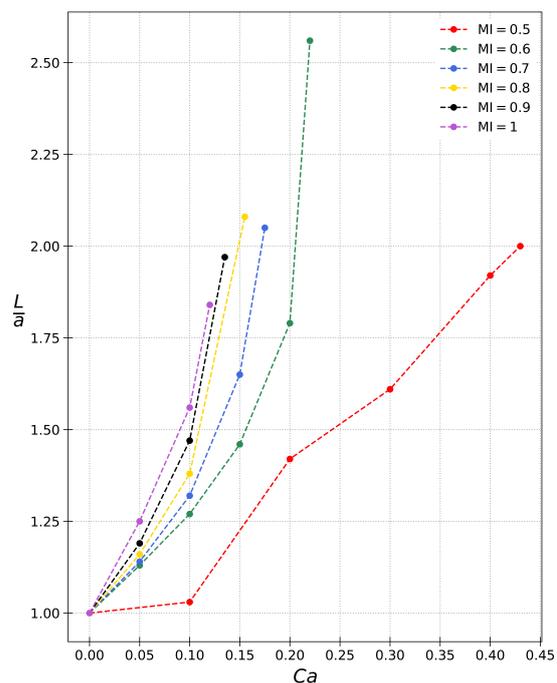
(c) Deformazione D della goccia al variare di *mixing index* e Ca per $\lambda = 0.1$



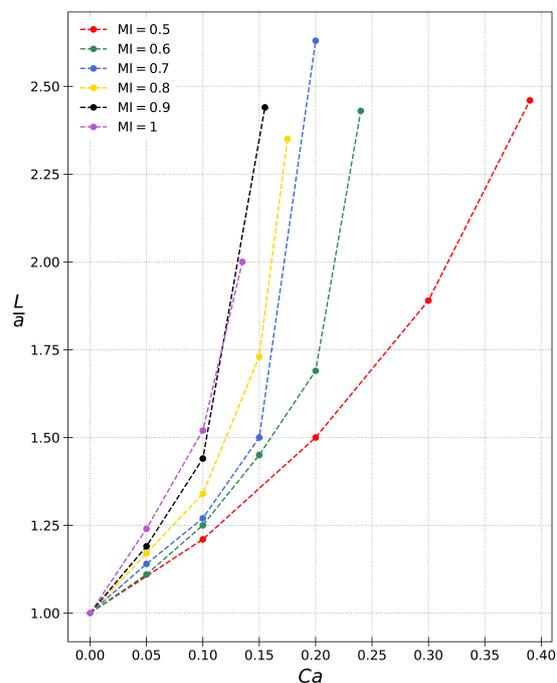
(d) Deformazione D della goccia al variare di *mixing index* e Ca per $\lambda = 0.01$

Figura 4.11: Deformazioni D delle gocce al variare delle condizioni analizzate

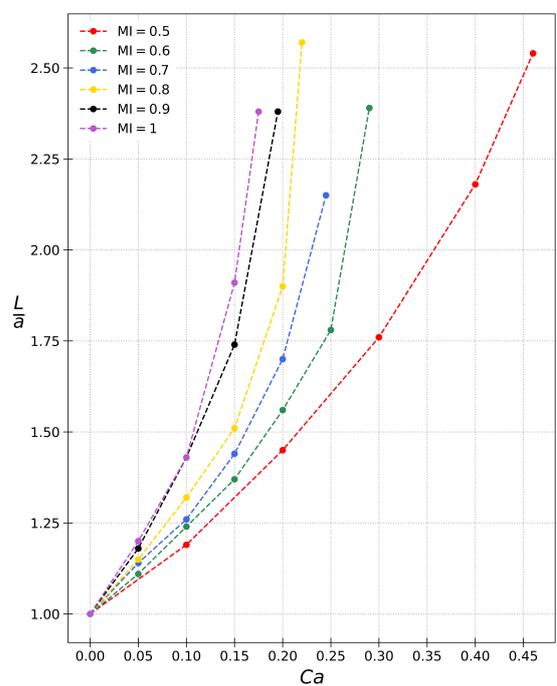
4.1. Simulazioni bidimensionali



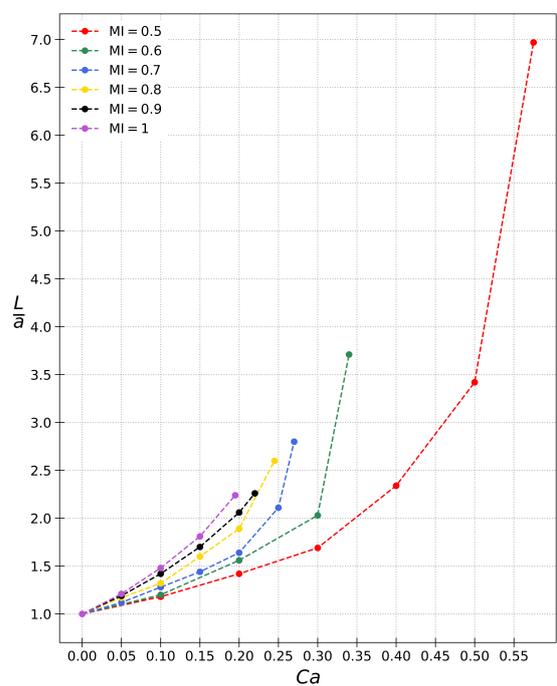
(a) Deformazione L/a della goccia al variare di *mixing index* e Ca per $\lambda = 4$



(b) Deformazione L/a della goccia al variare di *mixing index* e Ca per $\lambda = 1$



(c) Deformazione L/a della goccia al variare di *mixing index* e Ca per $\lambda = 0.1$



(d) Deformazione L/a della goccia al variare di *mixing index* e Ca per $\lambda = 0.01$

Figura 4.12: Deformazioni L/a delle gocce al variare delle condizioni analizzate

4.1.4 Dipendenza dal *mixing index* e campi di moto rotazionali

I risultati precedentemente esposti mostrano come il valore del numero di capillarità critico dipenda dal tipo di campo di moto in cui è immersa la goccia e, in particolare, dai dati riportati in letteratura sembra esserci una differenza crescente tra le condizioni critiche all'aumentare della componente rotazionale nel campo di moto, ovvero al diminuire del *mixing index*.

Per confermare questa tendenza le simulazioni effettuate comprendono anche due valori di *mixing index* difficili da ritrovare in letteratura. Si è analizzata quindi la condizione $MI = 0.55$, in modo da monitorare la transizione tra $MI = 0.5$ ed $MI = 0.6$, che presenta le differenze più ampie in termini di Ca_{crit} e la condizione $MI = 0.45$, in modo verificare se la tendenza continuasse anche per campi di moto con componente rotazionale più importante di quella elongazionale.

In figura 4.13 si può qualitativamente osservare come i punti che descrivono le condizioni critiche, a parità di rapporto di viscosità, si allontanino tra loro in maniera maggiore al diminuire del valore di *mixing index*.

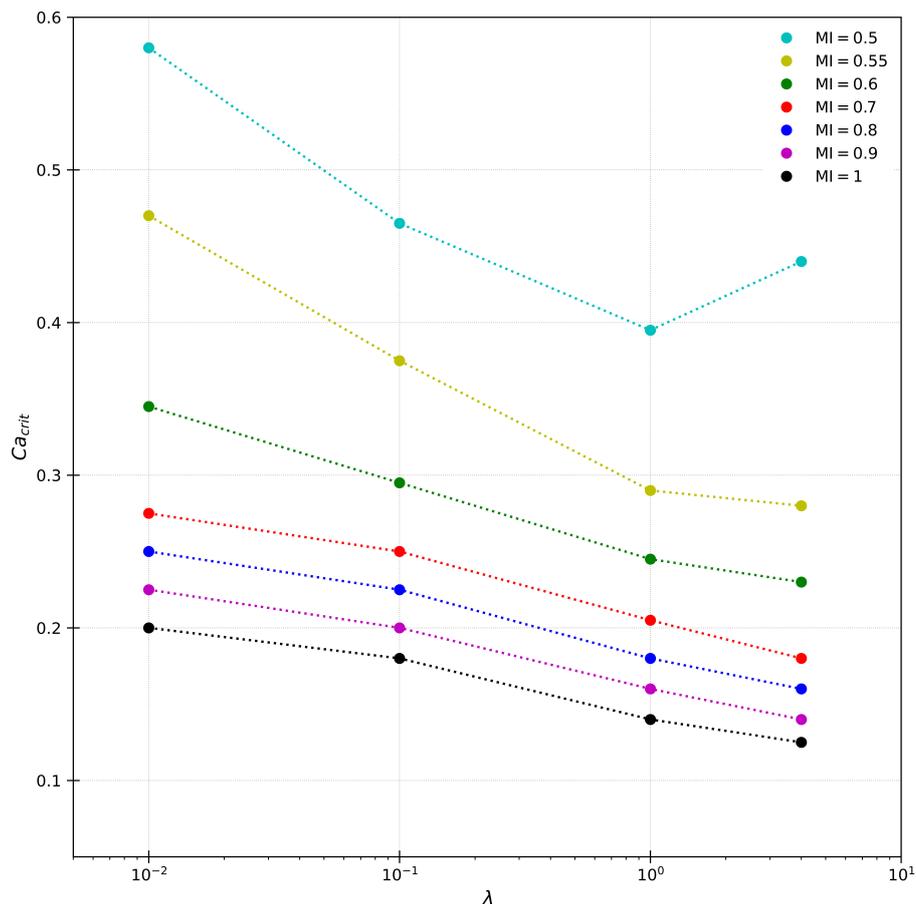


Figura 4.13: Risultati delle simulazioni VOF per campi di moto con componente elongazionale maggiore o uguale alla componente rotazionale

Un confronto quantitativo può essere effettuato in tabella 4.6, dove la tendenza è confermata, con valori più ravvicinati ad alti valori di MI e valori più distanti per bassi valori di MI.

Intervallo MI	ΔCa_{crit}			
	$\lambda = 4$	$\lambda = 1$	$\lambda = 0.1$	$\lambda = 0.01$
1 – 0.9	0.015	0.020	0,020	0.025
0.9 – 0.8	0.020	0.020	0,025	0.025
0.8 – 0.7	0.020	0.025	0,025	0.025
0.7 – 0.6	0.050	0.045	0,045	0.070
0.6 – 0.55	0.050	0.045	0,080	0.125
0.55 – 0.5	0.160	0.105	0,090	0.110

Tabella 4.6: Differenza tra i valori critici del numero di capillarità tra condizioni di campo di moto simulate differenti

Le simulazioni in campo rotazionale, con $MI = 0.45$, presentano delle difficoltà nell'individuazione corretta del valore critico per il numero di capillarità, a causa degli alti valori di *shear rate* richiesti e delle peculiarità del campo di moto, che stabilizzano inizialmente la goccia, ma successivamente tendono ad aumentarne lo spostamento, a causa delle presenze di correnti parassite, fino a portare la goccia ad interagire con i bordi del dominio. A causa di questi fenomeni l'incertezza nel determinare il punto di rottura è molto grande non è possibile effettuare un'analisi quantitativa valida. In figura 4.14 è possibile però notare qualitativamente come il fenomeno continua ed è accentuato dalla preponderanza della componente rotazionale su quella elongazionale. Le barre d'errore presenti indicano l'incertezza nell'identificare il valore di Ca_{crit} , che è più probabilmente corrispondente al punto indicato ma che non si può escludere essere uno dei punti all'interno dell'intervallo indicato.

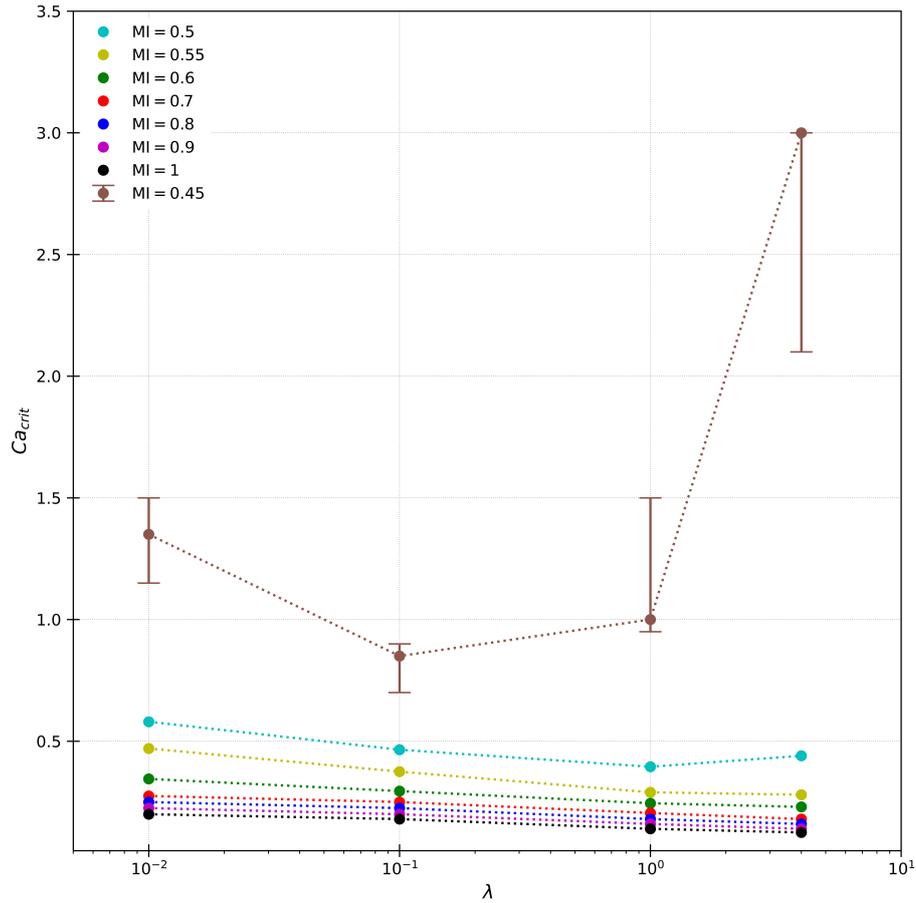


Figura 4.14: Risultati delle simulazioni VOF, sono inclusi i valori ottenuti per $MI = 0.45$.

4.2 Simulazioni tridimensionali

Le simulazioni bidimensionali hanno il grande vantaggio di avere un costo computazionale ridotto, ma non colgono pienamente la fisica del problema su cui viene applicata l'ipotesi tridimensionale. Nel caso analizzato in questo lavoro, ovvero la rottura di gocce, il limite maggiore delle simulazioni in 2D è dato dall'impossibilità di distinguere una sfera da un cilindro, in quanto entrambe queste forme presentano un cerchio come sezione trasversale. Per questo motivo sono stati analizzati alcuni casi in tre dimensioni, con l'obiettivo di convalidare i risultati ottenuti dalle simulazioni in due dimensioni, o da poterne quantificare le differenze.

L'elevato costo in termini di risorse computazionali ha portato alla scelta di adottare una griglia di calcolo relativamente grossolana, mantenendo un dominio cubico di 1.6 cm suddivisi in 80 celle per lato.

λ	Ca_{crit}	Ca_{crit} (De Bruijn)
4	0.37	∞
1	0.295	0.48
0.1	0.37	0.56

Tabella 4.7: Risultati ottenuti dalle simulazioni VOF tridimensionali con griglia grossolana per $MI = 0.5$, confrontati con i valori provenienti dalle relazioni sperimentali di De Bruijn [16].

In tabella 4.7 possiamo notare come le simulazioni presentino valori piuttosto lontani da quelli ottenuti dalle relazioni sperimentali delle equazioni 1.1 e 1.2. Un comportamento di questo tipo è sicuramente dovuto all'utilizzo di una griglia non abbastanza fitta, per cui si nota un migliore accordo tra le simulazioni bidimensionali, presentate nelle tabelle della sezione 4.1, e i dati sperimentali, rispetto a quanto ottenuto con le simulazioni in tre dimensioni.

L'effetto dell'utilizzo di una griglia più grossolana si nota anche nel distacco di materia dal corpo principale della goccia, per cui si ottiene una diminuzione di frazione volumica del fluido II massima di 8.1%, valore più elevato dei corrispettivi calcolati per le simulazioni in due dimensioni con griglia più fitta e lo stesso campo di moto, *i.e.* moto di taglio puro.

È stato possibile effettuare delle simulazioni tridimensionali con una griglia più fitta per un solo caso tridimensionale, con un rapporto di viscosità pari a quattro e in condizioni di moto di taglio puro. In questo caso, utilizzando un dominio cubico di 1.6 cm suddivisi in 160 celle per lato, il valore critico del numero di capillarità individuato è pari a $Ca_{crit} = 0.52$, maggiore rispetto a quello trovato per simulazioni bidimensionali, ma comunque non in linea con i valori sperimentali riportati in letteratura per valori di λ vicini al limite oltre il quale non è possibile ottenere rottura con $MI = 0.5$ [9], [16].

Capitolo 5

Conclusioni

L'obiettivo del lavoro presentato è stato valutare l'utilizzo di un approccio numerico basato sul metodo di fluidodinamica computazionale VOF (*Volume Of Fluid*) per determinare le condizioni critiche, *i.e.* il numero di capillarità critico, nella rottura di gocce disperse in un secondo fluido a bassi numeri di Reynolds. Le simulazioni, condotte attraverso l'utilizzo del software OpenFOAM, presentano una goccia inizialmente sferica al centro del dominio su cui viene imposto un campo di moto predefinito. Le condizioni operative utilizzate prevedono la valutazione delle condizioni critiche al variare del rapporto tra la viscosità dei due fluidi, λ , ed il tipo di campo di moto, caratterizzato attraverso il parametro *mixing index*, fatto variare in un intervallo che presenta come estremi il moto di taglio puro e il moto puramente elongazionale, detto anche moto irrotazionale.

Dai risultati delle simulazioni in due dimensioni, effettuate con dominio e griglia di calcolo descritti nel lavoro, si possono trarre le seguenti conclusioni:

- Non è stato possibile riprodurre correttamente il comportamento sperimentale delle gocce a valori prossimi al limite del rapporto di viscosità di $\lambda = 4$ nel caso di moto di taglio puro. La rottura si è infatti verificata a numeri di capillarità sensibilmente più bassi di quelli sperimentali, che tendono ad infinito.
- Gli andamenti del numero di capillarità critico al variare del rapporto di viscosità e del *mixing index* rispecchiano quanto presente in letteratura, pur presentando valori di Ca_{crit} differenti.
- Le dimensioni del dominio influiscono fortemente sul valore a cui avviene la rottura, in particolar modo per valori di λ molto bassi, nel campo della teoria del corpo snello, dove il dominio utilizzato non risulta sufficientemente grande rispetto alle deformazioni cui vanno incontro le gocce. I valori critici ottenuti per $\lambda = 0.01$ sono quindi quelli più lontani dai valori sperimentali riportati in letteratura.

Le simulazioni in tre dimensioni mostrano l'importanza della dimensione delle celle della griglia computazionale, che diventa fattore discriminante per ottenere risultati in accordo con i dati sperimentali. L'infittimento della griglia di calcolo è risultato essere importante anche nella riduzione dei problemi di diffusione numerica. Non è stato possibile verificare se le simulazioni tridimensionali permettessero di ottenere risultati migliori a parità delle altre variabili, a causa dell'elevato costo computazionale di queste simulazioni.

Per quanto riguarda uno dei principali svantaggi del metodo VOF, ovvero la presenza di correnti parassite, il metodo migliore per mitigare il loro effetto è risultato adottare un criterio temporale, *i.e.* numero di Courant, molto stringente. Riposizionare il campo di moto in base al tracciamento del centro di massa della goccia ha anche permesso di aggirare il problema del movimento della goccia dovuto alle correnti parassite.

Lo studio di questo problema attraverso simulazioni con metodo VOF ha dato risultati qualitativamente in linea con gli esperimenti, ma quantitativamente non del tutto soddisfacenti. La metodologia utilizzata ha infatti permesso di ottenere risultati accettabili per valori del rapporto di viscosità intermedi, ovvero per $0.1 < \lambda < 1$, mentre le condizioni più estreme sono predette con minore accuratezza. Studi futuri potrebbero concentrarsi su metodi numerici alternativi al VOF, che non siano afflitti dalle problematiche delle correnti parassite, che diventano particolarmente influenti per sistemi a velocità piccole, come quello studiato.

In alternativa il lavoro potrebbe essere migliorato attraverso uno studio delle prestazioni di schemi di avvezione differenti per il metodo VOF o studiando griglie di calcolo e domini computazionali ottimizzati per le varie condizioni simulate, a condizioni di disporre di maggiori risorse computazionali.

Appendici

Appendice A

Equazione di trasporto per α

In quest'appendice si riporta una derivazione semplificata per l'equazione di trasporto dell'indicatore di fase α , secondo il modello implementato in OpenFOAM. Per una trattazione più completa e rigorosa si consiglia di fare riferimento a Marquez [35], in cui le equazioni del metodo VOF sono ricavate partendo dalla teoria ASMM (*Algebraic Slip Mixture Model*).

Si parte da una descrizione separata dei fluidi, per ognuno dei quali è possibile scrivere l'equazione di trasporto dell'indicatore di fase:

$$\frac{\partial \alpha}{\partial t} + \nabla \cdot (\mathbf{U}^I \alpha) = 0 \quad (\text{A.1a})$$

$$\frac{\partial (1 - \alpha)}{\partial t} + \nabla \cdot [\mathbf{U}^{II} (1 - \alpha)] = 0 \quad (\text{A.1b})$$

Si assume che i contributi delle velocità all'evoluzione dell'interfaccia siano proporzionali alla frazione volumica dei due fluidi [38], in modo da scrivere la velocità effettiva come media pesata sull'indicatore di fase:

$$\begin{aligned} \mathbf{U} &= \alpha \mathbf{U}^I + (1 - \alpha) \mathbf{U}^{II} \\ &= \alpha \mathbf{U}^I + \mathbf{U}^{II} - \alpha \mathbf{U}^{II} \\ &= \alpha (\mathbf{U}^I - \mathbf{U}^{II}) + \mathbf{U}^{II} \end{aligned} \quad (\text{A.2})$$

Si definisce la velocità relativa \mathbf{U}_r :

$$\mathbf{U}_r = \mathbf{U}^I - \mathbf{U}^{II} \quad (\text{A.3})$$

Sostituendo quest'ultima nell'equazione A.2 e riarrangiando, si ottiene:

$$\mathbf{U}^{II} = \mathbf{U} - \alpha \mathbf{U}_r, \quad (\text{A.4})$$

che può essere direttamente sostituita nell'equazione A.1b, ottenendo la seguente

espressione:

$$\begin{aligned}
& \frac{\partial(1-\alpha)}{\partial t} + \nabla \cdot [(1-\alpha)(\mathbf{U} - \alpha\mathbf{U}_r)] = 0 \\
\Rightarrow & \frac{\partial 1}{\partial t} - \frac{\partial \alpha}{\partial t} + \nabla \cdot [(1-\alpha)\mathbf{U}] - \nabla \cdot [(1-\alpha)\alpha\mathbf{U}_r] = 0 \quad \text{con } \frac{\partial 1}{\partial t} = 0 \\
\Rightarrow & -\frac{\partial \alpha}{\partial t} + \nabla \cdot \mathbf{U} - \nabla \cdot (\alpha\mathbf{U}) - \nabla \cdot [(1-\alpha)\alpha\mathbf{U}_r] = 0 \quad \text{con } \nabla \cdot \mathbf{U} = 0 \\
\Rightarrow & -\frac{\partial \alpha}{\partial t} - \nabla \cdot (\alpha\mathbf{U}) - \nabla \cdot [(1-\alpha)\alpha\mathbf{U}_r] = 0 \\
\Rightarrow & \frac{\partial \alpha}{\partial t} + \nabla \cdot (\alpha\mathbf{U}) + \nabla \cdot [(1-\alpha)\alpha\mathbf{U}_r] = 0 \tag{A.5}
\end{aligned}$$

L'equazione A.5 di trasporto per α con il termine compressivo, che viene utilizzata nel metodo VOF di Weller, è così ottenuta, attraverso l'uso dell'equazione di continuità per i fluidi incomprimibili $\nabla \cdot \mathbf{U} = 0$.

Appendice B

Function object strainRate

Per poter verificare la corretta impostazione delle condizioni di bordo, è necessario calcolare il *mixing index*, secondo la formula

$$MI = \frac{\text{strainRate} + \text{mag}(\text{vorticity})}{\text{strainRate}} \quad (\text{B.1})$$

Nei listing seguenti si riporta il codice scritto per implementare la *function object* `strainRate` come applicazione di *post-processing* su OpenFOAM

```
1
2 #include "strainRate.H"
3 #include "fvcGrad.H"
4 #include "addToRunTimeSelectionTable.H"
5
6 // * * * * * Static Data Members * * * * *
7
8 namespace Foam
9 {
10 namespace functionObjects
11 {
12     defineTypeNameAndDebug(strainRate, 0);
13
14     addToRunTimeSelectionTable
15     (
16         functionObject,
17         strainRate,
18         dictionary
19     );
20 }
21 }
22
23
24 // * * * * * Private Member Functions * * * * *
25
26 bool Foam::functionObjects::strainRate::calc()
27 {
28     if (foundObject<volVectorField>(fieldName_))
29     {
30         return store
31         (
32             resultName_,
33             sqrt(2.0)*mag(symm(fvc::grad(lookupObject<
34 volVectorField>(fieldName_))))
```

```

34     );
35   }
36   else
37   {
38     return false;
39   }
40
41   return true;
42 }
43
44
45 // * * * * * Constructors * * * * * //
46
47 Foam::functionObjects::strainRate::strainRate
48 (
49     const word& name,
50     const Time& runTime,
51     const dictionary& dict
52 )
53 :
54     fieldExpression(name, runTime, dict, typeName, "U")
55 {
56 }
57
58
59 // * * * * * Destructor * * * * * //
60
61 Foam::functionObjects::strainRate::~~strainRate()
62 {}
63
64
65 // ***** //

```

Listing B.1: Contenuto di strainRate.C

```

1
2 #ifndef functionObjects_strainRate_H
3 #define functionObjects_strainRate_H
4
5 #include "fieldExpression.H"
6
7 // * * * * * //
8
9 namespace Foam
10 {
11     namespace functionObjects
12     {
13
14         /*-----*\
15                          Class strainRate Declaration
16         \*-----*/
17
18         class strainRate
19         :
20             public fieldExpression
21         {
22             // Private Member Functions
23

```

```

24     //- Calculate the strainRate field and return true if
    successful
25     virtual bool calc();
26
27
28 public:
29
30     //- Runtime type information
31     TypeName("strainRate");
32
33     // Constructors
34
35     //- Construct from Time and dictionary
36     strainRate
37     (
38         const word& name,
39         const Time& runTime,
40         const dictionary& dict
41     );
42
43
44     //- Destructor
45     virtual ~strainRate();
46 };
47
48
49 // * * * * *
50 // * * * * *
51 // * * * * *
52 } // End namespace functionObjects
53 } // End namespace Foam
54 // * * * * *
55 // * * * * *
56
57 #endif
58
59 // *****

```

Listing B.2: Contenuto di `strainRate.H`

Appendice C

Imposizione condizioni al contorno tramite codedFixedValue

La condizione `codedFixedValue` è utilizzata per impostare valori di condizioni di bordo che varino nel tempo e nello spazio secondo espressioni matematiche predefinite dall'utente. Grazie alla possibilità di accedere alla mesh ed ai valori dei campi calcolati nel processo di soluzione, viene implementata una condizione che calcoli il centro di massa della goccia, attraverso la formula:

$$\text{coord}_{\text{CoM}} = \frac{\sum_i \text{coord}_i \cdot (1 - \alpha_i)}{\sum_i \alpha_i} \quad \text{con coord} = x, y, z \text{ e } i \text{ celle.} \quad (\text{C.1})$$

A questo punto vengono impostate le equazioni

$$\begin{cases} U_x = G \cdot \theta \cdot (y - y_{\text{CoM}}) & (\text{C.2a}) \\ U_y = G \cdot (x - x_{\text{CoM}}) & (\text{C.2b}) \\ U_z = 0 & (\text{C.2c}) \end{cases}$$

che calcolano i valori da imporre sulla faccia di ogni cella.

Il codice riportato nel listing C.1, si riferisce ad uno solo dei bordi su cui è imposto questo tipo di condizione. Poichè le formule sono le stesse per tutti i bordi del dominio, lo stesso codice viene riscritto anche per le facce `west`, `north` e `south` in maniera identica.

```
1     east
2     {
3     type        codedFixedValue;
4     value       uniform (0 0 0);
5
6     name        eastProfile;
7
8     codeInclude
9     #{
10    #include "/home/fderoma/OpenFOAM/fderoma-8/run/3D/lambda4/
11    MI_05_061/input"
12    #include "fvCFD.H"
13    #include "fvMesh.H"
14    // #include "setRootCase.H"
15    // #include "createTime.H"
16    // #include "createMesh.H"
17    #};
```

```

17
18 codeOptions
19 #{
20     -I$(LIB_SRC)/finiteVolume/lnInclude \
21     -I$(LIB_SRC)/meshTools/lnInclude
22 #};
23
24 codeLibs
25 #{
26     -lfiniteVolume \
27     -lmeshTools
28 #};
29
30 code
31 #{
32     // This part calculate the centre of mass of the droplet
33     const volScalarField& alpha(db().objectRegistry::lookupObject<
volScalarField>("alpha.water"));
34     // We don't have direct access to the mesh, so we use a
workaround calling it from the field
35     // alpha
36     const fvMesh& mesh(alpha.mesh());
37     const scalar& cellsNum(mesh.C().size());
38     const volVectorField& centres(mesh.C());
39     const volScalarField xCoord(centres.component(0/*vector::X*/));
40     const volScalarField yCoord(mesh.C().component(vector::Y));
41     const volScalarField zCoord(mesh.C().component(vector::Z));
42
43     scalar denominator(0.), numeratorX(0.), numeratorY(0.),
numeratorZ(0.), COMX, COMY, COMZ;
44
45     for (label cellI = 0; cellI < cellsNum; cellI++){
46         numeratorX += (1 - alpha[cellI]) * xCoord[cellI];
47         numeratorY += (1 - alpha[cellI]) * yCoord[cellI];
48         numeratorZ += (1 - alpha[cellI]) * zCoord[cellI];
49         denominator += (1 - alpha[cellI]);
50     }
51
52     // reduce for parallel running
53     reduce(numeratorX, sumOp<scalar>());
54     reduce(numeratorY, sumOp<scalar>());
55     reduce(numeratorZ, sumOp<scalar>());
56     reduce(denominator, sumOp<scalar>());
57
58     COMX = numeratorX/denominator;
59     COMY = numeratorY/denominator;
60     COMZ = numeratorZ/denominator;
61
62     // This part calculates the value of the U field on the
boundary
63     const fvPatch& boundaryPatch = patch();
64     const vectorField& Cf = boundaryPatch.Cf();
65     vectorField& field = *this;
66
67     const scalar Gdot = gdot; // value of the shear rate
68     const scalar Beta = beta; // value linked to the mixing index
69     const scalar L = 0.005; // value of the half-lenght of the
domain

```

```
70
71  forAll(Cf, faceI)
72  {
73      const scalar x = Cf[faceI][0]; // .x();
74      const scalar y = Cf[faceI][1]; // .y();
75      const scalar z = Cf[faceI][2]; // .z();
76
77      field[faceI] = vector( Gdot*Beta*(y-COMY), (x-COMX)*Gdot, 0);
78  }
79
80 #};
81 }
```

Listing C.1: Implementazione della condizione codedFixedValue per il bordo east

Lista dei simboli

Simbolo	Descrizione	Unità di misura
a	raggio della goccia non deformata	m
B	asse minore della goccia deformata	m
\tilde{c}	funzione colore del metodo VOF	–
Ca	numero di capillarità	–
Ca_{crit}	numero di capillarità critico	–
Co	numero di Courant	–
D	deformazione della goccia	–
\mathbf{F}	flusso della generica proprietà trasportata	
F^C	flusso corretto	
F^H	flusso ottenuto con metodo di alto ordine	
F^L	flusso ottenuto con metodo di basso ordine	
f_σ	forza per unità di volume della tensione interfacciale	N
G	intensità dei gradienti di velocità	s^{-1}
\mathbf{g}	accelerazione di gravità	$m\ s^{-2}$
g_i	accelerazione di gravità in direzione i	$m\ s^{-2}$
i	indice numerico	–
j	indice numerico	–
L	asse maggiore della goccia deformata	m
MI	<i>mixing index</i>	–
\mathbf{n}	versore normale	–
n_i^I	versore normale alla superficie uscente dal fluido I in direzione i	–
n_i^{II}	versore normale alla superficie uscente dal fluido II in direzione i	–
p	pressione	Pa
p^I	pressione nel fluido I	Pa
p^{II}	pressione nel fluido II	Pa

p_{rgh}	pressione modificata	Pa
p_s	pressione di superficie	Pa
Pe	numero di Péclet	–
q_ϕ	generazione per unità di volume	
\bar{q}_ϕ	generazione media per unità di volume	
Re	numero di Reynolds	–
\mathcal{S}	superficie che racchiude il volume di controllo	m ²
t	tempo	s
t_i^α	componente i-esima del vettore tangente all'interfaccia	–
t_i^β	componente i-esima del vettore tangente all'interfaccia	–
\mathbf{U}	vettore velocità	m s ⁻¹
U^I	velocità caratteristica del fluido I	m s ⁻¹
U_i	velocità del fluido in direzione i	m s ⁻¹
U_i^I	velocità del fluido I in direzione i	m s ⁻¹
U_i^{II}	velocità del fluido II in direzione i	m s ⁻¹
U_j	velocità del fluido in direzione j	m s ⁻¹
U_j^I	velocità del fluido I in direzione j	m s ⁻¹
U_j^{II}	velocità del fluido II in direzione j	m s ⁻¹
\mathbf{U}_r	velocità relativa tra le due fasi	m s ⁻¹
U_x	velocità del fluido in direzione x	m s ⁻¹
U_y	velocità del fluido in direzione y	m s ⁻¹
U_z	velocità del fluido in direzione z	m s ⁻¹
We	numero di Weber	–
x	coordinata spaziale	m
x_i	coordinata spaziale i-esima	m
x_j	coordinata spaziale j-esima	m
y	coordinata spaziale	m
z	coordinata spaziale	m
α	indicatore di fase per il metodo VOF	–
$\dot{\gamma}$	tensore delle deformazioni	s ⁻¹
$\dot{\gamma}$	<i>strain rate</i>	s ⁻¹
$\dot{\gamma}_{ij}$	componente in posizione (i, j) del tensore delle deformazioni	s ⁻¹
θ	parametro legato al <i>mixing index</i>	–

κ	curvatura dell'interfaccia	m^{-1}
Λ	generico coefficiente di trasporto molecolare	$\text{m}^2 \text{s}^{-1}$
λ	rapporto di viscosità	–
λ_e	fattore di interpolazione lineare per schema CDS	–
λ_M	limitatore dell'algoritmo MULES	–
μ	viscosità dinamica	Pa s
μ^{I}	viscosità dinamica del fluido I	Pa s
μ^{II}	viscosità dinamica del fluido II	Pa s
ν	viscosità cinematica	$\text{m}^2 \text{s}^{-1}$
ν^{I}	viscosità cinematica del fluido I	$\text{m}^2 \text{s}^{-1}$
ν^{II}	viscosità cinematica del fluido II	$\text{m}^2 \text{s}^{-1}$
ρ	densità del fluido	kg m^{-3}
ρ^{I}	densità del fluido I	kg m^{-3}
ρ^{II}	densità del fluido II	kg m^{-3}
σ	tensione interfacciale	N m^{-1}
$\boldsymbol{\tau}$	tensore degli sforzi	Pa
τ_{ij}	componente in posizione (i, j) del tensore degli sforzi	Pa
τ_{ij}^{I}	componente in posizione (i, j) del tensore degli sforzi per il fluido I	Pa
τ_{ij}^{II}	componente in posizione (i, j) del tensore degli sforzi per il fluido II	Pa
Φ	flusso di proprietà in OpenFOAM	
Φ^{compr}	flusso compressivo in OpenFOAM	
Φ^{HO}	flusso ottenuto con metodo di alto ordine in OpenFOAM	
Φ^{UD}	flusso ottenuto con metodo di basso ordine in OpenFOAM	
ϕ	generica proprietà trasportata	
$\boldsymbol{\omega}$	tensore della vorticità	s^{-1}
Ω	volume di controllo	m^3
$\tilde{\omega}$	vorticità	s^{-1}
ω_{ij}	componente in posizione (i, j) del tensore della vorticità	s^{-1}

Bibliografia

- [1] B. J. Bentley and L. G. Leal, “An experimental investigation of drop deformation and breakup in steady, two-dimensional linear flows”, *Journal of Fluid Mechanics*, vol. 167, pp. 241–283, 1986. DOI: <http://dx.doi.org/10.1017/S0022112086002811>.
- [2] V. Cristini, S. Guido, A. Alfani, J. Bławdziewicz, and M. Loewenberg, “Drop breakup and fragment size distribution in shear flow”, *Journal of Rheology*, vol. 47, no. 5, pp. 1283–1298, 2003. DOI: <https://doi.org/10.1122/1.1603240>.
- [3] T. Kekesi, G. Amberg, and L. P. Wittberg, “Drop deformation and breakup in flows with shear”, *Chemical Engineering Science*, vol. 140, pp. 319–329, 2016. DOI: <https://doi.org/10.1016/j.ces.2015.10.019>.
- [4] A. Dubbelboer, J. J. M. Janssen, H. Hoogland, E. Zondervan, and J. Meuldijk, “Pilot-scale production process for high internal phase emulsions: Experimentation and modeling”, *Chemical Engineering Science*, no. 148, pp. 32–43, 2016. DOI: <https://doi.org/10.1016/j.ces.2016.03.014>.
- [5] G. I. Taylor, “The formation of emulsions in definable fields of flow”, *Proceedings of the Royal Society of London. Series A, containing papers of a mathematical and physical character*, vol. 146, no. 858, pp. 501–523, 1934. DOI: <https://doi.org/10.1098/rspa.1934.0169>.
- [6] J. Suhas, “Flow-induced breakup of drops and bubbles”, *arXiv e-print*, 2017. DOI: <https://arxiv.org/abs/1701.06157>.
- [7] G. I. Taylor, “The viscosity of a fluid containing small drops of another fluid”, *Proceedings of the Royal Society of London. Series A, Containing papers of a mathematical and physical character*, vol. 138, no. 834, pp. 41–48, 1932. DOI: <https://doi.org/10.1098/rspa.1932.0169>.
- [8] D. Barthes-Biesel and A. Acrivos, “Deformation and burst of a liquid droplet freely suspended in a linear shear field”, *Journal of Fluid Mechanics*, vol. 61, no. 1, pp. 1–22, 1973. DOI: <https://doi.org/10.1017/S0022112073000534>.
- [9] H. P. Grace, “Dispersion phenomena in high viscosity immiscible fluid systems and application of static mixers as dispersion devices in such systems”, *Chemical Engineering Communications*, vol. 14, no. 3-6, pp. 225–277, 1982. DOI: <https://doi.org/10.1080/00986448208911047>.
- [10] G. I. Taylor, “Conical free surfaces and fluid interfaces”, in *Applied Mechanics*, Springer, 1966, pp. 790–796. DOI: https://doi.org/10.1007/978-3-662-29364-5_104.

- [11] J. D. Buckmaster, “Pointed bubbles in slow viscous flow”, *Journal of Fluid Mechanics*, vol. 55, no. 3, pp. 385–400, 1972. DOI: <https://doi.org/10.1017/S0022112072001910>.
- [12] J. D. Buckmaster and J. E. Flaherty, “The bursting of two-dimensional drops in slow viscous flow”, *Journal of Fluid Mechanics*, vol. 60, no. 4, pp. 625–639, 1973. DOI: <https://doi.org/10.1017/S0022112073000388>.
- [13] A. Acrivos and T. S. Lo, “Deformation and breakup of a single slender drop in an extensional flow”, *Journal of Fluid Mechanics*, vol. 86, no. 4, pp. 641–672, 1978. DOI: <https://doi.org/10.1017/S0022112078001329>.
- [14] E. J. Hinch and A. Acrivos, “Long slender drops in a simple shear flow”, *Journal of fluid mechanics*, vol. 98, no. 2, pp. 305–328, 1980. DOI: <https://doi.org/10.1017/S0022112080000171>.
- [15] J. M. Rallison, “A numerical study of the deformation and burst of viscous drops in general shear flows”, *Journal of Fluid Mechanics*, no. 109, pp. 465–482, 1981. DOI: <https://doi.org/10.1017/S002211208100116X>.
- [16] R. A. De Bruijn, “Deformation and breakup of drops in simple shear flow”, Ph.D. dissertation, 1989.
- [17] J. M. Rallison, “The deformation of small viscous drops and bubbles in shear flows”, *Annual review of fluid mechanics*, vol. 16, no. 1, pp. 45–66, 1984. DOI: <https://doi.org/10.1146/annurev.fl.16.010184.000401>.
- [18] H. A. Stone, “Dynamics of drop deformation and breakup in viscous fluids”, *Annual Review of Fluid Mechanics*, vol. 26, no. 1, pp. 65–102, 1994. DOI: <https://doi.org/10.1146/annurev.fl.26.010194.000433>.
- [19] R. B. Bird, W. E. Stewart, and E. N. Lightfoot, *Transport phenomena*. John Wiley & Sons, 2006.
- [20] J. C. Berg, *An introduction to interfaces & colloids: the bridge to nanoscience*. World Scientific, 2010.
- [21] J. H. Ferziger, M. Perić, and R. L. Street, *Computational methods for fluid dynamics*. Springer, 2002.
- [22] H. K. Versteeg and W. Malalasekera, *An introduction to computational fluid dynamics: the finite volume method*. Pearson education, 2007.
- [23] B. Andersson, R. Andersson, L. Håkansson, M. Mortensen, R. Sudiyo, and B. Van Wachem, *Computational fluid dynamics for engineers*. Cambridge university press, 2011.
- [24] C. W. Hirt and B. D. Nichols, “Volume of fluid (vof) method for the dynamics of free boundaries”, *Journal of computational physics*, vol. 39, no. 1, pp. 201–225, 1981. DOI: [https://doi.org/10.1016/0021-9991\(81\)90145-5](https://doi.org/10.1016/0021-9991(81)90145-5).
- [25] V. R. Gopala and B. G. M. van Wachem, “Volume of fluid methods for immiscible-fluid and free-surface flows”, *Chemical Engineering Journal*, vol. 141, no. 1-3, pp. 204–221, 2008. DOI: <https://doi.org/10.1016/j.cej.2007.12.035>.
- [26] J. U. Brackbill, D. B. Kothe, and C. Zemach, “A continuum method for modeling surface tension”, *Journal of computational physics*, vol. 100, no. 2, pp. 335–354, 1992. DOI: [https://doi.org/10.1016/0021-9991\(92\)90240-Y](https://doi.org/10.1016/0021-9991(92)90240-Y).

- [27] S. S. Deshpande, L. Anumolu, and M. F. Trujillo, “Evaluating the performance of the two-phase flow solver interFoam”, *Computational science & discovery*, vol. 5, no. 1, p. 014016, 2012. DOI: <https://doi.org/10.1088/1749-4699/5/1/014016>.
- [28] C. Galusinski and P. Vigneaux, “On stability condition for bifluid flows with surface tension: Application to microfluidics”, *Journal of Computational Physics*, vol. 227, no. 12, pp. 6140–6164, 2008. DOI: <https://doi.org/10.1016/j.jcp.2008.02.023>.
- [29] V. Inguva, A. Schulz, and E. Y. Kenig, “On methods to reduce spurious currents within vof solver frameworks. part 1: A review of the static bubble/droplet”, *Chemical Product and Process Modeling*, 2020. DOI: <https://doi.org/10.1515/cppm-2020-0052>.
- [30] J. Li, Y. Y. Renardy, and M. Renardy, “Numerical simulation of breakup of a viscous drop in simple shear flow through a volume-of-fluid method”, *Physics of fluids*, vol. 12, no. 2, pp. 269–282, 2000. DOI: <https://doi.org/10.1063/1.870305>.
- [31] T. O. Foundation, *Openfoam v8 user guide*, Disponibile all’indirizzo <https://cfd.direct/openfoam/user-guide-v8> (26/07/2021).
- [32] S. Márquez Damián, “Description and utilization of interFoam multiphase solver”, *FinalWork, Computational Fluid Dynamics–Facultad de Ingeniería y Ciencias Hidricas, UNL* (<http://infofich.unl.edu.ar/upload/3be0e16065026527477b4b948c4caa7523c8ea52.pdf>), 2010.
- [33] J. P. Boris and D. L. Book, “Flux-corrected transport. i. shasta, a fluid transport algorithm that works”, *Journal of computational physics*, vol. 11, no. 1, pp. 38–69, 1973. DOI: [https://doi.org/10.1016/0021-9991\(73\)90147-2](https://doi.org/10.1016/0021-9991(73)90147-2).
- [34] S. T. Zalesak, “Fully multidimensional flux-corrected transport algorithms for fluids”, *Journal of computational physics*, vol. 31, no. 3, pp. 335–362, 1979. DOI: [https://doi.org/10.1016/0021-9991\(79\)90051-2](https://doi.org/10.1016/0021-9991(79)90051-2).
- [35] S. Márquez Damián, “An extended mixture model for the simultaneous treatment of short and long scale interfaces”, Ph.D. dissertation, 2013.
- [36] O. Ubbink, “Numerical prediction of two fluid systems with sharp interfaces”, Ph.D. dissertation, 1997.
- [37] H. Rusche, “Computational fluid dynamics of dispersed two-phase flows at high phase fractions”, Ph.D. dissertation, 2003.
- [38] E. Berberović, N. P. van Hinsberg, S. Jakirlić, I. V. Roisman, and C. Tropea, “Drop impact onto a liquid layer of finite thickness: Dynamics of the cavity evolution”, *Physical Review E*, vol. 79, no. 3, p. 036306, 2009. DOI: <https://doi.org/10.1103/PhysRevE.79.036306>.
- [39] S. Mukherjee, A. Zarghami, C. Haringa, K. van As, S. Kenjereš, and H. E. A. Van den Akker, “Simulating liquid droplets: A quantitative assessment of lattice boltzmann and volume of fluid methods”, *International Journal of Heat and Fluid Flow*, vol. 70, pp. 59–78, 2018. DOI: <https://doi.org/10.1016/j.ijheatfluidflow.2017.12.001>.

- [40] J. Roenby, H. Bredmose, and H. Jasak, “A computational method for sharp interface advection”, *Royal Society open science*, vol. 3, no. 11, p. 160 405, 2016. DOI: <https://doi.org/10.1098/rsos.160405>.

Ringraziamenti

Al termine di questo lavoro di tesi voglio ringraziare le persone che mi hanno sostenuto durante il mio percorso universitario.

Il primo ringraziamento va ai miei genitori, senza di loro questo traguardo non sarebbe stato possibile, il loro sostegno, sia economico che emotivo, e la loro presenza hanno contribuito a farmi diventare la persona che sono oggi. Grazie per esserci stati, per continuare ad esserci e per tutta la pazienza che avete dimostrato.

Un grande ringraziamento a mio fratello, sempre pronto per un confronto, per un consiglio o solo per scherzare e alleggerire una giornata un po' pesante. Anche se solitamente siamo lontati, il tuo contributo è sempre fondamentale.

Ringrazio i miei nonni, la tristezza con cui ci salutavamo prima di ogni mia partenza era forte solo quanto l'affetto di cui mi ricoprivate al mio ritorno. So con certezza quanto sareste orgogliosi di me se foste qui.

Ringrazio tutti gli amici che in questo lungo percorso mi hanno accompagnato, facendomi sentire a casa a Torino e rendendo sopportabili anche i periodi peggiori. Grazie a Antonio, Mimmo, Pio e Graziano. Ci siamo incontrati in una Torino sconosciuta e dopo averla esplorata insieme, portandoci dietro qualche vuoto di memoria, siamo rimasti legati, nonostante siamo ormai distanti. Non riesco a quantificare quanti ricordi ho con voi, i miei primi anni a Torino non sarebbero stati gli stessi senza il tempo speso insieme, in casa, al pub, al poli, o ai Murazzi. Adesso bisogna organizzare un viaggio se ricordo bene.

Grazie a Vincenzo, Betti, Jessica, Giusi, Carmine e Eugenio. Siete stati una costante del periodo della triennale, non si può dimenticare tutto il tempo insieme, anche se ora siamo un gruppo eterogeneo in cui dopo anni giustamente ognuno è "a fatti propri".

Ringrazio Liberale, con i suoi discorsi un po' nerd e a cui devo qualche esame superato.

Ringrazio i "Fulminati". Conosciuti durante la magistrale avete condiviso con me le grandi gioie e i grandi esaurimenti. Francesco, che mi ha introdotto al "gruppo 5" e mi dato una "X", Nicola, che è riuscito a rendere piacevoli anche le sessioni di studio infinite, Mariagrazia, esuberante come un martello pneumatico. Rimarrano impresse nella mente le cene per cui devo ringraziare Claudia e Sonia, le prime amiche incontrate a lezione, con Stefano che ci ha accettati come ospiti ed è diventato parte del gruppo. Gloria, sei e sarai sempre la nostra Lady, Alessio Mannalà, tanto preparato quanto bravo a cucinare, e Giorgio, con cui parlare di tecnologia e bere vino. Gaetano, quando sei entrato nel gruppo ci hai fatto notare quanto fortunati fossimo a farne parte, e Angelica, una buona amica trovata purtroppo solo alla fine del percorso. Gabriele, tanto matto quanto intelligente, Alessio Truncali e Martina, ormai fulminati emigrati. Grazie a voi questi ultimi anni rimarranno sempre nel mio cuore.

Ringrazio i miei relatori, i professori Buffo, Vanni, Marchisio e Boccardo, per avermi guidato nel corso del progetto di tesi e nella stesura di questo scritto.

Ringrazio Alexandra Elbakyan, chi ha lavorato ad una tesi in ambito scientifico probabilmente sa per quale motivo.

Infine, ringrazio in maniera speciale Ann, per essermi stata vicina nei momenti felici ed in quelli difficili, durante una pandemia, per aver creduto in me, accettato i miei fallimenti e festeggiato i miei traguardi. Mi hai dato la forza e mi hai fatto sentire compreso, grazie *schatje*.