

UNIVERSITÀ DEGLI STUDI DI CAMERINO  
FACOLTÀ DI SCIENZE MATEMATICHE, FISICHE E NATURALI

CORSO DI LAUREA IN MATEMATICA

DIPARTIMENTO DI MATEMATICA E INFORMATICA



Modelli “Lattice Boltzmann”  
e  
l’equazione di Navier-Stokes

*Tesi Sperimentale di Laurea  
in Analisi Numerica*

*Relatori*

**Prof. Luciano Misici**

**Prof. Riccardo Piergallini**

*Laureanda*

**Francesca Tosi**

---

Anno Accademico 2000-2001

*Alla Vita,  
A Chi me l'ha Donata  
e mi Aiuta a Viverla.  
Ad  
Anna, Mario e Antonio.*

# Indice

<b>Introduzione</b>	<b>1</b>
<b>1 Automi Cellulari</b>	<b>5</b>
1.1 Cenni storici . . . . .	5
1.1.1 L'ideatore: <i>von Neumann</i> . . . . .	5
1.1.2 "Game of life" . . . . .	7
1.1.3 Un passo fondamentale: dagli <i>AC</i> al <i>Lattice Gas Model</i>	9
1.1.4 ' <i>Lattice Boltzmann Method</i> ' e ' <i>Multiparticle Models</i> ' . .	11
1.2 Che cosa sono gli Automi Cellulari? . . . . .	12
1.3 Neighborhood . . . . .	13
1.4 Perché gli <i>AC</i> in fisica ? . . . . .	14
1.5 <i>HPP</i> : primo esempio di <i>LGCA</i> . . . . .	17
1.5.1 Breve descrizione del modello . . . . .	17
1.5.2 Coarse graining . . . . .	20
<b>2 <i>FHP Lattice Gas Model</i></b>	<b>23</b>
2.1 Le caratteristiche generali del metodo . . . . .	23
2.2 Caratteristiche microdinamiche . . . . .	29

2.2.1	Descrizione <i>booleana</i> . . . . .	29
2.2.2	Densità di massa e momento . . . . .	31
2.2.3	Conclusioni teoriche: $N_i^{eq}$ . . . . .	32
2.3	Derivazione delle equazioni macroscopiche:	
	<i>Eulero e Navier-Stokes</i> . . . . .	36
2.3.1	Equazione di <i>Boltzmann</i> . . . . .	36
2.3.2	Espansione di <i>Chapman-Enskog</i> . . . . .	39
2.3.3	' <i>Multi-Scale Analysis</i> ' . . . . .	46
2.4	Osservazioni conclusive sul <i>FHP</i> . . . . .	54
<b>3</b>	<b><i>Il Modello di Boltzmann</i></b> . . . . .	<b>57</b>
3.1	Sviluppo del <i>modello di Boltzmann</i> . . . . .	57
3.1.1	Dal <i>FHP</i> al <i>LBM</i> . . . . .	57
3.1.2	<i>Equazione ed approssimazione di Boltzmann</i> . . . . .	58
3.2	<i>LBM</i> bi-dimensionali . . . . .	61
3.2.1	Griglia bi-dimensionale . . . . .	62
3.2.2	La distribuzione di equilibrio . . . . .	63
3.2.3	Calcolo dei $\mathbf{W}_i$ . . . . .	65
3.2.4	Evoluzione del sistema . . . . .	66
3.3	Fasi per l'implementazione . . . . .	68
3.4	Derivazione delle equazioni macroscopiche:	
	<i>multi-scale analysis.</i> . . . . .	70
3.4.1	Equazioni macroscopiche . . . . .	70
3.5	Condizioni al bordo . . . . .	81
3.5.1	Condizioni al bordo no-slip . . . . .	81

---

3.6	Esperimenti numerici:	
	simulazione del “ <i>Cavity-Flow</i> ” bi-dimensionale. . . . .	84
<b>4</b>	<b><i>LBM</i> tri-dimensionali</b>	<b>91</b>
4.1	Scelta delle griglie: isotropia spaziale . . . . .	92
4.1.1	Tensori isotropi . . . . .	92
4.2	Calcolo della distribuzione di equilibrio nel senso di <i>Koelman</i> . . . . .	99
4.2.1	<i>LBM 3D</i> con 15 velocità . . . . .	100
4.2.2	<i>LBM 3D</i> con 19 velocità . . . . .	101
4.2.3	<i>Chen et al., (1992)</i> . . . . .	102
4.3	Distribuzione di equilibrio: <i>Ansatz-Method</i> . . . . .	103
4.4	Simulazione numerica tri-dimensionale . . . . .	107
4.4.1	<i>Miscelazione laminare tramite l’uso di lame rotanti.</i> . . . .	107
4.4.2	Formulazione matematica del problema della miscelazione	108
4.4.3	Calcolo del flusso sulle lame rotanti . . . . .	114
4.4.4	Discretizzazione del problema . . . . .	117
4.4.5	Implementazione del metodo <i>Lattice-Boltzmann</i> . . . . .	122
4.4.6	Misure qualitative e quantitative della miscelazione . . . . .	127
4.4.7	Risultati sperimentali . . . . .	131
4.5	Condizioni al contorno in presenza di bordi curvi . . . . .	151
<b>5</b>	<b><i>Fluidi binari e Gravità nei modelli LB</i></b>	<b>155</b>
5.1	<i>LBM</i> per <i>fluidi binari (binary-fluid)</i> . . . . .	155
5.1.1	<i>LBM</i> per fluidi immiscibili . . . . .	156
5.1.2	<i>LBM</i> per fluidi miscibili . . . . .	162

5.2	Gravità nel <i>modello di Boltzmann</i> . . . . .	164
5.2.1	La <i>classica equazione di Boltzmann</i> . . . . .	165
5.2.2	Combinazione del termine della gravità e quello della pressione . . . . .	166
5.2.3	Aggiungendo il termine della forza in quello che definisce il momento . . . . .	168
<b>A</b>	<b><i>Approfondimenti Teorici</i></b>	<b>171</b>
A.1	Calcolo Esplicito dei <i>Moltiplicatori di Lagrange</i> . . . . .	173
A.2	<i>H-Theorem</i> di Boltzmann . . . . .	180
<b>B</b>	<b><i>Listati dei programmi</i></b>	<b>185</b>
B.1	Listato del programma “ <i>anb.f90</i> ” sul modello <i>LB</i> bi-dimensionale . . . . .	187
B.2	Listato del programma “ <i>sim.f90</i> ” sul modello <i>LBM</i> tri-dimensionale . . . . .	201
	<b>Bibliografia</b>	<b>235</b>

# Introduzione

“ La risoluzione delle equazioni matematiche è il miglior modo per simulare i fenomeni naturali ? ”

Per molti anni si è pensato che il calcolo approssimato delle soluzioni delle equazioni che regolano i fenomeni reali, mediante i tradizionali algoritmi numerici, fosse il migliore approccio nello studio dei processi naturali.

Solo negli ultimi decenni è stato proposto lo studio dei sistemi estesi attraverso l'uso di modelli basati su semplici regole, mediante le quali si cerca di imitare il comportamento microdinamico del fenomeno.

Questa classe di modelli, noti come *Automati Cellulari*, rivoluziona radicalmente l'analisi numerica nello studio dei fenomeni fisici.

L'uso del termine *'imitare'* non è stata una fortunata coincidenza, ma un modo per sottolineare come gli *Automati Cellulari* simulano la dinamica dei sistemi fisici tramite equazioni che, apparentemente, non hanno nulla a che fare con essa.

In principio, gli *AC* furono introdotti, da *von Neumann*, con l'ambiziosa idea di poter usare questo metodo nella costruzione di una macchina digitale capace di auto-riprodursi.

Tale concezione fu riformulata ed applicata allo studio di processi negli ambiti più diversi, quali la comunicazione, il calcolo, la riproduzione e l'evoluzione.

Citiamo, solo a titolo di esempio, alcuni degli ambienti di ricerca nei quali gli *AC* sono impiegati:

**Fisica:** alcuni fenomeni naturali possono essere visti come sistemi fisici contenenti un numero elevato, pur sempre discreto, di elementi, la cui evoluzione nel tempo può essere modellata tramite una regola di iterazione locale;

**Biologia:** lo sviluppo di strutture ed i modelli per la crescita degli organismi viventi, risultano governati da semplici regole non lineari;

**Computer scienze:** gli *AC* possono anche essere considerati come un modello applicabile alla computazione parallela.

Fra la classe dei modelli *AC*, nel corso della tesi, si fa particolare riferimento ai *LBM*, in quanto scelti per lo studio del problema della simulazione del moto di un fluido laminare ed omogeneo sottoposto all'azione di lame rotanti, per la loro stabilità.

Il modello *LB* differisce dai tradizionali *AC* per alcune caratteristiche a livello implementativo, pur continuando a rispettarne le proprietà fondamentali.

In maniera più specifica, nel primo capitolo si hanno dei riferimenti storici sulla nascita e lo sviluppo degli studi riguardanti gli *Automati Cellulari*, prendendo, per primo, in considerazione il modello *HPP*, che però non mostra particolare convergenza e stabilità nella simulazione. Esso risulta, tuttavia, possedere tutte le principali caratteristiche degli *AC*, ed in questo risiede la sua impor-



tanza.

Nel secondo capitolo si continua nella trattazione, mostrando come la ricerca di una sempre maggiore aderenza alla realtà, porti ad una evoluzione nello studio, e quindi alla formulazione di un modello, *FHP*, che risulta convergere alle equazioni macroscopiche che regolano i fenomeni della fluido-dinamica, ovvero l'*equazione di continuità* e l'*equazione di Navier-Stokes*. Questa sicurezza, sulla consistenza del metodo *FHP*, la si acquisisce mediante il supporto di tutto uno studio teorico, che si sviluppa, a partire dall'espansione di *Chapman-Enskog*, e procede, attraverso le fasi della *multi-scale analysis*, parallelamente all'evoluzione dei modelli.

La stabilità numerica è, però, un ingrediente fondamentale in questo tipo di analisi, ed è la sua mancanza nei modelli del tipo *FHP* nello studio dei fenomeni turbolenti, che porta a considerare e definire il *modello di Boltzmann* bi-dimensionale nel terzo e tri-dimensionale nel quarto capitolo.

Accanto alla trattazione teorica, nella quale prendono forma le equazioni da associare ad ogni particolare problematica e che ci permette di mostrare la consistenza delle metodologie utilizzate, sono riportati gli studi e le simulazioni effettuate.

Nel primo caso si è considerata la simulazione numerica del *cavity-flow*, vista più come un primo passo nello studio di questo innovativo metodo risolutivo. È, tuttavia, nel quarto capitolo, cioè nell'ambito dello sviluppo dei *LBM* tri-dimensionali, che si svolge e si risolve la problematica principale, ovvero la simulazione e lo studio, sia qualitativo che quantitativo, del moto di un fluido sottoposto all'azione di lamine rotanti, nell'ambito della quale emerge la

potenza di un modello comunque discreto sia nello spazio che nel tempo.

Nel Quinto, ed ultimo, Capitolo si è cercato di ampliare il discorso verso due particolari direzioni. La prima, riguardante la possibilità di lavorare su sistemi estesi costituiti da due fluidi diversi (sia immiscibili che miscibili), è stata risolta mediante l'applicazione di due modelli, sulle tracce di quello proposto da *Zaleski, Zanetti e Rothman* per il primo tipo, e quello proposto da *Flekkøy* per il secondo, in modo da poter controllare il moto sull'interfaccia.

La seconda, invece, riguarda l'introduzione di un termine aggiuntivo, rispetto alle usuali equazioni di aggiornamento, che ci permette di considerare l'applicazione di una forza esterna agente sul sistema (ad esempio la forza di gravità). Per la risoluzione, più modelli vengono proposti, ognuno dei quali agisce attraverso diverse metodologie nell'affrontare tale problematica.

A conclusione del lavoro sono poste due appendici.

Nella prima sono presenti degli approfondimenti teorici, di alcune questioni discusse nel corso della trattazione.

Nella seconda sono riportati i listati dei programmi sviluppati per l'implementazione del *cavity-flow* (`amb.f90`) e della *miscelazione* mediante lamine rotanti (`sim.f90`). Entrambi sono scritti in FORTRAN90.

# Capitolo 1

## Automati Cellulari

“Cellular Automata are sufficient to allow detailed mathematical analysis, yet sufficient complexly to exhibit a wide variety of complicated phenomena.”

Wolfram(1983)

Questa citazione riassume in modo molto chiaro quello che è l'approccio al reale e lo scopo degli Automati Cellulari ed è ciò che cercheremo di mostrare in questo Capitolo.

### 1.1 Cenni storici

#### 1.1.1 L'ideatore: *von Neumann*

Gli Automati Cellulari sono una *idealizzazione* dei fenomeni fisici dove lo spazio ed il tempo sono discreti<sup>1</sup>.

Tale concetto (cfr. [4]) cominciò a prendere piede intorno agli anni '40 ed il

---

<sup>1</sup>In seguito vedremo in che modo.

pioniere fu *John von Neumann*. Il suo scopo iniziale era molto più ambizioso di quello che poi si riuscì realmente a realizzare. In quel periodo egli era coinvolto nella progettazione della prima macchina digitale ed il suo intento era quello di imitare il comportamento dell'intelligenza umana per poter costruire una macchina capace di risolvere problemi anche complessi.

Il suo concetto di *automa cellulare* era però soprattutto quello di una macchina che fosse “*capace di riprodurre se stessa*”, ovvero capace di auto-controllarsi ed auto-ripararsi dal momento in cui gli fossero stati dati gli input iniziali.

Tutto questo discorso di “*auto-riproduzione*” fu poi applicato nell'ambito della simulazione di processi fisici e biologici. In che modo?

*Von Neumann* pensò che l'universo potesse essere visto, a livello microscopico, come costituito da tutte *celle*, ognuna delle quali può essere caratterizzata da uno stato “interno intero”, cioè le informazioni in esse contenute potevano essere memorizzate completamente tramite un numero finito di bit, in modo che a questo livello non si avesse alcun troncamento ed approssimazione.

Ognuna di queste celle, inoltre, si evolve, secondo time-step discreti che, come un semplice automa, conosce solo le informazioni ad esso interne e quelle delle celle vicine<sup>2</sup>. Le attività di ogni cella hanno luogo simultaneamente, secondo delle semplici regole prestabilite, uguali e per tutte<sup>3</sup>.

Tutto questo avviene, come abbiamo sopra accennato, solo una volta stabilita

---

<sup>2</sup>Sono stati realizzati diversi tipi di *neighborhood*, quelli ai quali si fa maggiore riferimento (poi ne vedremo in modo più particolareggiato la definizione) sono stati realizzati da *von Neumann*, *Moore* e *Margolus*.

<sup>3</sup>vedremo poi che solo particolari celle, cioè quelle del bordo, non rispettano tali regole, o meglio, le rispettano solo parzialmente.

una configurazione iniziale del sistema, data la quale questo può evolversi anche verso inaspettati e complessi comportamenti che arrivano a simulare processi fisici reali.

Questo è, per sommi capi, il metodo che *von Neumann* idealizzò, e poi altri portarono avanti, e che è noto come *Automati Cellulari*.

### 1.1.2 “Game of life”

Già pochi anni dopo le entusiasmanti premesse di *von Neumann*, molti furono quelli che si avventurarono nel mondo degli *AC*<sup>4</sup> e già nel 1970 abbiamo un esempio concreto e significativo da parte di *John Conway*:

*‘Game of Life’*.

Questo è un *AC* bi-dimensionale, regolato da una semplicissima regola (che non è necessariamente l’unica), che simula l’evoluzione di una società reale.

Tutto parte da uno stato iniziale randomico e si evolve supponendo che una cella è viva o morta ed a seconda del numero di celle vive o morte nelle vicinanze allo stato precedente.

Potrebbe sembrare qualcosa di estremamente banale, ma questo “gioco” mostra come, il mondo creato da *Conway*, possa avere un’evoluzione e dei comportamenti complessi e ricchi, quasi come proprio una società reale.

Questi sono sorprendenti risultati che permisero di affermare che gli *AC* risultano essere un metodo molto efficace nella simulazione numerica, o, comunque, con un approccio completamente diverso al reale.

Altri esempi seguirono, negli anni ‘80, come quelli portati da *Wolfram*, che

---

<sup>4</sup>Da ora in poi useremo questa abbreviazione per indicare gli Automati Cellulari.

cominciò uno studio dettagliato degli *AC* uni-dimensionali<sup>5</sup>; tale studio mostrò aspetti interessanti degli *AC*: questi simulavano molto bene il comportamento di sistemi continui, ma con una struttura molto più semplice rispetto ai metodi usati fino a quel momento.

Il modo di operare degli *AC*, inoltre, porta ad un'elaborazione, da parte del compilatore, che risulta essere esatta, per la loro natura *booleana* (non si hanno, quindi, nè approssimazioni nè troncamenti, come avveniva nei tradizionali metodi numerici).

Altri importanti contributi furono portati, di seguito, da ricercatori quali Tommaso Toffoli, il quale assimilò gli *AC* ad un metodo capace di simulare processi fisici empirici tramite semplici regole locali in una struttura con spazio-tempo discreta. Egli, inoltre, con la collaborazione di *Margolus* e *Fredkin*, riconobbe l'importanza degli *AC* nella simulazione dell'evoluzione dei processi reali.

L'importanza del loro lavoro, e quindi il grande contributo che essi hanno portato, fu quella di riconoscere l'utilità della reversibilità delle operazioni nei processi, poiché se questi non sono reversibili rendono possibile il verificarsi di analoghi processi finali a partire da diverse configurazioni iniziali, con la conseguente impossibilità di simulare alcune situazioni.

Queste vengono anche chiamate *garden of eden configurations* (*Moore*, 1962; *Aggarwal*, 1973).

Un esempio di tutto questo fu "billiard ball".

---

<sup>5</sup>Questo tipo di *AC* risulta molto utile per mettere in evidenza i punti forti del metodo, ma anche per realizzare quali (e non sono pochi) sono i punti deboli dello stesso; tutto questo, comunque, per poter migliorare il metodo e per poi arrivare agli entusiasmanti risultati che sono stati raggiunti.

### 1.1.3 Un passo fondamentale: dagli *AC* al *Lattice Gas Model*

Fu sempre intorno agli anni '80 che fu compiuto un passo cruciale nello sviluppo degli *AC*: si riconobbe il così detto modello *HPP*<sup>6</sup> –lattice gas model– come un *AC*.

Questo modello consiste in una dinamica completamente discreta di particelle che collidono (ovvero ogni time-step è caratterizzato da due fasi: collisione e propagazione) e si muovono in una griglia bi-dimensionale in modo da conservare massa e momento. Questo modello fu idealizzato per simulare l'interazione fra particelle di un gas.

Contemporaneamente, ma sempre in maniera indipendente, *Maxwell* stava studiando un modello, sempre legato alla dinamica dei fluidi, con particelle interagenti, con velocità discrete; anche tale metodo fa parte dei così detti *Lattice Gas Model*<sup>7</sup> (*LGM*).

La possibilità di implementare gli *LGM* come *AC* porta a dei nuovi e fondamentali sviluppi concettuali che tendono a dare molta importanza agli aspetti microscopici delle leggi fisiche coinvolte nel fenomeno.

Possiamo, quindi, arrivare ad affermare che le principali differenze, che distinguono i nuovi modelli dai metodi usati in precedenza, sono:

- I fenomeni sono studiati a partire dal livello microscopico e non macroscopico, cioè non si studia direttamente l'equazione differenziale che li governa, e gli aggiornamenti avvengono in maniera simultanea (ovvero tutte le celle sono aggiornate contemporaneamente);

---

<sup>6</sup>Questo modello fu sviluppato da *Hardy, Pomeau e Pazzis* a partire dal 1970.

<sup>7</sup>Così detti anche per distinguerli dagli *AC* veri e propri.

- Le interazioni sono locali, cioè fanno riferimento solo alle celle ‘vicine’, cioè a quelle facenti parte del *neighborhood*;
- Tutte le operazioni sono reversibili;
- Gli aggiornamenti delle celle avvengono in due fasi ben distinte: *collisione* e *propagazione*, o anche *steaming-collision*.

Quindi gli *AC* costituiscono un modo alternativo di guardare al mondo macroscopico, che risulta essere raggiunto come uno sviluppo naturale e spontaneo dell’interazione fra particelle nel microscopico.

Lo scopo, naturalmente molto ambizioso, di questa fusione fra *LBM* e *AC* era quello di poter simulare la galleria del vento tramite un modello basato sulle semplici regole degli *AC*.

Un primo tentativo, abbastanza efficace, che cerca di raggiungere tale scopo, è il famoso *FHP*<sup>8</sup>, che segue il comportamento fisico microscopico dei fluidi, ma che a livello macroscopico arriva all’equazione di *Navier-Stokes*, che regola, appunto, tale moto.

Dal momento in cui i modelli *LGM* sono stati riconosciuti come *AC*, il loro studio si è sviluppato enormemente.

Nonostante le grandi aspettative che si avevano nei confronti di questi due metodi, essi, tuttavia, non mostrarono sufficiente stabilità numerica nel momento in cui si lavorava con alti numeri di *Reynolds*.

Questo fatto risultava essere molto limitante allo scopo che ci si era prefissi, in quanto nella simulazione della galleria del vento ci si trova in presenza di fenomeni turbolenti e non laminari.

---

<sup>8</sup>Questo modello fu ideato nel 1986 da *Frisch, Hasslacher e Pomeau*.



#### 1.1.4 ‘*Lattice Boltzmann Method*’ e ‘*Multiparticle Models*’

Abbiamo visto che gli *AC*, insieme agli *LGM*, portano molti vantaggi nelle simulazioni numeriche, come, ad esempio nel momento in cui ci troviamo di fronte a complesse condizioni al bordo, questi permettono un’implementazione molto più semplice rispetto ai metodi continui (come il metodo alle differenze). Tuttavia essi mostrano molti punti deboli, dovuti alla loro *natura discreta*:

- ci troviamo in presenza di *rumori statistici*;
- non si ha una grande flessibilità nel rappresentare più ampi processi fisici.

Tutti questi problemi sono stati risolti e superati con l’introduzione, da parte di *McNamara* e *Zanetti*, di un nuovo metodo che viene ancora riconosciuto come facente parte degli *AC*, ma che si discosta da essi per il fatto che lavora direttamente con numeri reali.

Questo è il così detto ‘*Lattice Boltzmann Method*’ (*LBM*), che è un metodo molto più adatto alla simulazione di processi che coinvolgono fluidi caratterizzati da alti *numeri di Reynolds* e questo è l’argomento che cercheremo di sviluppare nel corso di questa *Tesi*.

Fra gli *AC* e *LBM* possiamo trovare un metodo intermedio: ‘*multiparticle model*’ che preserva il concetto di *celle quantizzate* ma accetta un infinito numero di valori in input (è, tuttavia, ancora in fase di studio).

Da quanto detto, possiamo concludere che gli *Automati Cellulari* cercano di catturare il comportamento microscopico di particolari fenomeni e di tradurlo, attraverso una semplice formulazione, in un modello numerico effettivo, che simula processi fisici reali macroscopici.

## 1.2 Che cosa sono gli Automi Cellulari?

Fino a questo momento abbiamo parlato degli *AC*, senza però sapere cosa in realtà essi siano.

Vediamo ora quali sono le proprietà che debbono essere soddisfatte per poter affermare di essere in presenza di un *automa cellulare*.

Gli *AC* richiedono che:

- ci sia un posizionamento di ogni *cella* tramite l'uso di una griglia regolare;
- ci sia un numero finito di variabili *booleane*, riferite ad ogni sito della griglia, che ci danno lo stato della cella ad ogni passo;
- lo stato di ogni cella sia aggiornato simultaneamente, per ogni nodo della griglia;
- le regole per l'aggiornamento debbono essere deterministiche ed uniformi<sup>9</sup>, inoltre tali regole debbono dipendere, ad ogni passo, solo dallo stato dei *vicini*, oltre che quello della cella stessa, e dallo stato al tempo precedente.

Queste regole non sempre vengono completamente soddisfatte, come abbiamo già accennato in precedenza; tale fatto, però, non implica che non possiamo più parlare di *AC*, ma di particolari tipi di essi.

---

<sup>9</sup>Dire che tali regole sono *uniformi o omogenee* significa dire che esse non dipendono esplicitamente dalla posizione della cella all'interno della griglia. Si hanno, tuttavia, delle situazioni di non omogeneità nel momento in cui andiamo a considerare le particelle che si trovano sul bordo, per le quali si debbono necessariamente applicare regole di aggiornamento diverse.

### 1.3 Neighborhood

Spesso è capitato di parlare di ‘*vicini*’, senza darne, però, la definizione precisa. L’insieme del *neighborhood* rappresenta un concetto molto importante per gli *AC*, proprio perché nella fase dell’aggiornamento le celle prese in considerazione sono quelle facenti parte di tale regione.

Diversi sono stati quelli che si sono cimentati (naturalmente con successo) nella definizione di tale insieme; tutte le definizioni, però, sono accomunate dal fatto che l’insieme dei *vicini* non è troppo ampio, in quanto, in tal caso, si potrebbe correre il rischio di ottenere ad una complessità inaccettabile delle regole di aggiornamento (in quanto tale complessità cresce esponenzialmente con il numero delle celle presenti nel *neighborhood*).

Abbiamo detto che ci sono diverse definizioni di questi *intorni*, noi citeremo solo quelle bi-dimensionali, in quanto la generalizzazione a dimensioni superiori è banale e quelle mono-dimensionali sono, praticamente, definite implicitamente.

Le più importanti, o meglio, quelle usate più spesso, sono:

- *von Neumann neighborhood*, che consiste di una cella centrale (che è quella che deve essere aggiornata) e dalle quattro vicine che si trovano a *nord*, *sud*, *est* ed *ovest* rispetto ad essa.

Formalmente si ha:

$$N_{i,j}^{(vN)} := \left\{ (k, l) \in L \mid |k - i| + |l - j| \leq 1 \right\}$$

- *Moore neighborhood* che contiene, oltre alle celle presenti nella precedente, anche *quelle sugli angoli*, quindi globalmente ha nove celle.

Anche in questo caso si ha una definizione formale:

$$N_{i,j}^{(M)} := \left\{ (k, l) \in L \mid |k - i| \leq 1 \text{ e } |l - j| \leq 1 \right\}.$$

- Infine possiamo citare, senza definirla esplicitamente, quella di *Margolus*, che divide lo spazio in blocchi di celle due per due, fra di loro adiacenti, i quali blocchi sono sfasati in maniera alterna nel tempo (le più usate, tuttavia, sono le prime due).

#### 1.4 Perché gli *AC* in fisica ?

In Fisica, l'evoluzione nel tempo dei processi è governata da equazioni differenziali parziali non lineari, la cui soluzione può essere anche molto complessa, a causa proprio della loro non linearità.

In particolare, tale soluzione, può essere fortemente dipendente dalle condizioni iniziali e variare, quindi, rispetto ad esse.

Gli *AC* costituiscono un modo alternativo di studio di tali sistemi fisico-dinamici, che portano a soluzioni che sono prive di approssimazioni ed arrotondamenti, e che quindi possiamo ritenere 'esatte'. Possiamo dire, quindi, che gli *AC* sono utili perché, attraverso semplici regole, che sono comunque legate al sistema fisico che stiamo considerando, possono mostrare complessi comportamenti dinamici.

Un altro motivo per cui gli *AC* risultano essere particolarmente adatti alla simulazione di processi fisici, risiede nel fatto che essi riescono a cogliere quelli che sono *i diversi livelli della realtà*.

Più volte abbiamo accennato al modo di operare a livello microscopico degli

*AC*, al punto da poter sembrare ripetitivi, ma questo è il concetto fondamentale legato a tale modello.

Possiamo, infatti, proprio affermare che la realtà è caratterizzata dai ‘vari livelli’ attraverso i quali si manifesta; in particolare ci sono due principali scale di osservazione: quella *macroscopica* e quella *microscopica*.

Uno stesso fenomeno può apparire enormemente diverso se osservato a livelli diversi.

A livello *microscopico* un fenomeno è governato da complicate, spesso difficili da comprendere ed interpretare, regole.

A livello *macroscopico*, invece, nel sistema possiamo osservare solo gli effetti che si generano dall’aggregazione delle interazioni microscopiche ed i risultati sono molto più vicini a ciò che potremmo ‘aspettarci’ (cioè ai fenomeni visibili nel reale).

Il verificarsi di tali eventi fisici *macroscopici*, però, è solo apparentemente disconnesso a ciò che accade nel mondo microscopico, anzi potremmo affermare che ne discende direttamente e che tutti i dettagli di tale mondo si perdono nel momento in cui emergono le caratteristiche del collettivo.

È tutto questo discorso che dà senso allo studio in scala *microscopica*, e quindi agli *AC*, che ci mostrano come questi comportamenti, apparentemente così complessi, possono essere riprodotti attraverso dei semplici modelli dinamici.

I vari modelli giocano proprio sul fatto che non c’è una vera e propria relazione fra la natura microscopica delle interazioni ed i comportamenti macroscopici, in modo da poter creare modelli il più semplici possibile.

L’importante, infatti, è *catturare il fenomeno e la legge fisica che lo governa*,

cioè basta che questo sia riconosciuto e ridotto in una forma semplice da affrontare.

Da questo punto di vista, gli *AC* sono una *caricatura*, più che un ritratto, del reale.

Questo modo di procedere, tuttavia, è essenziale nello studio scientifico, poiché permette di trattare concetti anche difficoltosi, in modo semplice ed immediato; ecco perché gli *AC* sono particolarmente adatti all'implementazione di massa.

Oltretutto essi permettono di affrontare delle situazioni, che fino a quel momento erano di non facile soluzione, in modo quasi banale, come per esempio la naturale introduzione delle condizioni al contorno.

La loro *natura booleana*, inoltre, porta ad una quasi completamente fedele rappresentazione numerica.

Concludiamo dicendo che gli *AC* possono essere visti come un *sistema spazialmente esteso*<sup>10</sup>, il che è da considerarsi come un notevole pregio.

Questo particolare tipo di modello, tuttavia, mostra una grave pecca: porta al verificarsi di comportamenti collettivi macroscopici non banali, che risultano aggiungersi al processo reale che si sta simulando, senza che però se ne abbia una rispondenza reale.

---

<sup>10</sup>I *sistemi spazialmente estesi* sono dei particolari tipi di sistemi dove una o più dimensioni spaziali sono aggiunte; questi sono tipicamente definiti in una griglia dove ogni cella è caratterizzata da uno stato locale e da un'aggiornamento sincrono di tutti i nodi.

## 1.5 *HPP*: primo esempio di *LGCA*

Il primo Lattice-Gas Cellular Automata di particolare rilievo, *LGCA*, fu proposto nel 1973 da *Hardy, Pazzis e Pomeau*.

Questo prende il nome dalle iniziali dei tre ideatori: *HPP*.

Il modello *HPP* può essere considerato come uno dei più semplici *LGCA*. L'interesse riservato a tale modello è più storico che altro, in quanto esso, a livello macroscopico, non raggiunge nemmeno l'equazione di *Navier-Stokes*.

### 1.5.1 Breve descrizione del modello

*HPP* è un *LGCA* bi-dimensionale, definito su di una griglia quadrata, come quella mostrata in Figura (1.1), caratterizzata dai vettori  $\mathbf{c}_i$  ( $i = 1, \dots, 4$ ) che connettono le varie celle presenti. Questi vettori sono le direzioni della griglia e sono definite come:

$$\mathbf{c}_i = \sin \left[ \frac{\pi}{2} (i - 1) \right] \mathbf{i} + \cos \left[ \frac{\pi}{2} (i - 1) \right] \mathbf{j}. \quad (1.1)$$

Quindi ogni particella si muove lungo queste direzioni verso la cella vicina, con una certa velocità, che è definita come la direzione diviso il time-step  $\Delta t$ , ma generalmente questo viene scelto uguale ad uno. In questo modo, le direzioni della griglia coincidono con le velocità della stessa, pur essendo dimensionalmente differenti nella realtà. Inoltre, in ogni direzione può scorrere una sola particella, applicando il *principio di esclusione* (*Principio di Pauli*), quindi ad ogni cella possiamo associare una configurazione: libera o occupata. L'evoluzione del sistema nel tempo è deterministica e procede secondo due fasi ben distinte: *collisione* e *propagazione*.

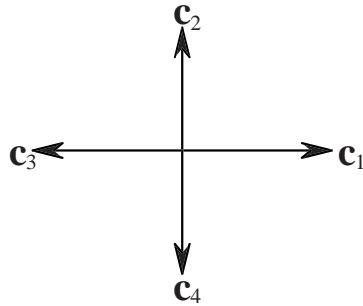


Figura 1.1: Nodo della griglia quadrata usata nel modello *HPP*; le quattro frecce rappresentano i vettori della stessa.

*Collisione:*

Coinvolge tutte le particelle che arrivano nella stessa cella in quel momento. Queste sono al più quattro (per il principio sopra citato) e giungono con una certa configurazione di entrata. Poi collidono, seguendo le regole di collisione stabilite, che nel caso del modello *HPP* sono quelle rappresentate nella Figura (1.2): se arrivano da direzioni opposte, escono in direzioni scambiate, altrimenti la loro direzione rimane inalterata, assumendo, così, una certa configurazione di uscita;

*Propagazione:*

Al contrario della precedente, questa fase coinvolge ogni singola particella lavorando direttamente sulla direzione di propagazione e non



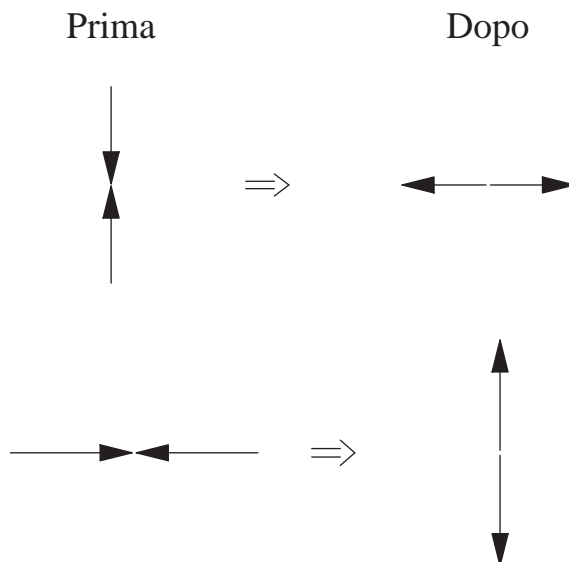


Figura 1.2: Le regole di collisione nel modello *HPP*.

sul nodo. Si parte, infatti, dalla configurazione di uscita della fase precedente, facendo “scorrere” la particella lungo il vettore che costituisce la griglia, in modo tale che questa possa arrivare al nodo successivo, e, quindi, preparando il sistema ad una nuova configurazione iniziale per il passo successivo, ed è così che ha luogo l’evoluzione nel tempo del processo.

Questo tipo di approccio cattura un aspetto importante: invarianza e reversibilità nel tempo.

Ad ogni particella, inoltre, è associata una quantità, che è il momento, il cui valore è  $m \mathbf{c}_i$ .

Tramite l’evoluzione dettata dalle regole del modello *HPP*, sia la massa della particella, che il momento totale, vengono conservati.

Nonostante ciò, tale modello non obbedisce, a livello macroscopico, all'equazione che regola l'idrodinamica (*Navier-Stokes*).

Questo risultato è imputabile all'insufficiente simmetria rotazionale della griglia, condizione che risulta essere necessaria per la simulazione del moto dei fluidi.

In aggiunta alla conservazione della massa e del momento, poi, ci sono delle altre quantità 'spurie', nel senso che queste non hanno la loro controparte nel mondo reale, il che restringe il grado di affidabilità del metodo.

### 1.5.2 Coarse graining

Non vogliamo troppo entrare nei particolari implementativi del modello, in quanto lo abbiamo citato solo come esempio; tuttavia non si può non far menzione del fenomeno del *coarse graining*.

Questo fenomeno si verifica, nel metodo *HPP*, nel momento in cui viene calcolato il valor medio per la massa e per il momento.

L'*HPP*, infatti, è caratterizzato dal fatto che, dopo l'inizializzazione dei nodi della griglia, che avviene assegnando in maniera randomica il valore zero o uno alle quantità  $n_i$ , che caratterizzano la presenza o l'assenza della particella nei vettori della griglia e sono chiamate *numeri di occupazione*, vengono calcolate delle *quantità medie*,  $N_i$ , appunto, mediando sui valori dei nodi vicini come segue:

$$N_i(t, \mathbf{x}) = \langle n_i(t, \mathbf{r}_j) \rangle \quad (1.2)$$

dove  $\mathbf{r}_j$  sono le coordinate del nodo.

Queste quantità medie vengono calcolate non rispetto ai 'vicini' nel *neighborhood*, ma rispetto ad un sub-insieme (in genere costituito da  $32 \times 32$  o  $64 \times 64$

nodii) in cui viene suddiviso l'intero dominio.

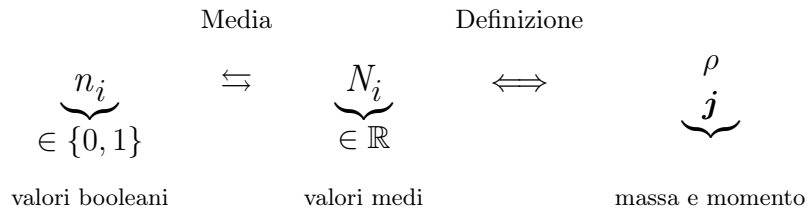
A partire, poi, dagli  $N_i$ , che assumono valori fra 0 e 1, sono definiti la densità di massa  $\rho(t, \mathbf{x})$  ed il momento  $\mathbf{j}(t, \mathbf{x})$  come:

$$\rho(t, \mathbf{x}) = \sum_{i=1}^4 N_i(t, \mathbf{x}) \quad (1.3)$$

e

$$\mathbf{j}(t, \mathbf{x}) = \rho \mathbf{u} = \sum_{i=1}^4 \mathbf{c}_i N_i(t, \mathbf{x}) \quad (1.4)$$

dove  $\mathbf{u}$  è la velocità del nodo. È da questi valori, poi, che si sviluppa tutto il metodo. Questa media, che è anche possibile rispetto al tempo ed allo spazio contemporaneamente, permette di rendere l'implementazione molto più veloce.



Questa Figura riassume in modo schematico quella che è la relazione fra il livello microscopico (*array booleani*  $n_i$ ) e quello macroscopico (*densità di massa e momento*).



## Capitolo 2

### *FHP Lattice Gas Model*

In questo capitolo cercheremo di trattare in maniera un po' più approfondita gli *LGCA* bi-dimensionali e mostreremo come già questi modelli arrivino, a livello macroscopico, all'equazione di *Navier-Stokes*.

A tale scopo, cominciamo con il modello *FHP* ([7]), che prende il nome dalle iniziali dei nomi di coloro che, nel 1986, lo idearono:

*Frisch, Hasslacher e Pomeau*. Essi riconobbero il fatto che il modello *lattice-gas*, implementato su di una griglia con una maggiore simmetria rispetto a quella quadrata, risulta possedere maggiore stabilità e simulare, a livello macroscopico, i fenomeni idrodinamici.

#### **2.1 Le caratteristiche generali del metodo**

L' *FHP* risulta essere un modello, da noi considerato nel caso bi-dimensionale, implementato su di una griglia esagonale, come quella mostrata nella Figura (2.1), o meglio su di una griglia costituita da triangoli che risultano essere

invarianti rispetto a rotazioni di  $60^\circ$  (*simmetria esagonale*). Questo significa che in ogni nodo possiamo trovare esattamente sei collegamenti, cioè le così dette *velocità della griglia*  $\mathbf{c}_i$ <sup>1</sup>, e, quindi, si hanno sei ‘vicini’. In tal modo si raggiunge una maggiore regolarità nella griglia, condizioni che risultano essere necessarie per poter ottenere la convergenza del metodo (motivo per cui il modello *HPP* non risultava essere adatto alle simulazioni idrodinamiche).

La necessità di tale simmetria risulta essere evidente nel momento in cui si calcolano i tre momenti associati alle velocità della griglia  $\mathbf{c}_i$ , che risultano assumere valori particolarmente semplici. Tutto questo si traduce come<sup>2</sup>:

$$\sum_i \mathbf{c}_i = 0 \quad (2.1)$$

$$\sum_i c_{i\alpha} c_{i\beta} = 3 \delta_{\alpha\beta} \quad (2.2)$$

$$\sum_i c_{i\alpha} c_{i\beta} c_{i\gamma} = 0 \quad (2.3)$$

In ogni nodo può, inoltre, essere presente una sola particella, come accadeva nel modello *HPP*, rispettando il *Principio di esclusione di Pauli*, il suo stato, quindi, può essere completamente definito tramite un numero finito di bit che stabiliscono se il nodo è *occupato* o è *vuoto*. Ogni particella, inoltre, è dotata di una massa  $m$  che risulta essere indistinguibile.

Un'altra importante caratteristica, che distingue i due metodi, è che il model-

---

<sup>1</sup>Queste sono così definite, pur essendo delle lunghezze, in quanto il time-step  $\Delta t$  è spesso scelto come unitario e quindi  $\mathbf{c}_i$  e  $\mathbf{c}_i/\Delta t$  hanno lo stesso valore numerico, anche se risultano dimensionalmente diversi. Inoltre, siccome tutte le particelle hanno la stessa massa,  $m=1$ ,  $\mathbf{c}_i$  è anche il momento della particella.

<sup>2</sup>Notiamo che, le lettere greche (usate come pedice) si riferiscono alle coordinate cartesiane, mentre le altre lettere (usate sempre come pedice), si riferiscono ai vettori collegamento della griglia, o meglio, al numero dei ‘link’ presenti nel nodo.

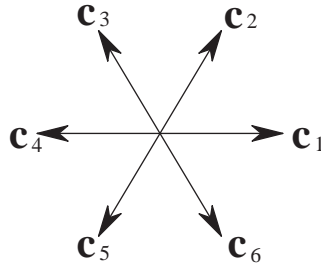


Figura 2.1: Griglia esagonale usata nel metodo *FHP*.

lo *FHP* supera le *regole deterministiche*. Ora, infatti, le possibili opzioni sulla configurazione di uscita sono stabilite attraverso metodi probabilistici.

La generazione di stati randomici, infatti, è un'operazione molto costosa, quindi si è cercato di optare per una scelta *pseudo-randomica*, che si attua invertendo l'orientazione dei vettori velocità della griglia ad ogni passo. Il tutto, quindi, si effettua attraverso l'uso di particolari collisioni che possono avvenire non solo fra due particelle entranti, come nel caso del modello *HPP*, ma anche fra più di due<sup>3</sup> (per i vari tipi di collisioni vedere i grafici annessi (2.2)).

In quel caso, infatti, le collisioni fra due particelle conservavano non solo la massa ed il momento, che sono l'ingrediente fondamentale per poter ottenere la dinamica di un fluido a livello macroscopico, ma anche la differenza fra il numero di particelle che scorrono lungo le varie direzioni della griglia. Questo fatto porta al crearsi delle così dette *invarianti spurie*, che generano, a livello

---

<sup>3</sup>In realtà nei grafici, come nel seguito del testo, verranno considerate solo le collisioni fra due o tre particelle. Questo non significa che nel nodo non possano arrivare un numero superiore di particelle, ma solo che se tale evento si verifica, la collisione è come se non avvenisse affatto; ovvero che le particelle proseguono nel loro cammino come se le altre non fossero presenti.

macroscopico, fenomeni inaspettati e non reali.

Questo tipo di situazione viene superata nel modello *FHP* aggiungendo le collisioni simmetriche fra tre particelle.

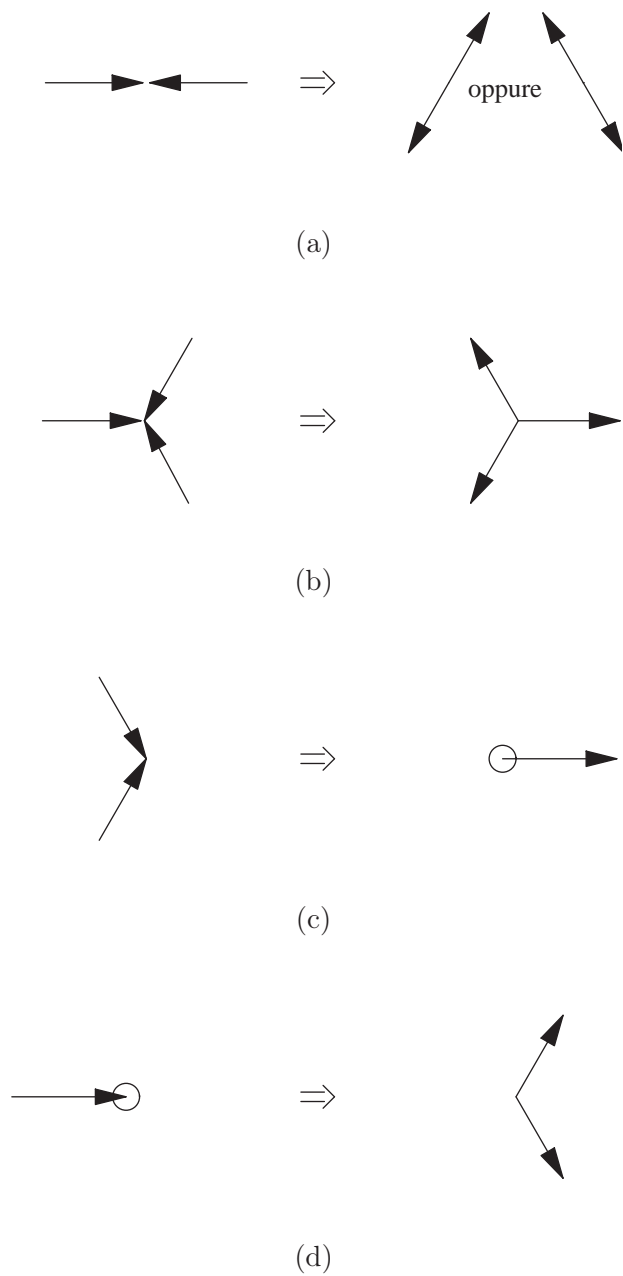
Il problema, in realtà, non è del tutto risolto, infatti, nonostante tale tipo di collisione sia simmetrico, in modo da eliminare alcune delle *invarianti spurie*, non lo è abbastanza da eliminarle tutte. Fu proprio *Zanetti*, infatti, nel 1989, a trovarne in tutte le varianti degli *FHP*; la loro presenza, tuttavia, non influenza in maniera eccessivamente evidente la dinamica macroscopica o, quanto meno, tali modelli simulano ancora con buona approssimazione il fenomeno reale.

Ora, tutte le collisioni citate costituiscono un'insieme minimale per il modello *FHP*, che viene detto *FHP-I* ([10]); a queste se ne possono aggiungere delle altre (come si può banalmente notare dalla simmetria della griglia stessa) che portano ad altre varianti del modello base:

***FHP-I* :**

Come abbiamo detto in precedenza, questo rappresenta il modello *FHP* base, che è, cioè, costituito dalle sole collisioni fra due e tre particelle (esempi (a) e (b) nella Figura (2.2)), senza considerare affatto la possibile presenza di particelle ferme all'interno del nodo. In tutte le considerazioni che saranno effettuate nel corso del seguente capitolo (riguardo i processi microscopici) si farà sempre riferimento a questo modello, per semplificare i calcoli e perché, comunque, per i nostri scopi, risulta essere sufficiente (in quanto la nostra attenzione sarà maggiormente diretta al modello *LBM*);



Figura 2.2: Collisioni nel metodo *FHP*.

***FHP- II :***

Questo modello introduce la possibilità di poter trovare una particella ferma nel nodo (come avviene negli esempi (c) e (d) della Figura (2.2)), oltre, naturalmente, alle collisioni già considerate nel modello precedente;

***FHP- III :***

Questo modello risulta essere un ampliamento dei due precedenti, nel senso che include la presenza di una particella ferma, non solo però nel caso in cui sia possibile avere una o due particelle entranti nel nodo, ma include anche la possibilità di avere tre ‘link’ contemporaneamente occupati. Il modello *FHP-III* completa, quindi, tutte le possibili collisioni che si possono incontrare nel *FHP*.

Dopo tale discorso, possiamo sintetizzare le principali caratteristiche del *FHP* come segue:

- la griglia presenta simmetria esagonale;
- ogni nodo è collegato a sei celle vicine attraverso dei vettori che hanno tutti la stessa lunghezza;
- i vettori che collegano i nodi hanno tutti lunghezza unitaria,  $|\mathbf{c}_i| = 1$ , e sono definiti come:

$$\mathbf{c}_i = \left( \cos \frac{\pi}{3} i, \sin \frac{\pi}{3} i \right), \quad i = 1, \dots, 6; \quad (2.4)$$

- ogni nodo può essere al più occupato da una particella, cioè vale anche in questo caso il *Principio di esclusione di Pauli*;

- tutte le particelle hanno la stessa massa indistinguibile  $m=1$ ;
- l'evoluzione del sistema, che indichiamo con  $\mathcal{E}$ , avviene attraverso la composizione di due fasi distinte: *collisione*  $\mathcal{C}$  e *propagazione*  $\mathcal{P}$ , in modo che si abbia:

$$\mathcal{E} = \mathcal{P} \circ \mathcal{C} \quad (2.5)$$

- le collisioni sono strettamente locali, cioè queste coinvolgono ogni singolo nodo e quelli presenti nel *neighborhood* considerata.

## 2.2 Caratteristiche microdinamiche

### 2.2.1 Descrizione booleana

Per poter conoscere lo stato di ogni cella ad ogni passo è sufficiente sapere se il vettore  $i$ -esimo che la collega al nodo vicino sia *libero* o *occupato*. Tale situazione è completamente descritta dal vettore *booleano*  $n_i(t, \mathbf{r})$ , definito come:

$$n_i(t, \mathbf{r}) = \begin{cases} 1 & \text{se il link è occupato} \\ 0 & \text{se il link è libero} \end{cases}$$

dove  $\mathbf{r}$  rappresenta il vettore posizione del nodo all'interno della griglia.

Questa quantità, tuttavia, ci specifica solo la situazione all'istante di tempo  $t$  e nella posizione  $\mathbf{r}$ . Per poter, quindi, ottenere l'evoluzione del sistema, dobbiamo conoscere il suo stato 'in ogni nodo' ed 'ad ogni iterata nel tempo'.

Dobbiamo, a tale scopo, specificare il valore della quantità  $n_i(t+1, \mathbf{r} + \mathbf{c}_i)$ , che, a livello microdinamico è descritta tramite delle relazioni che utilizzano l'algebra *booleana* e che, quindi, risultano essere di non facile lettura.

Allo scopo di semplificare i calcoli e di riportare il tutto ad una formulazione aritmetica (che è ciò che noi vogliamo), si introduce una particolare funzione detta *funzione-collisioni*  $\Delta_i$  :

$$\Delta_i = \begin{cases} 1 & \text{se la particella arriva al nodo lungo il vettore } \mathbf{c}_i \\ 0 & \text{se la particella non è nel nodo lungo il vettore } \mathbf{c}_i \\ -1 & \text{se la particella esce dal nodo lungo il vettore } \mathbf{c}_i \end{cases}$$

Quindi, come sopra accennato, l'evoluzione è descrivibile come :

$$n_i(t+1, \mathbf{r} + \mathbf{c}_i) = n_i(t, \mathbf{r}) + \Delta_i \quad i = 1, \dots, 6 \quad (2.6)$$

dove  $\mathbf{r} + \mathbf{c}_i$  indica la posizione dei nodi vicini al tempo  $t+1$ .

Notiamo come, esaminando la (2.6), vediamo essere rispettate tutte le caratteristiche del metodo *FHP* :

- 'l'evoluzione di una cella dipende solo dallo stato delle particelle presenti nei nodi vicini';
- 'lo stato al tempo  $t+1$  è determinato solo dalla configurazione assunta al tempo immediatamente precedente  $t$ '.

Come è noto, e come più volte abbiamo detto, la caratteristica fondamentale della dinamica dei fluidi incomprimibili (che sono quelli di cui ci stiamo occupando) è la conservazione, a livello *macroscopico*, della *massa* ed del *momento della quantità di moto*, che possono essere espresse attraverso le seguenti imposizioni su  $\Delta_i$  :

$$\sum_{i=1}^6 \Delta_i = 0 \quad (2.7)$$

e

$$\sum_{i=1}^6 \mathbf{c}_i \Delta_i = 0 \quad (2.8)$$

con  $v_i = m \mathbf{c}_i = \mathbf{c}_i$  ( $m = 1$ ) velocità della particella  $i$ -esima.

Queste due equazioni si traducono a livello *microscopico* tramite delle relazioni sulle quantità *booleane*  $n_i$ . Facendo, quindi, riferimento alla (2.6), si ottengono:

$$\sum_i n_i(t+1, \mathbf{r} + \mathbf{c}_i) = \sum_i n_i(t, \mathbf{r}) \quad (2.9)$$

e

$$\sum_i \mathbf{c}_i n_i(t+1, \mathbf{r} + \mathbf{c}_i) = \sum_i \mathbf{c}_i n_i(t, \mathbf{r}) \quad (2.10)$$

### 2.2.2 Densità di massa e momento

Per avere, tuttavia, l'evoluzione nel tempo del sistema, ciò che è fondamentale conoscere, sono le quantità medie calcolate rispetto alle  $n_i$ , e quindi, di conseguenza, la *densità di massa* e *del momento* ad esse associate. Qui di seguito vedremo come poter ottenere ciò.

Calcoliamo, partendo dalle  $n_i$ , i *valori medi di occupazione*, utilizzando quella che viene detta l'*equazione di Liouville*<sup>4</sup>. In generale, per ottenere tali medie, rispetto ad una funzione  $q(n(t, \dots))$ , si opera come segue:

$$\langle q(n(t, \dots)) \rangle := \sum_{s(\cdot) \in \Gamma} q(s(\cdot)) P(t, s(\cdot)) \quad (2.11)$$

e quindi nel nostro caso si ottiene:

$$N_i(t, \mathbf{r}) = \langle n_i(t, \mathbf{r}) \rangle \quad i = 1, \dots, b \quad (2.12)$$

<sup>4</sup>Non entrando troppo nei particolari, che riguardano la meccanica statistica, diciamo solo che la definizione di tale equazione è:  $P(t+1, \mathcal{E} s(\cdot)) = P(t, s(\cdot))$ , con  $s(\cdot) \in \Gamma$ , cioè lo spazio delle fasi (ovvero lo spazio di tutte le possibili configurazioni che il sistema può assumere).

dove  $b$  è il numero di collegamenti presenti nella griglia.

A partire da queste possiamo, quindi definire la densità di massa :

$$\rho(t, \mathbf{r}) := \sum_i N_i(t, \mathbf{r}) \quad (2.13)$$

ed il momento :

$$\mathbf{j}(t, \mathbf{r}) := \sum_i \mathbf{c}_i N_i(t, \mathbf{r}) \quad (2.14)$$

Facendo riferimento alle definizioni appena date di queste quantità, anche ricordando quelle che sono le loro dimensioni fisiche, possiamo direttamente affermare che vale la relazione *momento = massa · velocità*, quindi :

$$\mathbf{j}(t, \mathbf{r}) := \rho(t, \mathbf{r}) \mathbf{u}(t, \mathbf{r})$$

dove  $\mathbf{u}(t, \mathbf{r})$  è la velocità del flusso.

Abbiamo, quindi, definito le quantità medie che stavamo cercando, ed ora dobbiamo imporre che anche le (2.12) verifichino delle relazioni analoghe a quelle per le  $n_i$  per la conservazione della massa (2.9) e del momento (2.10) :

$$\sum_i N_i(t+1, \mathbf{r} + \mathbf{c}_i) = \sum_i N_i(t, \mathbf{r}) \quad (2.15)$$

$$\sum_i \mathbf{c}_i N_i(t+1, \mathbf{r} + \mathbf{c}_i) = \sum_i \mathbf{c}_i N_i(t, \mathbf{r}) \quad (2.16)$$

### 2.2.3 Conclusioni teoriche: $N_i^{eq}$

Vediamo, in questo paragrafo che conclude la descrizione del metodo vero e proprio, come attraverso le due condizioni sulla conservazione della massa e del momento, espresse dalle equazioni (2.15) e (2.16), sia possibile dare una regola di aggiornamento dei nodi facenti parte della nostra griglia, attraverso una formula analitica semplice e compatta, che a questo livello verrà usata

solo marginalmente (cioè nella derivazione delle equazioni macroscopiche). La sua reale e fondamentale utilità, verrà fuori quando parleremo del modello di Boltzmann e lì sarà usata non solo per tale motivo, ma soprattutto come regola di aggiornamento, ovvero servirà per ottenere l'evoluzione del sistema stesso. Questo è il risultato teorico veramente importante che viene fuori dall'analisi del metodo *FHP*, cioè la definizione dei così detti *numeri di occupazione media di equilibrio*:  $N_i^{eq}$ .

A tale scopo *Frisch et al.* (1987) dimostrarono il seguente teorema (del quale riportiamo solo l'enunciato):

**Teorema 2.2.1.** (*Frisch et al., 1987*)

*Le seguenti affermazioni sono equivalenti:*

1. *Gli  $N_i^{eq}$  soddisfano l'equazione di Liouville;*
2. *Gli  $N_i^{eq}$  sono soluzione delle equazioni:*

$$\Delta_i(N) = 0 \quad \forall i = 1, \dots, b \quad (2.17)$$

*dove  $b$  è il numero dei vettori che costituiscono il nodo;*

3. *Gli  $N_i^{eq}$  sono dati dalla distribuzione di Fermi-Dirac:*

$$N_i^{eq} = \frac{1}{1 + \exp(h + \mathbf{q} \cdot \mathbf{c}_i)} \quad (2.18)$$

*dove  $h$  è un numero reale e  $\mathbf{q}$  è un vettore  $D$ -dimensionale<sup>5</sup>.*

Nel teorema precedente, tuttavia, abbiamo detto che cosa rappresentano le quantità  $h$  e  $\mathbf{q}$ , ma non ne abbiamo specificato il valore. Per determinare

<sup>5</sup> $D$  è la dimensione dello spazio in cui stiamo lavorando.

tali *moltiplicatori di Lagrange*<sup>6</sup> facciamo, quindi, uso delle uniche condizioni che abbiamo fino ad ora a disposizione, cioè quelle sulla conservazione della densità di massa e del momento :

$$\rho = \sum_i N_i = \sum_i \frac{1}{1 + \exp(h + \mathbf{q} \cdot \mathbf{c}_i)} \quad (2.19)$$

e

$$\rho \mathbf{u} = \sum_i \mathbf{c}_i N_i = \sum_i \frac{\mathbf{c}_i}{1 + \exp(h + \mathbf{q} \cdot \mathbf{c}_i)} \quad (2.20)$$

La soluzione di tali equazioni (che ci permettono di calcolare i due moltiplicatori e quindi le quantità medie che vogliamo conoscere) sono note soltanto in alcuni casi particolari, al contrario di quanto accadeva nel modello *HPP*, cioè i *moltiplicatori di Lagrange* sono calcolabili in maniera esplicita, attraverso un'espansione, solo nel momento in cui ci troviamo in presenza di bassi *numeri di Mach*, che sono definiti come  $M_a := U/c_s$ , dove  $U = |\mathbf{u}|$ , cioè la velocità del flusso  $\mathbf{u}$ , deve essere molto al di sotto della così detta 'sound speed'  $c_s$ .

Tale sviluppo, nel caso del modello *FHP-I*<sup>7</sup> (che è quello che abbiamo considerato fino ad ora), è dato da :

$$N_i^{eq}(\rho, \mathbf{u}) = \frac{\rho}{6} + \frac{\rho}{3} \mathbf{c}_i \cdot \mathbf{u} + \rho G(\rho) Q_{i\alpha\beta} u_\alpha u_\beta + \mathcal{O}(u^3) \quad (2.21)$$

dove

$$G(\rho) = \frac{1}{3} \frac{6 - 2\rho}{6 - \rho} \quad (2.22)$$

---

<sup>6</sup>Così vengono definite  $h$  e  $\mathbf{q}$ , in quanto portano alla distribuzione di *Fermi-Dirac* attraverso le imposizioni sulla massa (2.15) e sul momento (2.16), che soddisfano le condizioni di massima entropia per il sistema (*Principio di Massima Entropia*).

<sup>7</sup>Il calcolo esplicito di  $N_i^{eq}$  sarà svolto nell'Appendice A (vista la sua lunghezza non volevo appesantire troppo il paragrafo, che deve essere una descrizione generale).



e

$$Q_{i\alpha\beta} = c_{i\alpha} c_{i\beta} - \frac{1}{2} \delta_{\alpha\beta}. \quad (2.23)$$

dove  $c_{i\alpha}$ , al variare di  $i$ , sono le componenti cartesiane del vettore velocità del lattice  $\mathbf{c}_i$ .

*Frisch et al.* mostrarono, in realtà, una formula valida per le quantità medie  $N_i^{eq}$  molto più generale, ossia:

$$N_i^{eq}(\rho, \mathbf{u}) = \frac{\rho}{b} + \frac{\rho D}{c^3 b} c_{i\alpha} u_{i\alpha} + \rho G(\rho) Q_{i\alpha\beta} u_\alpha u_\beta + \mathcal{O}(u^3) \quad (2.24)$$

con

$$G(\rho) = \frac{D^2}{2c^4 b} \frac{b - 2\rho}{b - \rho} \quad (2.25)$$

e

$$Q_{i\alpha\beta} = c_{i\alpha} c_{i\beta} - \frac{c^2}{D} \delta_{\alpha\beta}. \quad (2.26)$$

dove  $b$  è il numero dei vettori della griglia considerata e  $D$  è la dimensione dello spazio nel quale stiamo lavorando. Questa, inoltre, risulta essere valida non solo per il metodo *FHP-I* (per il quale è stato svolto il calcolo esplicito), ma anche per tutti gli altri, ovvero *HPP*, *FHP-II* e *FHP-III*. Prima di concludere il paragrafo, notiamo anche che nella (2.21) si ha la presenza del termine quadratico in  $\mathbf{u}$ . Vogliamo sottolineare tale fatto proprio perché è questa quantità che porta al termine non lineare presente nell'equazione di *Navier-Stokes* ed è questo il motivo per cui è possibile troncare lo sviluppo di  $N_i^{eq}$  al secondo ordine, ottenendo un'approssimazione comunque sufficiente.

## 2.3 Derivazione delle equazioni macroscopiche:

### *Eulero e Navier-Stokes*

La derivazione delle equazioni macroscopiche, e quindi dell'*equazione di continuità* e dell'*equazione di Navier-Stokes*, che governano il moto dei fluidi, viene detta *multi-scale analysis*, ed è uno dei traguardi raggiunti più importanti.

Si afferma ciò, in quanto questa rappresenta una conferma tangibile, e quindi una vera e propria dimostrazione, del fatto che il nostro modello simula ciò che noi realmente vogliamo: *l'idrodinamica dei fluidi incomprimibili*, che è per l'appunto regolata da tali equazioni. Tuttavia, prima di considerare tale analisi, dedichiamo una parte del nostro discorso allo studio della parte teorica che si è sviluppata a tale scopo, e quindi all'*equazione di Boltzmann* ed all'espansione di *Chapman-Enskog*, che sono fondamentali nella derivazione delle equazioni macroscopiche, sia in questo che nel momento in cui andremo a considerare il metodo di *Boltzmann (LBM)*, ecco perché cercheremo di trattarla in maniera dettagliata.

#### 2.3.1 Equazione di *Boltzmann*

Il moto dei fluidi può essere descritto a partire da diversi punti di vista. Uno dei modi più semplici è quello di utilizzare le equazioni di *Hamilton* per un insieme di particelle. Tuttavia, se pensiamo che in  $1\text{cm}^3$  di aria ci sono  $2,69 \cdot 10^{19}$  molecole, ci rendiamo conto che l'uso di tale approccio rende impensabile la costruzione di un modello che lo governi, per la quantità enorme di equazioni necessarie.

Ecco spiegata l'introduzione dell'equazione di *Boltzmann*, che viene fuori da una serie di approssimazioni applicate a tale idea di fondo. L'equazione che ne deriva risulta avere un'espressione integro-differenziale, rispetto ad una generica distribuzione  $f(t, \mathbf{x}, \mathbf{v})$ :

$$\partial_t f + \mathbf{v} \partial_{\mathbf{x}} f + \frac{\mathbf{K}}{m} \partial_{\mathbf{v}} f = Q(f, f) \quad (2.27)$$

dove  $m$  è la massa della particella,  $Q(f, f)$  è l'*integrale delle collisioni* e  $\mathbf{K}$  è la forza applicata<sup>8</sup>.

Osserviamo, inoltre, che l'*integrale delle collisioni*  $Q(f, f)$  deve verificare le *condizioni di invarianza per le collisioni*.

### Osservazione 2.1.

*Le equazioni che ci esprimono questa condizione, possono essere formulate tramite l'uso di cinque elementari funzioni  $\psi_k(\mathbf{v})$  ( $k = 0, \dots, 4$ ) nel senso che:*

$$\int Q(f, f) \psi_k(\mathbf{v}) d^3 v = 0, \quad k = 0, \dots, 4. \quad (2.28)$$

*Queste funzioni  $\psi_k(\mathbf{v})$  risultano essere proporzionali alla massa, al momento ed all'energia cinetica ed inoltre, nel caso particolare, sono esprimibili come:*

$$\psi_0 = 0; \quad (\psi_1, \psi_2, \psi_3) = \mathbf{v}; \quad \psi_4 = \mathbf{v}^2$$

---

<sup>8</sup>Tale termine non sarà considerato in ciò che verrà discusso nel seguente capitolo, ma sarà introdotto solo in seguito.

Invece la loro espressione generale, che indichiamo con  $\phi(\mathbf{v})$ , può essere scritta come una combinazione lineare delle sopra definite  $\psi_k(\mathbf{v})$ :

$$\phi(\mathbf{v}) = a \mathbf{b} \cdot \mathbf{v} + c \mathbf{v}^2$$

Le condizioni espresse dalla (2.28) sono soddisfatte se esiste una funzione  $f$  tale che:

$$Q(f, f) = 0$$

Un caso particolarmente interessante di funzione che rende nullo l'integrale delle collisioni, è dato dalla *distribuzione di Maxwell*:

$$f^{(M)} = f(t, \mathbf{r}, \mathbf{v}) = n \left( \frac{m}{2\pi k_B T} \right)^{3/2} \exp \left[ -\frac{m}{2k_B T} (\mathbf{v} - \mathbf{u})^2 \right] \quad (2.29)$$

dove  $\mathbf{v}$  è la velocità del flusso, ed  $\mathbf{u}$  è la velocità media.

Uno dei maggiori problemi che si incontrano nell'utilizzo dell'equazione di Boltzmann risiede nella complicata espressione dell'integrale delle collisioni. Ecco perché si è cercato di sostituire  $Q(f, f)$  con un operatore  $J(f)$ , per il quale debbono valere:

- $J(f)$  deve ancora soddisfare le *condizioni di invarianza per le collisioni*, sempre rispetto alle funzioni  $\psi_k(\mathbf{v})$ .

Questo significa che debbono valere le seguenti 5 relazioni integrali:

$$\int \psi_k J(f) d^3x d^3v = 0, \quad k = 0, \dots, 4.$$

- Il termine di collisione deve ancora avere l'andamento della *distribuzione di Maxwell*<sup>9</sup>.

---

<sup>9</sup>La validità di tale proprietà è dimostrata nel famoso *H-Theorem* di Boltzmann.

Questo verrà preso in considerazione nell'Appendice A.

Ambo le condizioni sono verificate dall'operatore :

$$J(f) = \omega \left[ f^{(M)}(\mathbf{x}, \mathbf{v}) - f(\mathbf{x}, \mathbf{v}) \right] \quad (2.30)$$

dove  $\omega$  è chiamata *frequenza di collisione*.

Questo termine viene utilizzato nella così detta *approssimazione BGK*<sup>10</sup> esprimibile tramite la seguente equazione:

$$\partial_t f + \mathbf{v} \partial_{\mathbf{x}} f + \frac{\mathbf{K}}{m} \partial_{\mathbf{v}} f = J(f) \quad (2.31)$$

### 2.3.2 Espansione di *Chapman-Enskog*

Questa tecnica è stata sviluppata da *Chapman* ed *Enskog* fra gli anni 1910 e 1920 ed inizialmente fu utilizzata anche per altri scopi, prima che se ne scoprisse il fondamentale uso nella *multi-scale analysis*.

Consideriamo, in modo del tutto generale, una distribuzione  $f = f(t, \mathbf{r}, \mathbf{u})$ , che dipende esplicitamente dal tempo  $t$ , dallo spazio  $\mathbf{r}$  e dalla velocità  $\mathbf{u}$ ; vedremo, poi, che il termine  $n$ -esimo  $f^{(n)}$  nell'espansione di *Chapman-Enskog* dipende solo implicitamente dal tempo, attraverso la densità locale, la velocità e la temperatura.

Cominciamo, quindi, considerando l'espansione, rispetto ad  $\epsilon$ , della distribuzione  $f(t, \mathbf{r}, \mathbf{u})$ :

$$f = f^{(0)} + \epsilon f^{(1)} + \epsilon^2 f^{(2)} + \dots \quad (2.32)$$

che esprime proprio l'espansione di *Chapman-Enskog*.

---

<sup>10</sup>Il metodo prende il nome dalle iniziali di coloro che lo idearono nel 1954: *Bhatnagar, Gross e Krook*.

Il simbolo  $\epsilon$  è usato con due significati:

1. Si parla di espansione come serie di potenze rispetto alla quantità, piccola,  $\epsilon$ , cioè  $|\epsilon| \ll 1$ . Nel caso dell'espansione di *Chapman-Enskog*, queste piccole quantità, sono i così detti *numeri di Knudsen*,  $K_n$ <sup>11</sup>;
2. Il parametro formale  $\epsilon$  nell'espansione rappresenta la traccia, nei relativi ordini, dei vari termini e variabili.

Come esempio consideriamo solamente  $f = f^{(0)} + \epsilon f^{(1)}$ .

Analogo sviluppo in  $\epsilon$  può essere considerato nella derivazione rispetto al tempo  $t$ , ottenendo:

$$\partial_t = \epsilon \partial_t^{(1)} + \epsilon^2 \partial_t^{(2)} + \dots$$

Se ora facciamo riferimento a quanto detto nel paragrafo precedente riguardo l'*approssimazione BGK*, all'equazione (2.32) ed al fatto che  $\rho = \rho(t, \mathbf{x}, \mathbf{u})$ ,  $j = j(t, \mathbf{x}, \mathbf{u})$  e  $\theta = \theta(t, \mathbf{x}, \mathbf{u})$ <sup>12</sup>, le leggi di conservazione possono essere riformulate come:

$$\begin{aligned} \partial_t \rho + \partial_{x_\alpha} (\rho u_\alpha) &= 0 \\ \rho \partial_t u_\alpha + \rho u_\beta \partial_{x_\beta} u_\alpha &= - \sum_{n=0}^{\infty} \epsilon^n \partial_{x_\alpha} \widehat{P}_{\alpha\beta}^{(n)} \end{aligned}$$

dove  $\widehat{P}_{\alpha\beta}^{(n)}$  è una quantità integrale definita, a partire dall'*equazione di Boltz-*

<sup>11</sup>Sono il rapporto fra la lunghezza media (cioè la distanza media fra due nodi) e la scala caratteristica del sistema usato. Quando il valore di tali  $K_n$  è dell'ordine dell'unità, il gas del sistema considerato non può essere descritto come un fluido.

<sup>12</sup> $\theta$  rappresenta il termine della temperatura che sarà trascurato nella trattazione seguente in quanto non influisce direttamente sul fenomeno, quindi nella simulazione, ed è quindi per questo motivo che tale termine non sarà più considerato.

mann, come segue :

$$\widehat{P}_{\alpha\beta}^{(n)} := m \int d^3 v f^{(n)} (v_\alpha - u_\alpha) (v_\beta - u_\beta) \quad (2.33)$$

Siccome  $f^{(n)}$ , come già accennato, dipende dal tempo  $t$  solo implicitamente attraverso  $\rho$  e  $\mathbf{u}$ , possiamo scrivere :

$$\partial_t f = \partial_\rho f \partial_t \rho + \partial_{u_\alpha} f \partial_t u_\alpha$$

Per poter continuare i calcoli, dopo la sostituzione dei termini considerati nell'equazione che ci da la derivata della funzione  $f$  rispetto al tempo, ovvero  $\partial_t f$ , dobbiamo specificare il valore dell'espansione di  $\partial_t \rho$  e  $\partial_t u_\alpha$ , in modo da ottenere delle quantità che siano consistenti con le leggi di conservazione espresse dalle (2.13) e (2.14) in ogni ordine in  $\epsilon$  :

$$\partial_t^{(1)} \rho := -\partial_{u_\alpha} (\rho u_\alpha) \quad (2.34)$$

$$\partial_t^{(n+1)} \rho := 0 \quad (n > 0) \quad (2.35)$$

$$\partial_t^{(1)} u_\alpha := -u_\alpha \partial_{x_\beta} u_\alpha - \frac{1}{\rho} \widehat{P}_{\alpha\beta}^{(0)} \quad (2.36)$$

$$\partial_t^{(n+1)} u_\alpha := -\frac{1}{\rho} \widehat{P}_{\alpha\beta}^{(n)} \quad (n > 0) \quad (2.37)$$

Applicando la (2.32) nelle derivate rispetto a  $\rho$  e  $u_\alpha$  si ottiene :

$$\begin{aligned} \partial_\rho f &= \partial_\rho f^{(0)} + \epsilon \partial_\rho f^{(1)} + \epsilon^2 \partial_\rho f^{(2)} + \dots \\ \partial_{u_\alpha} f &= \partial_{u_\alpha} f^{(0)} + \epsilon \partial_{u_\alpha} f^{(1)} + \epsilon^2 \partial_{u_\alpha} f^{(2)} + \dots \end{aligned}$$

Inoltre tali espansioni debbono essere definite in maniera tale che siano consistenti con le leggi di conservazione ad ogni livello rispetto ad  $\epsilon$ . Applicando

tutte queste considerazioni al termine  $\partial_t f$  si ottiene:

$$\begin{aligned}\partial_t f &= \left( \epsilon \partial_t^{(1)} + \epsilon^2 \partial_t^{(2)} + \dots \right) \left( f^{(0)} + \epsilon f^{(1)} + \epsilon^2 f^{(2)} + \dots \right) \\ &= \epsilon \partial_t^{(1)} f^{(0)} + \epsilon^2 \left( \partial_t^{(1)} f^{(1)} + \partial_t^{(2)} f^{(0)} \right) + \epsilon^3 \dots\end{aligned}$$

Sostituendo l'espansione di  $f$  nell'*integrale delle collisioni*  $Q(f, f)$  si ottiene:

$$\begin{aligned}Q(f, f) &= -\omega (f - f^{(0)}) \\ &= -\omega (\epsilon f^{(1)} + \epsilon^2 f^{(2)} + \dots) \\ &=: J^{(0)} + \epsilon J^{(1)} + \epsilon^2 J^{(2)} + \dots\end{aligned}\tag{2.38}$$

dove

$$\begin{aligned}J^{(0)}(f^{(0)}) &= 0 \\ J^{(1)}(f^{(0)}, f^{(1)}) &= J^{(1)}(f^{(1)}) = -\omega f^{(1)} \\ J^{(2)}(f^{(0)}, f^{(1)}, f^{(2)}) &= J^{(2)}(f^{(2)}) = -\omega f^{(2)} \\ &\dots\end{aligned}$$

e  $\omega$  è la *frequenza di collisione*, che assume valore costante.

Se ora consideriamo la derivata spaziale  $\partial_x$  come:

$$\partial_{x_\alpha} = \epsilon \partial_{x_\alpha}^{(1)}$$

ed eguagliamo i termini dello stesso ordine in  $\epsilon$  nell'equazione di *Boltzmann*, si ottiene:

$$\begin{aligned}J^{(0)}(f^{(0)}) &= 0 \tag{2.39} \\ \partial_t^{(1)} f^{(0)} + v_\alpha \partial_{x_\alpha}^{(1)} f^{(0)} &= J^{(1)}(f^{(0)}, f^{(1)}) = -\omega f^{(1)} \tag{2.40} \\ \partial_t^{(1)} f^{(1)} + \partial_t^{(2)} f^{(0)} + v_\alpha \partial_{x_\alpha}^{(1)} f^{(1)} &= J^{(2)}(f^{(0)}, f^{(1)}, f^{(2)}) = -\omega f^{(2)} \\ &\dots\end{aligned}$$



Sappiamo che la prima di queste (2.39) risulta essere soddisfatta se si effettua la particolare scelta  $f^{(0)} = f^{(M)}$ , poiché in tal caso  $J$  soddisfa la *condizione di invarianza per collisioni*, e quindi si annulla, se valutata rispetto a tale distribuzione (di tipo Maxwelliano). Tuttavia questa non è l'unica scelta possibile per tale termine che, a livello macroscopico, ci porta all'*equazione di continuità*. La (2.40) ci permette di calcolare  $f^{(1)}$  come:

$$\boxed{f^{(1)} = -\frac{1}{\omega} \left( \partial_t^{(1)} f^{(0)} + v_\alpha \partial_{x_\alpha}^{(1)} f^{(0)} \right)} \quad (2.41)$$

Per lo scopo che ci siamo prefissi il calcolo fino a quest'ordine è sufficiente.

Notiamo che, ora risulta essere possibile, grazie a tutti i risultati ottenuti fino ad ora, effettuare il calcolo della quantità  $\widehat{P}_{\alpha\beta}^{(n)}$  ed è ciò che ora andremo a fare.

Notiamo, tuttavia, che se ricordiamo che vale la (2.34), e quindi tutte le derivate rispetto al tempo della densità  $\rho$  di ordine superiore al secondo risultano essere nulle, l'unico tensore che ha senso calcolare è  $\widehat{P}_{\alpha\beta}^{(1)}$ .

Tale calcolo può essere effettuato facendo riferimento alla definizione stessa, data nella (2.33), di tale quantità ed al calcolo appena fatto di  $f^{(1)}$ .

Possiamo, quindi, scrivere:

$$\begin{aligned} \widehat{P}_{\alpha\beta}^{(1)} &= m \int d^3 v (v_\alpha - u_\alpha) (v_\beta - u_\beta) f^{(1)} \\ &= -\frac{m}{\omega} \int d^3 v (v_\alpha - u_\alpha) (v_\beta - u_\beta) (\partial_t^{(1)} f^{(0)} + v_\gamma \partial_{x_\gamma}^{(1)} f^{(0)}). \end{aligned}$$

Il calcolo delle derivate presenti nell'integrale avviene applicando la regola della catena alla  $f^{(0)}$ , in quanto dipende da  $t$  e da  $x_\gamma$  solo tramite  $\rho$  e  $\mathbf{u}$ , ed

usando le relazioni considerate in precedenza ed espresse dalle (2.34)-(2.37)<sup>13</sup>:

$$\begin{aligned}
\partial_t f^{(0)}(\rho, \mathbf{u}) &= \frac{\partial f^{(0)}}{\partial \rho} \frac{\partial \rho}{\partial t} + \frac{\partial f^{(0)}}{\partial u_\gamma} \frac{\partial u_\gamma}{\partial t} \\
&= -\frac{f^{(0)}}{m} \frac{\partial(\rho u_\gamma)}{\partial(x_\gamma)} + \frac{m}{k_B T} (v_\gamma - u_\gamma) f^{(0)} \left( u_\delta \frac{\partial u_\gamma}{\partial x_\delta} + \frac{1}{\rho} \frac{\widehat{P}_{\alpha\beta}^{(0)}}{\partial x_\delta} \right) \\
&= -\frac{\rho f^{(0)}}{m} \frac{\partial u_\gamma}{\partial(x_\gamma)} - \frac{f^{(0)}}{m} u_\gamma \frac{\partial \rho}{\partial(x_\gamma)} + \frac{m}{k_B T} (v_\gamma - u_\gamma) f^{(0)} u_\delta \frac{\partial u_\gamma}{\partial x_\delta} \\
&\quad + \frac{m}{k_B T} (v_\gamma - u_\gamma) f^{(0)} \frac{1}{\rho} \delta_{\gamma\delta} \frac{\delta p}{\partial x_\delta}
\end{aligned}$$

ponendo

$$\delta_{\gamma\delta} \frac{\delta p}{\partial x_\delta} = \frac{\widehat{P}_{\alpha\beta}^{(0)}}{\partial x_\delta}$$

che discende semplicemente dalla definizione di  $\widehat{P}_{\alpha\beta}^{(0)}$ .

Nei passaggi, oltre alle equazioni citate, si è tenuto conto anche del fatto che  $f^{(0)}$  è una distribuzione di tipo *maxwelliano*.

Inoltre l'altra derivata, nell'integrale considerato, si può calcolare con analoghi ragionamenti, ottenendo:

$$\begin{aligned}
v_\gamma \partial_{x_\gamma} f^{(0)} &= v_\gamma \frac{\partial f^{(0)}}{\partial \rho} \frac{\partial \rho}{\partial x_\gamma} + v_\gamma \frac{\partial f^{(0)}}{\partial u_\gamma} \frac{\partial u_\gamma}{\partial x_\gamma} \\
&= v_\gamma \frac{f^{(0)}}{\rho} \frac{\rho}{x_\gamma} + v_\gamma \frac{m}{k_B T} (v_\delta - u_\delta) f^{(0)} \frac{\partial u_\delta}{\partial x_\gamma}
\end{aligned}$$

---

<sup>13</sup>Per semplificare la notazione, non utilizziamo più l'indice (1) per indicare i diversi ordini di derivazione rispetto al tempo, anche perché consideriamo solo il primo.

Ora i diversi termini nell'integrale possono essere facilmente valutati:

$$\begin{aligned}
 & -\frac{1}{m} \frac{\partial u_\gamma}{\partial x_\gamma} \int d^3v (v_\alpha - u_\alpha) (v_\beta - u_\beta) f^{(0)} = -\delta_{\alpha\beta} n \frac{m}{k_B T} \frac{\partial u_\gamma}{\partial x_\gamma} \\
 & \frac{1}{m} \frac{\partial \rho}{\partial x_\gamma} \int d^3v (v_\alpha - u_\alpha) (v_\beta - u_\beta) (v_\gamma - u_\gamma) f^{(0)} = 0 \\
 & \frac{m}{k_B T} f^{(0)} \left( u_\gamma \frac{\partial u_\gamma}{\partial x_\delta} + \frac{1}{\rho} \delta_{\gamma\delta} \frac{\partial p}{\partial x_\delta} \right) \cdot \\
 & \cdot \int d^3v (v_\alpha - u_\alpha) (v_\beta - u_\beta) (v_\gamma - u_\gamma) f^{(0)} = 0 \\
 & \frac{m}{k_B T} \frac{\partial u_\delta}{\partial x_\gamma} \int d^3v (v_\alpha - u_\alpha) (v_\beta - u_\beta) u_\gamma (v_\delta - u_\delta) f^{(0)} \\
 = & (\delta_{\alpha\beta} \delta_{\gamma\delta} + \delta_{\alpha\gamma} \delta_{\beta\delta} + \delta_{\delta\alpha} \delta_{\beta\gamma}) n \frac{k_B T}{m} \frac{\partial u_\delta}{\partial x_\gamma}
 \end{aligned}$$

tenendo conto del fatto che valgono le già citate *condizioni di invarianza per collisioni* per  $J(f)$ , che sono espresse tramite le condizioni integrali (2.28).

E così possiamo concludere scrivendo,:

$$\begin{aligned}
 \widehat{P}_{\alpha\beta}^{(1)} &= -n \frac{k_B T}{\omega} \left[ (\delta_{\alpha\beta} \delta_{\gamma\delta} + \delta_{\alpha\gamma} \delta_{\beta\delta} + \delta_{\delta\alpha} \delta_{\beta\gamma}) \frac{\partial u_\delta}{\partial x_\gamma} - \delta_{\alpha\beta} \frac{\partial u_\gamma}{\partial x_\gamma} \right] \\
 &= n \frac{k_B T}{\omega} \begin{pmatrix} 2 \frac{\partial u}{\partial x} & \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} & \frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} \\ \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} & 2 \frac{\partial v}{\partial y} & \frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \\ \frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} & \frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} & 2 \frac{\partial w}{\partial z} \end{pmatrix} \quad (2.42)
 \end{aligned}$$

dove  $n$  è la costante che compare nella definizione della *distribuzione di Maxwell* e che ci indica il numero di collegamenti presenti nei nodi della griglia.

A questo punto, se trascuriamo la variazione della densità di massa e della temperatura (approssimazione possibile solamente a questo livello, ma non lo sarà più quando andremo a considerare il *modello di Boltzmann*), possiamo

calcolare la divergenza di  $\widehat{P}_{\alpha\beta}^{(1)}$ :

$$\begin{aligned} \frac{\partial \widehat{P}_{\alpha\beta}^{(1)}}{\partial x_\alpha} &= \mu \left[ 2 \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} + \frac{\partial^2 v}{\partial x \partial y} + \frac{\partial^2 w}{\partial x \partial z} \right] \mathbf{e}_x + \dots \\ &= \mu \left[ \frac{\partial}{\partial x_\beta} \left( \frac{\partial u_\alpha}{\partial x_\beta} \right) + \frac{\partial}{\partial x_\alpha} \left( \frac{\partial u_\beta}{\partial x_\beta} \right) \right] \\ &= \mu \left[ \nabla^2 \mathbf{u} + \nabla (\nabla \cdot \mathbf{u}) \right] \end{aligned}$$

dove

$$\mu = n \frac{k_B T}{\omega} \quad (2.43)$$

è la viscosità dinamica.

Si arriva in questo modo, a seconda dell'equazione di equilibrio presa in considerazione, all'equazione di *Navier-Stokes*.

Notiamo quindi, come fermandoci al primo ed al secondo ordine nell'espansione di *Chapman-Enskog* rispetto ad  $\epsilon$ , otteniamo le due equazioni che volevamo e che regolano completamente il moto del nostro sistema. Nel prossimo paragrafo vedremo esplicitamente tali calcoli nel caso del modello *FHP*, dove la funzione di equilibrio considerata è la (2.21).

### 2.3.3 'Multi-Scale Analysis'

Dopo la digressione teorica, effettuata nei precedenti paragrafi, cerchiamo ora di addentrarci nella *multi-scale analysis*, usando i concetti sopra introdotti.

Nel metodo *FHP*, i numeri di occupazione media  $N_i^{eq}$ , risultano essere dipendenti in maniera continua, dalle quantità medie della densità di massa e del momento. All'inizio della simulazione, cioè nel momento dell'inizializzazione dei nodi ( $t = 0$ ). Allo scorrere del tempo, si verificano in particolare tre fenomeni :

1. Rilassamento verso una situazione di equilibrio locale, che corrisponde, nella scala del tempo, all'ordine  $\epsilon^0$ : risulta, quindi, un processo molto veloce;
2. Perturbazione e variazione della densità di massa, che corrisponde al termine di ordine  $\epsilon^{-1}$  nel tempo;
3. Diffusione, corrispondente all'ordine  $\epsilon^{-2}$ .

A questo livello noi considereremo solo i processi che avvengono con una variazione nel tempo corrispondente all'ordine  $(\epsilon^{-1})^{14}$ . Facendo, quindi, riferimento agli sviluppi effettuati in precedenza, andiamo a considerare i vari ordini di derivazione rispetto al tempo:

$$\begin{aligned} t_* & \\ t_1 &= \epsilon t_* \\ t_2 &= \epsilon^2 t_* \end{aligned}$$

e rispetto allo spazio, per il quale consideriamo solo un ordine di derivazione, in quanto il fenomeno della perturbazione della massa e la propagazione, avvengono rispetto allo stesso ordine:

$$\begin{aligned} \mathbf{r}_* & \\ \mathbf{r}_1 &= \epsilon \mathbf{r}_* \end{aligned}$$

Ora andiamo ad applicare lo sviluppo di *Chapman-Enskog* ai valori medi  $N_i$ :

$$N_i = N_i^{(0)}(t, \mathbf{r}) + \epsilon N_i^{(1)}(t, \mathbf{r}) + \mathcal{O}(\epsilon^2) \quad (2.44)$$

---

<sup>14</sup>Solo nel caso del metodo *LBM* sarà trattato il caso rispetto ad  $\epsilon^{-2}$ , in quanto l'introduzione della distribuzione di equilibrio nel metodo stesso, rende i calcoli molto più semplici.

Il termine di ordine zero è, quindi, quello di equilibrio :

$$N_i^{(0)}(t, \mathbf{r}) \equiv N_i^{eq}(\rho, \mathbf{u}) \quad (2.45)$$

Se ora ci riferiamo alle equazioni (2.13) e (2.14), notiamo che il termine di correzione lineare in  $\epsilon$  non dà alcun contributo al calcolo della densità di massa e del momento, in quanto valgono :

$$\sum_i N_i^{(1)}(t, \mathbf{r}) = 0 \quad e \quad \sum_i \mathbf{c}_i N_i^{(1)}(t, \mathbf{r}) = 0 \quad (2.46)$$

Facendo, quindi, riferimento alle definizioni di  $\rho$  e  $\mathbf{j}$ , otteniamo le relazioni :

$$\rho = \sum_i N_i(t, \mathbf{r}) = \sum_i N_i^{(0)}(t, \mathbf{r}) \quad (2.47)$$

$$\mathbf{j} = \sum_i \mathbf{c}_i N_i(t, \mathbf{r}) = \sum_i \mathbf{c}_i N_i^{(0)}(t, \mathbf{r}) \quad (2.48)$$

Il punto di partenza per la *multi-scale analysis* si ha nel momento in cui  $\rho$  (2.47), e  $\mathbf{j}$ , (2.48), soddisfano le condizioni, date a livello microscopico, sulla conservazione della massa e del momento, (2.15) e (2.16).

Imponendole otteniamo :

$$\begin{aligned} \left\{ \begin{array}{c} 1 \\ \mathbf{c}_i \end{array} \right\} N_i(t+1, \mathbf{r} + \mathbf{c}_i) &= \left\{ \begin{array}{c} 1 \\ \mathbf{c}_i \end{array} \right\} \left[ N_i(t, \mathbf{r}) + \partial_t N_i + c_{i\alpha} \partial_{x_\alpha} N_i \right. \\ &\quad \left. + \frac{1}{2} \partial_t \partial_t N_i + \frac{1}{2} c_{i\alpha} c_{i\beta} \partial_{x_\alpha} \partial_{x_\beta} N_i N_i \right. \\ &\quad \left. + c_{i\alpha} \partial_t \partial_{x_\alpha} N_i + \mathcal{O}(\partial^3 N_i) \right] \end{aligned} \quad (2.49)$$

Le equazioni macroscopiche, quindi, si ottengono dalla (2.49) considerando una derivazione nello spazio e nel tempo, rispetto ad  $\epsilon$  come :

$$\begin{aligned} \partial_t &\longrightarrow \epsilon \partial_t^{(1)} + \epsilon^2 \partial_t^{(2)} \\ \partial_{x_\alpha} &\longrightarrow \epsilon \partial_{x_\alpha}^{(1)} \end{aligned}$$

Ora, inserendo tutte queste relazioni ottenute, nelle (2.15) e (2.16), si ha<sup>15</sup>:

$$\begin{aligned}
 & \sum_i \left\{ \begin{array}{c} 1 \\ \mathbf{c}_i \end{array} \right\} \left[ N_i(t+1, \mathbf{r} + \mathbf{c}_i) - N_i(t, \mathbf{r}) \right] \\
 = & \sum_i \left\{ \begin{array}{c} 1 \\ \mathbf{c}_i \end{array} \right\} \left[ \epsilon \partial_t^{(1)} N_i^{(0)} + \epsilon^2 \partial_t^{(1)} N_i^{(1)} + \epsilon^2 \partial_t^{(2)} N_i^{(0)} \right. \\
 & + \epsilon c_{i\alpha} \partial_{x_\alpha}^{(1)} N_i^{(0)} + \epsilon^2 c_{i\alpha} \partial_{x_\alpha}^{(1)} N_i^{(1)} + \frac{1}{2} \epsilon^2 \partial_t^{(1)} \partial_t^{(1)} N_i^{(1)} \\
 & \left. + \frac{1}{2} \epsilon^2 c_{i\alpha} c_{i\beta} \partial_{x_\alpha}^{(1)} \partial_{x_\beta}^{(1)} N_i^{(0)} + \epsilon^2 c_{i\alpha} \partial_t^{(1)} \partial_{x_\alpha}^{(1)} N_i^{(1)} \right] \\
 = & 0
 \end{aligned}$$

Le equazioni macroscopiche si ottengono proprio a partire da questo livello, imponendo l'uguaglianza a zero dei termini calcolati rispetto ai diversi ordini:

$$\partial_t^{(1)} \sum_i N_i^{(0)} + \partial_{x_\beta}^{(1)} \sum_i c_{i\beta} N_i^{(0)} = 0 \quad (2.50)$$

e

$$\partial_t^{(1)} \sum_i c_{i\alpha} N_i^{(0)} + \partial_{x_\beta}^{(1)} \sum_i c_{i\alpha} c_{i\beta} N_i^{(0)} = 0 \quad (2.51)$$

Sostituendo i valori della densità di massa e del momento, espressi dalle (2.47) e (2.48), la precedente si può riformulare come:

$$\partial_t^{(1)} \rho + \partial_{x_\beta}^{(1)} (\rho u_\beta) = 0 \quad (2.52)$$

$$\partial_t^{(1)} (\rho u_\alpha) + \partial_{x_\beta}^{(1)} P_{\alpha\beta}^{(0)} = 0 \quad (2.53)$$

Dove  $P_{\alpha\beta}^{(0)}$  è il  *tensore del flusso*  ed è dato da:

$$\begin{aligned}
 P_{\alpha\beta}^{(0)} & \equiv \sum_i c_{i\alpha} c_{i\beta} N_i^{(eq)} \\
 & = \rho \delta_{\alpha\beta} + \rho G(\rho) T_{\alpha\beta\gamma\delta}^{MA} u_\gamma u_\delta + \mathcal{O}(u^4)
 \end{aligned} \quad (2.54)$$

<sup>15</sup>Considerando solo i termini fino al secondo ordine in  $\epsilon$ .

mentre  $T^{MA}$  è un tensore del quarto ordine, definito come :

$$T_{\alpha\beta\gamma\delta}^{MA} = \frac{3}{4} (\delta_{\alpha\gamma} \delta_{\beta\delta} + \delta_{\alpha\delta} \delta_{\beta\gamma} - \delta_{\alpha\beta} \delta_{\gamma\delta}). \quad (2.55)$$

Se ora calcoliamo le varie componenti del *tensore del flusso* otteniamo :

$$\begin{aligned} P_{xx}^{(0)} &= \rho G(\rho) (u^2 - v^2) + \frac{\rho}{2} \\ P_{yy}^{(0)} &= \rho G(\rho) (v^2 - u^2) + \frac{\rho}{2} \\ P_{xy}^{(0)} &= P_{yx}^{(0)} = \rho G(\rho) 2uv \end{aligned} \quad (2.56)$$

tuttavia nel mondo reale (cioè nell'equazione di *Navier-Stokes*), queste sono date da :

$$\begin{aligned} P_{xx} &= \rho u^2 + p \\ P_{yy} &= \rho v^2 + p \\ P_{xy} &= P_{yx} = \rho uv \end{aligned} \quad (2.57)$$

La (2.57) si ottiene dalla seguente, considerando  $p = \rho/2 (1 - g(\rho) u^2)$

$$\begin{aligned} P_{xx}^{(0)} &= \rho g(\rho) u^2 + p \\ P_{yy}^{(0)} &= \rho g(\rho) v^2 + p \\ P_{xy}^{(0)} &= P_{yx}^{(0)} = \rho g(\rho) uv \end{aligned} \quad (2.58)$$

che è molto simile al *tensore del flusso* (2.56), se non per il fattore  $g$

$$g(\rho) = \frac{3}{2} G(\rho)$$

dove  $G(\rho)$  è definito nella (2.25).

Sostituendo tale definizione, ne possiamo dare il valore esplicito come :

$$g(\rho) = \frac{3 - \rho}{6 - \rho} = \frac{1}{2} \frac{6 - 2\rho}{6 - \rho}$$



Prima di continuare, notiamo che, per piccoli valori della velocità media  $\mathbf{u}^2$ , la pressione può essere espressa attraverso la relazione ‘*isoterma*’:

$$p = \frac{\rho}{2} = c_s^2 \rho$$

dove  $c_s = 1/\sqrt{2}$  è la *velocità del suono* nel fluido esaminato (anche detta *sound-speed*).

Dopo queste considerazioni, ritorniamo alle (2.50) e (2.51).

L’equazione che regola il sistema costituito da fluidi incomprimibili, discende da quella generale considerando la densità costante<sup>16</sup>,  $\rho = \rho_0$ , in tutti i termini, tranne che nel termine che contiene la pressione.

Se consideriamo, nella prima, la derivazione rispetto a tutte le componenti dello spazio, otteniamo:

$$\begin{cases} \partial_t \rho + \partial_x (\rho u_x) = 0 \\ \partial_t \rho + \partial_y (\rho u_y) = 0 \end{cases}$$

Espressione che può anche essere scritta come:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \mathbf{u} = 0$$

ma  $\rho$  è costante rispetto al tempo, quindi vale:

$$\nabla \cdot \mathbf{u} = 0$$

che è proprio l’*equazione di continuità*.

Se ora, invece consideriamo la seconda delle due equazioni (cioè la (2.51)) ed

---

<sup>16</sup>Questa approssimazione è possibile solo in questo caso particolare, in quanto nell’approssimazione, ci fermiamo a considerare solo i termini fino al primo ordine in  $\epsilon$ .

applichiamo le considerazioni fatte sul *tensore del flusso* e sulla densità  $\rho$ , all'equazione (2.53), si ottiene:

$$\rho_0 g(\rho_0) \frac{\partial \mathbf{u}}{\partial t} + \rho_0 g(\rho_0) (\mathbf{u} \cdot \nabla) \mathbf{u} = -\nabla \left( \frac{\rho}{2} - \frac{\rho}{2} g(\rho_0) \mathbf{u}^2 \right) \quad (2.59)$$

Come, però, abbiamo notato in precedenza, la (2.59) non è proprio uguale alle equazioni macroscopiche, ma differisce da esse per un termine  $g(\rho_0)$ .

Per arrivare ad esse, quindi, basta riscalarlo il tempo<sup>17</sup> rispetto al fattore  $g$

$$t \quad \longrightarrow \quad \frac{t}{g(\rho)} \quad (2.60)$$

e di conseguenza anche il termine della pressione ne rimane modificato.

Si ottiene, così, l'equazione di Eulero

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} = -\nabla P \quad (2.61)$$

dove si è indicato con  $P$  la *pressione cinematica*:

$$P = \left( \frac{\rho}{2 \rho_0 g(\rho_0)} - \mathbf{u}^2 \right) \quad (2.62)$$

### Osservazione 2.2.

*L'equazione di Eulero risulta essere l'equazione di Navier-Stokes senza alcun termine dissipativo.*

Se, ora, nello sviluppo di *Chapman-Enskog* consideriamo anche i termini in  $\epsilon^2$ , trascurando tutti gli altri, otteniamo le equazioni differenziali complete:

$$\boxed{\begin{aligned} \nabla \cdot \mathbf{u} &= 0 \\ \partial_t \mathbf{u} + (\mathbf{u} \cdot \nabla) \mathbf{u} &= -\nabla P + \nu \nabla^2 \mathbf{u} \end{aligned}} \quad (2.63)$$

<sup>17</sup>Operazione che non altera le equazioni di partenza.

dove  $\nu$  è la viscosità cinematica (per il cui valore, nei diversi modelli si può fare riferimento alla Figura (1)).

Siamo così riusciti ad ottenere ciò che volevamo: l'equazione di *Navier-Stokes*.

**Nota 2.1.**

*Nel momento in cui andiamo a riscaldare le equazioni che otteniamo con un'approssimazione del secondo ordine in  $\epsilon$ . Il termine da considerare non è, però, solo quello del tempo, ma bisogna anche tener conto della viscosità  $\nu$ :*

$$\nu \longrightarrow g(\rho)\nu \quad (2.64)$$

*Tuttavia, notiamo anche che quello sopra effettuato non è l'unico possibile riscalo, infatti possiamo anche considerare*

$$\mathbf{u} \longrightarrow \frac{\mathbf{u}}{g(\rho)} \quad (2.65)$$

In ogni caso, è possibile definire il *numero di Reynolds*

$$\mathcal{R}e = \frac{U L}{\nu(\rho_0)} \quad (2.66)$$

sostituendo

$$\mathcal{R}e = U L \frac{g(\rho_0)}{\nu(\rho_0)} \quad (2.67)$$

dove  $U$  ed  $L$  sono, rispettivamente, la velocità e la lunghezza caratteristiche del modello.

	<i>FHP-I</i>	<i>FHP-II</i>	<i>FHP-III</i>
$\rho$	$6d$	$7d$	$7d$
$c_s$	$\frac{1}{\sqrt{2}}$	$\sqrt{\frac{3}{7}}$	$\sqrt{\frac{3}{7}}$
$g(d)$	$\frac{1}{2} \frac{1-2d}{1-d}$	$\frac{7}{12} \frac{1-2d}{1-d}$	$\frac{7}{12} \frac{1-2d}{1-d}$
$\nu$	$\frac{1}{12} \frac{1}{d(1-d)^3} - \frac{1}{8}$	$\frac{1}{28} \frac{1}{d(1-d)^3} \frac{1}{1-4d/7} - \frac{1}{8}$	$\frac{1}{28} \frac{1}{d(1-d)} \frac{1}{1-8d(1-d)/7} - \frac{1}{8}$
$\eta$	0	$\frac{1}{98} \frac{1}{d(1-d)^4} - \frac{1}{28}$	$\frac{1}{98} \frac{1}{d(1-d)} \frac{1}{2d(1-d)} - \frac{1}{28}$

**Tabella 1.** Nella tabella,  $\rho$  è la densità media;  $\nu$  e  $\eta$  sono due particolari tipi di viscosità cinematiche (che possono anche coincidere in alcuni modelli); ‘d’ è la densità all’interno di ogni cella.

## 2.4 Osservazioni conclusive sul *FHP*

Riguardo tale metodo, le cose da aggiungere e sviluppare più dettagliatamente, sarebbero molte. Tuttavia, l’*FHP* è stato citato solo come un importante momento di passaggio dagli *Automati Cellulari* al *metodo di Boltzmann*, ecco perché faremo solo dei cenni ad argomenti, che poi discuteremo in maniera più dettagliata proprio nel modello *LBM*: **condizioni al bordo**

L’implementazione delle *CB*<sup>18</sup> è una parte molto importante in un qualsiasi modello, in particolare in questo caso, poiché la loro introduzione all’interno del codice è particolarmente semplice.

<sup>18</sup>Sigla usata per indicare le *condizioni al bordo*.

Se ne possono considerare diversi tipi :

- *CB periodiche* ;
- *CB “no-slip”*, cioè dove si considera nulla la velocità ( $\mathbf{u} = 0$ ) e vengono usate nel caso di bordi solidi ;
- *CB “slip”*, dove la componente normale della velocità è nulla ( $u_n = 0$ ), mentre la componente tangente soddisfa la seguente relazione sulla derivata prima:  $\partial u_t / \partial n = 0$ .

Osserviamo che nelle *CB “no-slip”*, che sono le più usate, anche nei listati dei programmi annessi, le particelle che arrivano nella zona di bordo, vengono rimandate indietro dalla stessa parte, cioè la direzione del loro moto è ruotata di  $180^\circ$ . Questa regola applicata a tali particelle viene anche detta *regola del ‘bounce-back’*. La relazione matematica usata è :

$$n_i(\mathbf{x}_b + \mathbf{c}_i, t + 1) = n_{i+3}(\mathbf{x}_b, t)$$

dove  $\mathbf{x}_b$  è il punto di bordo considerato. Questo metodo di implementazione risulta essere così semplice in quanto è indipendente dall’orientazione del bordo stesso.

#### ***Altri tipi di FHP***

Noi abbiamo, in particolare, considerato il modello *FHP-I*, facendo dei brevi cenni anche agli *FHP-II* e *FHP-III*. Questi, tuttavia, non esauriscono tutte le possibili varianti, in quanto ne sono state proposte alcune, che differiscono fra di loro non solo nelle collisioni, ma anche nella griglia (o meglio, nel numero di ‘link’ presenti nel nodo), che risulta assumere sempre maggiore simmetria, nel

numero di particelle ferme considerate (*multi-speed FHP*) e nella dimensione dello spazio in cui ci troviamo.

Come ultima considerazione, diciamo che l'estensione a dimensioni superiori, in particolare alla terza dimensione, è stata sviluppata pensando di usare delle geometrie sufficientemente regolari. Tuttavia, in tre dimensioni sono pochi i solidi che soddisfano tale condizione, e sono i così detti *solidi di Platone*. Un'altra possibilità è stata esaminata: considerare delle geometrie nella quarta dimensione (di molto più semplice definizione), la cui proiezione in  $3D$  ci consente di ottenere delle geometrie, in cui le griglie hanno dei vettori con la medesima lunghezza, ed inoltre con una sufficiente regolarità ed una certa simmetria spaziale.

## Capitolo 3

### *Il Modello di Boltzmann*

Il modello  $LB^1$  risulta essere un'evoluzione del metodo *FHP*, che supera i problemi ad esso associati. Nel seguente capitolo, quindi, vedremo come la somma di tutte queste modifiche ci porti alla definizione del modello di base, che sarà descritto nel caso particolare di un unico fluido (cioè nei nodi della griglia le particelle hanno tutte le stesse caratteristiche) al quale non sono applicate forze esterne.

#### 3.1 Sviluppo del *modello di Boltzmann*

##### 3.1.1 Dal *FHP* al *LBM*

Gli *Automati Cellulari* (e tutti i modelli che fino ad ora sono stati riconosciuti come tali) permettono di implementare, un certo fenomeno fisico, in maniera esatta, in quanto utilizzano quantità intere per definire lo stato di ogni nodo. Questo fatto, se da un lato è utile per l'accuratezza che si ottiene nella

---

<sup>1</sup>Sigla usata per indicare il *metodo Lattice Boltzmann*.

simulazione, dall'altro porta al manifestarsi di 'fluttuazioni' in sistemi costituiti da molte particelle. Il problema che affligge in particolar modo gli *AC* è il verificarsi di "rumori", oltre al fatto che la sua natura *booleana* lo rende poco flessibile e che il dover mediare su di un certo numero di nodi ( $32 \times 32$  generalmente) porta a dover considerare un numero esageratamente grande di collisioni.

Questo si verifica in quanto tutte le quantità in gioco sono calcolate come media solo dopo aver applicato il metodo. L'idea è, allora, quella di scambiare la valutazione dei due processi, ovvero sembrerebbe essere più vantaggioso applicare il metodo direttamente alle variabili mediate. Ciò comporta che gli ' $N_i$ ' non sono più variabili intere, ma reali comprese nell'intervallo  $[0, 1]$ . Comunque gli *LBM*, che si basano su tale nuova concezione, sono ancora considerati come facenti parte degli *AC*, in quanto la struttura ed il metodo di fondo sono gli stessi.

### 3.1.2 Equazione ed approssimazione di Boltzmann

La microdinamica degli *LGCA* è generalmente descritta (come visto nel capitolo precedente) da una relazione del tipo:

$$n_i(\mathbf{x} + \mathbf{c}_i \Delta t, t + \Delta t) = n_i(\mathbf{x}, t) + \Omega_i \quad (3.1)$$

dove gli  $\Omega_i$  rappresentano le collisioni del rispettivo modello.

Questo tipo di collisioni (che potremmo definire discrete) possono essere espresse anche usando l'*equazione di Boltzmann* (2.27):

$$\frac{\partial f}{\partial t} + \mathbf{v} \nabla f = Q \quad (3.2)$$



dove  $Q$  è il già citato integrale delle collisioni<sup>2</sup>.

La corrispondenza fra la (3.1) e la (3.2) risulta evidente se consideriamo lo sviluppo in serie di  $n_i(\mathbf{x} + \mathbf{c}_i\Delta t, t + \Delta t)$ :

$$n_i(\mathbf{x} + \mathbf{c}_i\Delta t, t + \Delta t) = n_i(\mathbf{x}, t) + \Delta t \frac{\partial n_i}{\partial t} + \mathbf{c}_i\Delta t \nabla n_i + \mathcal{O}\left((\Delta t)^2\right) \quad (3.3)$$

Trascurando i termini di ordine superiore al primo si ha:

$$\frac{\partial n_i}{\partial t} + \mathbf{c}_i \nabla n_i = \frac{\Delta n_i}{\Delta t} \quad (3.4)$$

dalla quale, ponendo:

$$n_i \longrightarrow f \qquad \mathbf{c}_i \longrightarrow \mathbf{v} \qquad \frac{\Delta n_i}{\Delta t} \longrightarrow Q$$

otteniamo proprio la (3.2).

Se oltretutto applichiamo anche l'*approssimazione di Boltzmann* (2.30) all'integrale delle collisioni  $Q(f, f)$ , otteniamo:

$$\frac{\partial f}{\partial t} + \mathbf{v} \nabla f = -\frac{1}{\tau} \left( f - f^{(eq)} \right). \quad (3.5)$$

dove  $f$  dipende in maniera continua dallo spazio, dalla velocità e dal tempo, che sono a loro volta variabili continue:  $f(\mathbf{x}, \mathbf{v}, t)$ .

Tuttavia il modello deve essere implementato su di un calcolatore, che necessita l'uso di variabili discrete (per la memorizzazione delle informazioni). A tale scopo, quindi, introduciamo l'insieme delle velocità  $\mathbf{v}_i$ , che ci permettono di discretizzare anche lo spazio, in quanto vale la relazione  $\mathbf{v}_i\Delta t = \mathbf{c}_i$  sulle direzioni della griglia, alle quali è associata la funzione  $f_i(\mathbf{x}, t)$ , che è regolata

---

<sup>2</sup>Nella (3.2) è stato trascurato il termine che fa riferimento alla forza esterna che sarà, quindi, considerata in seguito.

da un'equazione di *Boltzmann* discreta, analoga alla (3.5).

Se ora vogliamo ottenere un'espressione che sia adimensionale, in modo che risulti applicabile agli scopi più diversi, riscaliamo tutte le quantità introducendo la *lunghezza caratteristica*  $L$ , la *velocità caratteristica*  $U$ , la *densità di riferimento*  $\rho_r$  ed il tempo fra le collisioni  $\Delta t$ , ottenendo:

$$\frac{\partial F_i}{\partial \hat{t}} + \mathbf{c}_i \hat{\nabla} F_i = -\frac{1}{\hat{\tau}} \left( F_i - F_i^{(eq)} \right) \quad (3.6)$$

dove  $\mathbf{c}_i = \mathbf{v}_i / U$ ,  $\hat{\nabla} = L \nabla$ ,  $\hat{t} = t \cdot U / L$ ,  $\hat{\tau} = \tau / \Delta t$  ed  $F_i = f_i / \rho_r$ . Indichiamo con  $\epsilon$  il parametro  $\Delta t (U / L)$ .

L'equazione (3.6) può essere discretizzata nello spazio e nel tempo approssimando:

$$\begin{aligned} \frac{\partial F_i}{\partial \hat{t}} &\cong \frac{F_i(\hat{\mathbf{x}}, \hat{t} + \Delta \hat{t}) - F_i(\hat{\mathbf{x}}, \hat{t})}{\Delta \hat{t}} \\ \frac{\partial F_i}{\partial \hat{\mathbf{x}}} &\cong \frac{F_i(\hat{\mathbf{x}} + \Delta \hat{\mathbf{x}}, \hat{t} + \Delta \hat{t}) - F_i(\hat{\mathbf{x}}, \hat{t} + \Delta \hat{t})}{\Delta \hat{\mathbf{x}}} \end{aligned}$$

e ponendo

$$\mathbf{c}_i = (c_{ix}, c_{iy}, c_{iz})$$

con  $\Delta \hat{t} = \Delta t (U / L)$  e  $\Delta \hat{\mathbf{x}} / \Delta \hat{t} = \mathbf{c}_i$ .

Attraverso tale procedimento, con le opportune semplificazioni, otteniamo il *comportamento Lagrangiano* del sistema, che può essere descritto in modo semplice e compatto attraverso la seguente relazione:

$$\boxed{F_i(\mathbf{x} + \mathbf{c}_i \Delta t, t + \Delta t) - F_i(\mathbf{x}, t) = -\frac{1}{\tau} \left( F_i - F_i^{(eq)} \right)} \quad (3.7)$$

Quindi l'*integrale delle collisioni* può essere re-interpretato come:

$$\Omega_i = -\frac{1}{\tau} \left( F_i - F_i^{(eq)} \right) \quad (3.8)$$

Una *equazione di Boltzmann*, con tale operatore per le collisioni, viene appunto chiamata *BGK*, proprio per la sua somiglianza con la (2.30), dove  $\omega = 1/\tau$ . Il raffronto, però, non è fatto solo a livello di equazioni, ma anche la forma della distribuzione di equilibrio viene comparata e calcolata di conseguenza. Nella trattazione classica, questa è la *distribuzione di Maxwell* (come definita nella (2.29)), che viene usata per aver applicato il *Principio di Esclusione di Pauli* (per semplificare la trattazione). Questa costrizione porta ad avere specificato lo stato di ogni cella attraverso variabili booleane e limita il numero di collisioni possibili in ogni nodo.

Tali imposizioni non hanno alcuna giustificazione fisica!

Ecco perché nel caso della *BGK* si preferisce usare la *distribuzione di Fermi-Dirac*.

La (3.7) viene infatti usata proprio per ottenere l'evoluzione del sistema, e non solo nell'ambito della *multi-scale analysis*.

L'approccio attraverso l'equazione *BGK* (cioè l'uso della (3.7) per esprimere l'evoluzione del sistema) e la scelta di una distribuzione di equilibrio che soddisfi le corrette equazioni macroscopiche, ci portano alla formulazione di un metodo, *LBM* per l'appunto, che raggiunge a livello macroscopico l'*equazione di continuità* e l'esatta<sup>3</sup> *equazione di Navier-Stokes*. Questo sarà mostrato per esteso nel prossimo paragrafo.

## 3.2 LBM bi-dimensionali

Un *LBM* bi-dimensionale è caratterizzato da tre principali ingredienti:

- la griglia;

---

<sup>3</sup>Non l'*equazione di Eulero*, come quanto avveniva nel caso degli *LGCA*.

- la distribuzione di equilibrio;
- l'equazione cinetica che permette al sistema di evolversi (della quale abbiamo già parlato nel precedente paragrafo (3.7)).

Al contrario dei metodi *LCGA* che sono caratterizzati, oltre che dalla griglia, dalle sole collisioni.

### 3.2.1 Griglia bi-dimensionale

Il metodo *LBM* è stato definito ed implementato sin dall'inizio da *Koelman* (1991), su di una griglia con le seguenti velocità caratteristiche<sup>4</sup>  $\mathbf{c}_i$ :

$$\begin{aligned}\mathbf{c}_0^K &= (0, 0) \\ \mathbf{c}_{1,2,3,4}^K &= (\pm a, \pm b) \\ \mathbf{c}_{5,6}^K &= (\pm 2a, 0) \\ \mathbf{c}_{7,8}^K &= (0, \pm 2b)\end{aligned}$$

dove  $a$  e  $b$  debbono soddisfare la limitazione  $a^3/3 \leq b^2 \leq 3a^2$ .

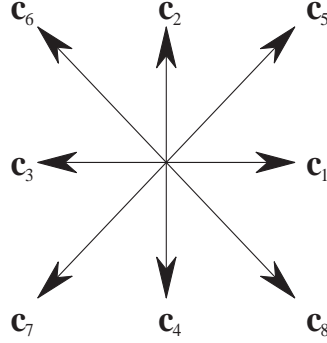
Noi facciamo la particolare scelta  $a = b = 1/\sqrt{2}c$ , che corrisponde ad una griglia con 'link' ruotati di  $45^\circ$  (come mostrato nella Figura 3.2.1), quindi con:

$$\begin{aligned}\mathbf{c}_0 &= (0, 0) \\ \mathbf{c}_{1,3}, \mathbf{c}_{2,4} &= (\pm c, 0), (0, \pm c) \\ \mathbf{c}_{5,6,7,8} &= (\pm c, \pm c)\end{aligned}$$

ed, in genere, si sceglie  $|\mathbf{c}| = 1$ .

---

<sup>4</sup>Anche in questo caso, come per il modello *FHP*, nel momento in cui parliamo degli  $\mathbf{c}_i$  li definiamo come velocità poiché si considera  $\Delta t$  come unitario.

Figura 3.1: Griglia bi-dimensionale del modello *BGK*.

### 3.2.2 La distribuzione di equilibrio

Lo stato di equilibrio, verso il quale i sistemi evolvono spontaneamente, e che quindi possiede caratteristiche di stabilità (cfr. [15]), è caratterizzato da un valore massimo per l'entropia.

Questa è l'osservazione centrale che ha ispirato il *Principio di Massima Entropia* ed è proprio questo il motivo per cui la *distribuzione di equilibrio* viene calcolata a partire da tale principio.

L'*entropia* di un sistema si definisce come:

$$S(\rho, \mathbf{j}) := -\frac{k}{m} \sum_i F_i^{(0)}(\rho, \mathbf{j}) \ln \frac{F_i^{(0)}(\rho, \mathbf{j})}{W_i}. \quad (3.9)$$

dove, facendo comunque riferimento al nostro modello, i valori di  $\rho$  e  $\mathbf{j}$  sono espressi da:

$$\rho = \sum_i F_i(\mathbf{x}, t) = \sum_i F_i^{(0)}(\rho, \mathbf{j}) \quad (3.10)$$

$$\mathbf{j} = \sum_i \mathbf{c}_i F_i(\mathbf{x}, t) = \sum_i \mathbf{c}_i F_i^{(0)}(\rho, \mathbf{j}). \quad (3.11)$$

Rendere massima l'entropia, significa calcolare i valori di  $F_i^{(0)}$ , e di conseguenza quelli di  $\rho$  e  $\mathbf{j}$ <sup>5</sup>, per cui la derivata prima della (3.9) si annulla. Tali valori sono quelli per cui  $F_i^{(0)}$  risulta assumere la seguente forma:

$$F_i^{(0)} = W_i \exp(A_0) \left\{ 1 + B_1 \mathbf{c}_i \cdot \mathbf{j} + \frac{B_1^2}{2} (\mathbf{c}_i \cdot \mathbf{j})^2 + A_2 \mathbf{j}^2 \right\}. \quad (3.12)$$

dove i valori dei termini presenti si calcolano imponendo la validità delle relazioni (3.10) e (3.11) per la  $F_i^{(0)}$  :

$$B_1 = \frac{1}{\exp(A_0)} \frac{m}{\rho_0 k_B T} \quad (3.13)$$

$$A_2 = -\frac{B_1^2 k_B T}{2} \frac{1}{m} + \frac{1}{\mathbf{j}^2} \underbrace{\left( \frac{\rho}{\rho_0 \exp(A_0)} - 1 \right)}_{(*)}$$

Affinchè  $A_0$  sia indipendente da  $\mathbf{j}$  si deve imporre  $(*) = 0$ , che ci permette di ottenere la terza condizione necessaria per il calcolo esplicito della distribuzione di equilibrio.

Segue, così, immediatamente che i parametri  $A_0$ ,  $B_1$  e  $A_2$  debbono soddisfare le relazioni:

$$\exp(A_0) = \frac{\rho}{\rho_0} \quad B_1 = \frac{m}{\rho k_B T} \quad A_2 = -\frac{1}{2\rho^2} \frac{m}{\rho k_B T}$$

Per cui, sostituendo tali valori nella (3.12), otteniamo la definizione esplicita della *distribuzione di equilibrio*, dipendente dai parametri  $\rho$  e  $\mathbf{j}$ , caratterizzanti il sistema:

$$F_i^{(0)} = \frac{W_i}{\rho_0} \left\{ \rho + \frac{m}{\rho k_B T} \mathbf{c}_i \cdot \mathbf{j} + \frac{m}{2\rho k_B T} \left[ \frac{m}{2\rho k_B T} (\mathbf{c}_i \cdot \mathbf{j}) - \mathbf{j}^2 \right] \right\}. \quad (3.14)$$

<sup>5</sup>L'entropia  $S(\rho, \mathbf{j})$  dipende dai parametri  $\rho$  e  $\mathbf{j}$  solo attraverso  $F_i^{(0)}$ .

### 3.2.3 Calcolo dei $W_i$

Se andiamo a considerare la nostra funzione di partenza  $F_i$ , i  $W_i$  rappresentano una piccola perturbazione da essa (facendo anche riferimento alla (3.14)), quindi:

$$F_i(\mathbf{x}, t) \simeq W_i(\mathbf{x}, t) \quad (3.15)$$

Per la (3.15) (facendo anche riferimento alle (2.13) e (2.14)) i  $W_i$ , che debbono assumere valori positivi per assicurare che la massa ed il momento siano a loro volta non negative, soddisfano le relazioni<sup>6</sup>:

$$\begin{aligned} \sum_i W_i c_{i\alpha} &= 0 \\ \sum_i W_i c_{i\alpha} c_{i\beta} c_{i\gamma} &= 0 \end{aligned}$$

valide per questi momenti dispari, e:

$$\sum_i W_i = \rho_0 \quad (3.16)$$

$$\sum_i W_i c_{i\alpha} c_{i\beta} = \rho_0 \frac{k_B T}{m} \delta_{\alpha\beta} \quad (3.17)$$

$$\sum_i W_i c_{i\alpha} c_{i\beta} c_{i\gamma} c_{i\delta} = \rho_0 \left( \frac{k_B T}{m} \right)^2 (\delta_{\alpha\beta} \delta_{\gamma\delta} + \delta_{\alpha\gamma} \delta_{\beta\delta} + \delta_{\alpha\delta} \delta_{\beta\gamma}) \quad (3.18)$$

valide, invece, per questi momenti pari.

#### Nota 3.1.

*Osserviamo che le costrizioni (3.16)-(3.18) imposte ai  $W_i$  sono molto più forti rispetto a quelle che si avrebbero considerando la sola condizione di isotropia spaziale per la griglia.*

---

<sup>6</sup>Le lettere greche utilizzate come pedice, stanno ad indicare le diverse componenti cartesiane del vettore a cui fanno riferimento.

Soluzioni ammissibili per il sistema costituito dalle (3.16)-(3.18) possono essere calcolate se il numero di collegamenti presenti nel nodo è abbastanza alto da garantire una sufficiente simmetria. Inoltre, per ogni tipo di griglia a cui ci si riferisce, si ottengono diversi valori per i  $W_i$ . Questo discende dalle relazioni che valgono per i vettori  $\mathbf{c}_i$ .

Se si considera la griglia nella Figura (3.2.1), e quindi quella per cui le particelle possono assumere nove possibili configurazioni all'interno del nodo, i valori che si ottengono sono:

$$\begin{aligned} W_0 / \rho_0 &= \frac{4}{9} \\ W_1 / \rho_0 &= \frac{1}{9} \\ W_2 / \rho_0 &= \frac{1}{36} \\ \frac{k_B T}{m} &= c_s^2 \end{aligned}$$

dove  $c_s^2 = c^2 / 3$ .

$\mathbf{c}_i^2$	link	particelle	$W_i$
0	0	1	$W_0$
1	1, 2, 3, 4	4	$W_1$
2	5, 6, 7, 8	4	$W_2$

### 3.2.4 Evoluzione del sistema

Possiamo, a questo punto, dare una regola effettiva che stabilisce l'evoluzione nel modello  $LB$  e, quindi, ci permette di poter effettuare l'applicazione dello stesso. Questa avviene attraverso due ben distinte fasi: *propagazione*, della distribuzione delle particelle verso i nodi adiacenti, e *collisione*, attraverso



l'uso della distribuzione di equilibrio.

La seconda fase può essere espressa tramite la *BGK* come segue:

$$\boxed{F_i(\mathbf{x} + \mathbf{c}_i \Delta t, t + \Delta t) = (1 - \omega) F_i(\mathbf{x}, t) + \omega F_i^{(0)}(\mathbf{x}, t) + \frac{\Delta t c_{i\alpha}}{12c^2} [K_\alpha(\mathbf{x}, t) + K_\alpha(\mathbf{x} + \mathbf{c}_i \Delta t, t + \Delta t)]} \quad (3.19)$$

dove  $\tau$  è il tempo di collisione, legato alla frequenza di collisione, dalla relazione  $\omega = \Delta t / \tau$ , mentre  $K_\alpha$  sono le componenti cartesiane della forza applicata, che indichiamo con  $\mathbf{K}$ .

Quindi l'equazione da noi usata per l'evoluzione del sistema è:

$$F_i(\mathbf{x} + \mathbf{c}_i \Delta t, t + \Delta t) = F_i(\mathbf{x}, t) - \omega \left( F_i(\mathbf{x}, t) - F_i^{(0)}(\mathbf{x}, t) \right) + \frac{\Delta t c_{i\alpha}}{12c^2} [K_\alpha(\mathbf{x}, t) + K_\alpha(\mathbf{x} + \mathbf{c}_i \Delta t, t + \Delta t)] \quad (3.20)$$

La distribuzione di equilibrio, dipende, come già accennato, solo dalla densità locale di massa e del momento:

$$F_i^{(0)}(\mathbf{x}, t) = F_i^{(0)}(\rho(\mathbf{x}, t), \mathbf{j}(\mathbf{x}, t)).$$

Questa si calcola, come visto nella (3.14), tramite la:

$$F_i^{(0)} = \frac{W_i}{\rho_0} \left\{ \rho + \frac{m}{\rho k_B T} \mathbf{c}_i \cdot \mathbf{j} + \frac{m}{2\rho k_B T} \left[ \frac{m}{2\rho k_B T} (\mathbf{c}_i \cdot \mathbf{j}) - \mathbf{j}^2 \right] \right\}. \quad (3.21)$$

Usando per i  $W_i$  i valori visti per la griglia con otto direzioni per le velocità (che è anche quella a cui ci siamo riferiti nelle implementazioni che abbiamo realizzato) nella (3.19).

Più esplicitamente si ha:

$$\begin{aligned}
 F_i^{(0)} &= \frac{4}{9} \rho \left[ 1 - \frac{3}{2} \frac{\mathbf{u}^2}{c^2} \right] & i &= 0 \\
 F_i^{(0)} &= \frac{1}{9} \rho \left[ 1 + 3 \frac{\mathbf{c}_i \cdot \mathbf{u}}{c^2} + \frac{9}{2} \frac{(\mathbf{c}_i \cdot \mathbf{u})^2}{c^4} - \frac{3}{2} \frac{\mathbf{u}^2}{c^2} \right] & i &= 1, 2, 3, 4 \\
 F_i^{(0)} &= \frac{1}{36} \rho \left[ 1 + 3 \frac{\mathbf{c}_i \cdot \mathbf{u}}{c^2} + \frac{9}{2} \frac{(\mathbf{c}_i \cdot \mathbf{u})^2}{c^4} - \frac{3}{2} \frac{\mathbf{u}^2}{c^2} \right] & i &= 5, 6, 7, 8
 \end{aligned}$$

Applicando la *multi-scale analysis* (che vedremo in seguito) si ottiene per la pressione il valore  $p = \rho k_B T / m$ , mentre per la viscosità cinematica si ha:

$$\nu = \frac{k_B T}{m} \left( \frac{1}{\omega} - \frac{1}{2} \right) \Delta t = \frac{c^2}{3} \left( \frac{1}{\omega} - \frac{1}{2} \right) \Delta t = \frac{c^2}{3} \left( \tau - \frac{\Delta t}{2} \right) \quad (3.22)$$

che risulta dipendere esplicitamente da  $\omega \Rightarrow \nu = \nu(\omega)$ .

### Nota 3.2.

*Anche in questo caso è possibile il calcolo esplicito del numero di Reynolds attraverso una formula analoga a quella vista nel caso del metodo FHP:*

$$\mathcal{R}_e = \frac{U L}{\nu(\omega)} \quad (3.23)$$

*dove  $U$  ed  $L$  sono le velocità caratteristiche del modello considerato, e  $\nu(\omega)$  è la viscosità cinematica definita nella (3.22).*

*Esplicitamente:*

$$\mathcal{R}_e = \frac{U L}{\Delta t} \frac{3}{c^2} \left( \frac{1}{\omega} - \frac{1}{2} \right)^{-1} \quad (3.24)$$

### 3.3 Fasi per l'implementazione

L'algoritmo che discende dai ragionamenti teorici fatti, e che risulta essere valido per ogni variante del modello *LB*, può essere riassunto tramite i seguenti "time-step":

**Step 1.**

Inizializzazione della densità di massa  $\rho(\mathbf{x}, t)$  e del momento  $\mathbf{j}(\mathbf{x}, t)$  in ogni nodo della griglia scelta;

**Step 2.**

Per questi valori di  $\rho(\mathbf{x}, t)$  e  $\mathbf{j}(\mathbf{x}, t)$ , calcolati al tempo  $t$ , si calcola la distribuzione di equilibrio scelta  $F_i^{(0)}(\rho(\mathbf{x}, t), \mathbf{j}(\mathbf{x}, t))$  usando la relazione (3.21) con gli opportuni valori di  $W_i$ , a seconda della griglia scelta;

**Step 3.**

A questo punto si applica il metodo vero e proprio, calcolando il valore che la distribuzione  $F_i$  assume al passo successivo, usando la relazione (3.19), che ci permette di ottenere l'evoluzione del sistema nel tempo;

**Step 4.**

Si effettua, a questo punto, la propagazione delle particelle presenti in ogni vettore velocità del nodo (tranne che per le particelle che si trovano ferme nel centro), verso quello adiacente;

**Step 5.**

Si calcolano, a questo punto, i nuovi valori per le distribuzioni della densità di massa  $\rho(\mathbf{x}, t)$  e del momento  $\mathbf{j}(\mathbf{x}, t)$ , attraverso le (3.10) e (3.11);

**Step 6.**

Il passo successivo, che è anche l'ultimo, inizia con il calcolo delle nuove distribuzioni di equilibrio. Poi si procede ripartendo dallo (**Step 2.**).

### 3.4 Derivazione delle equazioni macroscopiche:

#### *multi - scale analysis.*

La derivazione delle equazioni macroscopiche nel caso del modello *LB*, procede in maniera analoga al caso degli *LGCA*. La maggiore differenza che si incontra è che, in questo caso, il calcolo dei termini del secondo ordine, nell'espansione rispetto ad  $\epsilon$ , risulta essere molto più semplice ed immediato. Il punto di partenza, comunque, è sempre l'*espansione di Chapman-Enskog*.

#### 3.4.1 Equazioni macroscopiche

Le equazioni macroscopiche che si vogliono considerare, sono quelle che regolano il moto dei fluidi, ovvero l'*equazione di continuità* e l'*equazione di Navier-Stokes*.

Per poter ottenere, in modo completo, tale risultato, dobbiamo considerare, nello sviluppo di *Chapman-Enskog* i coefficienti non solo dei termini del primo ordine in  $\epsilon$ , ma bisogna estendere i calcoli fino ai termini in  $\epsilon^2$ .

Il calcolo dei termini del secondo ordine, che sarebbe risultato molto complicato negli *LGCA*, risulta quasi immediato nel presente caso ed, inoltre, ci permette di ottenere l'intera equazione regolante l'idrodinamica, non solo la parte non contenente i termini dissipativi, cioè l'equazione di *Eulero*.

Andiamo, quindi, a considerare l'espansione della nostra funzione  $F_i(\mathbf{x}, t)$  intorno alla distribuzione di equilibrio:

$$F_i(\mathbf{x}, t) = F_i^{(0)}(\mathbf{x}, t) + \epsilon F_i^{(1)}(\mathbf{x}, t) + \epsilon^2 F_i^{(2)}(\mathbf{x}, t) + \mathcal{O}(\epsilon^3) \quad (3.25)$$

con

$$\sum_i F_i^{(1)}(\mathbf{x}, t) = 0; \quad \sum_i \mathbf{c}_i F_i^{(1)}(\mathbf{x}, t) = 0$$

$$\sum_i F_i^{(2)}(\mathbf{x}, t) = 0; \quad \sum_i \mathbf{c}_i F_i^{(2)}(\mathbf{x}, t) = 0$$

ovvero, i termini in  $\epsilon$  ed in  $\epsilon^2$  non danno alcun contributo nel calcolo della densità di massa e del momento (quindi non interverranno nella derivazione, in quanto questa discende dalle imposizioni sulla conservazione di tali quantità). Consideriamo, ora, lo sviluppo in serie del primo membro della (3.20), troncato al secondo ordine, ponendo  $\Delta \mathbf{x}_i = \Delta t \cdot \mathbf{c}_i$ :

$$\begin{aligned} F_i(\mathbf{x} + \Delta t \cdot \mathbf{c}_i, t + \Delta t) &= F_i(\mathbf{x}, t) + \Delta t \partial F_i + \Delta t c_{i\alpha} \partial_{x_\alpha} F_i \\ &+ \frac{(\Delta t)^2}{2} [\partial_t \partial_t F_i + 2c_{i\alpha} \partial_t \partial_{x_\alpha} F_i + c_{i\alpha} \partial_{x_\alpha} c_{i\beta} \partial_{x_\beta} F_i] + \mathcal{O}(\partial^3 F_i) \end{aligned} \quad (3.26)$$

L'espansione del termine comprendente la forza  $\mathbf{K}$  è:

$$\begin{aligned} \frac{\Delta t}{c^2} \frac{c_{i\alpha}}{12} & [K_\alpha(\mathbf{x}, t) + K_\alpha(\mathbf{x} + \Delta t \cdot \mathbf{c}_i, t + \Delta t)] \\ &= \frac{\Delta t}{c^2} \frac{c_{i\alpha}}{6} K_\gamma(\mathbf{x}, t) + \frac{(\Delta t)^2}{c^2} \frac{c_{i\alpha}}{12} [\partial_t K_\gamma + c_{i\beta} \partial_{x_\beta} K_\gamma] \\ &+ \mathcal{O}((\Delta t)^3). \end{aligned}$$

Come nel caso già incontrato per l'*FHP*, introduciamo due scale di derivazione temporale, ed una sola spaziale:

$$\begin{aligned} \partial_t &\longrightarrow \epsilon \partial_t^{(1)} + \epsilon^2 \partial_t^{(2)} \\ \partial_{x_\alpha} &\longrightarrow \epsilon \partial_{x_\alpha}^{(1)}. \end{aligned}$$

Ora, attraverso diversi sviluppi, che considereremo man mano, si arriva alle equazioni macroscopiche, considerando tutti i termini e tutte le quantità a nostra disposizione.

### Conservazione della massa e del momento.

Sostituendo l'espansione sopra considerata nella (3.20):

$$0 = F_i(\mathbf{x} + \mathbf{c}_i \Delta t, t + \Delta t) - F_i(\mathbf{x}, t) + \omega \left[ F_i^{(0)}(\mathbf{x}, t) - F_i(\mathbf{x}, t) \right] - \frac{\Delta t}{c^2} \frac{c_{i\alpha}}{12} [K_\alpha(\mathbf{x}, t) + K_\alpha(\mathbf{x} + \Delta t \cdot \mathbf{c}_i, t + \Delta t)]$$

si ottiene:

$$\begin{aligned} 0 &= F_i(\mathbf{x}, t) + \Delta t \partial_t F_i + \Delta t c_{i\gamma} \partial_{x_\gamma} F_i \\ &+ \frac{(\Delta t)^2}{2} \left[ \partial_t \partial_t F_i + 2 c_{i\gamma} \partial_t \partial_{x_\gamma} F_i + c_{i\beta} c_{i\gamma} \partial_{x_\beta} \partial_{x_\gamma} F_i \right] \\ &+ \mathcal{O}(\partial^3 F_i) - F_i(\mathbf{x}, t) + \omega \left[ F_i(\mathbf{x}, t) - F_i^{(0)}(\mathbf{x}, t) \right] \\ &- \frac{\Delta t}{c^2} \frac{c_{i\gamma}}{6} K_\gamma(\mathbf{x}, t) - \frac{(\Delta t)^2}{c^2} \frac{c_{i\gamma}}{6} \left[ \partial_t K_\gamma + c_{i\beta} \partial_{x_\beta} K_\gamma \right] \\ &+ \mathcal{O}((\Delta t)^3) \\ &= +\epsilon \Delta t \left[ \partial_t^{(1)} F_i^{(0)} + c_{i\gamma} \partial_{x_\gamma}^{(1)} F_i^{(0)} \right] \\ &+ \epsilon^2 \Delta t \left[ \partial_t^{(1)} F_i^{(1)} + \partial_t^{(2)} F_i^{(0)} + c_{i\gamma} \partial_{x_\gamma}^{(1)} F_i^{(1)} \right] \\ &+ \epsilon^2 \frac{(\Delta t)^2}{2} \left[ \partial_t^{(1)} \partial_t^{(1)} F_i^{(0)} + 2 c_{i\gamma} \partial_t^{(1)} \partial_{x_\gamma}^{(1)} F_i^{(0)} \right. \\ &\left. + c_{i\beta} c_{i\gamma} \partial_{x_\beta}^{(1)} \partial_{x_\gamma}^{(1)} F_i^{(0)} \right] + \epsilon \omega F_i^{(1)} \\ &+ \epsilon^2 \omega F_i^{(2)} - \epsilon \frac{\Delta t}{c^2} \frac{c_{i\gamma}}{6} K_\gamma \\ &- \epsilon^2 \frac{(\Delta t)^2}{c^2} \frac{c_{i\gamma}}{12} \left[ \partial_t^{(1)} K_\gamma + c_{i\beta} \partial_{x_\beta}^{(1)} K_\gamma \right] \\ &+ \mathcal{O}(\epsilon^3) \end{aligned}$$

In maniera più sintetica:

$$0 = \epsilon E_i^{(0)} + \epsilon^2 E_i^{(1)} + \mathcal{O}(\epsilon^3)$$

con

$$E_i^{(0)} = \partial_t^{(1)} F_i^{(0)} + c_{i\gamma} \partial_{x_\gamma}^{(1)} F_i^{(0)} + \frac{\omega}{\Delta t} F_i^{(1)} - \frac{c_{i\gamma}}{6c^2} K_\gamma \quad (3.27)$$

$$\begin{aligned} E_i^{(1)} &= \partial_t^{(1)} F_i^{(1)} + \partial_t^{(2)} F_i^{(0)} + c_{i\gamma} \partial_{x_\gamma}^{(1)} F_i^{(1)} + \frac{\Delta t}{2} \partial_t^{(1)} \partial_t^{(1)} F_i^{(0)} \\ &= + \Delta t c_{i\gamma} \partial_t^{(1)} \partial_{x_\gamma}^{(1)} F_i^{(0)} + \frac{\Delta t}{2} c_{i\beta} c_{i\gamma} \partial_{x_\beta}^{(1)} \partial_{x_\gamma}^{(1)} F_i^{(0)} \\ &= + \frac{\omega}{\Delta t} F_i^{(2)} - \Delta t \frac{c_{i\gamma}}{12c^2} \partial_t^{(1)} K_\gamma - \Delta t \frac{c_{i\gamma}}{12c^2} c_{i\beta} \partial_{x_\beta}^{(1)} K_\gamma \end{aligned} \quad (3.28)$$

#### Termini del primo ordine in $\epsilon$ .

Se consideriamo i termini del primo ordine, cioè quelli  $E_i^{(0)}$ , imponendo la conservazione della massa e del momento, e calcolando:

$$\sum_i E_i^{(0)} \quad \sum_i c_{i\alpha} E_i^{(0)}$$

otteniamo<sup>7</sup>, come quanto sopra accennato, gli stessi risultati ottenuti nel caso degli *LGCA*, quindi le seguenti due equazioni:

$$\boxed{\partial_t \rho + \nabla \cdot \mathbf{j} = 0} \quad (3.29)$$

$$\boxed{\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} = -\frac{1}{\rho} \nabla p + \mathbf{K}} \quad (3.30)$$

In questo caso, tuttavia, non è così immediato il risultato finale, in quanto non possiamo affermare direttamente che vale la relazione  $\partial_t \rho = 0$  sulla densità,

<sup>7</sup>Senza ripetere tutti i calcoli in maniera esplicita, in quanto sono analoghi a quelli già visti nella *multi-scale analysis* per il metodo *FHP*.

poiché  $\rho$  dipende, nel caso del particolare modello considerato, dal tempo  $t$ . Il risultato finale, cioè l'equazione di continuità, discende dai valori che assumeranno le quantità presenti nell'equazione, che sono termini del secondo ordine nell'espansione in  $\epsilon$  (e quindi non da assunzioni preliminari).

### Calcolo di $P_{\alpha\beta}$ .

Con il calcolo esplicito del termine  $P_{\alpha\beta}$ , viene, per la prima volta, preso in considerazione il contributo che la distribuzione di equilibrio porta nella *multi-scale analysis*. È proprio questo che semplifica i passaggi nell'approssimazione al secondo ordine.

I termini che debbono essere presi in considerazione, sono quattro (proprio per come è stato definito il tensore  $P_{\alpha\beta}^{(0)}$ ).

Questi sono:

$$P_{\alpha\beta}^{(0.1)} := \frac{\rho}{\rho_0} \sum_i c_{i\alpha} c_{i\beta} W_i = \rho \underbrace{\frac{k_B T}{m}}_{= p} \delta_{\alpha\beta} = p \delta_{\alpha\beta}$$

con  $p = \rho k_B T / m = c_s^2$ .

$$P_{\alpha\beta}^{(0.2)} := \frac{m}{\rho_0 k_B T} \sum_i c_{i\alpha} c_{i\beta} (\mathbf{c}_i \cdot \mathbf{j}) W_i = 0$$



che va a zero poiché proporzionale al momento dispari (come considerato nel calcolo degli  $W_i$  nelle (3.16)-(3.18)).

$$\begin{aligned}
 P_{\alpha\beta}^{(0,3)} &:= \frac{1}{2\rho\rho_0} \left( \frac{m}{\rho_0 k_B T} \right)^2 \sum_i c_{i\alpha} c_{i\beta} (\mathbf{c}_i \cdot \mathbf{j})^2 W_i \\
 &= \frac{1}{2\rho} \mathbf{j}^2 \delta_{\alpha\beta} + \frac{1}{\rho} \begin{pmatrix} j_1^2 & j_1 j_2 \\ j_2 j_1 & j_2^2 \end{pmatrix} \\
 P_{\alpha\beta}^{(0,4)} &:= -\frac{1}{2\rho\rho_0} \frac{m}{\rho_0 k_B T} \mathbf{j}^2 \sum_i c_{i\alpha} c_{i\beta} W_i \\
 &= -\frac{1}{2\rho} \mathbf{j}^2 \delta_{\alpha\beta}
 \end{aligned}$$

Sommando, quindi, tutti i termini si ottiene la relazione finale che cercavamo e che ci esprime il tensore come:

$$P_{\alpha\beta}^{(0)} = \frac{1}{\rho} \begin{pmatrix} j_1^2 & j_1 j_2 \\ j_2 j_1 & j_2^2 \end{pmatrix} + p \delta_{\alpha\beta} \quad (3.31)$$

#### Termini del secondo ordine in $\epsilon$ .

Andiamo, ora, a considerare la parte che potremmo definire cruciale, cioè la vera e propria *multi-scale analysis* per il metodo *LBM*.

Allo scopo di ottenere le due equazioni macroscopiche, che includono anche tutti i termini dissipativi, che regolano il moto del nostro sistema, imponiamo alle quantità  $E_i^{(1)}$  di conservare, rispettivamente, la densità di massa e del momento.

**Conservazione della massa.**

Tale condizione equivale ad imporre che:

$$\begin{aligned}
0 &= \sum_i E_i^{(1)} \\
&= \sum_i \left[ \partial_t^{(1)} F_i^{(1)} + \partial_t^{(2)} F_i^{(0)} + c_{i\gamma} \partial_{x_\gamma}^{(1)} F_i^{(1)} + \frac{\Delta t}{2} \partial_t^{(1)} \partial_t^{(1)} F_i^{(0)} \right. \\
&\quad + \Delta t c_{i\gamma} \partial_t^{(1)} \partial_{x_\gamma}^{(1)} F_i^{(0)} + \frac{\Delta t}{2} c_{i\beta} c_{i\gamma} \partial_{x_\beta}^{(1)} \partial_{x_\gamma}^{(1)} F_i^{(0)} \\
&\quad \left. + \frac{\omega}{\Delta t} F_i^{(2)} - \Delta t \frac{c_{i\gamma}}{12 c^2} \partial_t^{(1)} K_\gamma - \Delta t \frac{c_{i\gamma}}{12 c^2} c_{i\beta} \partial_{x_\beta}^{(1)} K_\gamma \right]
\end{aligned}$$

Se ricordiamo la validità delle seguenti relazioni:

$$\begin{aligned}
\sum_i F_i &= \sum_i F_i^{(0)} = \rho \\
\partial_t^{(1)} \sum_i F_i^{(1)} &= 0 \\
\partial_{x_\gamma}^{(1)} \sum_i c_{i\gamma} F_i^{(1)} &= 0 \\
\frac{\omega}{\Delta t} \sum_i F_i^{(2)} &= 0 \\
\frac{\Delta t}{12 c^2} \partial_t^{(1)} K_\gamma \sum_i c_{i\gamma} &= 0
\end{aligned} \tag{3.32}$$

otteniamo:

$$\partial_t^{(2)} \sum_i F_i^{(0)} = \partial_t^{(2)} \rho$$

Il gradiente spaziale della forza  $\mathbf{K}$  assume la forma:

$$-\frac{c_{i\gamma}}{12 c^2} \partial_{x_{i\alpha}}^{(1)} K_\beta \sum_i c_{i\alpha} c_{i\beta} = \frac{\Delta t}{2} \partial_{x_\alpha}^{(1)} K_\alpha \tag{3.33}$$

Ora calcoliamo le somme dei diversi termini che compaiono nell'equazione, utilizzando anche le relazioni di Eulero ottenuta nel paragrafo precedente:

$$\begin{aligned}
 \frac{\Delta t}{2} \partial_t^{(1)} \partial_t^{(1)} F_i^{(0)} &= \frac{\Delta t}{2} \partial_t^{(1)} \partial_t^{(1)} \rho = -\frac{\Delta t}{2} \partial_t^{(1)} \partial_t^{(1)} \partial_{x_\beta}^{(1)} \rho u_\beta \\
 &= \frac{\Delta t}{2} \partial_{x_\alpha}^{(1)} \partial_{x_\beta}^{(1)} P_{\alpha\beta}^{(0)} - \frac{\Delta t}{2} \partial_{x_\alpha}^{(1)} K_\alpha \\
 \Delta t \partial_t^{(1)} \partial_{x_\alpha}^{(1)} \sum_i c_{i\alpha} F_i^{(0)} &= \Delta t \partial_{x_\alpha}^{(1)} \partial_t^{(1)} (\rho u_\alpha) \\
 &= -\Delta t \partial_{x_\alpha}^{(1)} \Delta t \partial_{x_\beta}^{(1)} P_{\alpha\beta}^{(0)} + \Delta t \partial_{x_\alpha}^{(1)} K_\alpha \\
 \frac{\Delta t}{2} \partial_{x_\alpha}^{(1)} \partial_{x_\beta}^{(1)} \sum_i c_{i\alpha} c_{i\beta} F_i^{(0)} &= \frac{\Delta t}{2} \partial_{x_\alpha}^{(1)} \partial_{x_\beta}^{(1)} P_{\alpha\beta}^{(0)}
 \end{aligned}$$

Sommando tutte queste quantità (senza farlo esplicitamente), vediamo che tutti i termini si cancellano e rimane solo la relazione:

$$\boxed{\partial_t^{(2)} \rho = 0} \tag{3.34}$$

che era ciò che ci serviva.

Sostituendo, infatti, tale relazione nella (3.29) otteniamo proprio l'*equazione di Continuità*:  $\nabla \cdot \mathbf{u} = 0$ .

#### Conservazione del momento.

Questa si ottiene imponendo che:

$$\begin{aligned}
 0 &= \sum_i E_i^{(1)} \\
 &= \sum_i \left[ c_{i\alpha} \partial_t^{(1)} F_i^{(1)} + c_{i\alpha} \partial_t^{(2)} F_i^{(0)} + c_{i\alpha} c_{i\beta} \partial_{x_\beta}^{(1)} F_i^{(1)} + \frac{\Delta t}{2} c_{i\alpha} \partial_t^{(1)} \partial_t^{(1)} F_i^{(0)} \right. \\
 &\quad + \Delta t c_{i\alpha} c_{i\gamma} \partial_t^{(1)} \partial_{x_\gamma}^{(1)} F_i^{(0)} + \frac{\Delta t}{2} c_{i\alpha} c_{i\beta} c_{i\gamma} \partial_{x_\beta}^{(1)} \partial_{x_\gamma}^{(1)} F_i^{(0)} \\
 &\quad \left. + \frac{\omega}{\Delta t} c_{i\alpha} F_i^{(2)} - \Delta t c_{i\alpha} \frac{c_{i\gamma}}{12c^2} \partial_t^{(1)} K_\gamma - c_{i\alpha} c_{i\beta} c_{i\gamma} \frac{\Delta t}{12c^2} \partial_{x_\beta}^{(1)} K_\gamma \right]
 \end{aligned}$$

Tale equazione ci permette di calcolare un'approssimazione del termine  $F_i^{(1)}$

$$F_i^{(1)}(\mathbf{x}, t) = -\frac{\Delta t}{\omega} \partial_t^{(1)} F_i^{(0)} - \frac{\Delta t}{\omega} c_{i\gamma} \partial_{x_\gamma}^{(1)} F_i^{(0)} + \frac{\Delta t c_{i\gamma}}{6 c^2 \omega} K_\gamma \quad (3.35)$$

Ora, sempre supponendo di trovarci in presenza di bassi numeri di Mach, ovvero sotto la condizione che la velocità media del fluido sia più piccola della *sound-speed*,  $|\mathbf{u}| \ll c_s$ , possiamo continuare esplicitamente i calcoli:

$$\begin{aligned} \partial_t^{(1)} \sum_i c_{i\alpha} F_i^{(1)} &= 0 \\ \partial_t^{(2)} \sum_i c_{i\alpha} F_i^{(0)} &= \partial_t^{(2)} j_\alpha \\ \\ \partial_{x_\beta}^{(1)} \sum_i c_{i\alpha} c_{i\beta} F_i^{(1)} &= \underbrace{-\frac{\Delta t}{\omega} \partial_t^{(1)} \partial_{x_\beta}^{(1)} \sum_i c_{i\alpha} c_{i\beta} F_i^{(0)}}_{(*)} \\ &\quad - \underbrace{\frac{\Delta t}{\omega} \partial_{x_\beta}^{(1)} \partial_{x_\gamma}^{(1)} \sum_i c_{i\alpha} c_{i\beta} c_{i\gamma} F_i^{(0)}}_{(**)} \\ &\quad + \frac{\Delta t}{6 c^2 \omega} \partial_{x_\beta}^{(1)} K_\gamma \underbrace{\sum_i c_{i\alpha} c_{i\beta} c_{i\gamma}}_{=0} \\ \\ \frac{\Delta t}{2} \partial_t^{(1)} \partial_t^{(1)} \sum_i c_{i\alpha} F_i^{(0)} &= \frac{\Delta t}{2} \partial_t^{(1)} \partial_t^{(1)} j_\alpha \\ &= -\frac{\Delta t}{2} \partial_t^{(1)} \partial_{x_\beta}^{(1)} P_{\alpha\beta}^{(0)} \\ &\approx -\frac{\Delta t}{2} \frac{k_B T}{m} \partial_t^{(1)} \partial_{x_\beta}^{(1)} \rho \delta_{\alpha\beta} \end{aligned}$$

Se esprimiamo il tutto in forma vettoriale si ottiene:

$$\frac{\Delta t}{2} \frac{k_B T}{m} \nabla \nabla \cdot \mathbf{j} + \frac{\Delta t}{2} \partial_t \mathbf{K} \quad (3.36)$$

Quindi i due termini (\*) e (\*\*) si riducono a:

$$\underbrace{\Delta t \partial_{x_\beta}^{(1)} \partial_t^{(1)} \sum_i c_{i\alpha} c_{i\beta} F_i^{(0)}}_{(*)} \quad \underbrace{\frac{\Delta t}{2} \partial_{x_\beta}^{(1)} \partial_{x_\gamma}^{(1)} \sum_i c_{i\alpha} c_{i\beta} c_{i\gamma} F_i^{(0)}}_{(**)}$$

ed inoltre:

$$\frac{\omega}{\Delta t} \sum_i c_{i\alpha} F_i^{(2)} = 0$$

In questo caso, se consideriamo i termini della forza che contengono la derivazione nel tempo, il loro valore si può esprimere nella, estremamente semplice, seguente forma:

$$-\frac{\Delta t}{12 c^2} \partial_t^{(1)} K_\gamma \sum_i c_{i\alpha} c_{i\beta} = -\frac{\Delta t}{12 c^2} \partial_t^{(1)} K_\alpha$$

Considerando tutte le componenti, otteniamo l'espressione:

$$-\frac{\Delta t}{2} \partial_t \mathbf{K}$$

dove si è usata la relazione:

$$\sum_i c_{i\alpha} c_{i\beta} = 6 c^2 \delta_{\alpha\beta}$$

Questo termine, quindi, non influenza i calcoli, in quanto si cancella con l'analogo nell'equazione (3.36).

Concludendo, quindi, il calcolo di tutti i termini attraverso l'uso delle (\*) e (\*\*) si ottiene la relazione vettoriale:

$$(*) \Rightarrow -\Delta t \left(1 - \frac{1}{\omega}\right) \frac{k_B T}{m} \nabla \nabla \cdot \mathbf{j}$$

$$(**) \Rightarrow -\Delta t \left(\frac{1}{2} - \frac{1}{\omega}\right) \frac{k_B T}{m} (\nabla^2 \mathbf{j} + 2 \nabla \nabla \cdot \mathbf{j})$$

Se riconsideriamo la relazione di partenza alla luce dei risultati ottenuti, finalmente si arriva a scrivere:

$$\partial_t^{(2)} \mathbf{j} = \Delta t \left( \frac{1}{\omega} - \frac{1}{2} \right) \frac{k_B T}{m} (\nabla^2 \mathbf{j} + 2 \nabla \nabla \cdot \mathbf{j}) \quad (3.37)$$

Il termine

$$\mu_S = \mu_K = \Delta t \left( \frac{1}{\omega} - \frac{1}{2} \right) \frac{k_B T}{m} \quad (3.38)$$

rappresenta, rispettivamente, la viscosità dinamica e di compressione. La (3.38) può essere usata per la scelta del parametro  $\omega$ . Tuttavia tale scelta deve essere effettuata in maniera tale che la viscosità risulti sempre positiva, ovvero:

$$\left( \frac{1}{\omega} - \frac{1}{2} \right) > 0 \quad \Rightarrow \quad 0 < \omega < 2$$

che risulta essere un'imposizione necessaria se vogliamo che il modello *LB* risulti convergere.

### Le equazioni macroscopiche.

Finalmente possiamo concludere il discorso sulla *multi-scale analysis*.

Se infatti consideriamo la somma dei termini ottenuti rispetto al primo ed al secondo ordine in  $\epsilon$ , cioè le equazioni (3.34) e (3.37), si hanno:

$$\nabla \cdot \mathbf{u} = 0 \quad (3.39)$$

$$\partial_t \mathbf{u} + (\mathbf{u} \cdot \nabla) \mathbf{u} = -\frac{1}{\rho} \nabla p + \nu \nabla^2 \mathbf{u} + \mathbf{K} \quad (3.40)$$

rispettivamente l'equazione di continuità e l'equazione di Navier-Stokes.

Il termine

$$\nu = \Delta t \left( \frac{1}{\omega} - \frac{1}{2} \right) \frac{c^2}{3} = \frac{c^2}{3} \left( \tau - \frac{\Delta t}{2} \right)$$

è la viscosità cinematica  $\nu = \nu(\omega)$  che ci è utile anche nel calcolo del *numero di Reynolds*:

$$\mathcal{R}_e = \frac{U L}{\nu(\omega)} \quad (3.41)$$

dove, come sempre,  $U$  è la velocità caratteristica del nostro sistema, e  $L$  è la lunghezza caratteristica.

### 3.5 Condizioni al bordo

Il modello di Boltzmann ha avuto un così grande sviluppo non solo per il fatto di essere un metodo alternativo e promettente nella simulazione numerica del moto dei fluidi (cfr. [14]), ma specialmente perché permette di implementare, con notevole semplicità, le condizioni al bordo con geometrie complicate. In generale, i nodi che si trovano nelle zone di bordo, vengono aggiornati in maniera diversa rispetto a quelli interni, possiamo, quindi, affermare che l'effetto del bordo rispetto al moto globale del fluido, dipende da come viene aggiornata la distribuzione in tali celle, e quindi dalla scelta della particolare distribuzione di equilibrio.

#### 3.5.1 Condizioni al bordo no-slip

Questo tipo di  $CB^8$  è quello che viene usato nel momento in cui ci troviamo di fronte a bordi solidi, per i quali è richiesta la condizione che la velocità della particella sia nulla in tali nodi.

Quando una particella arriva, ad un certo istante di tempo, ad un nodo facente parte del bordo, questa viene rimandata indietro nella stessa direzione da cui

---

<sup>8</sup>Stà per Condizioni al Bordo.

è arrivata, e la distribuzione ad essa associata rimane inalterata, in modulo. Questo tipo di *CB* viene anche definito, per tale motivo, *condizioni 'bounce-back'*.

Lo schema *bounce-back* completo, è così schematizzabile:

$$\text{stato di entrata} : F_0, F_1, F_2, F_3, F_4, F_5, F_6, F_7, F_8$$

$$\text{stato di uscita} : F_0, F_3, F_4, F_1, F_2, F_7, F_8, F_5, F_6$$

Si vede, da un lato, come questo risulta una via estremamente semplice per fissare il valore della distribuzione di equilibrio in tali punti, la cui espressione sarebbe, altrimenti, non nota. Dall'altro lato, però, è stato mostrato (Courbert et al. (1991)) che lo schema ci da un'ordine di accuratezza numerica solo del primo ordine. Per cercare di ovviare a questo problema, sono state proposte altre soluzioni:

- Un metodo con ordine di accuratezza numerica più elevato, fu studiato da Skordos nel (1993). Questo utilizza uno schema agli elementi finiti e richiede la conoscenza del gradiente della velocità nei nodi facenti parte del bordo.
- Nel (1996) Maier propose di modificare il metodo *bounce-back* introducendo una condizione sul momento (annullando la componente tangente e conservando quella normale) invece che sulle velocità;
- Zou e He nel (1997) proposero di estendere le condizioni *bounce-back* anche a quella parte della distribuzione che non era di equilibrio.

Come possiamo notare, quindi, molti sono stati i tentativi rivolti a risolvere questo problema. Tuttavia, le applicazioni pratiche hanno mostrato che questi



metodi danno risultati soddisfacenti nella simulazione idrodinamica, solo nel caso in cui ci si trova in presenza di bordi sufficientemente regolari. La loro estensione a superfici con contorni arbitrari, risulta essere notevolmente complicata.

Condizioni di ordine più alto e che sono di facile implementazione nei diversi casi, sono le:

*Condizioni al bordo di Dirichlet*

Queste sono sempre espresse in termini della velocità che il fluido ha sul bordo (cfr. [10]).

Un modo per imporre queste condizioni, per bordi fissati, viene dato su di una griglia esagonale (per semplicità), ma lo stesso approccio può essere adottato per una griglia qualsiasi.

Le condizioni da imporre sulla velocità sono che  $u_x = u_y = 0$ . Queste sono soddisfatte se la distribuzione, che stabilisce lo stato di ogni nodo, soddisfa le seguenti:

$$F_1 + 2F_2 + F_3 - F_4 - 2F_5 - F_6 = 0$$

$$F_1 - F_2 - F_3 + F_6 = 0$$

Questo risulta essere diverso rispetto alle tradizionali condizioni *bounce-back*, ed in particolare differisce da esse in quanto non si parte dall'ipotesi di avere conservate la densità di massa e del momento, ma sono calcolate attraverso degli algoritmi diversi per i nodi facenti parte del bordo.

### 3.6 Esperimenti numerici:

#### simulazione del “*Cavity-Flow*” bi-dimensionale.

Uno dei primi esperimenti da noi effettuati nell’ambito della *simulazione idrodinamica*, è rivolto ad un esempio nel bi-dimensionale.

Nel seguente paragrafo cercheremo di mostrare le diverse scelte verso le quali abbiamo optato e che riteniamo siano, quanto meno, sufficientemente giuste, anche alla luce dei risultati ottenuti.

##### *Griglia.*

Il modello è stato costruito su di una griglia bi-dimensionale come quella mostrata nella Figura (3.2.1), costituita da otto vettori velocità di riferimento per ogni particella che arriva nel nodo. Si è considerata, inoltre, la possibilità di poterne trovare una ferma nel centro della griglia stessa (“*rest-particle*”).

Questo tipo di scelte effettuate, sembrano rendere il modello il più generale possibile, almeno nell’ambito considerato.

Andiamo, ora, a considerare la diverse fasi nell’implementazione, cercando di descrivere, in modo breve ma completo, il programma, il cui listato (scritto in *Fortran-90*) è allegato nell’Appendice B.

##### *Inizializzazione.*

L’inizializzazione dei nodi della griglia, ovvero l’assegnazione dei valori di partenza alla massa che le particelle hanno in ogni ‘link’ della cella, avviene in modo uniforme in tutti i nodi che la costituiscono. Si considera, infatti, una densità iniziale  $\rho_0$ , scelta a seconda del tipo di fluido di cui si vuole simulare il moto, che deve essere conservata, secondo il criterio sempre utilizzato di *con-*

servazione locale della densità di massa. Questo si verifica scegliendo in modo opportuno i ‘pesi’ delle particelle  $W_i$  lungo i diversi collegamenti, cioè tali che valga:

$$\sum_{i=0}^9 W_i = \rho_0.$$

Tale condizione è verificata, nella griglia a nove velocità, se si scelgono

$$\begin{aligned} W_i &= 4/9 \rho_0 & i &= 0 \\ W_i &= 1/9 \rho_0 & i &= 1, \dots, 4 \\ W_i &= 1/36 \rho_0 & i &= 5, \dots, 9 \end{aligned}$$

che sono, per l’appunto, i valori della distribuzione iniziale di massa.

***Evoluzione del sistema.***

Conseguenza, abbastanza logica ed obbligata dalle scelte fatte in precedenza, è la fase centrale del modello, cioè l’evoluzione della situazione iniziale sopra impostata.

Questa avviene come è stato esplicitamente e chiaramente mostrato nelle fasi descritte nel paragrafo (3.3).

L’unica cosa, lì citata, che qui rimane da specificare, è la particolare distribuzione di equilibrio utilizzata. Facendo riferimento ai calcoli teorici fatti nell’ambito della *multi-scale analysis*, possiamo scrivere la nostra distribuzione come:

$$\begin{aligned} F_i^{(0)} &= \frac{4}{9} \rho \left[ 1 - \frac{3}{2} \frac{\mathbf{u}^2}{c^2} \right] & i &= 0 \\ F_i^{(0)} &= \frac{1}{9} \rho \left[ 1 + 3 \frac{\mathbf{c}_i \cdot \mathbf{u}}{c^2} + \frac{9}{2} \frac{(\mathbf{c}_i \cdot \mathbf{u})^2}{c^4} - \frac{3}{2} \frac{\mathbf{u}^2}{c^2} \right] & i &= 1, 2, 3, 4 \\ F_i^{(0)} &= \frac{1}{36} \rho \left[ 1 + 3 \frac{\mathbf{c}_i \cdot \mathbf{u}}{c^2} + \frac{9}{2} \frac{(\mathbf{c}_i \cdot \mathbf{u})^2}{c^4} - \frac{3}{2} \frac{\mathbf{u}^2}{c^2} \right] & i &= 5, 6, 7, 8 \end{aligned}$$

*Conclusioni pratiche.*

Il modello, sopra descritto, è stato implementato nel caso, particolarmente interessante, del *Cavity-Flow*.

Ciò significa che, abbiamo considerato un contenitore, quadrato prima e rettangolare poi, nel quale si trova un fluido con densità costante. A questo è applicata una velocità, diretta orizzontalmente rispetto al sistema di riferimento considerato, sulla faccia superiore.

Si può notare che, a partire da una certa velocità iniziale, sufficientemente elevata (questa può essere scelta nel momento in cui si lancia il programma), il sistema evolve verso situazioni caratteristiche di moti turbolenti, cioè in presenza di alti *numeri di Reynolds*.

Andiamo, ora, a considerare e descrivere nel particolare le due simulazioni, i cui risultati ottenuti sono mostrati, per ambo le scelte possibili nella geometria<sup>9</sup> del sistema, nelle due Figure (3.2) e (3.3), anche per quanto riguarda considerazioni puramente matematiche.

Nel primo caso è stata considerata la geometria quadrata ( $l_x = l_y$ ), le cui dimensioni reali sono specificate nel momento in cui si definiscono:

$$L = \Delta x \quad T = \Delta t \quad (3.42)$$

dove  $L$  è la lunghezza caratteristica della griglia, mentre  $T$  è il tempo caratteristico, ovvero l'intervallo di tempo fra ogni iterata.

Nel caso particolare la velocità applicata alla parete superiore vale:

$$U = \chi_v l_x \frac{L}{U} = \chi_v l_x \frac{\Delta x}{\Delta t} \quad (3.43)$$

---

<sup>9</sup>Le due geometrie non sono le uniche possibili, ma sono quelle per cui noi abbiamo optato. Ottenerne altre è estremamente semplice per la generalità con cui è stato sviluppato il modello.

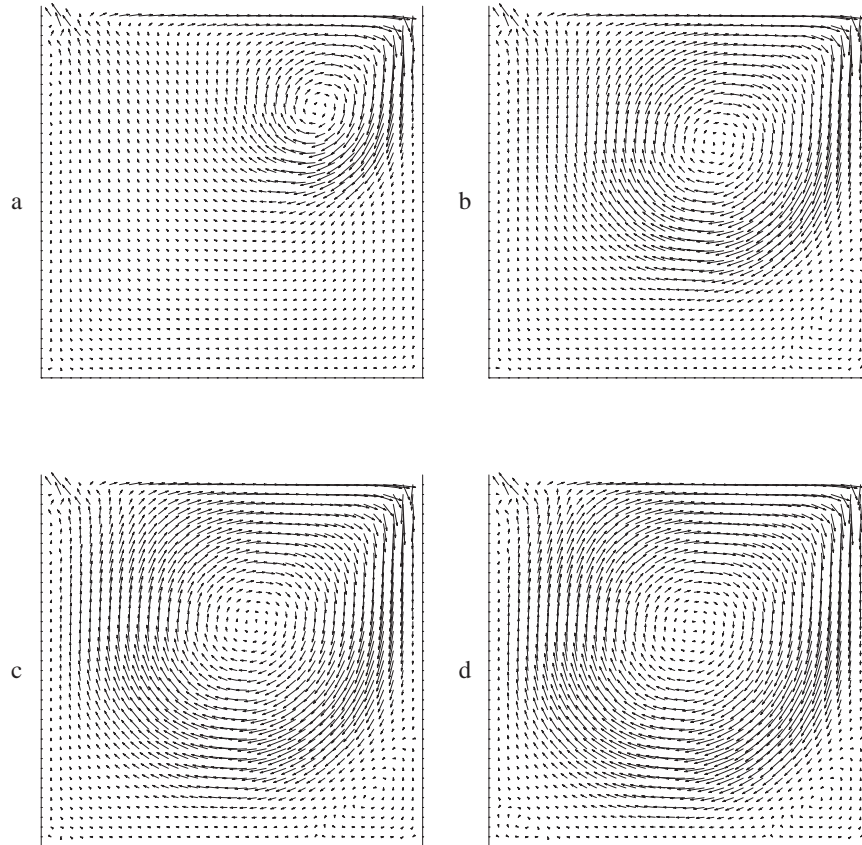


Figura 3.2: *Cavity-Flow* su griglia quadrata. a:  $t = 1000T$ ; b:  $t = 2000T$ ;  
c:  $t = 3500T$ ; d:  $t = 5000T$  iterate.

dove  $\chi_v$  è il valore numerico che dobbiamo noi assegnare dall'esterno, per poter quantificare tale velocità (valore che non deve superare la sopra citata ‘*sound-speed*’), mentre  $l_x$  è il numero di suddivisioni nella coordinata x.

La nostra scelta per tali due quantità, in ambo i casi la stessa, è stata:

$$\begin{aligned} \chi_v &= 0.005 & \left( < c_s = \frac{1}{3l_x} \right) \\ l_x &= 40. \end{aligned}$$

Queste scelte ci permettono di ottenere l'evoluzione del sistema nel tempo, risultati mostrati nella Figura (3.2).

Nella (a) si nota, infatti, l'incipiente formarsi di un vortice che sembra diventare stabile nella (d), passando attraverso le due fasi intermedie (b) e (c).

Nella seconda geometria considerata, invece, il contenitore ha le stesse dimensioni lungo la direzione  $x$  ( $l_x = 40$ ) degli assi coordinati, mentre assume valore doppio lungo la direzione  $y$  ( $l_y = 2l_x$ ).

Tutte le altre caratteristiche ed i valori delle variabili, sono stati scelti allo stesso modo, così da poter effettuare un confronto anche grafico fra le due simulazioni.

Questo raffronto è anche possibile nelle nostre figure in quanto i tempi in cui sono state stampate le velocità mostrate, sono gli stessi.

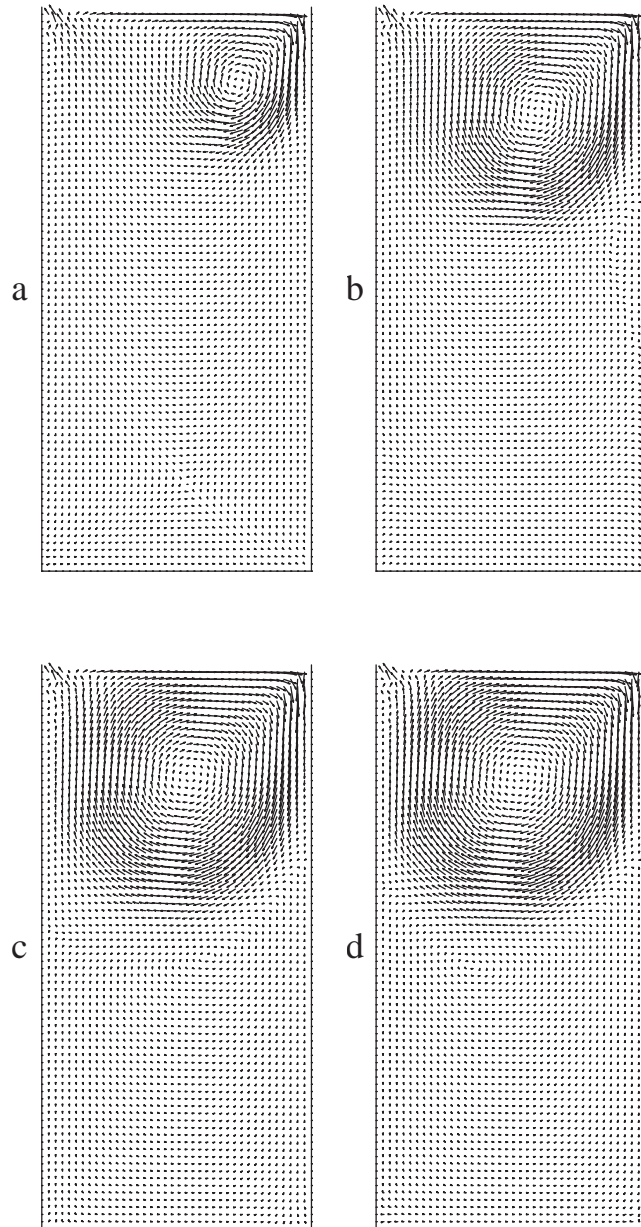


Figura 3.3: *Cavity-Flow* su griglia rettangolare. a:  $t = 1000 T$ ; b:  $t = 2000 T$ ;  
c:  $t = 3500 T$ ; d:  $t = 5000 T$  iterate.





## Capitolo 4

# *LBM* tri-dimensionali

Nel capitolo precedente abbiamo considerato e discusso i *LBM* bi-dimensionali, in particolare quelli implementati sulla griglia con otto vettori delle velocità e con un'unica particella ferma al centro del nodo. A partire da tali condizioni di base, si è cercato di completare la scelta dei termini per l'implementazione calcolando opportunamente la distribuzione di equilibrio, in modo che il metodo risultasse convergere alle equazioni macroscopiche che lo governano.

L'estensione, di tutti questi ragionamenti e calcoli, alla terza dimensione (cfr. [7]) risulta essere un passo praticamente immediato. Bisogna, infatti, solo fare un'opportuna scelta della griglia e della distribuzione di equilibrio.

Questa proprietà dei *LBM*, cioè il fatto di essere così versatili, è un'altra caratteristica che li rende così importanti e, per alcuni versi, straordinari.

## 4.1 Scelta delle griglie: isotropia spaziale

Tutti i *LGCA* sono modellati a partire da uno studio basato sullo spazio delle fasi, che risulta essere estremamente discretizzato.

Il problema si incontra, quindi, nel momento in cui dobbiamo effettuare la scelta della distribuzione e della discretizzazione nel nostro modello. Questa, infatti, non può essere mai minore rispetto a quella che si incontra nello spazio delle “configurazioni possibili”. Ci si pone di fronte, quindi, la domanda di quali siano le condizioni da imporre per poter ottenere una griglia che abbia sufficiente simmetria.

Queste vengono fuori nel momento in cui si calcolano i tensori della stessa, i quali debbono soddisfare opportune condizioni di isotropia spaziale, ovvero, debbono risultare invarianti rispetto ad alcune trasformazioni ortogonali (quali, ad esempio, le rotazioni).

Possiamo, quindi, concludere dicendo che:

“Condizione necessaria per ottenere delle equazioni macroscopiche ‘ragionevoli’, è che i *tensori della griglia* (che qui di seguito definiremo) del secondo e del quarto ordine siano isotropi”.

### 4.1.1 Tensori isotropi

**Definizione 4.1.** *Un tensore  $T_{\alpha_1, \alpha_2, \dots, \alpha_n}$  di ordine  $n$ , viene detto isotropo se risulta invariante rispetto a trasformazioni ortogonali  $O$  (rotazioni e riflessioni), ovvero:*

$$T_{\alpha_1, \alpha_2, \dots, \alpha_n} = T_{\beta_1, \beta_2, \dots, \beta_n} O_{\alpha_1, \beta_1} O_{\alpha_2, \beta_2} \dots O_{\alpha_n, \beta_n}$$

Uno dei tensori del quarto ordine, il più generale possibile, è definibile tramite il seguente teorema:

**Teorema 4.1.1.** (Jeffreys, (1965))

1. Non ci sono tensori isotropi di rango 1;
2. I tensori isotropi di rango 2 sono proporzionali a  $\delta_{\alpha\beta}$ ;
3. I tensori isotropi di rango 3 sono proporzionali ai simboli di Levi-Civita  $\epsilon_{\alpha\beta\gamma}$ <sup>1</sup>;
4. Esistono tre diversi tipi di tensori di rango 4

$$\delta_{\alpha\beta} \delta_{\gamma\delta}, \quad \delta_{\alpha\gamma} \delta_{\beta\delta}, \quad \delta_{\alpha\delta} \delta_{\beta\gamma},$$

che possono essere combinati, nella forma più generale, come:

$$T_{\alpha\beta\gamma\delta} = a \delta_{\alpha\beta} \delta_{\gamma\delta} + b \delta_{\alpha\gamma} \delta_{\beta\delta} + c \delta_{\alpha\delta} \delta_{\beta\gamma},$$

con  $a$ ,  $b$  e  $c$  costanti arbitrarie.

Possiamo, quindi, definire un generico tensore di rango  $n$ , detto anche *tensore della griglia*, in quanto ad essa associato, come:

$$L_{\alpha_1, \alpha_2, \dots, \alpha_n} = \sum_i c_{i\alpha_1} c_{i\alpha_2} \dots c_{i\alpha_n}$$

dove  $c_{i\alpha_\nu}$  sono le componenti cartesiane del vettore delle velocità della griglia  $\mathbf{c}_i$  ed  $n$  è l'ordine del tensore stesso.

A partire da questi, quindi, possiamo scegliere le griglie bi-dimensionali nei

---

<sup>1</sup>Notiamo che  $\delta_{\alpha\beta}$  è la delta di Kronecker. Inoltre i simboli di Levi-Civita  $\epsilon_{\alpha\beta\gamma}$  sono definiti come:  $\epsilon_{123} = \epsilon_{231} = \epsilon_{312} = 1$ ,  $\epsilon_{132} = \epsilon_{321} = \epsilon_{213} = -1$  e zero in tutti gli altri casi.

modelli *LGCA* fra quelle già considerate.

Nel caso dei modelli *LB*, tuttavia, possiamo incontrare dei problemi, nel senso che, il tensore del quarto ordine, in alcuni casi, può non risultare isotropo. Per ovviare a tale inconveniente e raggiungere comunque la necessaria isotropia spaziale, si possono aggiungere dei pesi  $\omega_i$ , in modo da ottenere un tensore generalizzato della forma:

$$\boxed{G_{\alpha_1, \alpha_2, \dots, \alpha_n} = \sum_i \omega_i c_{i\alpha_1} c_{i\alpha_2} \dots c_{i\alpha_n}} \quad (4.1)$$

Tali pesi  $\omega_i$  corrispondono ai *numeri di occupazione media* per le diverse direzioni della griglia, calcolati in maniera tale che la somma delle velocità macroscopiche, rispetto a tutti i collegamenti presenti, siano nulle.

Risolvendo, quindi, il sistema associato alle condizioni che debbono venire imposte nella (4.1) nei diversi casi (che dipendono dal numero di collegamenti presenti), otteniamo le diverse griglie (anche nelle diverse dimensioni):

***Griglia a 9 velocità:***

Questa è quella già utilizzata nel precedente capitolo, i cui vettori (Figura (3.2.1)) sono definiti come:

$$\begin{aligned} c_0 &= (0, 0), \\ c_{1,3}, c_{2,4} &= (\pm 1, 0), (0, \pm 1), \\ c_{5,6,7,8} &= (\pm 1, \pm 1). \end{aligned}$$

Il tensore della griglia del secondo ordine è dato da:

$$L_{\alpha\beta} = 6 \delta_{\alpha\beta}$$

in quanto

$$\sum_i c_{i\alpha} c_{i\beta} = 6 \delta_{\alpha\beta}$$

Tuttavia  $L_{\alpha\beta\gamma\delta}$ , ovvero il tensore del quarto ordine, non risulta soddisfare le condizioni di isotropia.

Siamo, quindi, obbligati ad introdurre i pesi  $\omega_i = 1$  per velocità unitarie ( $i = 1, \dots, 4$ ) e  $\omega_i = 1/4$  per velocità con valore  $\sqrt{2}$  ( $i = 5, \dots, 8$ ), ottenendo i tensori generalizzati del secondo ordine

$$G_{\alpha\beta} = 3 \delta_{\alpha\beta}$$

e del quarto ordine

$$G_{\alpha\beta\gamma\delta} = \delta_{\alpha\beta} \delta_{\gamma\delta} + \delta_{\alpha\gamma} \delta_{\beta\delta} + \delta_{\alpha\delta} \delta_{\beta\gamma},$$

che, ora soddisfano la condizione richiesta  $\Rightarrow$  la griglia a nove velocità usata, attraverso il contributo di  $G_{\alpha\beta}$  e  $G_{\alpha\beta\gamma\delta}$ , è adatta all'uso preposto.

### ***Griglia a 15 velocità:***

Il modello *LB* (che viene anche detto *modello a più velocità* o anche *modello multi-speed*<sup>2</sup>) tri-dimensionale può essere implementato su di una griglia i cui vettori possono essere definiti come:

$$\begin{aligned} c_0 &= (0, 0, 0), \\ c_{1,2}, c_{3,4}, c_{5,6} &= (\pm 2, 0, 0), (0, \pm 2, 0), (0, 0, \pm 2) \\ c_{7,\dots,14} &= (\pm 1, \pm 1, \pm 1). \end{aligned}$$

---

<sup>2</sup>Come tutti quei modelli per cui le velocità hanno un vettore di riferimento che non ha modulo unitario, o, quanto meno, uguale in tutte le direzioni prese in considerazione.

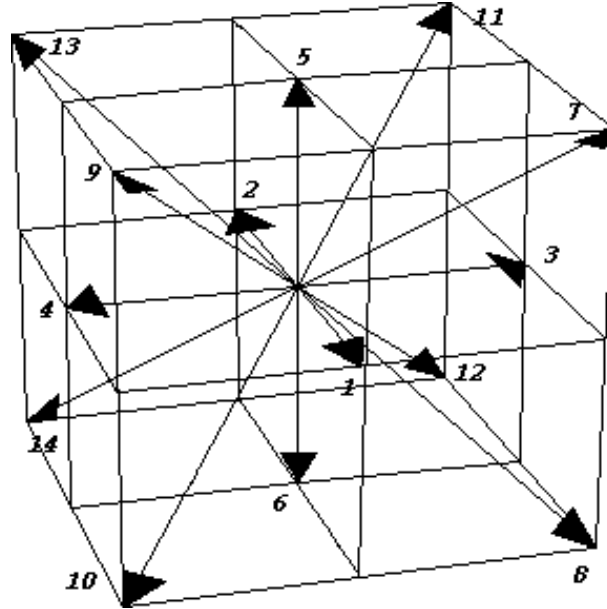


Figura 4.1: Griglia con 15 vettori velocità

come quella mostrata nella Figura (4.1). In questo caso il tensore della griglia di ordine due (che è isotropo) è:

$$L_{\alpha\beta} = 16 \delta_{\alpha\beta}$$

e quello del quarto ordine:

$$L_{\alpha\beta\gamma\delta} = 8 (\delta_{\alpha\beta} \delta_{\gamma\delta} + \delta_{\alpha\delta} \delta_{\beta\gamma} + \delta_{\alpha\gamma} \delta_{\beta\delta}) + 16 \delta_{\alpha\beta\gamma\delta}$$

che, tuttavia, non risulta essere isotropo.

Bisogna introdurre, anche in questo caso, per eliminare tale problema, i pesi  $\omega_i = 2$  per  $i = 7, \dots, 14$  e  $\omega_i = 1$  per  $i = 1, \dots, 6$ . Si ottengono in tal modo i tensori generalizzati di ordine due

$$G_{\alpha\beta} = 24 \delta_{\alpha\beta}$$

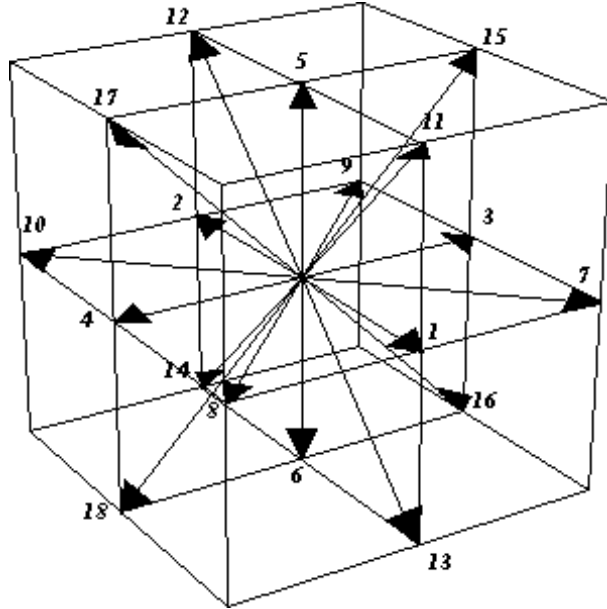


Figura 4.2: Griglia con 19 vettori velocità

e di ordine quattro

$$G_{\alpha\beta\gamma\delta} = 16 (\delta_{\alpha\beta} \delta_{\gamma\delta} + \delta_{\alpha\delta} \delta_{\beta\gamma} + \delta_{\alpha\gamma} \delta_{\beta\delta})$$

***Griglia a 19 velocità:***

Introduciamo, infine, la griglia che otteniamo considerando una diversa simmetria nella posizione dei vettori, in maniera tale da ottenere 19 possibili vettori velocità (proposta da *d'Humières, (1986)*).

Questo tipo di griglia non esaurisce, insieme alle precedenti, tutte le possibili scelte. Si vuole, tuttavia, sottolinearne l'importanza in quanto è stata da noi scelta per l'implementazione del modello *LG* tri-dimensionale. La nostra preferenza è andata in questa direzione per il semplice motivo che tale griglia risulta essere quella che mostra maggiore stabilità nell'implementazione.

Si può, oltretutto, osservare che essa rappresenta la più semplice e diretta estensione alla terza dimensione della griglia a 9 direzioni vista nel bi-dimensionale<sup>3</sup>. Nel modello a 19 velocità, quindi, i vettori presi in considerazione sono definiti come segue:

$$\begin{aligned} c_0 &= (0, 0, 0), \\ c_{1,2}, c_{3,4}, c_{5,6} &= (\pm 1, 0, 0), (0, \pm 1, 0), (0, 0, \pm 1) \\ c_{7,\dots,10}, c_{11,\dots,14}, c_{15,\dots,18} &= (\pm 1, \pm 1, 0), (\pm 1, 0, \pm 1), (0, \pm 1, \pm 1) \end{aligned}$$

La griglia associata può essere osservata nella Figura (4.2).

Il tensore della griglia di rango due

$$L_{\alpha\beta} = 10 \delta_{\alpha\beta}$$

è isotropo, mentre (come nei casi precedenti) non lo è quello di rango quattro

$$L_{\alpha\beta\gamma\delta} = 4(\delta_{\alpha\beta} \delta_{\gamma\delta} + \delta_{\alpha\delta} \delta_{\beta\gamma} + \delta_{\alpha\gamma} \delta_{\beta\delta}) - 2 \delta_{\alpha\beta\gamma\delta}$$

Anche in questo caso dobbiamo andare a considerare il tensore generalizzato da poter associare alla nostra griglia. Questo può essere definito a partire dai pesi  $\omega_i = 2$  per velocità<sup>4</sup> unitarie (cioè per  $i = 1, \dots, 6$ ) e  $\omega_i = 1$  per velocità pari a  $\sqrt{3}$  (ovvero per  $i = 7, \dots, 18$ ) che ci danno un tensore generalizzato di ordine due

$$G_{\alpha\beta} = 12 \delta_{\alpha\beta}$$

<sup>3</sup>Visto l'esito positivo dei risultati ottenuti nel bi-dimensionale, abbiamo ritenuto opportuno mantenere una certa simmetria e analogia con tale situazione

<sup>4</sup>Quando parliamo di velocità, ci riferiamo alla lunghezza del vettore della griglia. Questa rappresenta anche la direzione della velocità che la particella stessa assume lungo quel collegamento.



e del quarto ordine

$$G_{\alpha\beta\gamma\delta} = 4(\delta_{\alpha\beta}\delta_{\gamma\delta} + \delta_{\alpha\delta}\delta_{\beta\gamma} + \delta_{\alpha\gamma}\delta_{\beta\delta})$$

che ora soddisfano la condizione di isotropia spaziale.

## 4.2 Calcolo della distribuzione di equilibrio nel senso di *Koelman*

A seguire, il calcolo della  $F_i^{(eq)}$  sarà effettuato seguendo il così detto *metodo di Koelman (1991)*.

Nel caso bi-dimensionale, la distribuzione di equilibrio è stata calcolata imponendo la condizione di *Massima Entropia* del sistema, ovvero massimizzando il funzionale (3.9) e quindi calcolando i  $W_i$ , imponendo che valessero le (3.16)-(3.18) sulla griglia scelta<sup>5</sup>.

Nel caso  $3D$  si agisce in modo del tutto analogo, in particolare ora che sono già state effettuate alcune possibili scelte riguardo le grigliature più adatte alla nostra situazione.

Andiamo a definire, quindi, le distribuzioni di equilibrio che si possono considerare ed ottenere nelle diverse situazioni.

---

<sup>5</sup>La scelta della griglia bi-dimensionale (quella da noi effettuata è rappresentata nella Figura (3.2.1)) o tri-dimensionale che voglia essere, è un passo fondamentale nel calcolo degli  $W_i$ , e di conseguenza della distribuzione di equilibrio, in quanto essa stabilisce le relazioni esistenti fra le quantità  $c_i$ , e quindi fra i momenti della griglia stessa.

### 4.2.1 LBM 3D con 15 velocità

Se si considera la griglia con 15 vettori di collegamento fra i nodi (come in Figura (4.1)) si possono calcolare, rispetto ad essa, i *momenti*, che sono le uniche quantità che è necessario conoscere nel calcolo della distribuzione di equilibrio.

Come nel caso bi-dimensionale (grazie alla condizione di isotropia spaziale) quelli di ordine dispari sono tutti nulli. Mentre quelli di ordine pari si calcolano tramite le relazioni (3.16)-(3.18), imponendo le quali si ottiene un sistema le cui equazioni sono:

$$\begin{aligned} \sum_i W_i &= W_0 + 6W_2 + 8W_3 = \rho_0 \\ \sum_i c_{i\alpha}^2 W_i &= 8W_2 + 8W_3 = \rho_0 \frac{kT}{m} \\ \sum_i c_{i\alpha}^4 W_i &= 32W_2 + 8W_3 = 3\rho_0 \left(\frac{kT}{m}\right)^2 \\ \sum_i c_{i\alpha}^2 c_{i\beta}^2 W_i &= 8W_3 = 3\rho_0 \left(\frac{kT}{m}\right)^2 \end{aligned}$$

Le soluzioni di tale sistema sono:

$$\begin{aligned} W_0 &= \rho_0 \frac{7}{18} \\ W_2 &= \rho_0 \frac{1}{16} \left(\frac{kT}{m}\right)^2 = \rho_0 \frac{1}{36} \\ W_3 &= \rho_0 \frac{1}{8} \left(\frac{kT}{m}\right)^2 = \rho_0 \frac{1}{18} \\ \frac{kT}{m} &= \frac{2}{3} \end{aligned}$$

Ora è possibile anche dare le relazioni che definiscono esplicitamente le distribuzioni che volevamo:

$$\begin{aligned}
 F_i^{(eq)} &= \frac{7}{18}\rho \left[ 1 - \frac{3}{4}\mathbf{u}^2 \right] & i &= 0 \\
 F_i^{(eq)} &= \frac{1}{36}\rho \left[ 1 + \frac{3}{2}(\mathbf{c}_i \cdot \mathbf{u}) + \frac{9}{8}(\mathbf{c}_i \cdot \mathbf{u})^2 - \frac{3}{4}\mathbf{u}^2 \right] & i &= 1, \dots, 6 \\
 F_i^{(eq)} &= \frac{1}{18}\rho \left[ 1 + \frac{3}{2}(\mathbf{c}_i \cdot \mathbf{u}) + \frac{9}{8}(\mathbf{c}_i \cdot \mathbf{u})^2 - \frac{3}{4}\mathbf{u}^2 \right] & i &= 7, \dots, 14
 \end{aligned}$$

### 4.2.2 LBM 3D con 19 velocità

Se ora, invece, viene considerata la griglia con 19 velocità (cfr. Figura (4.2)), il calcolo dei  $W_i$  avviene in maniera del tutto analoga al caso precedente, cioè basta imporre le (3.16) - (3.18) ai momenti pari (risultando ancora nulli tutti i momenti dispari).

Tali costrizioni ci portano ad ottenere un sistema le cui equazioni sono:

$$\begin{aligned}
 \sum_i W_i &= W_0 + 6W_1 + 12W_2 = \rho_0 \\
 \sum_i c_{i\alpha}^2 W_i &= 2W_1 + 8W_2 = \rho_0 \frac{kT}{m} \\
 \sum_i c_{i\alpha}^4 W_i &= 2W_1 + 8W_2 = 3\rho_0 \left( \frac{kT}{m} \right)^2 \\
 \sum_i c_{i\alpha}^2 c_{i\beta}^2 W_i &= 4W_2 = \rho_0 \left( \frac{kT}{m} \right)^2
 \end{aligned}$$

La cui soluzione è data da

$$\begin{aligned}
 W_0 &= \rho_0 \frac{1}{3} \\
 W_1 &= \rho_0 \frac{1}{2} \left( \frac{kT}{m} \right)^2 = \rho_0 \frac{1}{18} \\
 W_2 &= \rho_0 \frac{1}{4} \left( \frac{kT}{m} \right)^2 = \rho_0 \frac{1}{36} \\
 \frac{kT}{m} &= \frac{1}{3}
 \end{aligned}$$

Quindi, l'esplicita espressione dell'equazione di equilibrio è:

$$\begin{aligned}
 F_i^{(eq)} &= \frac{1}{3}\rho \left[ 1 - \frac{3}{2}\mathbf{u}^2 \right] & i &= 0 \\
 F_i^{(eq)} &= \frac{1}{18}\rho \left[ 1 + 3(\mathbf{c}_i \cdot \mathbf{u}) + \frac{9}{2}(\mathbf{c}_i \cdot \mathbf{u})^2 - \frac{3}{2}\mathbf{u}^2 \right] & i &= 1, \dots, 6 \\
 F_i^{(eq)} &= \frac{1}{36}\rho \left[ 1 + 3(\mathbf{c}_i \cdot \mathbf{u}) + \frac{9}{2}(\mathbf{c}_i \cdot \mathbf{u})^2 - \frac{3}{2}\mathbf{u}^2 \right] & i &= 7, \dots, 18
 \end{aligned}$$

#### Nota 4.1.

*Abbiamo già fatto cenno al fatto che le distribuzioni di equilibrio, calcolate attraverso il principio di Massima Entropia, non esauriscono tutte le possibili alternative. È per questo motivo che qui di seguito andremo a considerare delle distribuzioni calcolate in modo diverso.*

#### 4.2.3 Chen et al., (1992)

La distribuzione di equilibrio che andiamo a considerare, e che è stata proposta da *Chen et al.*, è stata scelta per essere implementata su di una griglia del tipo a 15 velocità.

Questa è:

$$\begin{aligned}
 F_i^{(eq)} &= d^{(0)} + \delta^{(0)}\mathbf{v}^2 & i &= 0 \\
 F_i^{(eq)} &= d^{(1)} + \beta^{(1)}(\mathbf{c}_i \cdot \mathbf{v}) + \gamma^{(1)}(\mathbf{c}_i \cdot \mathbf{v})^2 + \delta^{(1)}\mathbf{v}^2 & i &= 1, \dots, 6 \\
 F_i^{(eq)} &= d^{(2)} + \beta^{(2)}(\mathbf{c}_i \cdot \mathbf{v}) + \gamma^{(2)}(\mathbf{c}_i \cdot \mathbf{v})^2 + \delta^{(2)}\mathbf{v}^2 & i &= 7, \dots, 14
 \end{aligned}$$

dove

$$\begin{aligned}
 d^{(0)} &= d^{(1)} = \frac{\rho}{11}, & d^{(2)} &= \frac{\rho}{22} \\
 \beta^{(1)} &= \frac{\rho}{24}, & \beta^{(2)} &= \frac{\rho}{12} \\
 \gamma^{(1)} &= \frac{\rho}{32}, & \gamma^{(2)} &= \frac{\rho}{16} \\
 \delta^{(0)} &= -\frac{7}{24}\rho, & \delta^{(1)} &= -\frac{\rho}{48}, & \delta^{(2)} &= -\frac{\rho}{24}
 \end{aligned}$$

### 4.3 Distribuzione di equilibrio: *Ansatz-Method*

“I’ve been shown cellular automata that make surprisingly good models of hydrodynamics, . . . , but I’m left with the impression that these are all *ad hoc* models, arrived by some sort of *magic*.

I’m a scientist, not a magician!”

*Toffoli e Margolus, (1990).*

Il problema che *Toffoli e Margolus* hanno voluto sottolineare, con tali parole, è il fatto che fino ad ora la distribuzione di equilibrio è stata calcolata in stretta analogia con la distribuzione di Maxwell. Questo tipo di distribuzione porta all’equazione macroscopica di *Navier-Stokes* rispettando, però delle condizioni, come l’isotropia spaziale e l’invarianza Galileiana, dalle quali il moto del fluido non dipende esplicitamente.

Ecco perchè ho deciso di dedicare un intero paragrafo alla questione che ha così tanto interessato *Toffoli e Margolus*: la ricerca di una distribuzione di equilibrio che fosse il più generale possibile, ovvero per la quale non fosse necessario

imporre i valori di alcune delle costanti usate per ottenere la convergenza del metodo. Questi, infatti, possono anche essere assegnati in un secondo tempo senza avere alcun tipo di problema nell'ambito della *multi-scale analysis*.

Tutto ciò suggerisce la seguente alternativa nella scelta di una distribuzione di equilibrio generale<sup>6</sup>, ovvero con dei parametri liberi:

$$\begin{aligned} F_i &= A_0 + D_0 \mathbf{u}^2 & i &= 0 \\ F_i &= A_1 + B_1 (\mathbf{c}_i \cdot \mathbf{u}) + C_1 (\mathbf{c}_i \cdot \mathbf{u})^2 + D_1 \mathbf{u}^2 & i &= 1, \dots, 4 \\ F_i &= A_2 + B_2 (\mathbf{c}_i \cdot \mathbf{u}) + C_2 (\mathbf{c}_i \cdot \mathbf{u})^2 + D_2 \mathbf{u}^2 & i &= 5, \dots, 8 \end{aligned}$$

La definizione della densità di massa e del momento ci portano tre imposizioni scalari su tali parametri:

$$\begin{aligned} \rho &:= \sum_i F_i = \underbrace{A_0 + 4(A_1 + A_2)}_{=\rho} + \mathbf{u}^2 \underbrace{(D_0 + 4D_1 + 4D_2 + 2C_1 + 4C_2)}_{=0} \\ \mathbf{j} &:= \sum_i \mathbf{c}_i F_i = \underbrace{(2B_1 + 4B_2)}_{=\rho} \mathbf{u}. \end{aligned}$$

Attraverso la *multi-scale analysis*, inoltre, otteniamo delle ulteriori imposizioni, che però ci lasciano comunque una certa libertà nella scelta.

Tale analisi procede, nella prima parte, sostanzialmente come quanto visto nel paragrafo (2.3.3) nel caso bi-dimensionale, in quanto per lo sviluppo dei calcoli annessi, non entra in gioco la scelta specifica fatta per la distribuzione di equilibrio. L'unico calcolo nel quale si deve considerare tale decisione è quello

---

<sup>6</sup>Tutte le considerazioni sono fatte nel bi-dimensionale, per semplificare i calcoli. Tuttavia l'estensione alla terza dimensione non porta alcuna complicazione pratica.

che include il momento  $P_{\alpha\beta}^{(0)}$ , e quindi l'unica parte della *multi-scale analysis* che riconsideriamo è proprio quella che include tale termine.

$$\begin{aligned} P_{\alpha\beta}^{(0)} &:= \sum_i c_{i\alpha} c_{i\beta} F_i^{(0)} \\ &= (2A_1 + 4A_2) \delta_{\alpha\beta} + (2C_1 u_\alpha^2 + 4C_2 \mathbf{u}^2) \delta_{\alpha\beta} \\ &= +2C_1 u_\alpha u_\beta (1 - \delta_{\alpha\beta}) + (2D_1 + 4D_2) \mathbf{u}^2 \delta_{\alpha\beta}. \end{aligned}$$

Il risultato che vorremmo ottenere è una forma standard per  $P_{\alpha\beta}^{(0)}$ , ovvero:

$$\rho \begin{pmatrix} u^2 & uv \\ uv & v^2 \end{pmatrix} + p \delta_{\alpha\beta}. \quad (4.2)$$

Questo risultato lo si raggiunge se sono verificate alcune condizioni sui parametri utilizzati. In particolare, il termine del tensore  $P_{\alpha\beta}^{(0)}$  indipendente dalla velocità del flusso  $\mathbf{u} = (u, v)$ , ci da il valore per determinare la pressione

$$p = 2A_1 + 4A_2.$$

I termini restanti ci portano alla formulazione delle seguenti relazioni:

$$4C_2 + 2D_1 + 4D_2 = 0$$

$$2C_1 = \rho$$

$$8C_2 = \rho$$

e quindi

$$C_1 = \frac{\rho}{2} \quad e \quad C_2 = \frac{\rho}{8}.$$

Ora, la scelta dei valori che debbono assumere tutte le rimanenti quantità, è sottoposta alle sole imposizioni già considerate

$$\begin{aligned} A_0 + 4(A_1 + A_2) &= \rho \\ 2B_1 + 4B_2 &= \rho \\ D_0 + 4D_1 + 4D_2 = -2C_1 - 4C_2 &= -\frac{3}{2}\rho \\ 2D_1 + 4D_2 &= -\frac{1}{2}\rho \end{aligned}$$

Come abbiamo accennato in precedenza, con l'uso di tale metodo abbiamo una maggiore generalità e libertà nella scelta della distribuzione di equilibrio.

#### Osservazione 4.1.

*Possiamo osservare che la distribuzione di Koelman è ottenibile come caso particolare di quella appena considerata, a conferma dei discorsi fatti.*

*L'ulteriore imposizione che bisogna effettuare, a tale scopo è la seguente:*

$$\frac{A_0}{A_1} = \frac{A_1}{A_2} = \frac{B_1}{B_2} = \frac{D_0}{D_1} =: r$$

*con  $r$  parametro libero. L'ulteriore condizione*

$$\frac{D_1}{D_2} = r$$

*ci permette di avere dei valori espliciti per  $r$ :  $r = -2$  e  $r = 4$ .*

*Il valore  $r = 4$ <sup>7</sup> ci dà delle soluzioni per le costanti<sup>8</sup> tali da ottenere l'e-*

---

<sup>7</sup>L'altro valore non lo consideriamo poiché ci porta a delle distribuzioni di equilibrio negative, valori non accettabili nella nostra situazione.

<sup>8</sup>Non facciamo esplicitamente tali calcoli in quanto semplici e per non appesantire troppo la trattazione, che vuole essere solo un esempio dei diversi modi attraverso i quali sia possibile 'imitare' il moto microscopico dei sistemi fisici reali.



*quazione (3.21) calcolata a suo tempo imponendo il Principio di Massima Entropia.*

Concludiamo il paragrafo osservando che la distribuzione di equilibrio è stata calcolata regolando i parametri liberi presenti, che vengono fuori dall'*ansatz-method*, in modo che le equazioni macroscopiche considerate siano soddisfatte. Tutte le altre costrizioni imposte sono arbitrarie e totalmente libere. La scelta fatta nell'osservazione è infatti volta a mostrare come le precedenti equazioni siano da esse generalizzabili.

## 4.4 Simulazione numerica tri-dimensionale

### 4.4.1 *Miscelazione laminare tramite l'uso di lame rotanti.*

Nell'ambito dei *LBM* tri-dimensionali, uno dei maggiori risultati da noi ottenuti è stata la simulazione numerica della miscelazione di un fluido viscoso incomprimibile (Newtoniano) con densità uniforme, posto all'interno di un recipiente, al quale viene applicato un forzante attraverso delle lame rotanti poste al suo interno (cfr. [12]).

Questo moto è studiato utilizzando le traiettorie (calcolate attraverso il metodo *LB*) di alcune particelle del fluido per poi visualizzarne la distribuzione all'interno del contenitore considerato.

Qui di seguito studieremo nel dettaglio la formulazione matematica del problema in questione e le fasi implementative.

#### 4.4.2 Formulazione matematica del problema della miscelazione

Come più volte è stato detto, l'idrodinamica dei fluidi viscosi ed incomprimibili si fonda sull'equazione di continuità e sulle equazioni di Navier-Stokes:

$$\operatorname{div}(\mathbf{u}) = 0 \quad (4.3)$$

$$\frac{\partial \mathbf{u}}{\partial t} - \frac{1}{\mathcal{R}_e} \Delta \mathbf{u} + (\mathbf{u} \cdot \nabla) \mathbf{u} + \nabla p = \mathbf{f} \quad (4.4)$$

dove,  $\forall t \in [0, T]$  e  $\forall \mathbf{x} = (x_1, x_2, x_3) \in \Omega$ ,  $\mathbf{u}(\mathbf{x}, t) = (u_1, u_2, u_3)$  è la velocità del fluido considerato all'interno di un ben definito dominio  $\Omega \times [0, T] \subset \mathbb{R}^3 \times \mathbb{R}$ .

#### Nota 4.2.

*Queste equazioni (cfr. [2]) sono state integrate solo in casi estremamente semplici, quali, ad esempio, quello di uno strato limite di una parete piana. Tutti i fluidi possono essere considerati come mezzi continui in movimento e per molti di essi la condizione di compressibilità può essere trascurabile, ovvero ci troviamo in presenza di fluidi incomprimibili (cioè di quella classe che intendiamo considerare).*

*Questa condizione è verificata tutte le volte che la velocità del fluido, relativa ai corpi che esso incontra nel suo moto, è piccola di fronte alla velocità con cui il suono si propaga nel fluido stesso (questa è quella che noi abbiamo definito sound-speed ed indicato con  $c_s$ ). Quando tale condizione non è soddisfatta, si entra più propriamente nell'aerodinamica. Più esattamente, le leggi idrodinamiche valgono fino a che il numero di*

*Mach*<sup>9</sup> non oltrepassa un valore limite, che è la  $c_s$  nell'aria in condizioni normali (300 Km / h).

Se poi nella Formula (4.4) introduciamo il *coefficiente di viscosità dinamica*  $\mu$  e la *densità assoluta del fluido*  $\rho$  (supponendo che assumano valore uniforme all'interno del dominio), si può definire un coefficiente adimensionale

$$\mathcal{R}_e = \frac{\rho L U}{\mu} \quad (4.5)$$

conosciuto come il *numero di Reynolds*.

Le quantità  $L$  ed  $U$  rappresentano, rispettivamente, la lunghezza e la velocità caratteristica del corpo considerato.

Inoltre se poniamo

$$\nu = \frac{\mu}{\rho}$$

otteniamo il coefficiente di viscosità cinematica caratteristico del fluido.

### **Nota 4.3.**

*Il numero di Reynolds (cfr. [3]), che rappresenta in un certo senso il rapporto tra le forze di inerzia e le forze di viscosità, definisce, nella meccanica dei fluidi, le condizioni di similitudine di scorrimento in condotti geometricamente simili. Nel momento in cui la compressibilità di un fluido può essere trascurata, tutti i coefficienti adimensionali che caratterizzano uno scorrimento dipendono soltanto da tale numero.*

*Nell'idrodinamica, quindi, ha grande importanza il valore che esso assume ogni volta, in particolare esiste un valore critico, che è definito*

---

<sup>9</sup>Quantità il cui valore è stato definito nel paragrafo (2.2.3).

*come la soglia corrispondente alla transizione dal moto laminare a quello turbolento del fluido. Quando si raggiunge tale valore critico si osserva una brusca variazione dei coefficienti che caratterizzano il moto del fluido stesso. Tale valore critico del numero di Reynolds, per una data corrente fluida, può variare entro limiti molto ampi; esso dipende, in particolare, dal tipo di superficie solida su cui scorre il fluido, dalla turbolenza della corrente che investe la superficie e dall'esistenza di perturbazioni che si producono nel fluido stesso.*

In questa formulazione, completamente adimensionale del problema, e quindi generalizzabile alle situazioni più diverse, il campo delle velocità del fluido è definito da  $\mathbf{u}U$ , mentre la posizione è specificata dalla quantità  $\mathbf{x}L$ , la pressione del fluido è  $p\rho U^2$  e la forza per unità di massa è  $fU^2/L$ .

La condizione imposta attraverso l'equazione di continuità (4.3), ci dice che il fluido considerato è incomprimibile, ovvero che il volume occupato da un certo numero di particelle non risulta essere variabile nel tempo.

Tuttavia, anche se sono stati dati questi precisi 'riferimenti matematici', il problema non risulta ancora essere completamente definito, ma lo sarà solo nel momento in cui verranno stabilite le condizioni al bordo che dovranno essere utilizzate e verrà ben specificato il dominio su cui agire.

Per quanto riguarda la prima cosa da chiarire, possiamo affermare che le *C-B* utilizzate sono quelle di *Dirichlet*, che possono essere schematizzate come:

$$\mathbf{u}(\mathbf{x}, t) = \mathbf{g}(\mathbf{x}, t) \quad \forall t, \forall \mathbf{x} \in \partial\Omega, \quad (4.6)$$

dove  $\mathbf{g}$  è una funzione che deve soddisfare la condizione

$$\int_{\partial\Omega} \mathbf{g} \cdot \mathbf{n} \, dS = 0 \quad (4.7)$$

ovvero deve avere divergenza nulla.

Per quanto riguarda la geometria del problema, e quindi la seconda delle argomentazioni necessarie per poter specificare completamente la situazione, cercheremo ora di individuare il dominio all'interno del quale le equazioni (4.3) e (4.4) risultano essere definite. Tale regione è stata espressa come  $\Omega \times [0, T]$  ( $\Omega \subset \mathbb{R}^3$ ).

Nel problema da noi considerato, questa è definita come

$$\Omega = T_C / C$$

con

$$T_C = \left\{ (x_1, x_2, x_3) \in \mathbb{R}^3 : 0 < x_3 < z_s, x_1^2 + x_2^2 < \left( \frac{r_s - r_i}{z_s} x_3 + r_i \right)^2 \right\}$$

e

$$C = \left\{ (x_1, x_2, x_3) \in \mathbb{R}^3 : z_m \leq x_3 \leq z_M, x_1^2 + x_2^2 \leq (r_C)^2 \right\}.$$

A questo punto dobbiamo distinguere, a seconda della scelta di alcuni dei parametri, due situazioni possibili:

#### **Bicchiere Dritto:**<sup>10</sup>

Questo (mostrato nella Figura (4.3)-(a)) si ha se si considerano, nel tronco di cono circolare  $T_C$ , le quantità  $r_i$  e  $r_s$  rispettivamente come il raggio di base inferiore e superiore;

<sup>10</sup>Li definiamo, in ambo i casi, *Bicchieri* solo per la loro forma caratteristica.

**Bicchiere Rovesciato:**

Questo, invece, (mostrato nella Figura (4.3)-(b)) lo si ottiene se si considerano, nel medesimo tronco di cono circolare  $T_C$ , le quantità  $r_i$  e  $r_s$  rispettivamente come il raggio di base superiore e inferiore, cioè invertite.

Queste sono le due geometrie da noi considerate nel problema della miscelazione.

Ora, nella definizione del dominio, abbiamo anche la quantità  $z_s$ , che rappresenta l'altezza del tronco di cono  $T_C$  e  $C$  è un cilindretto circolare di raggio  $r_C$  ed altezza  $z_M - z_m$ .

Le ulteriori relazioni esistenti fra le diverse quantità considerate, oltre quelle che impongono la fondamentale distinzione fra *Bicchiere Dritto* e *Bicchiere Rovesciato*, sono:

$$\begin{aligned} r_i > 0, \quad r_s > 0, \\ z_s > z_M > z_m > 0, \\ \frac{r_s - r_i}{z_s} z_m + r_i > r_C > 0. \end{aligned}$$

Il bordo del dominio  $\Omega$  è  $\partial\Omega = \Gamma_i \cup \Gamma_0 = \Gamma$ , dove:

$$\Gamma_i = \partial C, \quad \Gamma_0 = \partial T_C.$$

Le *condizioni al bordo* che debbono essere imposte su  $\Gamma$  non sono uniformi, ma si differenziano a seconda della zona considerata.

Se, infatti, ci troviamo in  $\Gamma_0$  vengono imposte *C-B no-slip*, in quanto questo viene considerato come un bordo rigido, ovvero le particelle debbono assumere su di esso velocità nulla:

$$\mathbf{u}(\mathbf{x}, t) = 0 \quad \forall \mathbf{x} \in \Gamma_0, \forall t$$

Se, invece, ci troviamo in  $\Gamma_i$ , le condizioni da imporre sono diverse, in quanto siamo sulle lame rotanti.

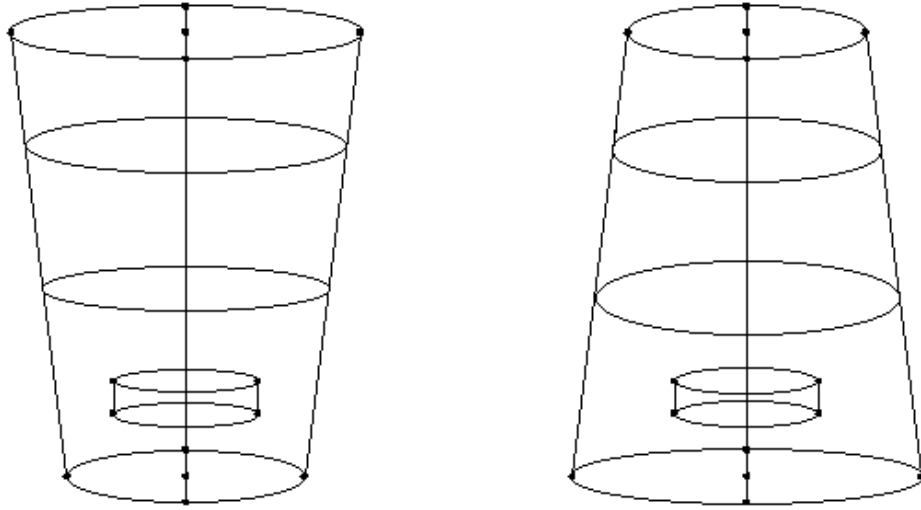
(a) *Bicchiere Dritto*(b) *Bicchiere Rovesciato*

Figura 4.3: Geometrie utilizzate nella simulazione della miscelazione.

In tale zona, le particelle assumono una velocità che dipende dalla spinta impressa da quest'ultime, che si può esprimere attraverso la funzione  $\mathbf{g}$ , considerata nella (4.6), in maniera tale che:

$$\mathbf{u}(\mathbf{x}, t) = \mathbf{g} \quad \forall \mathbf{x} \in \Gamma_i, \forall t.$$

Inoltre  $\mathbf{g}$  deve soddisfare anche la condizione (4.7)<sup>11</sup>.

La condizione iniziale è imposta supponendo che il campo delle velocità sia ovunque nullo.

<sup>11</sup>Nel prossimo paragrafo descriveremo la funzione utilizzata nell'implementazione a tale scopo e che soddisfa la condizione  $\operatorname{div} \mathbf{g} = 0$

### 4.4.3 Calcolo del flusso sulle lame rotanti

Cerchiamo, ora, di calcolare e dare una precisa formulazione della funzione che ci permette di stabilire ed imporre le velocità che debbono assumere le particelle che si trovano in prossimità delle lame rotanti, ovvero sul bordo della regione  $C$  sopra definita, nella quale il fluido si trova sotto l'azione delle lamine.

Data la simmetria cilindrica del problema, le coordinate cilindriche, appunto, sembrano essere le più adatte al calcolo che andremo ad effettuare.

In tali coordinate, si può definire:

$$\mathbf{V}(\rho, \theta, z) = (-\rho \sin(\theta))V_\rho + \cos(\theta)V_\theta, \rho \cos(\theta)V_\rho + \sin(\theta)V_\theta, V_z),$$

dove

$$\begin{cases} V_\rho(\rho, \theta, z) = f(\theta) \\ V_\theta(\rho, \theta, z) = \frac{k-1}{2} f'(\theta) \rho + g(\theta) \phi(\rho) \\ V_z(\rho, \theta, z) = -kz f'(\theta) - zg(\theta) \left[ \phi'(\rho) + \frac{\phi(\rho)}{\rho} \right] + h(\theta) \varphi(\rho) \end{cases}$$

Nella definizione, abbiamo la somma di tre termini,  $f$ ,  $g\phi$  e  $h\varphi$ , con divergenza nulla e di conseguenza  $\text{div } \mathbf{V} = 0$ .

Per poter definire le condizioni da imporre sul bordo del dominio determinato dalle lame rotanti, dobbiamo introdurre la funzione periodica

$$l(l_1, l_0, A, a) = \begin{cases} l_1 + \frac{l_0 - l_1}{2} \left( \cos\left(\frac{2\pi a}{A}\right) + 1 \right), & \text{se } \cos(a) \geq \cos(A/2) \\ l_1, & \text{se } \cos(a) < \cos(A/2) \end{cases}$$

Osserviamo che per  $a = 0$ ,  $l = l_0$ .

Notiamo, inoltre, che la posizione di ogni pala, al tempo iniziale, coincide con



quella di un segmento la cui collocazione, in coordinate cilindriche, è data da  $\theta_i = (i\pi)/2$ , per  $i = 0, \dots, 3$ ,  $z_0 = (z_m + z_M)/2$ ,  $r \leq l_b$ , dove  $l_b$  rappresenta la lunghezza di ogni lama.

Nella simulazione da noi implementata, sono state distinte diverse fasi nell'utilizzo di tale funzione, in modo da partire da una situazione iniziale di completa stazionarietà, per poi arrivare ad uno stato finale, per cui la velocità angolare assunta sulle pale è pari a  $\omega_b = 8\pi$  (così che al tempo  $t$  esse si trovano nella posizione  $\theta_i(t) = \theta_i + \omega_b t$ ), senza alcuna discontinuità.

In altre parole, questo significa che per  $t \geq 0.5$  la velocità di trascinamento del fluido è  $\omega_f l_b$  con  $\omega_f = 2\pi$ , alla quale si aggiunge un'ulteriore velocità radiale, quando questo si trova sulle lame, che risulta essere inversamente proporzionale alla distanza rispetto alle pale e alla posizione radiale di ogni particella che lo costituisce. Il valore massimo che tale quantità può assumere è  $(\omega_b - \omega_f)l_b = 6\pi l_b$ .

Se invece supponiamo che  $t \leq 0.5$ , la velocità applicata sulle pale è riscalata di un certo fattore, in modo da arrivare dal valore iniziale nullo, al valore finale, raggiunto al tempo  $t = 0.5$ , di 4 giri al secondo.

Tutto questo è possibile attraverso l'uso della funzione continua

$$s(t) = \begin{cases} \frac{1 - \cos(2\pi t)}{2}, & \text{se } t \leq 0.5 \\ 1, & \text{se } t > 0.5 \end{cases}$$

Quindi la velocità impressa al fluido passa da un valore iniziale a quello finale di  $8\pi l_b$ , in un intervallo di tempo  $[0, 0.5]$ .

Nella simulazione da noi implementata, le lame rotanti danno al fluido una spinta sia in direzione radiale che verticale e questo fatto si riflette nella

definizione della funzione  $\mathbf{g}$  che determina tale spinta.

Sotto tali ipotesi, quindi, la funzione che assegna le velocità del fluido al tempo  $t$  nel punto  $p \in \partial C$  di coordinate  $p = (\rho, \theta, z)$ , assume la forma  $\mathbf{g}(\rho, \theta, z) = \mathbf{V}(\rho, \theta, z - z_0)$  con:

$$\left\{ \begin{array}{l} f(\theta) = s(t) (\omega_f + \sum_{i=0}^3 f_i(\theta, t)) \\ g(\theta) = 0 \\ h(\theta) = s(t) (\sum_{i=0}^3 h_i(\theta, t)) \\ \phi(\rho) = 1 \\ \varphi(\rho) = \tilde{\varphi}(\rho) \\ k = 1 \end{array} \right. \quad (4.8)$$

dove

$$f_i(\theta, t) = l(0, \omega_b - \omega_f, \frac{\pi}{4}, \theta - \theta_i(t))$$

in questo caso  $A = \pi/4$ .

Inoltre si hanno:

$$h_i(\theta, t) = \begin{cases} l(0, v_i, \pi, \theta - \theta_i(t)), & \text{se } \sin(\theta - \theta_i(t)) < 0 \\ l(0, v_i, \pi/4, \theta - \theta_i(t)), & \text{se } \sin(\theta - \theta_i(t)) \geq 0 \end{cases}$$

e

$$\tilde{\varphi}(\rho) = \begin{cases} \left(1 - \cos\left(\frac{\pi\rho}{l_b}\right)\right), & \text{se } \rho < l_b \\ 1, & \text{se } \rho \geq l_b \end{cases}$$

dove  $v_i = (-1)^i \tilde{s}$ .

In questo modo possiamo imprimere al fluido, attraverso le pale, un qualsiasi valore di  $\tilde{s}$ , che ci indica la spinta verticale.

Nel problema studiato si è scelto  $\tilde{s} = 5$ .

#### 4.4.4 Discretizzazione del problema

Tutte le considerazioni fatte, nei paragrafi precedenti, riguardo il problema che vogliamo andare a risolvere, sono state, comunque, pensate e proposte in maniera continua, i volumi considerati sono i veri volumi fisici.

Tuttavia, per poter implementare il metodo *LB* dobbiamo suddividere il dominio  $\Omega$  in un insieme di cubetti. Va ricordato che in tal modo si può fornire solo un'approssimazione del dominio fisico.

Il primo passo, in questo senso, si effettua considerando una prima suddivisione del *dominio complessivo*  $P$ , che rappresenta il parallelepipedo che racchiude il bicchiere, come mostrato in Figura (4.4), definito, in coordinate cartesiane, come:

$$P = \left\{ (x_1, x_2, x_3) \in \mathbb{R}^3 : 0 \leq x_1 \leq 2r_s, 0 \leq x_2 \leq 2r_s, 0 \leq x_3 \leq z_s \right\}$$

con  $r_s, z_s$  definiti come in precedenza.

Questo è costruito in maniera tale che  $\Omega \subset P$ . È stato considerato tale insieme  $P$ , che costituisce un ampliamento del dominio che a noi interessa realmente, in quanto il calcolo risulta essere enormemente semplificato. Tuttavia nell'implementazione del problema i nodi che saranno stimati essere, secondo alcuni criteri che di seguito andremo a definire, esterni al bicchiere, non saranno valutati nel corso del processo evolutivo, in modo da rendere i calcoli il meno pesanti possibile.

La partizione che abbiamo effettuato, la cui proiezione orizzontale è mostrata nella Figura (4.5), considerata rispetto alle coordinate cartesiane, in quanto la griglia utilizzata (cioè quella con 19 vettori velocità e rappresentata nella

Figura (4.2)<sup>12</sup>) è costruita rispetto ad esse, permette di suddividere  $P$  rispettivamente lungo le tre direzioni in  $n_x$ ,  $n_y$  e  $n_z$  componenti.

Ora, stabilito di voler utilizzare la griglia con 19 vettori, possiamo affermare che i nodi, che formano tale griglia, si trovano nei punti  $\mathbf{x} = (x_i, y_j, z_k) \in P$  con  $i = 0, \dots, n_x$ ,  $j = 0, \dots, n_y$  e  $k = 0, \dots, n_z$ .

A questo punto, tuttavia, dobbiamo stabilire quali sono le relazioni che questi punti debbono soddisfare, per poter dire che si trovano all'interno del nostro dominio  $\Omega$  o su parti di esso già considerate in precedenza (come sul bordo esterno o sulle lame rotanti).

Consideriamo, quindi, un punto  $\mathbf{x} \in P_{disc}$  (sotto-insieme di  $P$  che contiene solo i punti facenti parte della discretizzazione) con  $\mathbf{x} = (x_1, x_2, x_3)$ .

Se, ora, indichiamo con  $l_x$ ,  $l_y$  e  $l_z$  rispettivamente le lunghezze reali del dominio  $\Omega$ , e con  $\delta_x = \frac{l_x}{n_x - 1}$  la lunghezza caratteristica della griglia considerata, le coordinate della particella considerata nel punto  $\mathbf{x}$  all'interno del dominio  $P$  sono  $p = (p_1, p_2, p_3)$  con

$$\begin{cases} p_1 = -\frac{l_x}{2} + (x_1 - 1) \delta_x \\ p_2 = -\frac{l_x}{2} + (x_2 - 1) \delta_x \\ p_3 = (x_3 - 1) \delta_x \end{cases}$$

Una volta determinata, quindi, la posizione della particella, possiamo, finalmente, dettare le condizioni che ci permettono di stabilirne la posizione:

---

<sup>12</sup>La scelta è caduta su tal griglia, piuttosto che un'altra, soltanto, come già detto in altre occasioni, per una questione di continuità rispetto alle scelte fatte in precedenza a livello bi-dimensionale.

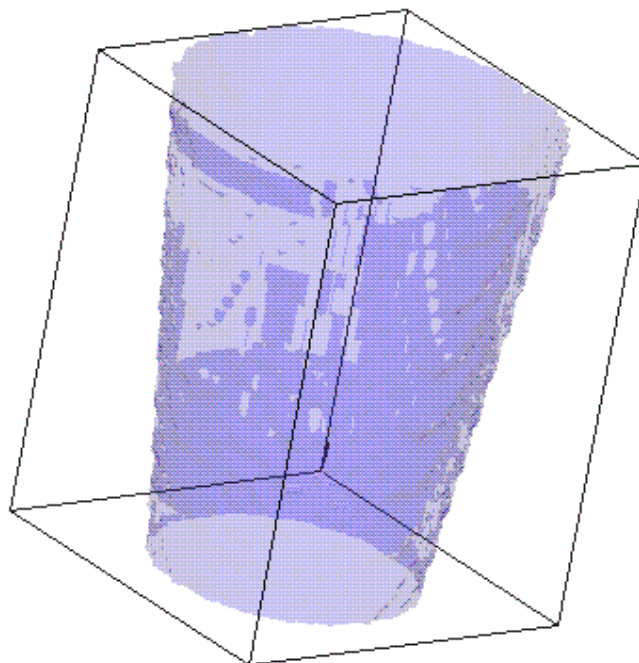


Figura 4.4: Parallelepipedo che contiene il bicchiere.

$$\text{se } \underbrace{\{(p_1)^2 + (p_2)^2\}}_{\text{vera}} < \underbrace{\left\{ p_3 \frac{r_s - r_i}{l_z} + r_i \right\}^2}_{\text{vera}}$$

$$\Downarrow$$

$$p \in \Omega \subset P$$

Possiamo quindi affermare di trovarci all'interno del bicchiere.



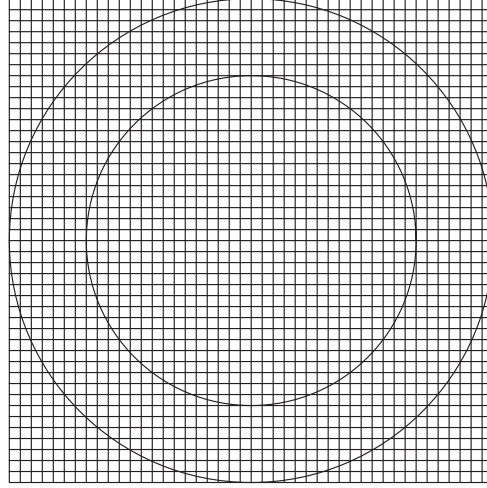


Figura 4.5: Proiezione orizzontale della griglia.

Se una delle ulteriori condizioni seguenti è verificata

$$\begin{cases} p_3 = l_z \\ p_3 = 1 \text{ e } \left\{ p_3 \frac{r_s - r_i}{l_z} + r_i \right\} + \epsilon \delta_x > \left\{ (p_1)^2 + (p_2)^2 \right\}^{1/2} \\ \left\{ p_3 \frac{r_s - r_i}{l_z} + r_i \right\} + \epsilon \delta_x > \left\{ (p_1)^2 + (p_2)^2 \right\}^{1/2} \end{cases}$$

possiamo affermare di trovarci in una zona di bordo, ovvero che  $p \in \Gamma \subset \partial\Omega$ .  
Invece se sono verificate le seguenti, possiamo dire di trovarci in prossimità delle lame, cioè  $p \in \partial C = \Gamma_i$

$$x_i < x_1 < x_s, \quad y_i < x_2 < y_s, \quad r_m < p_3 < r_M,$$

e

$$\left\{ (p_1)^2 + (p_2)^2 \right\} \leq r_C^2$$

dove  $x_i$ ,  $x_s$ ,  $y_i$  e  $y_s$  sono, rispettivamente, le lunghezze del dischetto contenente le pale, lungo le direzioni dell'asse  $x$  e  $y$ .

Abbiamo, in questo modo, discretizzato completamente il dominio di partenza considerato, siamo quindi in grado di andare ad applicarvi il *metodo di Boltzmann* per poter risolvere completamente il problema.

#### 4.4.5 Implementazione del metodo *Lattice-Boltzmann*

Quello che ora vogliamo andare a calcolare, attraverso il metodo *LB*, sono le velocità che le particelle assumono in quanto sottoposte alle condizioni sul *bordo* ed a quelle sulle *lame rotanti* specificate all'inizio del paragrafo. Ovvero vogliamo risolvere l'*equazione di Navier-Stokes* e l'*equazione di continuità*  $\forall p \in \Omega$ .

A tale scopo, abbiamo scelto di utilizzare un *LBM* tri-dimensionale, implementato su di un dominio ampiamente descritto nel paragrafo precedente.

Come sappiamo, una volta stabilite le condizioni sulla griglia, o meglio, una volta che questa è stata scelta, risulta facile la definizione della *distribuzione di equilibrio*<sup>13</sup> da associare al modello, in quanto i momenti  $W_i$  sono completamente definiti dai vettori del 'lattice'. Il loro valore nel caso della griglia a 19 velocità è:

$$W_0 = \rho_0 \frac{1}{3} \quad (4.9)$$

$$W_1 = \rho_0 \frac{1}{2} \left( \frac{kT}{m} \right)^2 = \rho_0 \frac{1}{18} \quad (4.10)$$

$$W_2 = \rho_0 \frac{1}{4} \left( \frac{kT}{m} \right)^2 = \rho_0 \frac{1}{36} \quad (4.11)$$

$$\frac{kT}{m} = \frac{1}{3}$$

---

<sup>13</sup>Notiamo che, nella nostra implementazione tutte le scelte fatte, per la distribuzione di equilibrio, si sono basate sul modello di *Koelman*.



dove  $\rho_0$  è la densità di massa iniziale in ogni nodo.

A partire da questi valori si arriva con facilità a stilare la relazione che stabilisce la distribuzione di equilibrio:

$$\begin{aligned} F_i^{(eq)} &= \frac{1}{3}\rho \left[1 - \frac{3}{2}\mathbf{u}^2\right] & i &= 0 \\ F_i^{(eq)} &= \frac{1}{18}\rho \left[1 + 3(\mathbf{c}_i \cdot \mathbf{u}) + \frac{9}{2}(\mathbf{c}_i \cdot \mathbf{u})^2 - \frac{3}{2}\mathbf{u}^2\right] & i &= 1, \dots, 6 \\ F_i^{(eq)} &= \frac{1}{36}\rho \left[1 + 3(\mathbf{c}_i \cdot \mathbf{u}) + \frac{9}{2}(\mathbf{c}_i \cdot \mathbf{u})^2 - \frac{3}{2}\mathbf{u}^2\right] & i &= 7, \dots, 18 \end{aligned}$$

Le condizioni generali per l'applicazione del metodo, quindi, sono state date. Rimangono, a questo punto, solo da considerare in maniera più dettagliata i diversi *step* necessari all'implementazione.

### Step 1.

*Il primo passo compiuto nella stesura del programma e nella simulazione della miscelazione, è stato l'inizializzazione dei nodi della griglia mediante l'uso delle relazioni (4.9)-(4.11), che stabiliscono il valore dei momenti, rispetto alla griglia considerata.*

*Al tempo  $t = 0$  si suppone, inoltre, che il fluido sia, oltre che fermo, uniformemente distribuito all'interno del dominio  $\Omega$ . Questo equivale ad imporre che la densità di massa locale assuma in ogni nodo del lattice lo stesso valore iniziale  $\text{density} = \rho_0 = 0.1$ .*

*Nel programma `sim.f90` (il cui listato è allegato nell'Appendice B) questo processo si effettua all'interno della 'subroutine' `init_density` che inizializza i nodi della griglia, memorizzati nella matrice quadri-dimensionale `node(i,x,y,z)`, utilizzando le relazioni (4.9)-(4.11) come segue:*

$$\begin{aligned} \text{node}(i, \mathbf{x}, y, z) = F_i(\mathbf{x}, 0) &= \rho_0 \frac{1}{3}, & \text{se } i = 0 \\ \text{node}(i, \mathbf{x}, y, z) = F_i(\mathbf{x}, 0) &= \rho_0 \frac{1}{18}, & \text{se } i = 1, \dots, 6 \\ \text{node}(i, \mathbf{x}, y, z) = F_i(\mathbf{x}, 0) &= \rho_0 \frac{1}{36}, & \text{se } i = 7, \dots, 18 \end{aligned}$$

dove  $\mathbf{x} = (x, y, z)$  è la posizione del nodo all'interno del dominio.

### Step 2.

A questo punto si può effettuare la seconda fase della simulazione.

Abbiamo più volte detto che nei modelli facenti parte degli FHP, l'evoluzione del sistema si ottiene attraverso la composizione di due processi:

$$\mathcal{E} = \mathcal{C} \circ \mathcal{P}$$

dove  $\mathcal{C}$  rappresenta la collisione e  $\mathcal{P}$  è la fase della propagazione, che è quella che vogliamo considerare in questo passo.

Questa consiste nello 'scorrimento' delle particelle che si trovano lungo un collegamento del nodo considerato, verso il nodo adiacente.

Tale processo è regolato dalla relazione:

$$F_i(\mathbf{x} + \mathbf{c}_i \Delta t, t + \Delta t) = F_i(\mathbf{x}, t) \quad (4.12)$$

dove  $F_i$  è il momento associato alla particella che scorre lungo il collegamento  $i$ -esimo del nodo che si trova nella posizione  $\mathbf{x}$ .

Nel programma la 'subroutine' `propagate` assolve questa funzione, mediante l'uso di una matrice di appoggio `n_hlp(i, x, y, z)`.

### Step 3.

Il passo successivo nell'implementazione del metodo, è quello della collisione, che nel modello LB viene effettuata tramite l'uso dell'equazione BGK, espressa

dalla (3.7), e mediante una distribuzione di equilibrio opportuna, come quella vista nel paragrafo (4.2.2). Per questo motivo, quindi, quella che stiamo considerando risulta essere la fase cruciale.

Tuttavia è necessario fare delle distinzioni rispetto alla posizione assunta dalla particella  $p$  all'interno del dominio, le equazioni da applicare, infatti, sono diverse e variano a seconda del punto in cui  $p$  si trova ( $p \in \Gamma_0$  o  $p \in \text{Int } \Omega$ <sup>14</sup>), al tempo  $t$  considerato, durante il suo moto.

### **Bordo.**

Se la particella  $p$  si trova in un punto del bordo di  $T_C$ , ovvero  $p \in \Gamma_0$ , la sua velocità risulta essere nota, in quanto su  $\Gamma_0$ , considerato come una parete rigida, sono imposte condizioni no-slip, ovvero  $\mathbf{u}(\mathbf{x}, t) = 0$  (dove con  $\mathbf{x}$  indichiamo la posizione di  $p$ ).

Osserviamo che, come più volte è stato detto, il metodo LB, così come tutti gli AC, risulta essere omogeneo ed isotropo. Questa fondamentale proprietà, però, si perde nel momento in cui ci troviamo in particolari posizioni, come nelle regioni che costituiscono il bordo del dominio considerato.

In esse non si possono, quindi, applicare le classiche regole di aggiornamento, ma si debbono usare dei metodi alternativi.

Quello generalmente utilizzato nella situazione nella quale ci troviamo, è il metodo bounce-back, che riassegna ad ogni particella la stessa velocità di arrivo, in modulo e direzione, ma con verso opposto.

---

<sup>14</sup>Con 'Int  $\Omega$ ' abbiamo indicato l'interno di  $\Omega$ .

Schematicamente questo può essere espresso come:

$$\begin{aligned} \text{prima:} & \quad F_0, F_1, F_2, F_3, F_4, F_5, F_6, \dots \\ \text{dopo:} & \quad F_0, F_2, F_1, F_4, F_3, F_6, F_5, \dots \end{aligned}$$

e per il calcolo di tutti i restanti, basta fare riferimento, procedendo in maniera del tutto analoga, alla Figura (4.2).

### **Internamente al Bicchiere.**

In questo caso la velocità della particella è  $\mathbf{u} = (u_x, u_y, u_z)$ , dove tale quantità viene calcolata come:

$$\begin{aligned} u_x &= \frac{1}{\rho_l} \{ F_1 + F_7 + F_8 + F_{11} + F_{13} \\ &\quad - (F_2 + F_9 + F_{10} + F_{12} + F_{14}) \} \\ u_y &= \frac{1}{\rho_l} \{ F_3 + F_7 + F_9 + F_{15} + F_{16} \\ &\quad - (F_4 + F_8 + F_{10} + F_{17} + F_{18}) \} \\ u_z &= \frac{1}{\rho_l} \{ F_5 + F_{11} + F_{12} + F_{15} + F_{17} \\ &\quad - (F_6 + F_{13} + F_{14} + F_{16} + F_{18}) \} \end{aligned}$$

dove  $F_i$ ,  $\forall i = 1, \dots, 18$ , è il valore, lungo uno dei diversi collegamenti del nodo, del momento locale e  $\rho_l$  è la densità di massa locale in ogni nodo.

Tuttavia se ci troviamo sulle pale, ovvero  $p \in \Gamma_i = \partial C$ , il calcolo della velocità avviene in modo leggermente diverso, infatti,  $\mathbf{u} = (\tilde{u}_x, \tilde{u}_y, \tilde{u}_z)$ , con:

$$\begin{aligned} \tilde{u}_x &= \frac{g_x - u_x}{\omega} + u_x \\ \tilde{u}_y &= \frac{g_y - u_y}{\omega} + u_y \\ \tilde{u}_z &= \frac{g_z - u_z}{\omega} + u_z \end{aligned}$$

dove  $(g_x, g_y, g_z)$  sono le tre componenti cartesiane della velocità impressa dalle lamine rotanti e  $\omega$  è il termine che ci dà la frequenza di collisione.

In questo modo, tuttavia, abbiamo dato le formule mediante le quali, all'interno del programma `sim.f90`, sono state calcolate le velocità delle particelle all'interno del bicchiere.

Se invece vogliamo considerare la fase delle collisioni vera e propria, la quale ci permette di aggiornare la direzione ed il verso delle stesse, dobbiamo fare riferimento all'equazione di Boltzmann

$$F_i(\mathbf{x}, t + 1) = F_i(\mathbf{x}, t) + \omega ( F_i^{(eq)}(\mathbf{x}, t) - F_i(\mathbf{x}, t) )$$

che viene applicata a tutti i nodi interni, rispettando, in questo caso, la condizione di omogeneità spaziale.

La 'subroutine' che implementa tutto questo ragionamento all'interno del nostro programma `sim.f90`, è `relaxation`<sup>15</sup>.

#### Osservazione 4.2.

*Concludiamo il paragrafo osservando che l'evoluzione del sistema si ottiene applicando iterativamente le fasi descritte negli **Step 2.** e **Step 3.***

#### 4.4.6 Misure qualitative e quantitative della miscelazione

Il modello implementato e descritto nel paragrafo precedente, ci ha permesso di calcolare il campo delle velocità del flusso all'interno del dominio considerato. Quello che, a questo punto, rimane da fare, è analizzare i risultati ottenuti

<sup>15</sup>Nei listati annessi è possibile trovare anche una `relax_wr`, che differisce dalla precedente solo nel fatto che in questa vengono stampati tutti i valori necessari per le valutazioni quantitative e qualitative della simulazione.

in modo da poter controllare ed esaminare il fenomeno della miscelazione studiato.

Un modo per poter ottenere ciò è quello di calcolare le traiettorie seguite dalle particelle del fluido (naturalmente le particelle considerate saranno solo una parte, anche se molto grande, di tutte quelle presenti all'interno del contenitore) durante il fenomeno.

Il metodo usato per effettuare l'integrazione delle equazioni delle traiettorie è quello di *Adams-Bashforth* a due passi.

Se indichiamo con  $\boldsymbol{\xi}^p(t)$  la posizione del generico punto  $p$  al tempo  $t$  e  $\boldsymbol{\xi}_0^p$  la sua posizione iniziale, si hanno le seguenti equazioni per le traiettorie:

$$\begin{cases} \frac{d\boldsymbol{\xi}^p}{dt} = \mathbf{v}(\boldsymbol{\xi}^p, t), & \forall t \in [0, T] \\ \boldsymbol{\xi}^p(0) = \boldsymbol{\xi}_0^p \end{cases}$$

dove  $\mathbf{v}(\boldsymbol{\xi}^p, t)$  rappresenta la velocità della particella  $p$  (o meglio che si trova nella posizione  $p$ ) al tempo  $t$ , e la si ottiene come:

$$\mathbf{v}(\mathbf{x}, t) = \begin{cases} \mathbf{u}(\mathbf{x}, t), & \forall t \in [0, T], \forall \mathbf{x} \in \Omega \\ \mathbf{g}(\mathbf{x}, t), & \forall t \in [0, T], \forall \mathbf{x} \in C \end{cases}$$

A questo punto suddividiamo il dominio temporale  $[0, T]$  in  $N_T$  sotto-intervalli in modo da avere

$$\Delta t = \frac{T}{N_T}, \quad t_n = n \Delta t \quad \text{con} \quad 0 \leq n \leq N_T$$

Ora possiamo applicare il sopra citato metodo di *Adams-Bashforth*, ottenendo  $\forall 1 \leq n \leq N_T$ :

$$\boldsymbol{\xi}^p(t_{n+1}) = \boldsymbol{\xi}^p(t_n) + \frac{\Delta t}{2} (3\mathbf{v}(\boldsymbol{\xi}^p(t_n), t_n) - \mathbf{v}(\boldsymbol{\xi}^p(t_{n-1}), t_{n-1})).$$

Osserviamo che per  $t = 0$  si assume  $\mathbf{v}(\mathbf{x}, 0) = 0 \quad \forall \mathbf{x}$ , ovvero, questo sottolinea il fatto che inizialmente il fluido è fermo, come dalle ipotesi fatte.

In questo modo sappiamo dove è posizionata, all'interno del bicchiere, la particella (e di conseguenza l'insieme di tutte le particelle che si è deciso di osservare)  $\forall t \in [0, T]$  che si muove sotto l'effetto delle *lamine rotanti*.

Come abbiamo più volte accennato, solamente una parte del fluido viene seguita nel suo moto.

A tale scopo si considera la seguente suddivisione di  $\Omega \cup C$ , con un passo  $\delta_g = 0.015$ :

$$G = \{\mathbf{x} \in \Omega \cup C : x_1 = -r_m + i_1\delta_g, x_2 = -r_m + i_2\delta_g, x_3 = i_3\delta_g, i_1, i_2, i_3 \in \mathbb{N}\},$$

dove  $r_m = \max\{r_i, r_s\}$ .

Noi calcoliamo la traiettoria di un insieme di particelle, che indichiamo con  $I_p$ , all'interno dell'insieme  $S \cap G$  al tempo  $t = \Delta t$ , dove

$$S = \{\mathbf{x} \in \Omega \cup C : \frac{z_m + z_M}{2} - 0.135 \leq x_3 \leq \frac{z_m + z_M}{2} + 0.135, 0.0 \leq x_1, x_2 \leq r_m\}.$$

A questo punto, dopo aver effettuato il calcolo delle traiettorie delle particelle di cui siamo interessati, possiamo pensare di visualizzare il fenomeno della miscelazione. Questo può essere effettuato rappresentando (attraverso l'uso di [9]) le isosuperfici della funzione  $\delta : G \rightarrow \mathbb{R}$  definita nel seguente modo:

$$\delta(g) = \sum_{p \in I_p} \delta_p(g)$$

con

$$\delta_p(g) = \begin{cases} \exp \frac{d_{g,p}}{d_{g,p} - \varepsilon}, & \text{se } d_{g,p} < \varepsilon \\ 0, & \text{se } d_{g,p} \geq \varepsilon \end{cases}$$

dove  $d_{g,p}$  è la distanza fra la particella  $p$  e  $g$ , mentre  $\varepsilon = 2\delta_g$ .

Quanto detto fino a questo momento ci da una misura qualitativa del fenomeno, rappresentando graficamente la variazione nel tempo della massa considerata. Tuttavia se vogliamo avere una valutazione quantitativa dello stesso, dobbiamo considerare delle altre quantità.

A tale scopo definiamo l'insieme  $B_s$ , con  $s = 1, \dots, 12$ , nel quale rimane suddiviso parte del dominio, come

$$B_s = \left\{ \left( i d_1 \cos\left(\frac{k\pi}{8}\right), i d_1 \sin\left(\frac{k\pi}{8}\right), j d_2 + \frac{d_2}{2} \right) : 0 \leq k \leq 7, \right. \\ \left. 3 + 6k \leq i \leq 50, 3 + 12(s-1) \leq j \leq 3 + 12s - 1 \right\},$$

dove

$$d_1 = \frac{\max\{r_s, r_i\}}{75}, \quad e \quad d_2 = \frac{z_s}{150}.$$

Indichiamo, inoltre, con  $P_s$  l'insieme delle particelle che al tempo  $t = 0$  si trovano all'interno di ciascuno degli  $B_s$  sopra definiti,  $\forall s = 1, \dots, 12$ .

Questi insiemi assumono particolare importanza in quanto indicano le particelle che saranno considerate nel calcolo degli indici che a seguito saranno definiti.

A tale scopo, suddividiamo l'insieme  $\Omega \cup C$  in un numero  $N_b^{tot}$  di cubetti (nel nostro caso abbiamo considerato  $N_b^{tot} = 1100$ ) ciascuno di altezza  $3d_2$  e base  $0.13 \times 0.13$ .

Ciascuno di questi box, al tempo  $t$ , lo si dice *occupato* se in esso è presente almeno una particella dell'insieme  $P_{tot}$ , dove  $P_{tot} = \cup_{s=1}^{12} P_s$  è l'insieme delle particelle seguite che allo scorrere del tempo si saranno spostate all'interno del dominio considerato.



In caso contrario, il box lo si dice *vuoto*.

Definiamo  $N_b(t)$  come il numero di scatole occupate al tempo  $t$ .

A questo punto, per poter ottenere una valutazione quantitativa del fenomeno della miscelazione, andiamo a definire l'*indice di dispersione*  $\sigma(t)$  nel seguente modo:

$$\sigma(t) = \frac{N_b(t)}{N_b^{tot}}, \quad (4.13)$$

Notiamo che, in tal modo, ovvero dividendo per il numero totale dei cubetti in cui il dominio rimane partizionato, otteniamo una quantità nell'intervallo  $(0, 1]$  (il valore 1 lo si ottiene nel momento in cui tutti i "box" sono occupati). Continuiamo nello studio quantitativo pensando ad una ulteriore suddivisione di  $\Omega \cup C$  in un numero  $L_{tot}$  di piani (nella nostra simulazione si è scelto  $L_{tot} = 72$ )  $L_k$  definiti come:

$$L_k = \left\{ \mathbf{x} \in \Omega \cup C : \left( 2k + \frac{3}{2} \right) d_2 \leq x_3 < \left( 2k + \frac{7}{2} \right) d_2 \right\} \quad k = 1, \dots, L_{tot}$$

Se, in maniera analoga a quanto fatto per l'indice di dispersione, definiamo  $N_s(t, k)$  come l'insieme delle particelle di  $P_s$  che al tempo  $t$  si trovano nell'insieme  $L_k$ , possiamo arrivare a definire l'*indice di diffusione*  $\rho_s(t, k)$  come:

$$\rho_s(t, k) = \frac{N_s(t, k)}{10 V_k}, \quad (4.14)$$

dove con  $V_k$  abbiamo indicato il volume del piano  $L_k$ .

#### 4.4.7 Risultati sperimentali

In questo paragrafo, che conclude le considerazioni sul modello  $LB$  implementato e la descrizione del problema che ci siamo posti e che abbiamo risolto, riportiamo i parametri effettivi utilizzati e alcuni dei risultati ottenuti.

La simulazione è stata effettuata per un tempo  $T = 14$  secondi ed  $N_T = 1400$ , dove tale quantità rappresenta il numero di suddivisioni in cui si partiziona l'intervallo  $[0, T]$ , ponendo, oltretutto, il *time-step*  $\Delta t = \frac{T}{N_T}$ .

Per quanto riguarda, invece, le posizioni fatte sulla geometria del sistema, si è supposto di suddividere il dominio  $P$  ponendo  $n_x = n_y = 45$  e  $n_z = 57$ .

Le dimensioni degli elementi interni, invece, sono state scelte come:  $r_i = 0.45$ ,  $r_s = 0.66$ ,  $z_s = 1.68$ ,  $l_b = r_C = 0.275$ ,  $z_m = 0.27$  e  $z_M = 0.36$ .

Si è inoltre supposto di simulare il moto di un fluido caratterizzato da un valore del *numero di Reynolds* pari a 250, ovvero  $\mathcal{R}_e = 250$ , dove tale coefficiente adimensionale è espresso da:

$$\mathcal{R}_e = 3 \frac{U L}{\Delta t c^2} \left( \frac{1}{\omega} - \frac{1}{2} \right).$$

Tale relazione ci permette anche il calcolo del valore, da utilizzare nella simulazione, che deve essere assegnato ad  $\omega$  (ricordando che tale quantità deve rimanere sotto la soglia del valore 2).

I risultati qualitativi, ottenuti dall'applicazione del metodo *LB*, sono mostrati nelle Figure (4.8)-(4.10), attraverso la visualizzazione delle isosuperfici, tramite [9], nelle due geometrie considerate ed a diversi istanti di tempo (sono state affiancate proprio con lo scopo di poter effettuare dei confronti a stessi istanti di tempo).

Mentre nelle Figure (4.11)-(4.13) sono riportati, agli stessi istanti di osservazione, i risultati ottenuti dalla risoluzione del medesimo problema di miscelazione mediante l'applicazione del metodo agli elementi finiti, implementato con le stesse condizioni, in modo da poter effettuare un confronto qualitativo fra i due modelli.

I risultati quantitativi, sono invece mostrati, mediante la visualizzazione degli indici di diffusione definiti dalla (4.14), scegliendo  $s = 2, 4, 6$ , nella Figura (4.14).

Mentre, come quanto fatto per le isosuperfici, nella Figura (4.15) sono riportati gli indici di diffusione, calcolati considerando gli stessi valori per l'indice  $s$ , ottenuti con il metodo agli elementi finiti.

La seguente Figura (4.6) mostra la variazione dell'indice di dispersione, definito mediante la (4.13) e calcolato mediante l'applicazione del metodo  $LB$ , al passare del tempo, per il bicchiere dritto, nella (a), e rovesciato, nella (b).

Mentre la Figura (4.7) ci permette di fare un raffronto con l'indice di dispersione calcolato attraverso il metodo degli elementi finiti.

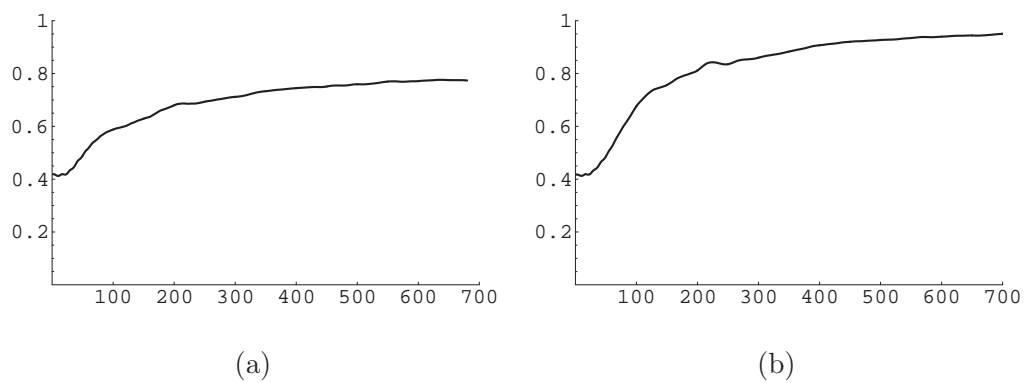


Figura 4.6: Indici di dispersione  $\sigma(t)$  calcolati con il metodo *LB*.

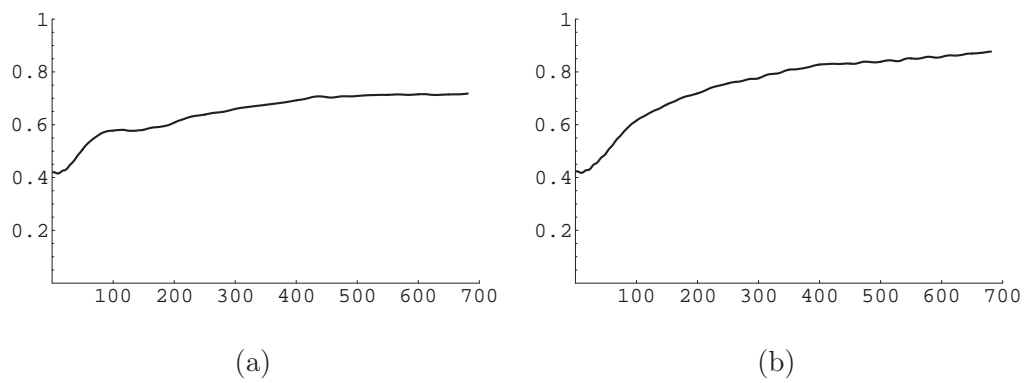


Figura 4.7: Indici di dispersione  $\sigma(t)$  calcolati con il metodo agli elementi finiti.

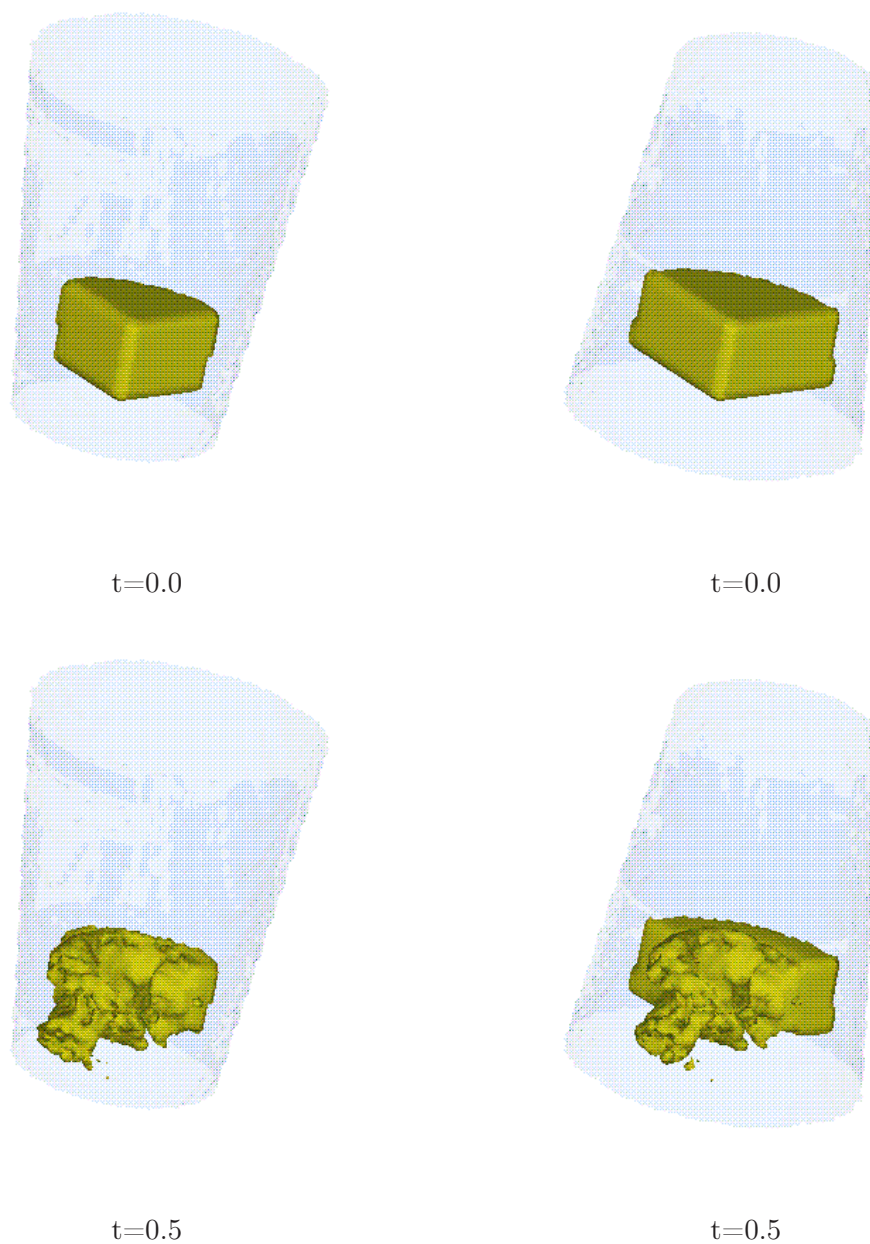


Figura 4.8: Isosuperfici nella miscelazione con il metodo  $LB$ , con  $\delta = 0.61$ .



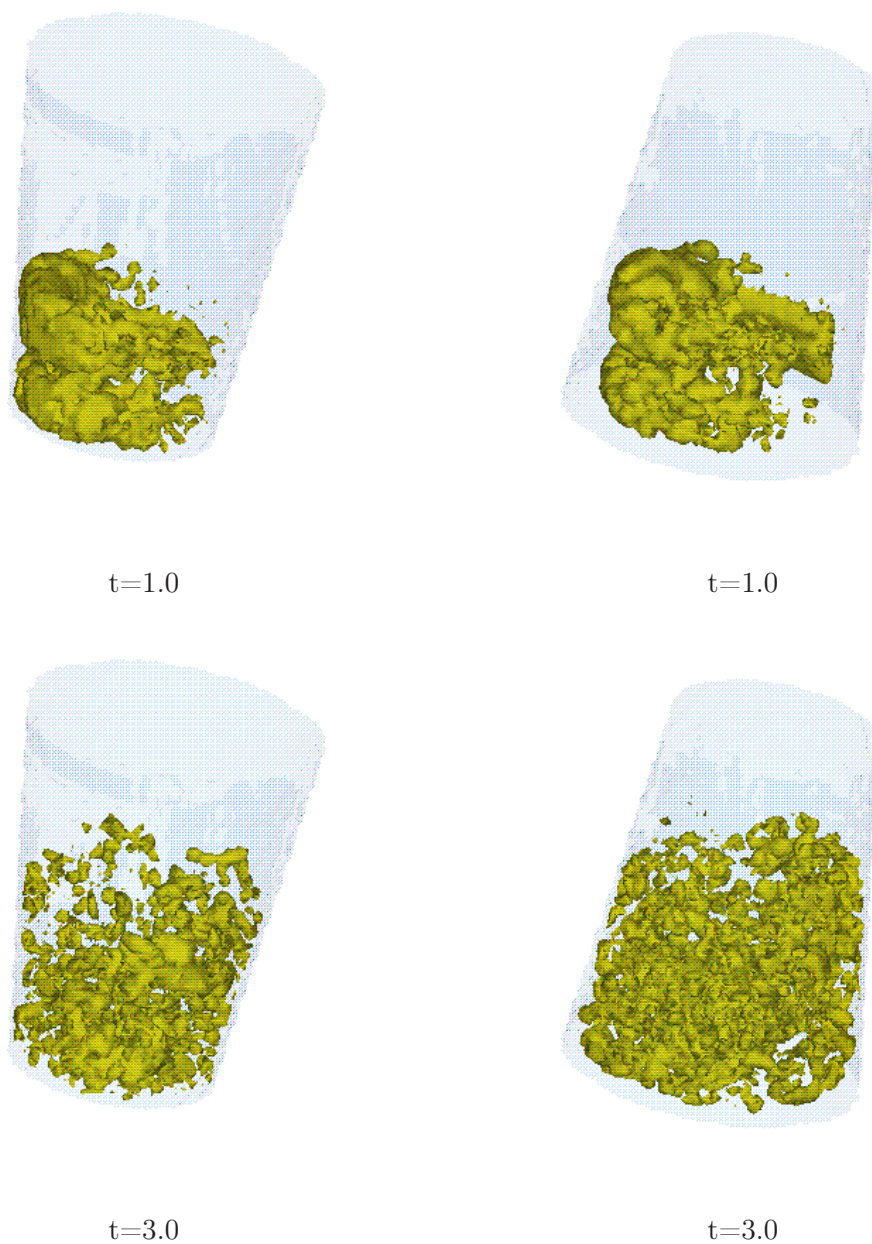


Figura 4.9: Isosuperfici nella miscelazione con il metodo  $LB$ , con  $\delta = 0.61$ .





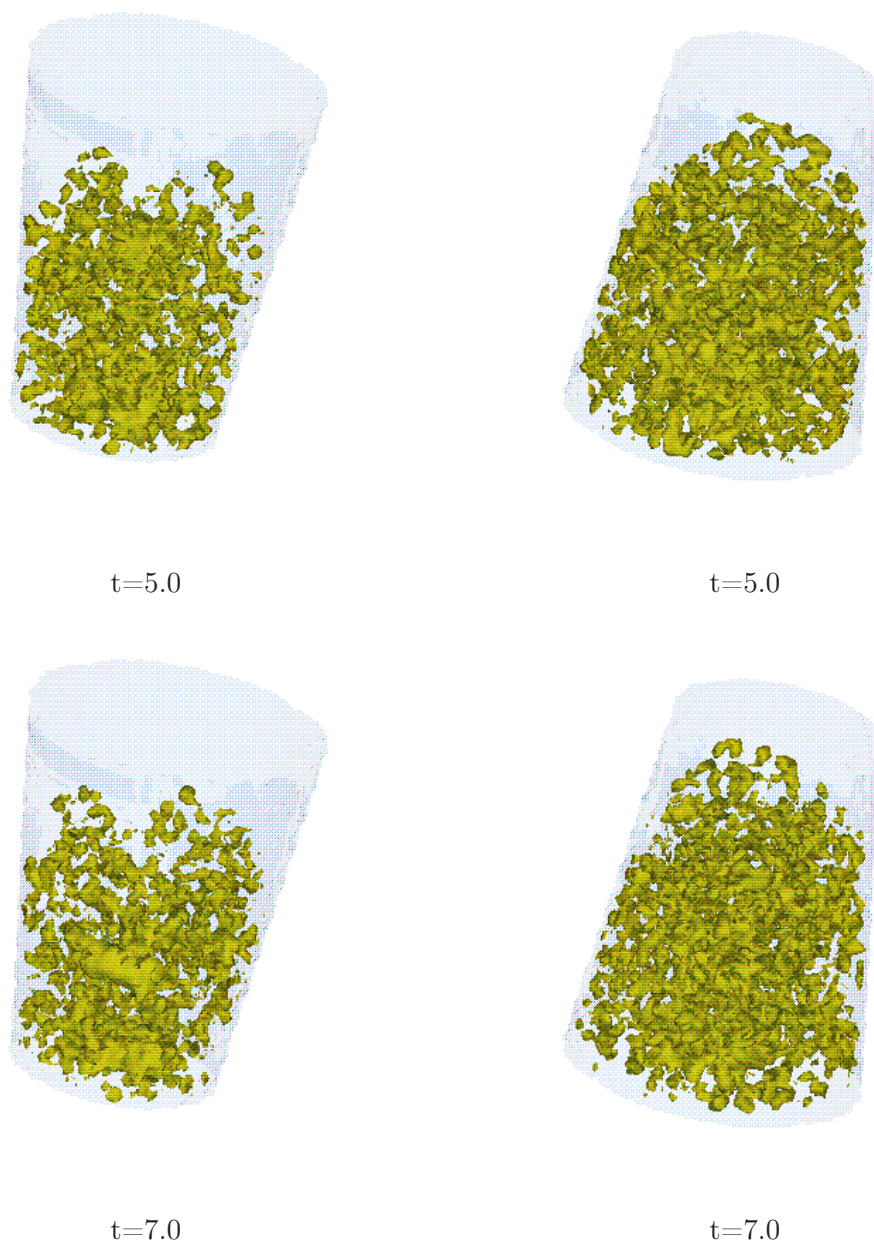


Figura 4.10: Isosuperfici nella miscelazione con il metodo  $LB$ , con  $\delta = 0.61$ .



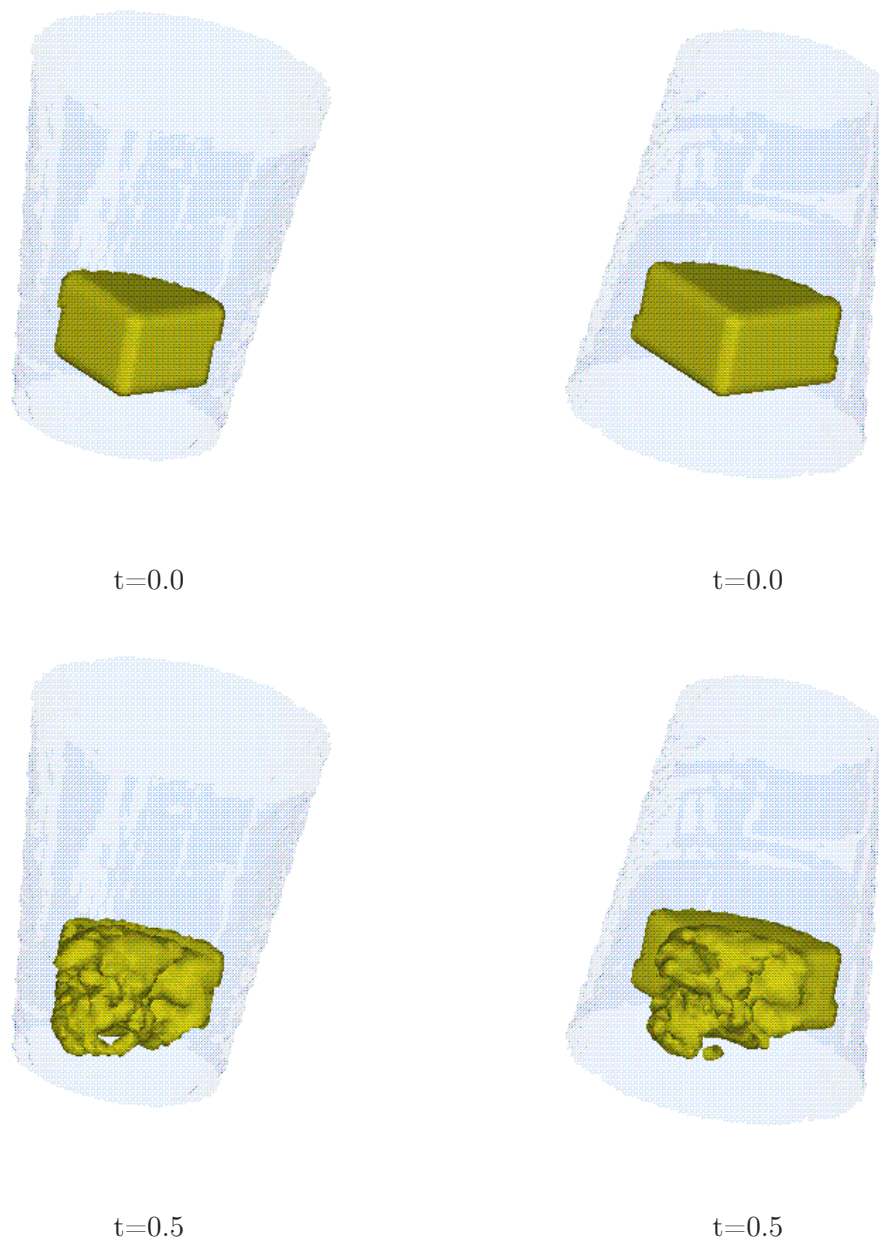


Figura 4.11: Isosuperfici nella miscelazione con il metodo agli elementi finiti, con  $\delta = 0.61$ .



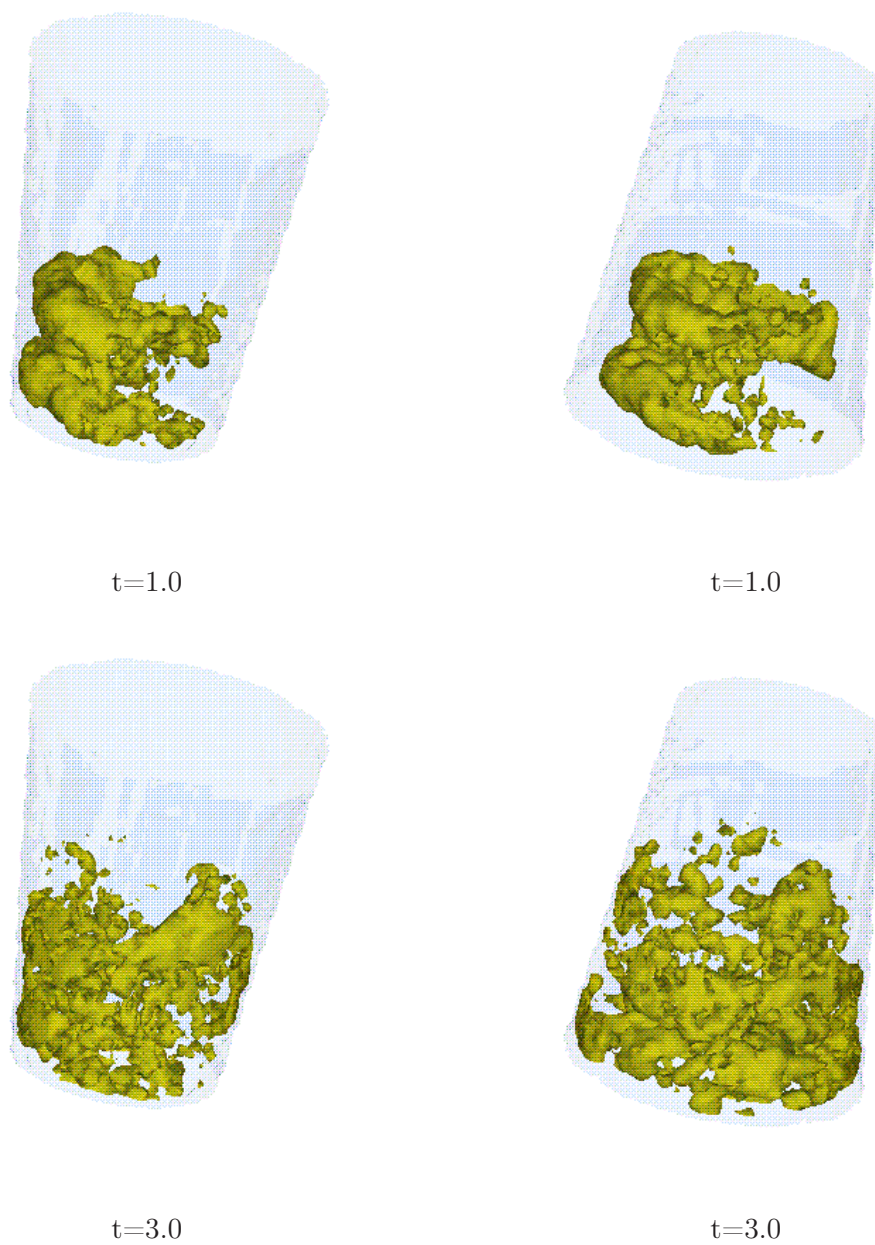


Figura 4.12: Isosuperfici nella miscelazione con il metodo agli elementi finiti, con  $\delta = 0.61$ .



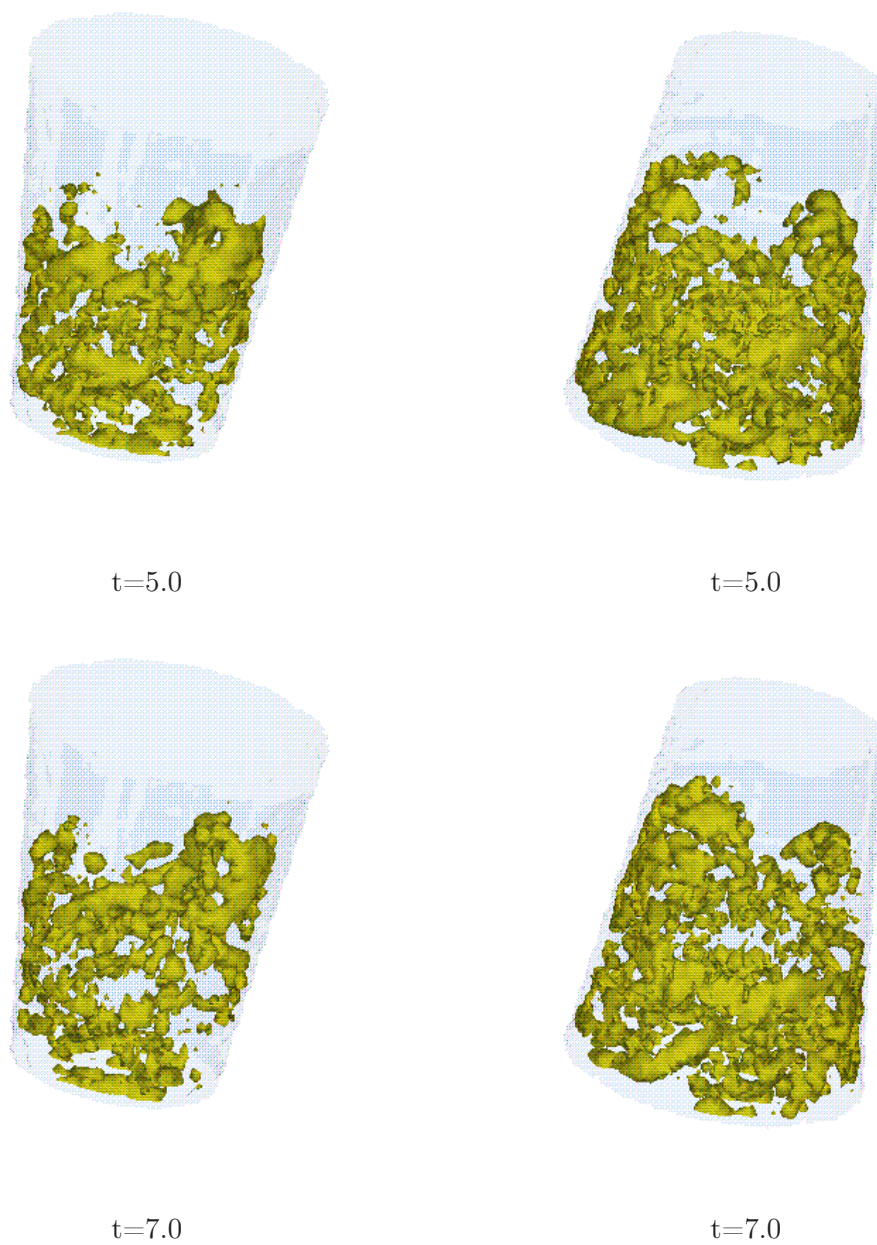


Figura 4.13: Isosuperfici nella miscelazione con il metodo agli elementi finiti, con  $\delta = 0.61$ .





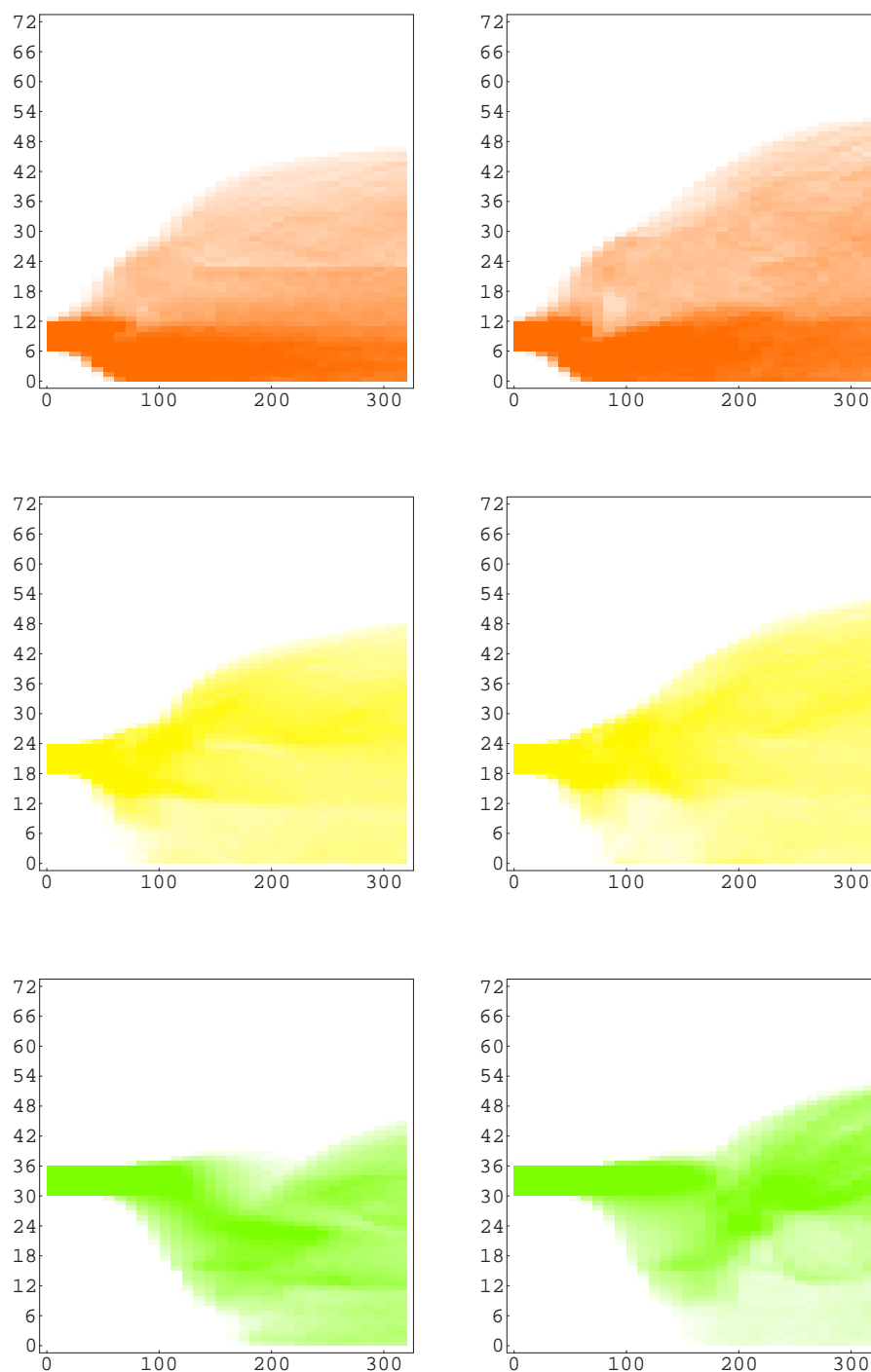


Figura 4.14: Indici di diffusione calcolati con il metodo  $LB$ , per il bicchiere dritto e rovesciato, rispettivamente.



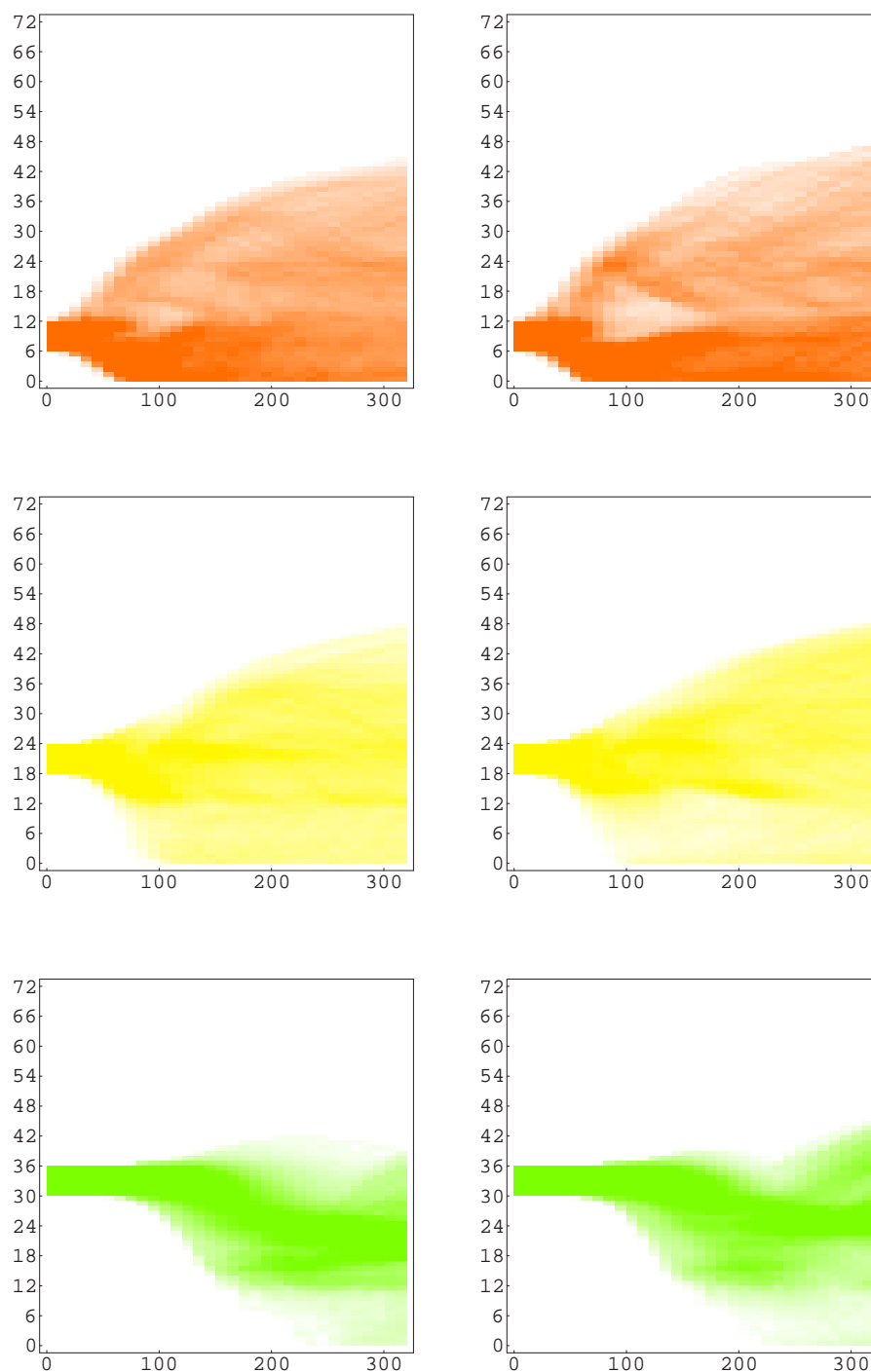


Figura 4.15: Indici di diffusione calcolati con il metodo agli elementi finiti, per il bicchiere dritto e rovesciato, rispettivamente.



## 4.5 Condizioni al contorno in presenza di bordi curvi

Nel precedente paragrafo, che descrive l'applicazione del modello *LB* nel caso particolare del problema considerato delle turbine rotanti all'interno di un contenitore, abbiamo visto come per l'implementazione del modello stesso sui nodi facenti parte del bordo, considerato come superficie solida, siano state utilizzate delle condizioni di tipo *bounce-back*, come usualmente si fa per tale metodo. Queste non fanno altro che invertire il verso del momento della particella che in quell'istante si trova a collidere con tale superficie, mantenendone invariata la direzione ed il modulo.

Tutti i nostri modelli sono stati, quindi, risolti con l'utilizzo di tale regola, che è risultata essere sufficientemente esatta. Tuttavia, secondo un'analisi più dettagliata, le condizioni al bordo di tipo *bounce-back* risultano avere una convergenza con ordine due. Questa accuratezza, oltretutto, la si può ottenere solo se si pensa che tale superficie sia posta, in maniera fittizia, proprio equidistante rispetto a due nodi vicini.

Questo fatto, quindi, impedisce, da un certo punto di vista, l'implementazione del modello *LB* sulla superficie di delimitazione, attraverso la regola *bounce-back*, se questa risulta possedere una certa curvatura.

Allo scopo di superare tale difficoltà, *Mei e Shyy*<sup>16</sup> (cfr. [13]) hanno cercato di risolvere l'*equazione di Boltzmann* in coordinate cilindriche utilizzando un metodo alle differenze finite nel momento in cui si trattava di risolvere l'e-

---

<sup>16</sup>Citiamo questo nuovo approccio solo come un'approfondimento teorico e come itazione, in quanto all'interno del programma `sim` le condizioni al bordo sono state implementate attraverso le regole di tipo *bounce-back*.

quazione che definisce  $\tilde{F}_\alpha$ . Questa quantità, che è la distribuzione che si ottiene applicando l'equazione (3.7) per ottenere l'evoluzione del sistema, è calcolata prima di effettuare la propagazione delle particelle lungo i collegamenti della griglia.

Attraverso una formulazione matematica più precisa, questo si può esprimere come:

$$\begin{aligned} \text{step - collisioni :} \quad & \tilde{F}_i(\mathbf{x}, t) = F_i(\mathbf{x}, t) - \omega [F_i(\mathbf{x}, t) - F_i^{(eq)}(\mathbf{x}, t)], \\ \text{step - propagazione :} \quad & F_i(\mathbf{x} + \mathbf{c}_i \Delta t, t + \Delta t) = \tilde{F}_i(\mathbf{x}, t) \end{aligned}$$

dove  $F_i$  è la distribuzione che ci definisce il momento locale posseduto dalla particella nel nodo  $\mathbf{x}$  e lungo il collegamento  $i$ -esimo,  $i = 0, \dots, b$ .

Questo si può attuare utilizzando delle *coordinate curvilinee adattate*, con un'interpolazione su tutti i nodi facenti parte della 'mesh', escludendo soltanto quelli del bordo ai quali vengono applicate le regole di collisione *bounce-back*.

In lavori che si riferiscono a studi più recenti, sono state utilizzate delle coordinate cartesiane per un'interpolazione applicata ai soli nodi considerati come facenti parte del bordo. Quindi i discorsi e le regole che di seguito verranno esposte, sono riferite a questa seconda formulazione.

Riferendoci alla Figura (4.16), che risulta rappresentare una proiezione nella seconda dimensione di una parte di bordo tri-dimensionale, possiamo dire che la prima fase del modello *LB*, cioè quella della propagazione, richiede la conoscenza di  $\tilde{F}_i(\mathbf{x}_B, t)$ , dove  $\mathbf{x}_B$  è la posizione di un nodo facente parte del bordo, sopra definita, nella quale  $\mathbf{c}_i = -\mathbf{c}_i$ , nel punto  $B$  di bordo, in modo da poter calcolare ed ottenere  $F_i(\mathbf{x}_f, \Delta t)$ , calcolata nel nodo della griglia  $f$  facente parte della zona del fluido, per il quale la posizione si calcola come

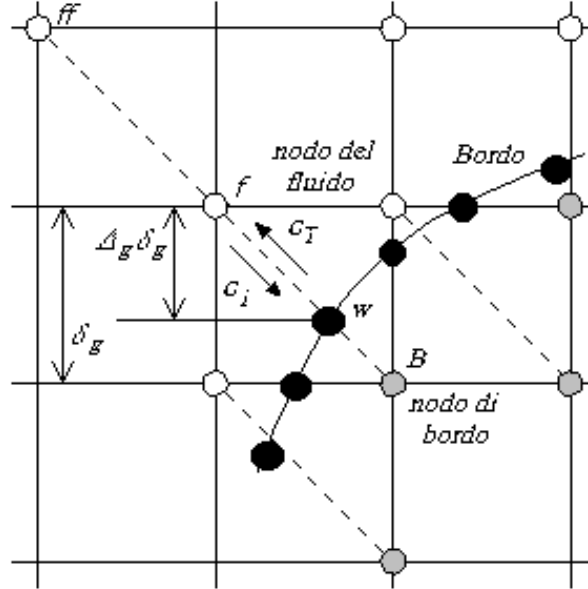


Figura 4.16: Proiezione bi-dimensionale di un bordo regolare curvo tri-dimensionale.

$$\mathbf{x}_f = \mathbf{x}_B + \mathbf{c}_i \Delta t.$$

Ora definiamo:

$$\Delta = \frac{|\mathbf{x}_f - \mathbf{x}_w|}{|\mathbf{x}_f - \mathbf{x}_B|}$$

che risulta essere una sezione di un collegamento fra due nodi.

Risulta essere chiaro, anche dal grafico, che  $\Delta$  soddisfa la relazione  $0 \leq \Delta \leq 1$  ed inoltre la distanza verticale fra due nodi, la cui posizione è data da  $\mathbf{x}_B$  ed  $\mathbf{x}_w$  la si può ottenere come  $(1 - \Delta) \delta$  nella griglia tri-dimensionale.

Le equazioni che ci permettono di implementare tutta la formulazione effettuata da *Mei et al.* sulle *bordature curve*, sono le seguenti:

$$\tilde{F}_i(\mathbf{x}_B, t) = (1 - \chi) \tilde{F}_i(\mathbf{x}_f, t) + \chi F_i^{(*)}(\mathbf{x}_f, t) + 2W(i) \rho \frac{3}{c^2} \mathbf{c}_i \cdot \mathbf{u}_w \quad (4.15)$$

con

$$F_i^{(*)}(\mathbf{x}_B, t) = W(i) \rho(\mathbf{x}_f, t) \left[ 1 + \frac{3}{c^2} \mathbf{c}_i \cdot \mathbf{u}_{Bf} + \frac{9}{2c^4} (\mathbf{c}_i \cdot \mathbf{u}_f)^2 - \frac{3}{2c^2} \mathbf{u}_f \cdot \mathbf{u}_f \right]$$

e

$$\text{se } \Delta \geq 1/2 \quad \begin{cases} \mathbf{u}_{Bf} = \frac{(\Delta - 1) \mathbf{u}_f}{\Delta} + \frac{\mathbf{u}_w}{\Delta} \\ \chi = \frac{(2\Delta - 1)}{\tau} \end{cases}$$

invece

$$\text{se } \Delta < 1/2 \quad \begin{cases} \mathbf{u}_{Bf} = \mathbf{u}_{ff} = \mathbf{u}_f(\mathbf{x}_f + \mathbf{e}_{\bar{\alpha}} \delta t) \\ \chi = \frac{(2\Delta - 1)}{(\tau - 2)} \end{cases}$$

dove  $\tau = \frac{1}{\omega}$ .

Concludiamo il paragrafo, e con esso il capitolo, osservando che le scelte sopra effettuate, in particolare quella fatta per  $\mathbf{u}_{Bf}$ , migliorano la stabilità computazionale nel caso in cui si considerano  $\tau < 1$  e  $\Delta < 1/2$ .

Inoltre il fatto che le equazioni, sopra proposte, sono formulate in maniera vettoriale, rende la loro estensione ad un bordo tri-dimensionale, estremamente semplice.



## Capitolo 5

# *Fluidi binari e Gravità nei modelli $LBM$*

### 5.1 *LBM per fluidi binari (binary-fluid)*

Cerchiamo, in questa parte, di generalizzare il più possibile, l'uso che si può fare del modello *LBM* nella simulazione della dinamica dei fluidi. Un primo passo può essere effettuato considerando un sistema nel quale si hanno due masse distinte.

Se volessimo dare una caratterizzazione cromatica, potremmo dire di trovarci in presenza di un sistema costituito per una parte da particelle *rosse* e per l'altra da particelle *blu*.

Esaminiamo, in particolare, due situazioni:

- La prima consiste nel supporre che i due tipi di fluidi abbiano caratteristiche completamente diverse fra di loro, ovvero sono *immiscibili*;

- La seconda, invece, nell'ipotesi di trovarci in presenza di particelle di due fluidi, con caratteristiche similari, e quindi *miscibili*.

### 5.1.1 LBM per fluidi immiscibili

Il modello che abbiamo costruito per la simulazione di fluidi immiscibili, si basa, essenzialmente, sulla netta suddivisione delle particelle considerate, e quindi della massa totale, in due classi ben distinte. A seguire, vedremo come le equazioni che ci permettono di applicare il metodo e, quindi, di far evolvere il sistema, sono modificate in maniera tale da ottenere, nella zona di separazione fra i due fluidi, un termine aggiuntivo dipendente dalla tensione superficiale, il quale porta ad una redistribuzione delle masse, che rimangono, comunque, separate e ben distinte.

A tale scopo, si delinea un modello, sulle tracce di quello proposto da *Zaleski, Zanetti e Rothman* (cfr. [1]), che si sviluppa, per quanto riguarda la fase della collisione, attraverso due momenti:

- '*Single-Phase collision*', ovvero la classica collisione considerata fino a questo momento, che avviene nelle zone dove si trovano particelle di un unico fluido;
- '*Two-Phase collision*', della quale ora ci occuperemo, che viene applicata ai punti considerati come facenti parte dell'interfaccia.

L'algoritmo può essere sintetizzato come segue:

**Step 1.**

*Si calcola la funzione, che definisce la distribuzione della massa locale, ovvero quella presente in ogni nodo della griglia, come:  $F_i = R_i + B_i$ ;*

**Step 2.**

*Alla distribuzione  $F_i$  è applicata la "collisione", attraverso l'equazione (3.20), ma non la fase di propagazione;*

**Step 3.**

*In ogni nodo in cui  $\|\mathbf{f}\| > \epsilon$ , ovvero dove il Gradiente Colore <sup>1</sup> è non trascurabile, la distribuzione  $F'_i(\mathbf{x}, t) = F_i(\mathbf{x} + \mathbf{c}_i, t + 1)$  viene perturbata in una ulteriore distribuzione  $F''_i$ , che ci permette di controllare il moto delle particelle rosse e di quelle blu nella zona di separazione.*

*Viceversa, se ci troviamo nella situazione per cui  $\|\mathbf{f}\| < \epsilon$ , cioè all'interno di un unico fluido, allora tutto rimane invariato, basta, infatti, porre  $F''_i = F'_i$ ;*

**Nota 5.1.**

*La condizione sul Gradiente Colore,  $\mathbf{f}$ , ci permette di determinare quali siano i nodi facenti parte della superficie di separazione fra i due fluidi e quali no. Questo risultato è molto importante, in quanto ci permette di ottimizzare il programma.*

---

<sup>1</sup>La definizione del *Gradiente Colore* sarà data in maniera chiara e precisa, nel seguito della trattazione.

**Step 4.**

A questo punto avviene il calcolo della redistribuzione dei due fluidi, rosso e blu, all'interno dei nodi. Questo risultato lo si raggiunge mediante la risoluzione di una massimizzazione, che ci permette di ottenere le nuove distribuzioni  $R_i''$  e  $B_i''$  in ogni nodo della griglia;

**Step 5.**

Avviene la fase di propagazione per le due nuove distribuzioni calcolate;

**Step 6.**

Il processo si conclude nel momento in cui si calcolano le distribuzioni dei due colori nel nodo, al time-step successivo, ovvero i valori delle celle vengono aggiornati.

Questo lo si ottiene attraverso la posizione:

$$R_i(\mathbf{x} + \mathbf{c}_i, t + 1) = R_i''(\mathbf{x}, t)$$

$$B_i(\mathbf{x} + \mathbf{c}_i, t + 1) = B_i''(\mathbf{x}, t)$$

Solo a questo punto si può affermare di avere terminato la fase della collisione e, quindi, una singola iterata del metodo.

Notiamo che la prima delle due fasi, in cui è stata suddivisa la collisione, ovvero la “*single-phase collision*” è del tutto analoga al metodo trattato fino ad ora, per questo motivo si ritiene opportuno non dedicarle ulteriore spazio, in quanto già descritta in maniera sufficientemente estesa, e focalizzare la nostra attenzione alla seconda.

La ‘*Two-Phase collision*’ può, a sua volta, essere suddivisa in due momenti: *perturbazione e redistribuzione* (o *ricolorazione*).

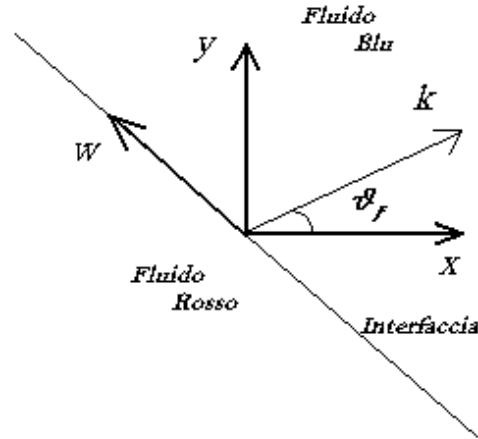


Figura 5.1: Geometria dell'Interfaccia nell'Analisi della regola delle collisioni, dove  $k$  è l'asse perpendicolare al piano dell'interfaccia,  $w$  è ad esso parallelo, mentre  $x$  e  $y$  sono gli usuali assi cartesiani.

***Two-phase collision: perturbazione.***

La prima fase consiste nella perturbazione dell'equazione di aggiornamento, attraverso un termine aggiuntivo, nel momento in cui si considera un nodo vicino all'interfaccia.

Questo è possibile utilizzando la definizione (che ora daremo) del *gradiente colore*,  $\mathbf{f}(\mathbf{x})$ , sopra citato:

$$\mathbf{f}(\mathbf{x}) := \sum_{i=1}^b \mathbf{c}_i \left[ [R_0(\mathbf{x} + \mathbf{c}_i) - B_0(\mathbf{x} + \mathbf{c}_i)] + \sum_{j=1}^b [R_j(\mathbf{x} + \mathbf{c}_i) - B_j(\mathbf{x} + \mathbf{c}_i)] \right], \quad (5.1)$$

dove  $R_i$  e  $B_i$  indicano le componenti del fluido rosso e blu, rispettivamente, in ciascun nodo, nella direzione  $\mathbf{c}_i$ .

Notiamo che, il valore assunto dalla quantità  $\| \mathbf{f} \|$  ci dice se la particella presa in considerazione si trova in un nodo facente parte della zona di separazione fra i due fluidi, o no.

Quindi possiamo intuire l'importanza che assume tale termine nel corso della trattazione.

Altrettanta importanza può essere attribuita a:

$$\theta_f = \tan^{-1} \left( \frac{f_y}{f_x} \right),$$

ovvero l'angolo del *gradiente locale* (come rappresentato in Figura (5.1)).

A questo punto iniziamo, come già mostrato, il metodo, con la posizione  $F_i = R_i + B_i$  ed il calcolo dell'operatore delle collisioni linearizzato

$$F'_i(\mathbf{x}, t) = F_i(\mathbf{x} + \mathbf{c}_i, t + 1) = F_i - \omega (F_i - F_i^{(eq)}).$$

Ora utilizziamo la definizione data nella (5.1), poichè la vera e propria fase perturbativa ha luogo solo nei nodi in cui  $\| \mathbf{f} \| > \epsilon$ .

In tali zone si calcola una variazione rispetto alla  $F'_i$ , in modo da poter controllare le quantità  $R_i$  e  $B_i$ .

Si definisce, a questo punto,  $F''_i$  in maniera tale che:

$$\begin{aligned} F''_i(\mathbf{x}, t) &= F'_i(\mathbf{x}, t) + A \| \mathbf{f} \| \cos [2(\theta_i - \theta_f)] \\ &\equiv F'_i(\mathbf{x}, t) + C_i^b \end{aligned} \quad (5.2)$$

dove  $\theta_i = (i - 1) 2 \pi / b$  è l'angolo formato dal vettore velocità della griglia  $\mathbf{c}_i$  con l'asse  $x$ , mentre  $C_i^b$  è la tensione superficiale indotta dalla perturbazione dei due fluidi. Questa è determinabile a partire dalla costante  $A$  che da esplicitamente la tensione superficiale dell'intero sistema.

Il valore che assume il termine  $C_i^b$  è molto importante in quanto ci permette di redistribuire le diverse masse lungo i nodi facenti parte della superficie di separazione, togliendo delle particelle di un tipo da un ‘link’ ed aggiungendone su un altro.

Si può osservare che il metodo si sviluppa in maniera tale da avere sempre rispettate le condizioni di conservazione della densità di massa e del momento, tramite l’uso delle equazioni (2.13) e (2.14). Questo lo si ottiene mediante una semplice somma, su tutti gli  $\mathbf{c}_i$ , nella (5.2).

Possiamo, così concludere dicendo che il metodo è consistente, cioè soddisfa, anche in questo caso, le equazioni che regolano la fluido-dinamica, ovvero l’equazione di continuità e quella di Navier-Stokes, (2.63).

### *Two-phase collision: ricolorazione.*

Dopo aver stimato i valori perturbati per le distribuzioni globali dei due fluidi, dobbiamo valutare in che modo siano cambiate le quantità  $R_i$  e  $B_i$  al passo successivo.

Questo si consegue assegnando loro il valore della soluzione del seguente *problema di massimizzazione*:

$$M \left( R'_i, B'_i \right) = \max_{R'', B''} \left[ \sum_{i=1}^b \left( R''_i - B''_i \right) \mathbf{c}_i \right] \cdot \mathbf{f} \quad (5.3)$$

soggetto alle condizioni:

$$R_0 + \sum_{i=1}^b R''_i = R_T \quad (5.4)$$

$$R''_i + B''_i = F''_i \quad (5.5)$$

dove  $R_T$  è la somma totale della massa rossa presente in ogni nodo della griglia.

Queste condizioni permettono di avere ancora la conservazione locale della massa rossa, con la (5.4), e conseguentemente di quella blu, con la (5.5), e fanno sì che il problema di massimizzazione (5.3) sia ben posto, cioè ammetta soluzione unica.

### 5.1.2 *LBM* per fluidi miscibili

Il secondo caso che andiamo a considerare è il modello *LB* che viene utilizzato nelle simulazioni in presenza di due fluidi miscibili.

Tale metodo è stato proposto da Flekkøy (cfr. [8]) e noi ne riportiamo le caratteristiche principali per completezza, nonostante nelle nostre simulazioni questo non sia stato implementato.

In questo caso, ciò che si segue durante la simulazione non è solamente la massa presente su ciascun collegamento che si ha fra i nodi, ma si considera anche la variazione della stessa.

Se, quindi, supponiamo di avere la presenza di due fluidi, che caratterizziamo ancora come *rosso* e *blu*, le quantità che ci permettono di controllare il moto delle particelle dei due colori nel tempo sono:

$$\begin{aligned}\rho &= \rho_r + \rho_b \\ \Delta\rho &= \rho_r - \rho_b\end{aligned}$$

A questo punto, sono quindi necessarie due ben distinte *equazioni di Boltzmann*, ciascuna delle quali ci permetta di poter controllare l'evoluzione, rispet-



tivamente, della densità di massa e della sua variazione. Si ha allora:

$$F_i(\mathbf{x} + \mathbf{c}_i, t + 1) - F_i(\mathbf{x}, t) = -\omega_\nu \left[ F_i(\mathbf{x}, t) - F_i^{(eq)}(\mathbf{x}, t) \right]$$

$$\Delta_i(\mathbf{x} + \mathbf{c}_i, t + 1) - \Delta_i(\mathbf{x}, t) = -\omega_D \left[ \Delta_i(\mathbf{x}, t) - \Delta_i^{(eq)}(\mathbf{x}, t) \right]$$

dove  $\omega_\nu$  e  $\omega_D$  sono, rispettivamente, i parametri di rilassamento che possono essere determinati attraverso il termine che definisce la viscosità, come nella (5.7), ed il coefficiente di diffusione, come definito nella (5.6).

Se ora, invece, supponiamo di metterci nel caso particolare di una griglia con simmetria esagonale<sup>2</sup>, siamo in grado di determinare esplicitamente il valore delle due distribuzioni di equilibrio<sup>3</sup>

$$F_i^{(eq)} = \frac{\rho}{6} (1 + u_\alpha c_{i\alpha} + G Q_{i\alpha\beta} u_\alpha u_\beta)$$

$$\Delta_i^{(eq)} = \frac{\rho}{6} (i + u_\alpha c_{i\alpha})$$

dove  $G$  è il termine di invarianza definito nella (2.22).

Inoltre la densità  $\rho$ , il momento  $\rho u_\alpha$  e la variazione di densità  $\Delta \rho$  sono definiti come:

$$\rho = \sum_i F_i, \quad \rho u_\alpha = \sum_i F_i c_{i\alpha}, \quad \Delta \rho = \sum_i \Delta_i.$$

Il tensore  $Q_{i\alpha\beta}$  è, invece, definito tramite la relazione:

$$Q_{i\alpha\beta} = c_{i\alpha} c_{i\beta} - c_s^2 \delta_{\alpha\beta},$$

<sup>2</sup>Il modello, in questo caso, è descritto per una griglia esagonale, solo per semplicità nei calcoli e nell'esposizione. Tuttavia, la generalizzazione a situazioni più complesse è quasi immediata.

<sup>3</sup>Come sappiamo, ma lo ripetiamo per completezza, la distribuzione di equilibrio può essere determinata solo una volta nota la griglia sulla quale si intende implementare il modello.

analoga alla (2.23), dove  $c_s$  è la *sound-speed*.

Usando queste relazioni, il modello soddisfa l'*equazione di diffusione*

$$(\partial_t + u_\alpha c_{i\alpha}) \Delta \rho = D \left( \partial_\alpha \partial_\alpha \Delta \rho - \frac{1}{c_s^2} u_\alpha u_\beta \partial_\alpha \partial_\beta \Delta \rho \right)$$

e l'*equazione di Navier-Stokes*

$$\partial_t u_\alpha + u_\beta \partial_\beta u_\alpha = -\frac{1}{\rho} \partial_\alpha p + \nu \partial_\beta \partial_\beta u_\alpha,$$

dove il coefficiente di diffusione  $D$ , la viscosità  $\nu$  e la pressione  $p$  sono calcolabili, rispettivamente, attraverso le seguenti relazioni:

$$D = -c_s^2 \left( \frac{1}{\omega_D} + \frac{1}{2} \right) \quad (5.6)$$

$$\nu = -\frac{1}{4} \left( \frac{1}{\omega_\nu} + \frac{1}{2} \right) \quad (5.7)$$

$$p = \frac{1}{2} \rho (1 - |\mathbf{u}|^2) \quad (5.8)$$

## 5.2 Gravità nel *modello di Boltzmann*

In questo paragrafo cercheremo di illustrare diversi metodi (cfr. [10]), anche già incontrati, allo scopo di poter introdurre all'interno del modello *LB* un termine atto a simulare la presenza di una forza agente sul sistema.

Si noterà che, nel momento in cui un termine aggiuntivo verrà introdotto nell'equazione di evoluzione in un modello isotropo, la forza agente su di esso, risulterà essere indipendente dall'orientazione della griglia stessa.

Notiamo, oltretutto, che si intende affrontare questo discorso in quanto nella maggior parte delle simulazioni numeriche la forza, sia essa di gravità oppure no, gioca un ruolo particolarmente importante nell'evoluzione del sistema.

### 5.2.1 La classica equazione di Boltzmann

Già nel terzo capitolo, nel momento in cui è stata affrontata la *multi-scale analysis* per il modello *LB* bi-dimensionale, si è potuto notare come se nell'equazione di Boltzmann si tiene conto anche del termine dissipativo, ovvero si considera:

$$\partial_t f + \mathbf{v} \partial_x f + \frac{\mathbf{K}}{m} \partial_v f = Q(f, f) \quad (5.9)$$

(dove  $\mathbf{K}$  è la forza applicata), si ottiene nell'equazione un valore aggiuntivo che ci permette di ottenere l'evoluzione del sistema, data dalla (3.20), considerando tutte le quantità che in esso entrano in gioco.

Il termine della forza risulta essere completamente indipendente sia dalla distribuzione di equilibrio, sia dalla distribuzione che determina i momenti della griglia lungo le diverse direzioni del lattice.

L'equazione (3.20) risulta essere<sup>4</sup>:

$$\begin{aligned} F_i(\mathbf{x} + \mathbf{c}_i \Delta t, t + \Delta t) &= F_i(\mathbf{x}, t) - \omega \left( F_i(\mathbf{x}, t) - F_i^{(0)}(\mathbf{x}, t) \right) \\ &+ \frac{\Delta t c_{i\alpha}}{12 c^2} [K_\alpha(\mathbf{x}, t) + K_\alpha(\mathbf{x} + \mathbf{c}_i \Delta t, t + \Delta t)] \end{aligned} \quad (5.10)$$

Notiamo, ulteriormente, che se la forza di cui si vuole tenere conto è quella di gravità, basta andare ad aggiungere tale termine sulle direzioni lungo le quali la forza agisce, indipendentemente dall'orientazione della griglia. In particolare se la griglia è orientata secondo le classiche coordinate cartesiane, gli unici vettori le cui componenti vengono modificate, sono quelli verticali.

Quindi, se si fa riferimento alla classica tecnica di aggiornamento del modello

<sup>4</sup> $K_\alpha$  è la componente cartesiana della forza applicata  $\mathbf{K}$ .

LB, la considerazione e l'introduzione del termine comprendente la forza, è un passo pressochè immediato. Questo fatto, sottolinea ancora una volta la grande versatilità dei modelli discussi nel corso di questa tesi, ovvero dei LBM.

### 5.2.2 Combinazione del termine della gravità e quello della pressione

Cominciamo la trattazione, osservando che la classica *equazione di Navier-Stokes*

$$\frac{\partial \mathbf{u}}{\partial t} = -\frac{1}{\rho} \nabla p + \nu \nabla^2 \mathbf{u} - \zeta (\mathbf{u} \cdot \nabla) \mathbf{u} + \frac{\mathbf{K}}{\rho} \quad (5.11)$$

può essere riscritta come

$$\frac{\partial \mathbf{u}}{\partial t} = -\frac{1}{\rho_0} \nabla P + \nu \nabla^2 \mathbf{u} - \zeta (\mathbf{u} \cdot \nabla) \mathbf{u} + \frac{\delta \rho}{\rho_0} \mathbf{g} \quad (5.12)$$

dove si considera  $P = p + \rho_0 \phi$ ,  $\mathbf{K} = \rho \mathbf{g}$ ,  $\mathbf{g} = -\nabla \phi$  ed infine  $\rho = \rho_0 + \delta \rho$ .

Osserviamo che l'equazione (5.12) è analoga alla (5.11), solo che, invece di considerare la forza agente  $\mathbf{K}$ , si considera la variazione della pressione  $p$ , attraverso la funzione  $\phi$ .

Inoltre, notiamo che se si suppone che l'incremento  $\delta \rho = 0$  ciò significa che nessuna forza entra in gioco nel sistema.

Altrimenti, continuando con questo tipo di approccio, possiamo andare a considerare una nuova distribuzione di equilibrio che, in questo caso, al contrario di quanto accadeva nella trattazione classica, vista nel paragrafo precedente, ne rimane modificata.

Infatti, il termine che ci permette di implementare ed introdurre nel modello la forza, non risulta più essere una semplice quantità aggiuntiva, nell'equazione

dell'aggiornamento, come nella (5.10), ma entra a far parte del modello *LB* vero e proprio, mediante la perturbazione della stessa distribuzione di equilibrio.

Definiamo, quindi, tale distribuzione,  $F_i^{(A)}(\rho, \mathbf{u}) = F_i^{(eq)}$ , che potremmo definire 'Alterata', come:

$$F_i^{(A)} = \begin{cases} \rho \left( \frac{1-d_0}{b} + \frac{\phi D}{bc^2} + \frac{D}{bc^2} \mathbf{c}_i \cdot \mathbf{u} + \frac{D(D+2)}{2bc^4} (\mathbf{c}_i \cdot \mathbf{u})^2 - \frac{D\mathbf{u}^2}{2bc^2} \right) & i = 1, \dots, b \\ \rho \left( d_0 - \frac{\phi D}{c^2} - \frac{\mathbf{u}^2}{c^2} \right) & i = 0 \end{cases}$$

dove, come sempre,  $b$  è il numero dei nodi della griglia, mentre  $D$  è la dimensione dello spazio nel quale si sta agendo.

Senza fare tutti i passaggi in maniera esplicita, per non appesantire troppo un discorso che voleva solo essere una citazione teorica, possiamo supporre di utilizzare questa distribuzione di equilibrio nell'*espansione di Chapman-Enskog*, espressa dalla (2.32), ed all'interno dei passaggi che comprendono la *multi-scale analysis*, considerata nel paragrafo (2.3.3), notando che tutti i termini, quali la densità di massa, definita dalla relazione (2.47), il momento del primo e del terzo ordine, come visti nella (2.48), rimangono invariati. L'unico termine che risulta esserne modificato è il momento del secondo ordine, che assume la forma:

$$\sum_i F_i c_{i\alpha} c_{i\beta} = \frac{\rho(1-d_0)c^2}{D} \delta_{\alpha\beta} + \rho u_\alpha u_\beta + \rho \phi \delta_{\alpha\beta}. \quad (5.13)$$

Inoltre, l'ulteriore quantità che ne risulta modificata, è quella espressa dall'equazione (3.35), (naturalmente escludendo il termine della forza) solo nella

parte che include la pressione.

In tutte le relazioni citate basta sostituire, quindi, la (5.13).

In tal modo si ottiene un'equazione di Navier-Stokes analoga alla (5.12) ed alla (5.11), dove la viscosità  $\zeta$  assume il valore:

$$\zeta = \left(\frac{1}{\omega} - \frac{1}{2}\right) \left[ \frac{2c^2}{(D+2)} - \frac{c^2(1-d_0)}{D - \rho\phi\delta_{\alpha\beta}} \right]$$

Questo metodo ha lo svantaggio che bisogna supporre la densità di massa costante, ovvero  $\delta\rho = 0$ , nel momento in cui non si ha l'applicazione di alcuna forza esterna al sistema.

### 5.2.3 Aggiungendo il termine della forza in quello che definisce il momento

Questo modello, suggerito da *Swift et al.*, introduce la forza nel sistema che si considera, attraverso una variazione della relazione che definisce il momento (3.11), mediante l'introduzione di un termine perturbativo

$$\sum_i F_i c_{i\alpha} = \rho u_\alpha - K_\alpha \quad (5.14)$$

La presenza di tale "termine aggiuntivo"  $K_\alpha$ , tuttavia, deve essere presa in considerazione anche nell'equazione di moto ed in tutti i termini nell'*espansione di Chapman-Enskog*, (2.32), oltre che in tutti i passaggi della *multi-scale analysis*, considerati nel paragrafo (3.4).

Questo significa che le modifiche da apportare nei calcoli teorici sono moltissime, in quanto la definizione del termine 'momento' risulta essere costantemente presente in tutto il corso della trattazione.

Inoltre è bene sottolineare il fatto che, questo tipo di approccio può essere

usato con consistenza, solo nel momento in cui  $\mathbf{K}$ , ovvero la forza applicata, è sufficientemente piccola.





Appendice A

*Approfondimenti Teorici*



### A.1 Calcolo Esplicito dei *Moltiplicatori di Lagrange*

In questa Appendice vorremmo dedicare una parte del nostro discorso al calcolo dei *moltiplicatori di Lagrange*, che abbiamo citato nel Paragrafo (2.2.3) nell'ambito del calcolo della distribuzione di equilibrio per il modello *FHP-I* e per quello *HPP*.

Cominciamo tale discorso facendo notare che, nel momento in cui andiamo a considerare velocità molto piccole, che sono approssimativamente nulle, i *Numeri di Occupazione Media*, definiti nella (2.12), risultano essere tutti uguali fra di loro, ovvero vale la relazione:

$$\mathbf{u} = 0 \quad \Longrightarrow \quad N_i = \frac{\rho}{b} = d \quad (\text{A.1})$$

dove abbiamo indicato con  $b$  il numero di collegamenti presenti in ogni nodo (6 nel caso del modello *FHP-I* e 4 per l'*HPP*) mentre con  $d$  la densità di massa locale in ogni vettore del lattice. Da questa, quindi, ricordando la definizione, dei *numeri di occupazione media di equilibrio*, data nel Teorema (2.1) di *Frisch et al.*, possiamo scrivere:

$$N_i(\rho, 0) = \frac{1}{1 + \exp[h(\rho, 0)]} = d \quad (\text{A.2})$$

e

$$\mathbf{q}(\rho, 0) = 0. \quad (\text{A.3})$$

Ora, dall'invarianza dei numeri di occupazione  $N_i$  rispetto a trasformazioni pari, possiamo scrivere che valgono le seguenti:

$$\mathbf{u} \longrightarrow -\mathbf{u}, \quad \mathbf{c}_i \longrightarrow -\mathbf{c}_i \quad (\text{A.4})$$

e da ciò segue che

$$h(\rho, -\mathbf{u}) = h(\rho, \mathbf{u}) \quad (\text{A.5})$$

e

$$\mathbf{q}(\rho, -\mathbf{u}) = \mathbf{q}(\rho, \mathbf{u}) \quad (\text{A.6})$$

Per ottenere il valore esplicito dei moltiplicatori di Lagrange  $h$  e  $\mathbf{q}$ , ne calcoliamo l'espansione, fino al secondo ordine, rispetto alla velocità  $\mathbf{u}$ :

$$h(\rho, \mathbf{u}) = h_0(\rho) + h_2(\rho) \mathbf{u}^2 + \mathcal{O}(\mathbf{u}^4) \quad (\text{A.7})$$

$$\mathbf{q}(\rho, \mathbf{u}) = q_1(\rho) \mathbf{u} + \mathcal{O}(\mathbf{u}^3) \quad (\text{A.8})$$

Notiamo che nell'espansione di  $h$  sono presenti solo gli ordini pari, rispetto alla  $\mathbf{u}$ , mentre in quello di  $\mathbf{q}$  solo quelli dispari. Questo fatto dipende dalla condizione di parità soddisfatta dai numeri di occupazione media ed espressa dalla (A.4).

Introduciamo, a questo punto, i valori ottenuti nelle (A.7)-(A.8) nell'espansione della distribuzione di *Fermi-Dirac* (2.18), calcolata nel Teorema sopra citato, fino al secondo ordine in  $\mathbf{u} = (u, v)$ , in un opportuno intorno del punto  $\mathbf{u} = 0$ :

$$\begin{aligned} N_i(\mathbf{u}) &= N_i(\mathbf{u} = 0) + \frac{\partial N_i}{\partial u} u + \frac{\partial N_i}{\partial v} v \\ &+ \frac{1}{2} \frac{\partial^2 N_i}{\partial u^2} u^2 + \frac{\partial^2 N_i}{\partial u \partial v} uv + \frac{1}{2} \frac{\partial^2 N_i}{\partial v^2} v^2 + \mathcal{O}(\mathbf{u}^3) \end{aligned} \quad (\text{A.9})$$

Se, quindi, consideriamo

$$\begin{aligned} N_i(\mathbf{u}) &= \frac{1}{1 + \exp[x(\mathbf{u})]}, \\ x(\mathbf{u}) &= h_0 + h_2 \mathbf{u}^2 + q_1 \mathbf{u} \mathbf{c}_i, \\ x(\mathbf{0}) &= h_0, \\ N_i(h_0) &= \frac{1}{1 + \exp[h_0]} = d, \end{aligned}$$

si ottengono tutti i valori necessari al calcolo:

$$\exp[h_0] = \frac{1-d}{d},$$

$$h_0 = \ln\left(\frac{1-d}{d}\right),$$

$$\frac{\partial N_i}{\partial u_\alpha} = \frac{\partial N_i}{\partial x} \frac{\partial x}{\partial u_\alpha} \underset{\mathbf{u}=\mathbf{0}}{\rightarrow} d(d-1) q_1 c_{i\alpha},$$

$$\frac{\partial N_i}{\partial x} = -\frac{\exp[x]}{(1 + \exp[x])^2} \underset{\mathbf{u}=\mathbf{0}}{\rightarrow} \frac{d-1}{d} d^2 = d(d-1)$$

$$\frac{\partial x}{\partial u_\alpha} = 2h_2 u_\alpha + q_1 c_{i\alpha} \rightarrow q_1 c_{i\alpha},$$

Questo per quanto riguarda i termini del primo ordine presenti nella (A.9).

Invece, per quanto concerne i termini dello sviluppo, calcolati fino al secondo

ordine in  $\mathbf{u}$ , si ottengono i valori che seguono :

$$\begin{aligned} \frac{\partial^2 N_i}{\partial u_\alpha^2} &= \frac{\partial}{\partial u_\alpha} \left[ \frac{\partial N_i}{\partial x} \cdot \frac{\partial x}{\partial u_\alpha} \right] \\ &= \frac{\partial^2 N_i}{\partial x \partial u_\alpha} \cdot \left( \frac{\partial x}{\partial u_\alpha} \right)^2 + \frac{\partial N_i}{\partial x} \frac{\partial^2 x}{\partial u_\alpha^2} \\ &\rightarrow d(d-1)(2d-1)q_1^2 c_{i\alpha}^2 + d(d-1)2h_2, \end{aligned}$$

$$\begin{aligned} \frac{\partial^2 N_i}{\partial x^2} &= - \frac{\exp[x](1 + \exp[x])^2 - \exp[x] \cdot (1 + \exp[x]) \exp[x]}{(1 + \exp[x])^4} \\ &= \frac{\exp[x](\exp[x] - 1)}{(1 + \exp[x])^3} \\ &\rightarrow \frac{1-d}{d} \left( \frac{1-d}{d} - 1 \right) d^3 = d(d-1)(2d-1), \end{aligned}$$

$$\frac{\partial^2 x}{\partial u_\alpha^2} = 2h_2.$$

Supponendo  $\alpha \neq \beta$  possiamo anche scrivere :

$$\begin{aligned} \frac{\partial^2 N_i}{\partial u_\alpha \partial u_\beta} &= \frac{\partial}{\partial u_\beta} \left[ \frac{\partial N_i}{\partial x} \cdot \frac{\partial x}{\partial u_\alpha} \right] \\ &= \frac{\partial^2 N_i}{\partial x \partial u_\beta} \cdot \frac{\partial x}{\partial u_\alpha} + \frac{\partial N_i}{\partial x} \frac{\partial^2 x}{\partial u_\alpha \partial u_\beta} \\ &= \frac{\partial^2 N_i}{\partial x^2} \cdot \frac{\partial x}{\partial u_\alpha} \cdot \frac{\partial x}{\partial u_\beta} + \frac{\partial N_i}{\partial x} \cdot \frac{\partial x^2}{\partial u_\alpha \partial u_\beta} \\ &\rightarrow d(d-1)(2d-1)q_1^2 c_{i\alpha} c_{i\beta} \end{aligned}$$

$$\frac{\partial^2 x}{\partial u_\alpha \partial u_\beta} = 0.$$

Abbiamo, in questo modo, calcolato tutti i valori necessari per il calcolo di  $N_i$ , che può ottenersi sostituendo tali quantità nella (A.9):

$$N_i(\mathbf{u}) = d + d(d-1)q_1 c_i \cdot \mathbf{u} + \frac{1}{2}d(d-1)(2d-1)q_1^2 c_{i\alpha}^2 u_\alpha^2 + d(d-1)h_2 \mathbf{u}^2 \quad (\text{A.10})$$

Notiamo che, questa espressione, in forma polinomiale, per i numeri di occupazione media  $N_i$ , risulta essere molto più semplice ed agevole da utilizzare, al contrario di quella che avevamo nella (A.9) con funzioni fratte con funzioni esponenziali al denominatore.

A questo punto, però, il nostro lavoro non risulta essere completamente compiuto, in quanto non è ancora stato effettuato il calcolo dei termini che volevamo, cioè dei *moltiplicatori di Lagrange*  $h_2$  e  $q_1$ . Tale risultato può essere raggiunto, usando le quantità sopra calcolate, ed imponendo le ‘usuali’ condizioni sulla conservazione della densità di massa e del momento:

$$\rho = \sum_i N_i \quad e \quad \mathbf{j} = \rho \mathbf{u} = \sum_i N_i \mathbf{c}_i$$

Dalla prima possiamo ottenere

$$\begin{aligned} \rho &= \sum_i N_i \\ &= \underbrace{\sum_i d}_{=\rho} + \underbrace{\sum_i d(d-1)q_1 c_i \cdot \mathbf{u}}_{=0} \\ &\quad + \underbrace{\sum_i \frac{1}{2}d(d-1)(2d-1)q_1^2 c_{i\alpha}^2 u_\alpha^2}_{=\frac{1}{2}3d(d-1)(2d-1)q_1^2 \mathbf{u}^2} \\ &\quad + \underbrace{\sum_i d(d-1)h_2 \mathbf{u}^2}_{=6d(d-1)h_2 \mathbf{u}^2} \end{aligned}$$

Raccogliendo tutti i termini si ottiene :

$$\begin{aligned}\rho &= \rho + \frac{3}{2}d(d-1)(2d-1)q_1^2 \mathbf{u}^2 \\ &\quad + 6d(d-1)h_2 \mathbf{u}^2 \\ (2d-1)q_1^2 &= 4h_2\end{aligned}$$

e quindi

$$h_2 = \frac{1}{4}(2d-1)q_1^2$$

Mentre dalla seconda imposizione si ottiene :

$$\begin{aligned}\mathbf{j} = \rho \mathbf{u} &= \sum_i N_i \mathbf{c}_i \\ &= \underbrace{\sum_i d \mathbf{c}_i}_{=0} + \underbrace{\sum_i d(d-1)q_1(\mathbf{c}_i \cdot \mathbf{u}) \mathbf{c}_i}_{=3d(d-1)q_1 \mathbf{u}} \\ &\quad + \underbrace{\sum_i \frac{1}{2}d(d-1)(2d-1)q_1^2 c_{i\alpha}^2 u_\alpha^2 \mathbf{c}_i}_{=0} \\ &\quad + \underbrace{\sum_i d(d-1)h_2 \mathbf{u}^2 \mathbf{c}_i}_{=0}\end{aligned}$$

La somma di tutti i termini, che in questo caso risulta essere molto più semplice, ci da :

$$\rho \mathbf{u} = \rho \mathbf{u} + 3d(d-1)q_1 \mathbf{u}$$

quindi

$$q_1 = \frac{2}{d-1}$$



A questo punto possiamo concludere scrivendo gli effettivi valori delle quantità cercate, ovvero dei due *moltiplicatori di Lagrange* :

$$h_2 = \frac{1}{4}(2d - 1)q_1^2$$

$$q_1 = \frac{2}{d - 1}$$

Ed inoltre si ha :

$$\begin{aligned} N_i(\mathbf{u}) &= d + 2d\mathbf{c}_i \cdot \mathbf{u} + 2d\frac{1 - 2d}{1 - d}c_{i\alpha}^2u_\alpha^2 - d\frac{1 - 2d}{1 - d}\mathbf{u}^2 \\ &= \frac{\rho}{6} + \frac{\rho}{3}\mathbf{c}_i \cdot \mathbf{u} + G(\rho)Q_{i\alpha\beta}u_\alpha u_\beta, \end{aligned}$$

con

$$G(\rho) = \frac{16 - 2\rho}{36 - \rho} \quad e \quad Q_{i\alpha\beta} = c_{i\alpha}c_{i\beta} - \frac{1}{2}\delta_{\alpha\beta}$$

## A.2 *H*-Theorem di Boltzmann

**Teorema A.2.1.** (*Boltzmann, (1872)*)

*Si dimostra che, se si definisce*

$$H(t) := \int d^3v d^3x f(\mathbf{x}, \mathbf{v}, t) \ln(f(\mathbf{x}, \mathbf{v}, t)) \quad (\text{A.11})$$

dove  $f(\mathbf{x}, \mathbf{v}, t)$  è una qualsiasi funzione che soddisfa l'equazione di Boltzmann, allora  $H(t)$  soddisfa la seguente relazione

$$\frac{dH(t)}{dt} \leq 0 \quad (\text{A.12})$$

ed  $f$  è la distribuzione di Maxwell.

### Dimostrazione.

Assumiamo, innanzitutto, che non vi siano forze esterne applicate, in modo che la funzione  $f(\mathbf{x}, \mathbf{v}, t)$  possa obbedire alla seguente equazione di Boltzmann:

$$\frac{\partial f}{\partial t} + \mathbf{v} \cdot \nabla f = Q(f, f). \quad (\text{A.13})$$

Differenziando la (A.11) otteniamo:

$$\frac{dH(t)}{dt} = \int d^3v d^3x \frac{\partial f}{\partial t}(\mathbf{x}, \mathbf{v}, t) [1 + \ln(f(\mathbf{x}, \mathbf{v}, t))]. \quad (\text{A.14})$$

Mentre, inserendo l'equazione di Boltzmann (A.13) si ottiene:

$$\begin{aligned} \frac{dH(t)}{dt} = & - \int \mathbf{v} \cdot \nabla [f(\mathbf{x}, \mathbf{v}, t) \ln(f(\mathbf{x}, \mathbf{v}, t))] d^3v d^3x \\ & + \int d^3v_1 d^3v_2 d^3x dg \varphi(g) [f'_2 f'_1 - f_2 f_1] \cdot |\mathbf{v}_2 - \mathbf{v}_1| [1 + \ln(f_1)] \end{aligned} \quad (\text{A.15})$$

dove abbiamo indicato con  $\varphi(g)$  la funzione, dipendente dalla griglia, che trasforma le velocità della particella  $\{\mathbf{v}, \mathbf{v}_1\}$  prima della collisione, nelle velocità  $\{\mathbf{v}', \mathbf{v}'_1\}$ , dopo l'avvenuta collisione. Inoltre abbiamo posto  $f_1 = f(\mathbf{x}, \mathbf{v}_1, t)$ ,

$$f_2 = f(\mathbf{x}, \mathbf{v}_2, t), \quad f'_1 = f(\mathbf{x}, \mathbf{v}'_1, t) \quad \text{e} \quad f'_2 = f(\mathbf{x}, \mathbf{v}'_2, t).$$

Il primo integrale che compare nella somma, può essere trasformato in un integrale di superficie

$$- \int \mathbf{n} \cdot \mathbf{v} f(\mathbf{x}, \mathbf{v}, t) \ln(f(\mathbf{x}, \mathbf{v}, t)) d^3 v d\Omega \quad (\text{A.16})$$

dove  $\mathbf{n}$  è la normale (uscente) alla superficie  $\Omega$  che racchiude il fluido considerato. Senza aggiungere ulteriori dettagli, assumiamo che tale integrale sia nullo.

Per quanto riguarda il secondo integrale, questo risulta essere invariante rispetto a trasformazioni di  $\mathbf{v}_1$  e  $\mathbf{v}_2$ , in quanto la stessa  $\varphi(g)$  è invariante rispetto alle medesime trasformazioni. Quindi si ottiene:

$$\frac{dH(t)}{dt} = \int d^3 v_1 d^3 v_2 d^3 x d g \varphi(g) |\mathbf{v}_2 - \mathbf{v}_1| (f'_2 f'_1 - f_2 f_1) [1 + \ln(f_1)] \quad (\text{A.17})$$

Sommando la metà della (A.15) e la metà della (A.17) otteniamo:

$$\frac{dH(t)}{dt} = \frac{1}{2} \int d^3 v_1 d^3 v_2 d^3 x d g \varphi(g) |\mathbf{v}_2 - \mathbf{v}_1| (f'_2 f'_1 - f_2 f_1) [2 + \ln(f_1 f_2)]$$

Anche questo integrale è invariante rispetto a trasformazioni di  $\{\mathbf{v}_1, \mathbf{v}_2\}$  e  $\{\mathbf{v}'_1, \mathbf{v}'_2\}$ .

Quindi si ottiene:

$$\frac{dH(t)}{dt} = \frac{1}{2} \int d^3 v'_1 d^3 v'_2 d^3 x d g \varphi'(g) |\mathbf{v}'_2 - \mathbf{v}'_1| (f_2 f_1 - f'_2 f'_1) [2 + \ln(f'_1 f'_2)]$$

ed essendo  $d^3 v'_1 d^3 v'_2 = d^3 v_1 d^3 v_2$ ,  $|\mathbf{v}'_2 - \mathbf{v}'_1| = |\mathbf{v}_2 - \mathbf{v}_1|$  e  $\varphi'(g) = \varphi(g)$ :

$$\frac{dH(t)}{dt} = \frac{1}{2} \int d^3 v_1 d^3 v_2 d^3 x d g \varphi(g) |\mathbf{v}_2 - \mathbf{v}_1| (f_2 f_1 - f'_2 f'_1) [2 + \ln(f'_1 f'_2)]$$

Sommando la metà della prima e dell'ultima di queste relazioni arriviamo ad avere:

$$\frac{dH(t)}{dt} = \frac{1}{2} \int d^3 v_1 d^3 v_2 d^3 x d g \varphi(g) |\mathbf{v}_2 - \mathbf{v}_1| (f'_2 f'_1 - f_2 f_1) [\ln(f_1 f_2) - \ln(f'_1 f'_2)]$$

Osserviamo, a questo punto, che la funzione integranda non è mai positiva, in quanto vale la seguente disuguaglianza :

$$(b - a) \cdot (\ln(a) - \ln(b)) > 0 \quad \forall a \neq b > 0,$$

in questa maniera siamo arrivati a dimostrare che

$$\frac{dH(t)}{dt} \leq 0$$

che è una delle tesi che volevamo mostrare.

Questa quantità risulta, comunque, essere nulla nel momento in cui  $(f'_2 f'_1 - f_2 f_1) = 0$ , quindi, se tale condizione si verifica, vale anche  $\frac{\partial f(\mathbf{x}, \mathbf{v}, t)}{\partial t} = 0$ .

Quindi

$$\frac{dH(t)}{dt} = 0 \quad \iff \quad f(\mathbf{v}'_1) f(\mathbf{v}'_2) - f(\mathbf{v}_1) f(\mathbf{v}_2) = 0$$

per tutte le  $\mathbf{v}'_1, \mathbf{v}'_2$  che discendono da  $\mathbf{v}_1, \mathbf{v}_2$  per collisioni.

Applicando la funzione 'logaritmo' ad ambo i membri della relazione precedente, otteniamo la seguente relazione, valida per la funzione  $f(\mathbf{v})$  stessa :

$$\ln(f(\mathbf{v}'_1)) + \ln(f(\mathbf{v}'_2)) = \ln(f(\mathbf{v}_1)) + \ln(f(\mathbf{v}_2))$$

cioè la funzione  $\ln(f(\mathbf{v}))$  è additiva ed invariante per collisioni, così può essere espressa come combinazione lineare delle funzioni  $\psi_k$ <sup>1</sup>:

$$\ln(f(\mathbf{x}, \mathbf{v})) = a(\mathbf{x}) + \mathbf{b}(\mathbf{x}) \cdot \mathbf{v} + c(\mathbf{x}) \mathbf{v}^2. \quad (\text{A.18})$$

Da questa segue immediatamente che :

$$f(\mathbf{x}, \mathbf{v}) = C(\mathbf{x}) e^{-\frac{m(\mathbf{v} - \mathbf{u}(\mathbf{x}))}{2k_B T(\mathbf{x})}} \quad (\text{A.19})$$

<sup>1</sup>Le funzioni *invarianti per collisioni*  $\psi_k$ , sono state definite nell'Osservazione (2.1).

dove i termini  $C(\mathbf{x})$ ,  $\mathbf{u}(\mathbf{x})$  e  $T(\mathbf{x})$  sono indipendenti da  $\mathbf{v}$ .

Tuttavia la relazione (A.19) rappresenta un sistema che si trova in una situazione di equilibrio se  $f(\mathbf{x}, \mathbf{v})$  è indipendente da  $\mathbf{x}$ , così si ottiene:

$$f^{(eq)}(\mathbf{x}, \mathbf{v}) = f^{(eq)}(\mathbf{v}) = C e^{-\frac{m(\mathbf{v} - \mathbf{u}(\mathbf{x}))^2}{2k_B T}} \quad (\text{A.20})$$

dove, però, in questo caso le quantità  $C$ ,  $\mathbf{u}$  e  $T$  assumono valori costanti.

Se si considera un sistema isolato, nel quale la velocità media  $\mathbf{u}$  risulta essere nulla, si ha:

$$f^{(eq)}(\mathbf{v}) = C e^{-\frac{m \mathbf{v}^2}{2k_B T}}. \quad (\text{A.21})$$

La relazione (A.21) è la famosa *distribuzione di Maxwell*. (c.v.d.)



## Appendice B

### *Listati dei programmi*





## B.1 Listato del programma “anb.f90” sul modello *LB* bi-dimensionale

```

!-----
!-----PROGRAMMA PRINCIPALE-----
!-----
program anb
implicit none
integer :: lx,ly      !....dimensioni della griglia
                    !....nella dimensione x e y
real(8) :: density   !....densità del fluido
real(8) :: omega     !....parametro di rilassamento
real(8) :: vbor,vvera
integer :: t_max     !....numero massimo di iterate
real(8) :: r_rey     !....dimensioni lineari
                    !....el numero di Reynoods
integer :: time,alfa,nfile,ifile
logical :: error
logical,dimension(:,:),allocatable :: obst
                    !....array contenente indicazioni
                    !....sull'ostacolo
!-----
!---abbiamo usato una griglia a 9 velocità,-----
!---anche se sono possibili-----
!---altre geometrie-----
!-----
!           6   2   5
!           \ | /
!           3 - 0 - 1
!           / | \
!           7   4   8
!-----
!---rappresenta la numerazione-----
!---delle direzioni della griglia-----
!-----

real(8), dimension(:, :, :), allocatable :: node,n_hlp
real(8), dimension(:, :), allocatable :: por
real(8) :: vel,mseg
integer :: x,y

!-----
!---Inizio del programma con la richiesta-----
!---dei primi dati sulla dimensione della griglia-----
!-----

```

```

write(*,*) 'Introdurre il numero di suddivisioni:'
read(*,*) lx
!ly=2*lx
ly=lx
allocate(node(0:8,lx,ly),n_hlp(0:8,lx,ly)
          obst(lx,ly),por(lx,ly))

!-----
!---comincia l'inizializzazione-----
!-----

error = .false.
write(*,*) 'Introdurre:Numero di Iterate,&
          &Velocita Effetiva(<',1.d0/(3.d0*lx),'
read(*,*) t_max,vvera

density=0.1
omega=1.9

vbor=vvera*lx
r_rey=1

call crea_ostacolo(obst,lx,ly)
call init_density(lx,ly,density,node)
call init_por(lx,ly,por)

open(10,file='anb.out')
open(20,file='part0.out')
open(30,file='part1.out')

write(20,*) ((por(x,y),x=1,lx),y=1,ly)
mseg=0.d0
do x=1,lx
  do y=1,ly
    mseg=mseg+por(x,y)*(node(0,x,y)+node(1,x,y)&
      &+node(2,x,y)+&
      & node(3,x,y)+node(4,x,y)+node(5,x,y)+&
      & node(6,x,y)+node(7,x,y)+node(8,x,y))
  end do
end do
write(*,*) 'massa iniziale seguita=',mseg

!-----
!---fine dell'inizializzazione-----
!-----

```

```

!-----
!---iniziamo ad iterare sul tempo-----
!-----

    alfa=int(t_max/10.d0)
    nfile=1
    ifile=5
    do time = 1, t_max
      if (time .ge. 10 .and. mod(time,t_max/10) .eq. 0) then
        call check_density(lx,ly,node,time)
      end if
      call cond_bordo(lx,ly,node,vbor,density)
      !....controlliamo se il nodo è di bordo
      if (time==3000) call init_por(lx,ly,por)
      call propagate(lx,ly,node,n_hlp,por,density,vbor)
      !...Propagazione delle particelle
      call bounceback(lx,ly,obst,node,n_hlp)
      !..se i nodi sono di bordo facciamo il bounce-back
      call relaxation(density,omega,lx,ly,node,n_hlp,obst)
      !..applicazione del metodo di Boltzmann
      if (time==t_max-ifile*200+200) then
        !...stampiamo le velocità ogni 200 iterate
        if(ifile==5) then
          open(21,file='pos1.out')
          open(22,file='vel1.out')
        else if(ifile==4) then
          open(21,file='pos2.out')
          open(22,file='vel2.out')
        else if(ifile==3) then
          open(21,file='pos3.out')
          open(22,file='vel3.out')
        else if(ifile==2) then
          open(21,file='pos4.out')
          open(22,file='vel4.out')
        else if(ifile==1) then
          open(21,file='pos5.out')
          open(22,file='vel5.out')
        end if
        write(21,*) ((por(x,y),x=1,lx),y=1,ly)
        call write_results(lx,ly,obst,node,density,22)
        ifile=ifile-1
        close(21)
        close(22)
      end if
    end do
    write(30,*) ((por(x,y),x=1,lx),y=1,ly)
    mseg=0.d0

```

```

do x=1,lx
do y=1,ly
mseg=mseg+por(x,y)*(node(0,x,y)+node(1,x,y)+&
& node(2,x,y)+&
& node(3,x,y)+node(4,x,y)+node(5,x,y)+&
& node(6,x,y)+node(7,x,y)+node(8,x,y))
end do
end do
write(*,*) 'massa finale seguita=',mseg
call comp_rey(lx,ly,vbor,omega,r_rey)
goto 999
990 write (*,*) '!!! error: program stopped &
& during iteration =', time
write (*,*) '!!!'

999 continue
close(10)
write (*,*) '***** end *****'

stop
!-----
!---Fine del Programma Principale-----
!-----
end

!---Inizia la subroutine 'crea_ostacolo'-----

subroutine crea_ostacolo(obst,lx,ly)

implicit none
integer lx,ly
logical obst(lx,ly)
integer x,y,I

!...Se il nodo fa parte dell'ostacolo
!.....viene posto = .true.
!.....altrimenti = .false.
do y = 1, ly
do x = 1, lx
obst(x,y) = .false.
end do
end do
do i=1,lx
obst(i,1)=.true.
end do
do i=1,ly
obst(1,i)=.true.

```

```

        obst(lx,i)=.true.
    end do
return
end

!---Inizia la subroutine 'init_density'-----

subroutine init_density(lx,ly,density,node)

implicit none
integer lx,ly
real(8):: density,node(0:8,lx,ly)
integer x,y
real(8):: t_0,t_1,t_2

!...calcoliamo i pesi che dipendono dalla
!...geometria della griglia

t_0 = density * 4.d0 / 9.d0
t_1 = density / 9.d0
t_2 = density / 36.d0
!...loop sul dominio
do x = 1, lx
    do y = 1, ly
        !...nodo centrale con velocità nulla
        node(0,x,y) = t_0
        !...densità di equilibrio per le velocità
        !...parallele agli assi
        node(1,x,y) = t_1
        node(2,x,y) = t_1
        node(3,x,y) = t_1
        node(4,x,y) = t_1
        !..densità di equilibrio per le velocità
        !...lungo le diagonali
        node(5,x,y) = t_2
        node(6,x,y) = t_2
        node(7,x,y) = t_2
        node(8,x,y) = t_2
    end do
end do
return
end

!---Inizia la subroutine 'init_por'-----

subroutine init_por(lx,ly,por)
!.....inizializziamo la porzione di

```

```

!.....particelle seguite

implicit none
integer :: lx,ly
real(8) :: por(lx,ly)
integer :: x,y,yi,cont

yi=ly-ly/3
por=0.d0
cont=0
do x =lx-7 ,lx-4
  do y =ly-16 ,ly-6
    cont=cont+1
    por(x,y) = 1.d0
  end do
end do
write(*,*) 'punti seguiti=',cont
return
end

!---Inizia la subroutine 'check_density'-----

subroutine check_density(lx,ly,node,time)

!...subroutine che ci permette di calcolare
!...la densità nel nodo considerato

implicit none
integer :: lx,ly,time
real(8) :: node(0:8,lx,ly)
integer :: x,y,n
real(8) :: n_sum

n_sum = 0.d0
!...sul dominio
do y = 1, ly
  do x = 1, lx
    !...loop sui link del nodo
    do n = 0, 8
      !..sommiamo le densità
      n_sum = n_sum + node(n,x,y)
    end do
  end do
end do
write (*,*) '*** Iteration number = ', time
write (*,*) '*** Integral density = ', n_sum

```

```
write (*,*) '***'
return
end

!---Inizia la subroutine 'cond_bordo'-----

subroutine cond_bordo(lx,ly,node,vbor,density)

!...se il nodo è di bordo calcoliamo la distribuzione
!...di equilibrio per il caso particolare

implicit none
integer :: lx,ly
real(8) :: node(0:8,lx,ly),vbor,density
integer :: x
real(8) :: t_0,t_1,t_2

t_0=density*4.d0/9.d0
t_1=density/9.d0
t_2=density/36.d0

do x=2,lx-1
  node(0,x,ly)=t_0
  node(1,x,ly)=t_1*(1+3*vbor+3*vbor**2)
  node(2,x,ly)=t_1*(1-3*vbor**2/2.d0)
  node(3,x,ly)=t_1*(1-3*vbor+3*vbor**2)
  node(4,x,ly)=t_1*(1-3*vbor**2/2.d0)
  node(5,x,ly)=t_2*(1+3.d0*vbor+3*vbor**2)
  node(6,x,ly)=t_2*(1-3.d0*vbor+3*vbor**2)
  node(7,x,ly)=t_2*(1-3.d0*vbor+3*vbor**2)
  node(8,x,ly)=t_2*(1+3.d0*vbor+3*vbor**2)
end do

node(0,1,ly)=t_0
node(1,1,ly)=t_1*(1+3*vbor/2+3*vbor**2/4)
node(2,1,ly)=t_1*(1-3*vbor**2/8.d0)
node(3,1,ly)=t_1*(1-3*vbor/2+3*vbor**2/4)
node(4,1,ly)=t_1*(1-3*vbor**2/8.d0)
node(5,1,ly)=t_2*(1+3.d0*vbor/2+3*vbor**2/4)
node(6,1,ly)=t_2*(1-3.d0*vbor/2+3*vbor**2/4)
node(7,1,ly)=t_2*(1-3.d0*vbor/2+3*vbor**2/4)
node(8,1,ly)=t_2*(1+3.d0*vbor/2+3*vbor**2/4)
node(0,lx,ly)=t_0
node(1,lx,ly)=t_1*(1+3*vbor/2+3*vbor**2/4)
node(2,lx,ly)=t_1*(1-3*vbor**2/8.d0)
node(3,lx,ly)=t_1*(1-3*vbor/2+3*vbor**2/4)
node(4,lx,ly)=t_1*(1-3*vbor**2/8.d0)
```

```

node(5,lx,ly)=t_2*(1+3.d0*vbor/2+3*vbor**2/4)
node(6,lx,ly)=t_2*(1-3.d0*vbor/2+3*vbor**2/4)
node(7,lx,ly)=t_2*(1-3.d0*vbor/2+3*vbor**2/4)
node(8,lx,ly)=t_2*(1+3.d0*vbor/2+3*vbor**2/4)
return
end

!---Inizia la subroutine 'bounceback'-----

subroutine bounceback(lx,ly,obst,node,n_hlp)

implicit none
integer lx,ly
logical obst(lx,ly)
real(8):: node(0:8,lx,ly),n_hlp(0:8,lx,ly)
integer x,y

do x = 1, lx
do y = 1, ly
if (obst(x,y)) then
node(2,x,y) = n_hlp(4,x,y) !...north
node(4,x,y) = n_hlp(2,x,y) !...south
node(1,x,y) = n_hlp(3,x,y) !...east
node(3,x,y) = n_hlp(1,x,y) !...west
node(0,x,y) = n_hlp(0,x,y)
node(5,x,y) = n_hlp(7,x,y) !...north-east
node(6,x,y) = n_hlp(8,x,y) !...north-west
node(7,x,y) = n_hlp(5,x,y) !...south-west
node(8,x,y) = n_hlp(6,x,y) !...south-east
end if
end do
end do

return
end

!---Inizia la subroutine 'propagate'-----

subroutine propagate(lx,ly,node,n_hlp,por,density,vbor)
!...applichiamo la propagazione delle particelle
!...in ogni nodo facente parte della griglia

implicit none
integer :: lx,ly
real(8) :: node(0:8,lx,ly),n_hlp(0:8,lx,ly)
real(8) :: por(lx,ly),p_hlp(lx,ly)

```



```

integer  :: x,y,x_e,x_w,y_n,y_s,l
real(8)  :: t1,t2,density,beta(8,lx,ly),d_loc,massat(lx,ly)
real(8)  :: vx,vy,tol,vsqu,mrim,mesc,marr,vbor,fatt,teta,pgr

pgr=acos(-1.d0)
tol=1.d-10*vbor
t1=density/9.d0
t2=density/36.d0

do x=1,lx
  do y=1,ly
    massat(x,y) = 0.d0
    do l = 0, 8
      massat(x,y) =massat(x,y) + node(l,x,y)
    end do
    vx= (node(1,x,y) + node(5,x,y) + node(8,x,y) &
      & -(node(3,x,y) + node(6,x,y) + node(7,x,y)))&
      & / massat(x,y)
    vy= (node(2,x,y) + node(5,x,y) + node(6,x,y) &
      & -(node(4,x,y) + node(8,x,y) + node(7,x,y)))&
      & / massat(x,y)
    vsqu=vx**2+vy**2
    if (x==1 .or. x==lx .or. y==1 .or. y==ly) vsqu=0.d0
    beta(:,x,y)=0.d0
    if (vsqu>tol ) then
      if( vsqu<=1.d0+tol) then
        fatt=vsqu

      else
        fatt=1.d0
      end if
    end if
    if (vx>tol) then

      teta=atan(vy/vx)
    else if (vx<-tol) then
      teta=atan(vy/vx)+pgr
    else if (vy>tol) then
      teta=pgr/2.d0
    else
      teta=3.d0*pgr/2.d0
    end if
    if (teta<=-pgr/4.d0) then
      beta(4,x,y)=1.d0-4.d0*(teta+pgr/2.d0)/pgr
      beta(8,x,y)=4.d0*(teta+pgr/2.d0)/pgr
    else if (teta<=0.d0) then
      beta(1,x,y)=4.d0*(teta+pgr/4.d0)/pgr
      beta(8,x,y)=1-4.d0*(teta+pgr/4.d0)/pgr
    end if
  end do
end do

```

```

else if (teta<=pgr/4.d0) then
  beta(1,x,y)=1-4.d0*teta/pgr
  beta(5,x,y)=4.d0*teta/pgr
else if (teta<=pgr/2.d0) then
  beta(2,x,y)=4.d0*(teta-pgr/4.d0)/pgr
  beta(5,x,y)=1-4.d0*(teta-pgr/4.d0)/pgr
else if (teta<=3.d0*pgr/4.d0) then
  beta(2,x,y)=1-4.d0*(teta-pgr/2.d0)/pgr
  beta(6,x,y)=4.d0*(teta-pgr/2.d0)/pgr
else if (teta<=pgr) then
  beta(3,x,y)=4.d0*(teta-3.d0*pgr/4.d0)/pgr
  beta(6,x,y)=1-4.d0*(teta-3.d0*pgr/4.d0)/pgr
else if (teta<=5.d0*pgr/4.d0) then
  beta(3,x,y)=1-4.d0*(teta-pgr)/pgr
  beta(7,x,y)=4.d0*(teta-pgr)/pgr
else if (teta<=3.d0*pgr/2.d0) then
  beta(4,x,y)=4.d0*(teta-5.d0*pgr/4.d0)/pgr
  beta(7,x,y)=1-4.d0*(teta-5.d0*pgr/4.d0)/pgr
end if
end if
end do
beta(1,x,ly)=vbor !vsqu=vbor<1/3
end do

do x = 1, lx
do y = 1, ly
  !...calcolo dei vicini del nodo che si trovano
  !...sopra ed a destra
  y_n = mod(y,ly) + 1
  x_e = mod(x,lx) + 1
  !...calcolo dei vicini che si trovano
  !...sotto ed a sinistra
  y_s = ly - mod(ly + 1 - y, ly)
  x_w = lx - mod(lx + 1 - x, lx)

  !...propagazione della densità
  n_hlp(0,x ,y ) = node(0,x,y) !....zero: just copy
  n_hlp(1,x_e,y ) = node(1,x,y) !....east
  n_hlp(2,x ,y_n) = node(2,x,y) !....north
  n_hlp(3,x_w,y ) = node(3,x,y) !....west
  n_hlp(4,x ,y_s) = node(4,x,y) !....south
  n_hlp(5,x_e,y_n) = node(5,x,y) !....north-east
  n_hlp(6,x_w,y_n) = node(6,x,y) !....north-west
  n_hlp(7,x_w,y_s) = node(7,x,y) !....south-west
  n_hlp(8,x_e,y_s) = node(8,x,y) !....south-east

mesc=0.d0

```

```

do l=1,8
  mesc=mesc+beta(1,x,y)*node(1,x,y)
end do

mesc=por(x,y)*mesc
mrim=por(x,y)*massat(x,y)-mesc
marr=por(x_e,y )*beta(3,x_e,y )*node(3,x_e,y ) + &
& por(x ,y_n)*beta(4,x ,y_n)*node(4,x ,y_n) + &
& por(x_w,y )*beta(1,x_w,y )*node(1,x_w,y ) + &
& por(x ,y_s)*beta(2,x ,y_s)*node(2,x ,y_s) + &
& por(x_e,y_n)*beta(7,x_e,y_n)*node(7,x_e,y_n) + &
& por(x_w,y_n)*beta(8,x_w,y_n)*node(8,x_w,y_n) + &
& por(x_w,y_s)*beta(5,x_w,y_s)*node(5,x_w,y_s) + &
& por(x_e,y_s)*beta(6,x_e,y_s)*node(6,x_e,y_s)
if (x==1 .and. y==ly) marr=0.d0
p_hlp(x,y) = (marr+mrim)/&
& (node(0,x ,y )+node(3,x_e,y )+node(4,x ,y_n)+ &
& node(1,x_w,y )+node(2,x ,y_s)+node(7,x_e,y_n)+ &
& node(8,x_w,y_n)+node(5,x_w,y_s)+node(6,x_e,y_s))
end do
end do

do x = 1, lx
  do y = 1, ly
    por(x,y)=p_hlp(x,y)
  end do
end do

return
end

!---Inizia la subroutine 'relaxation'-----

subroutine relaxation(density,omega,lx,ly,node,n_hlp,obst)
!...applicazione di una singola fase del
!...metodo di Boltzmann

implicit none
integer :: lx,ly
logical :: obst(lx,ly)
real(8) :: density,omega
real(8) :: node(0:8,lx,ly),n_hlp(0:8,lx,ly)
integer :: x,y,i
real(8) :: c_squ,t_0,t_1,t_2,u_x,u_y,u_n(8)
real(8) :: n_equ(0:8),u_squ,d_loc

```

```

t_0 = 4.d0 / 9.d0  !...pesi
t_1 = 1.d0 / 9.d0
t_2 = 1.d0 / 36.d0
c_squ = 1.d0 / 3.d0

do x = 1, lx
  do y = 1, ly
    !... sono considerati solo i nodi liberi
    if (.not. obst(x,y)) then
      !...calcoliamo la densità locale
      !...e inizializziamo la variabile d_loc
      d_loc = 0.d0
      do i = 0, 8
        d_loc = d_loc + n_hlp(i,x,y)
      end do
      !...componenti x- e y- della velocità
      u_x = (n_hlp(1,x,y) + n_hlp(5,x,y) + n_hlp(8,x,y) &
        &-(n_hlp(3,x,y) + n_hlp(6,x,y) + n_hlp(7,x,y)))&
        & / d_loc
      u_y = (n_hlp(2,x,y) + n_hlp(5,x,y) + n_hlp(6,x,y) &
        &-(n_hlp(4,x,y) + n_hlp(7,x,y) + n_hlp(8,x,y)))&
        & / d_loc

      u_squ = u_x * u_x + u_y * u_y  !...velocità al quadrato
      !...n- componenti della velocità
      !...n =numero dei vettori velocità della griglia

      u_n(1) =  u_x
      u_n(2) =      u_y
      u_n(3) = - u_x
      u_n(4) =      - u_y
      u_n(5) =  u_x + u_y
      u_n(6) = - u_x + u_y
      u_n(7) = - u_x - u_y
      u_n(8) =  u_x - u_y
      !...densità di equilibrio
      !...densità della particella ferma nel centro
      n_equ(0) = t_0 * d_loc * (1.d0 - u_squ / (2.d0 * c_squ))
      !...densità relative alle
      !...velocità parallele agli assi
      !...(fattore: t_1)
      n_equ(1) = t_1 * d_loc * (1.d0 + u_n(1) / c_squ      &
        &+ u_n(1) ** 2.d0 / (2.d0 * c_squ ** 2.d0) &
        &- u_squ / (2.d0 * c_squ))
      n_equ(2) = t_1 * d_loc * (1.d0 + u_n(2) / c_squ      &

```

```

      &+ u_n(2) ** 2.d0 / (2.d0 * c_squ ** 2.d0) &
      &- u_squ / (2.d0 * c_squ))
n_equ(3) = t_1 * d_loc * (1.d0 + u_n(3) / c_squ      &
      &+ u_n(3) ** 2.d0 / (2.d0 * c_squ ** 2.d0) &
      &- u_squ / (2.d0 * c_squ))
n_equ(4) = t_1 * d_loc * (1.d0 + u_n(4) / c_squ      &
      &+ u_n(4) ** 2.d0 / (2.d0 * c_squ ** 2.d0) &
      &- u_squ / (2.d0 * c_squ))
!...densità relative alle
!...velocità diagonali
!...(fattore: t_2)
n_equ(5) = t_2 * d_loc * (1.d0 + u_n(5) / c_squ      &
      &+ u_n(5) ** 2.d0 / (2.d0 * c_squ ** 2.d0) &
      &- u_squ / (2.d0 * c_squ))
n_equ(6) = t_2 * d_loc * (1.d0 + u_n(6) / c_squ      &
      &+ u_n(6) ** 2.d0 / (2.d0 * c_squ ** 2.d0) &
      &- u_squ / (2.d0 * c_squ))
n_equ(7) = t_2 * d_loc * (1.d0 + u_n(7) / c_squ      &
      &+ u_n(7) ** 2.d0 / (2.d0 * c_squ ** 2.d0) &
      &- u_squ / (2.d0 * c_squ))
n_equ(8) = t_2 * d_loc * (1.d0 + u_n(8) / c_squ      &
      &+ u_n(8) ** 2.d0 / (2.d0 * c_squ ** 2.d0) &
      &- u_squ / (2.d0 * c_squ))
!...fase di rilassamento

do i = 0, 8
  node(i,x,y) = n_hlp(i,x,y) &
    &+ omega * (n_equ(i) - n_hlp(i,x,y))
end do
end if
end do
end do

return
end

!---Inizia la subroutine 'write_results'-----

subroutine write_results(lx,ly,obst,node,density,nfile)
!...calcolo delle velocità risultanti

implicit none
integer  :: lx,ly,nfile
real(8)  :: node(0:8,lx,ly),density
logical  :: obst(lx,ly)
integer  :: x,y,i

```

```

real(8)  :: u_x,u_y,d_loc,press,c_squ

c_squ = 1.d0 / 3.d0  !...quadrata 'speed of sound'

do y = 1, ly
  do x = 1, lx
    if (obst(x,y)) then
      u_x = 0.d0
      u_y = 0.d0
      press = density * c_squ
    else
      d_loc = 0.d0
      do i = 0, 8
        d_loc = d_loc + node(i,x,y)
      end do

      u_x = (node(1,x,y) + node(5,x,y) + node(8,x,y) &
        &-(node(3,x,y) + node(6,x,y) + node(7,x,y)))&
        & / d_loc
      u_y = (node(2,x,y) + node(5,x,y) + node(6,x,y) &
        &-(node(4,x,y) + node(7,x,y) + node(8,x,y)))&
        & / d_loc
      press = d_loc * c_squ
    end if

    write(nfile,*) u_x, u_y, press

  end do
end do

return
end

!---Inizia la subroutine 'comp_rey'-----

subroutine comp_rey(lx,ly,vbor,omega,r_rey)
!...calcolo del numero di Reynolds

implicit none
integer  :: lx,ly
real(8)  :: omega,r_rey
real(8)  :: vbor,visc,rey

visc = 1.d0 / 6.d0 * (2.d0 / omega - 1.d0)
rey = vbor * r_rey / visc

```

```

write (*,*) '*** Calcolo terminate, risultati:'
write (*,*) '***'
write (*,*) '*** viscosità = ', visc
write (*,*) '*** velocità di bordo = ', vbor/lx
write (*,*) '*** Numero di Reynolds = ', rey
write (*,*) '***'
write (*,*) '*** Nel file anb.dat, si trovano tutte le'
write (*,*) '*** informazioni sul flusso considerato.'
write (*,*) '***'
write (*,*) '*** Nel file anb_qx.out, si trovano le velocità'
write (*,*) '*** medie del flusso considerate come funzioni di t.'
write (*,*) '***'

return
end

```

## B.2 Listato del programma “sim.f90” sul modello *LBM* tri-dimensionale

```

module DATI

  implicit none
  real(4) :: tcorr,deltat,deltax,lx,ly,lz
  real(4) :: tol,density,rbi,rbs,mlx,mly
  real(4) :: rdisc2,rdisc,cz_disc
  !...raggio al quadrato, raggio e altezza del
  !...centro del dischetto
  integer :: nx,ny,nz,time,npunti_seg,npunti_persi
  integer :: nzi,nzs
  !...limite inferiore e superiore del dischetto in z
  real(4) :: hzi,hzs
  !...limite inferiore e superiore dell'altezza del
  !...dischetto
  integer :: nxyi,nxys
  !...limite inferiore e superiore del dischetto in x e y
  real(4) :: c(0:18,3),a1(0:18),b1(0:18),c1(0:18),d1(0:18)
  real(4) :: dx_dt,pgr,mbordo,vmaxdisc
  real(4),dimension(:,:,:,),allocatable:: node,n_hlp,vnode

  !-----per seguire i punti-----

```

```

real(4)  :: rsfera2,rsfera,c_s
real(4)  :: va(0:1,0:1,0:1,3),vb(0:1,0:1,0:1,3)
real(4),dimension(:,:,:),allocatable :: mas_seg
real(4),dimension(:,:),allocatable  :: posizione,vel3v
real(4)  :: deltaxsim

!-----miscelazione-----

integer  :: nzone,nliv,nscxy,nscz
real(4), dimension(:), allocatable  :: vetz,vetxy,invvol
real(4),dimension(:,:,:), allocatable:: pospt,vel3vpt
real(4)  :: eps
integer  :: nsct,npsegric,npunti(0:7)
integer, dimension(:), allocatable :: nsc,npzona
logical, dimension(:,:,:), allocatable :: inbic,scatola
logical, dimension(:,:,:), allocatable :: scatolat,scatolan
logical, dimension(:,:,:), allocatable :: scatolan1
integer, dimension(:,:,:), allocatable :: nps_liv

!-----Variabili Globali-----

end module DATI

module PROCEDURE

!-----Modulo che contiene le 'Function'-----
!-----e le 'Subroutine' utilizzate nel-----
!-----Programma Principale-----

use DATI

implicit none

CONTAINS

!-----Inizia la 'Function LAMA'-----

function LAMA(a0,w0,w,x,t,vz)
real(4),dimension(3) ::LAMA
real(4),intent(in)  ::a0,w0,w,t,vz
real(4),dimension(3),intent(in) ::x
real(4),parameter  ::k=1.e0,rr=0.e0,rz=0.e0
real(4)  ::a,r,fat
real(4)  ::f,g,h,df,pr,pz,dpr

a=atan2(x(2),x(1))

```



```

r=sqrt(x(1)**2+x(2)**2)
fat=4.e0
f=L(0.e0,w-w0,pgr/4.e0,a-a0-w*t)
g=L(0.e0,0.e0,pgr,a-a0-w*t+rr)
if (sin(a-a0-w*t+rz)>=0) then
  h=L(0.e0,vz,pgr/4,a-a0-w*t+rz)
else
  h=L(0.e0,vz,fat*pgr/4,(a-a0-w*t+rz))
end if
df=DL(0.e0,w-w0,pgr/4.e0,a-a0-w*t)
pr=P(0.e0,1.e0,1.e0,1.e0,r)
pz=P(0.e0,0.e0,rdisc,1.e0,r)
!---'rdisc' sono le dimensioni del dischetto---
dpr=DP(0.e0,1.e0,1.e0,1.e0,r)

LAMA(1)=f
LAMA(2)=pr*g-0.5e0*(1.e0-k)*r*df
LAMA(3)=pz*h-x(3)*g*(pr/r+dpr)-k*x(3)*df

CONTAINS

!-----Inizia la 'Function L'-----

real(4) :: function L(v0,v1,amp,a)
real(4),intent(in) :: v0,v1,amp,a
real(4),parameter :: esp=1.e0

L=v0
if (cos(a)>cos(amp/2.e0)) &
  & L=L+(v1-v0)*((cos(2*pgr*a/amp)+1)/2.e0)**esp
end function L

real(4) :: function DL(v0,v1,amp,a)
real(4),intent(in) :: v0,v1,amp,a
real(4),parameter :: esp=1.e0

DL=0.e0
if (cos(a)>cos(amp/2.e0)) DL=-pgr/amp*esp*(v1-v0)*&
  & sin(2*pgr*a/amp)
!-----& ((cos(2*pgr*a/amp)+1)/2.e0)**&
!-----& (esp-1.e0)*sin(2*pgr*a/amp)

end function DL

!-----Inizia la 'Function P'-----

```

```

real(4) function P(r1,v1,r2,v2,r)

real(4),intent(in) ::r1,v1,r2,v2,r

if (r<=r1) then
  P=v1
else if (r>=r2) then
  P=v2
else
  P=v1+(v2-v1)*((1.e0-cos(pgr*(r-r1)/(r2-r1)))/2.e0)
end if

end function P

!-----Inizia la 'Function DP'-----

real(4) function DP(r1,v1,r2,v2,r)

real(4),intent(in) :: r1,v1,r2,v2,r

  DP=0.e0
  if (r>r1 .and. r<r2) &
    & DP=0.5e0*(v2-v1)*pgr/(r2-r1)* &
    & sin(pgr*(r-r1)/(r2-r1))

end function DP

end function LAMA

!-----Inizia la 'Function ROTORE'-----

function ROTORE (velang,xx,t,plama1,plama2,plama3,plama4)

real(4),intent(in) :: velang,t
real(4),dimension(3),intent(in) :: xx
real(4),dimension(2),intent(in),optional:: plama1,plama2
real(4),dimension(2),intent(in),optional:: plama3,plama4
real(4),dimension(3) :: ROTORE,rotore0
real(4),dimension(:,:),allocatable ::p_rot
real(4) :: r,pa_t
integer :: k,i

if (present(plama1)) k=1

```

```

if (present(plama2)) k=2
if (present(plama3)) k=3
if (present(plama4)) k=4

allocate(p_rot(k,2),stat=i)

if (i/=0) stop 'ROTORE:allocazione fallita!'
if (k>=1) p_rot(1,:)=plama1
if (k>=2) p_rot(2,:)=plama2
if (k>=3) p_rot(3,:)=plama3
if (k==4) p_rot(4,:)=plama4
rotore0=(/velang,0.e0,0.e0/)
pa_t=pa(2*t)
do i=1,2
    rotore0=rotore0+LAMA(p_rot(i,1),velang,&
        & p_rot(i,2),xx,t,5.e0)
end do
!---5 fattore di spinta verticale delle pale---

do i=3,4
    rotore0=rotore0+LAMA(p_rot(i,1),      &
        & velang,p_rot(i,2),xx,t,-5.e0)
end do

rotore0=pa_t*rotore0
!----Cambiamento di base----

r=sqrt(xx(1)**2+xx(2)**2)

ROTORE(1)=-xx(2)*rotore0(1)+xx(1)*rotore0(2)/r
ROTORE(2)=xx(1)*rotore0(1)+xx(2)*rotore0(2)/r
ROTORE(3)=rotore0(3)

CONTAINS

!-----Inizia la 'Function pa'-----

real(4) function pa(t)

real(4),intent(in) :: t

    if (t>=1.e0) then
        pa=1.e0
    else
        pa=(1.e0-cos(pgr*t))/2.e0
    end if

```

```

end function pa

end function ROTORE

!---Assegnamo le velocità sul Dischetto-----
!---Inizia la 'Function VBNDRY'-----

function VBNDRY(p)

!---Valore nel dischetto della velocità-----
!---v(k) nel pto p-----

real(4),dimension(3),intent(in) :: p
real(4),dimension(3) :: VBNDRY,rot,xx
real(4),dimension(2) :: plama1,plama2
real(4),dimension(2) :: plama3,plama4

!---Componenti cartesiane della-----
!---Velocità angolare-----

plama1=(/0.e0,8*pgr/)
plama3=(/pgr/2.e0,8*pgr/)
plama2=(/pgr,8*pgr/)
plama4=(/1.5e0*pgr,8*pgr/)

xx(1:2)=p(1:2);xx(3)=p(3)-cz_disc
rot=ROTORE(2.e0*pgr,xx,tcorr,plama1,&
           & plama2,plama3,plama4)
VBNDRY=(deltat/deltax)*rot

end function VBNDRY

!---Con 'VBNDRY' terminano le 'Function'----
!---per l'assegnazione delle velocità-----
!---sul Dischetto-----

!---Inizia la 'Subroutine init-density'-----

subroutine init_density()

real(4) :: p(3),r,m,p1(3),r1
real(4) :: vdisc(3)
real(4) :: rbic,rbic1,dist
integer :: x,y,z,i,x1,y1,z1,i1

```

```

integer  ::  xinf,xsup,yinf,ysup,zinf,zsup

mas_seg=0.e0
m=(rbs-rbi)/lz
do x=1,nx
  p(1)=-mlx+(x-1)*deltax
  do y=1,ny
    p(2)=-mly+(y-1)*deltax
    r=sqrt(p(1)**2+p(2)**2)
    do z=1,nz
      p(3)=(z-1)*deltax
      do i=0,18
        node(i,x,y,z)=a1(i)*density
        n_hlp(i,x,y,z)=node(i,x,y,z)
      end do
      rbic=p(3)*m+rbi
      if( z>=nzi-4 .and. z<=nzs+4      &
        & .and. x>=1 .and. x<= nx/2+1 &
        & .and. y>=1 .and. y<= ny/2+1 &
        & .and. r<=rbic) then

        !...p e' uno dei punti che seguiamo...

        xinf=max(floor((p(1)-rsfera+mlx)/deltaxsim+1),1)
        xsup=min(floor((p(1)+rsfera+mlx)/deltaxsim+1),nx)
        yinf=max(floor((p(2)-rsfera+mly)/deltaxsim+1),1)
        ysup=min(floor((p(2)+rsfera+mly)/deltaxsim+1),ny)
        zinf=max(floor((p(3)-rsfera)/deltaxsim+1),1)
        zsup=min(floor((p(3)+rsfera)/deltaxsim+1),nz)

        do x1=xinf,xsup
          p1(1)=-mlx+(x1-1)*deltaxsim
          do y1=yinf,ysup
            p1(2)=-mly+(y1-1)*deltaxsim
            r1=sqrt(p1(1)**2+p1(2)**2)
            do z1=zinf,zsup
              p1(3)=(z1-1)*deltaxsim
              rbic1=p1(3)*m+rbi
              dist=sqrt( sum((p-p1)**2) )
              if( dist<=rsfera-1.d-7 .and. r1<=rbic1) &
                & mas_seg(x1,y1,z1)=mas_seg(x1,y1,z1) &
                &+exp(dist/(dist-rsfera))
            end do
          end do
        end do
      end if
    end do
  end do
end do

```

```

        end do
    end do

end subroutine init_density

!---Inizia la 'Subroutine propagate'-----
subroutine propagate()

real(4)  :: p(3),m,eqz,r,rbic
integer  :: x,y,z,x_a,x_d,y_d,y_s,z_u,z_d

m=(rbs-rbi)/lz
do x=1,nx
    p(1)=-mlx+(x-1)*deltax
    do y=1,ny
        p(2)=-mly+(y-1)*deltax
        r=sqrt(p(1)**2+p(2)**2)
        do z=1,nz
            p(3)=(z-1)*deltax
            rbic=p(3)*m+rbi
            eqz=rbic+1.42e0*deltax
            if(r<eqz) then
                x_a=mod(x,nx)+1
                y_d=mod(y,ny)+1
                z_u=mod(z,nz)+1
                x_d=nx-mod(nx+1-x,nx)
                y_s=ny-mod(ny+1-y,ny)
                z_d=nz-mod(nz+1-z,nz)
                n_hlp(0,x,y,z)=node(0,x,y,z)
                n_hlp(1,x_a,y,z)=node(1,x,y,z)
                n_hlp(2,x_d,y,z)=node(2,x,y,z)
                n_hlp(3,x,y_d,z)=node(3,x,y,z)
                n_hlp(4,x,y_s,z)=node(4,x,y,z)
                n_hlp(5,x,y,z_u)=node(5,x,y,z)
                n_hlp(6,x,y,z_d)=node(6,x,y,z)
                n_hlp(7,x_a,y_d,z)=node(7,x,y,z)
                n_hlp(8,x_a,y_s,z)=node(8,x,y,z)
                n_hlp(9,x_d,y_d,z)=node(9,x,y,z)
                n_hlp(10,x_d,y_s,z)=node(10,x,y,z)
                n_hlp(11,x_a,y,z_u)=node(11,x,y,z)
                n_hlp(12,x_d,y,z_u)=node(12,x,y,z)
                n_hlp(13,x_a,y,z_d)=node(13,x,y,z)
                n_hlp(14,x_d,y,z_d)=node(14,x,y,z)
                n_hlp(15,x,y_d,z_u)=node(15,x,y,z)
                n_hlp(16,x,y_d,z_d)=node(16,x,y,z)
                n_hlp(17,x,y_s,z_u)=node(17,x,y,z)
                n_hlp(18,x,y_s,z_d)=node(18,x,y,z)
            end if
        end do
    end do
end do

```

```

        end if
    end do
end do
end do
end subroutine propagate

!---Inizia la 'Subroutine relax_wr'-----

subroutine relax_wr(omega)

real(4)  :: omega,ricop,ric_p
integer  :: x,y,z,i
real(4)  :: massa_loc,u_x,u_y,u_z,u_squ
real(4)  :: u_equ(0:18),d_equ(0:18)
real(4)  :: cupcond,c_i,b_i
real(4)  :: mtot,m,r,p(3),eqz,vdisc(3),dist
real(4)  :: p1(3),somma,maxval,maxvel
real(4)  :: r1,rbic,rbic1
integer  :: x1,y1,z1,i1,xm,ym,zm,xvm,yvm,zvm
integer  :: npun,npun_out
integer  :: xinf,xsup,yinf,ysup,zinf,zsup

m=(rbs-rbi)/lz

!---riempio mas_seg e stampo-----

npun_out=0
do npun=1,npunti_seg
    r=sqrt(posizione(npun,1)**2+posizione(npun,2)**2)
    rbic=posizione(npun,3)*m+rbi
    if(r<=rbic .and. posizione(npun,3)>=0.e0 &
        & .and. posizione(npun,3)<=lz ) then
        xinf=max(floor((posizione(npun,1)-
            & rsfera+mlx)/deltaxsim+1)+1,1)    &
        xsup=min(floor((posizione(npun,1)+
            & rsfera+mlx)/deltaxsim+1),nx)    &
        yinf=max(floor((posizione(npun,2)-
            & rsfera+mly)/deltaxsim+1)+1,1)    &
        ysup=min(floor((posizione(npun,2)+
            & rsfera+mly)/deltaxsim+1),ny)    &
        zinf=max(floor((posizione(npun,3)-
            & rsfera)/deltaxsim+1)+1,1)    &
        zsup=min(floor((posizione(npun,3)+
            & rsfera)/deltaxsim+1),nz)    &
        do x1=xinf,xsup
            p1(1)=-mlx+(x1-1)*deltaxsim

```

```

do y1=yinf,ysup
  p1(2)=-mly+(y1-1)*deltaxsim
  r1=sqrt(p1(1)**2+p1(2)**2)
  do z1=zinf,zsup
    p1(3)=(z1-1)*deltaxsim
    rbic1=p1(3)*m+rbi
    dist=sqrt( sum((posizione(npun,:)-p1)**2))
    if( dist<=rsfera-1.d-7 .and. r1<=rbic1) &
      & mas_seg(x1,y1,z1)=mas_seg(x1,y1,z1) &
      &+exp(dist/(dist-rsfera))
  end do
end do
end do
else
  npun_out=npun_out+1 !numero di punti persi
end if
end do

write(*,*) 'punti seguiti=',npunti_seg,' &
           & punti usciti=',npun_out
write(30,*) 'simulazione',npunti_seg,npun_out
maxval=0.e0
maxvel=0.e0
u_squ=0.e0

mtot=0.e0
do y=ny,1,-1
  p(2)=-mly+(y-1)*deltax
  do x=1,nx
    p(1)=-mlx+(x-1)*deltax
    r=sqrt(p(1)**2+p(2)**2)
    do z=1,nz
      p(3)=(z-1)*deltax
      rbic=p(3)*m+rbi
      eqz=rbic+1.42e0*deltax
      if(z>=2 .and. z<=nz-1 .and. r<=rbic) then
        massa_loc=sum(n_hlp(:,x,y,z))
        !---Siamo all'interno del Bicchiere-----
        !---calcoliamo le velocità-----

        u_x=(n_hlp(1,x,y,z)+n_hlp(13,x,y,z)+ &
          & n_hlp(11,x,y,z)+n_hlp(8,x,y,z)+ &
          & n_hlp(7,x,y,z)-(n_hlp(2,x,y,z)+ &
          & n_hlp(12,x,y,z)+n_hlp(14,x,y,z)+ &
          & n_hlp(10,x,y,z)+n_hlp(9,x,y,z)))/ &
          & massa_loc
        u_y=(n_hlp(3,x,y,z)+n_hlp(7,x,y,z)+ &

```



```

&   n_hlp(9,x,y,z)+n_hlp(15,x,y,z)+ &
&   n_hlp(16,x,y,z)-(n_hlp(4,x,y,z)+ &
&   n_hlp(17,x,y,z)+n_hlp(18,x,y,z)+ &
&   n_hlp(10,x,y,z)+n_hlp(8,x,y,z))/ &
&   massa_loc
u_z=(n_hlp(5,x,y,z)+n_hlp(12,x,y,z)+ &
&   n_hlp(11,x,y,z)+n_hlp(15,x,y,z)+ &
&   n_hlp(17,x,y,z)-(n_hlp(6,x,y,z)+ &
&   n_hlp(13,x,y,z)+n_hlp(14,x,y,z)+ &
&   n_hlp(16,x,y,z)+n_hlp(18,x,y,z)))/ &
&   massa_loc
if( x>=nxyi .and. x<=nxys .and. y>=nxyi &
&   .and. y<=nxys .and. &
&   z>=nzi .and. z<=nzs .and. r*r<= &
&   rdisc2 ) then

!---siamo nel dischetto-----
!---assegnamo le velocita'-----

    if (r>tol) then
        vdisc=vbndry(p)
    else
        vdisc=0.e0
    end if
    u_x=(vdisc(1)-u_x)/omega+u_x
    u_y=(vdisc(2)-u_y)/omega+u_y
    u_z=(vdisc(3)-u_z)/omega+u_z
    vnode(x,y,z,:)=dx_dt*vdisc(:)
else
    vnode(x,y,z,1)=u_x*dx_dt
    vnode(x,y,z,2)=u_y*dx_dt
    vnode(x,y,z,3)=u_z*dx_dt
end if
u_squ=u_x**2+u_y**2+u_z**2
do i=0,18
    c_i=c(i,1)*u_x+c(i,2)*u_y+c(i,3)*u_z
    b_i=(a1(i)+b1(i)*c_i+c1(i)* &
&   c_i**2+d1(i)*u_squ)
    u_equ(i)=massa_loc*b_i
    node(i,x,y,z)=n_hlp(i,x,y,z)+omega* &
&   (u_equ(i)-n_hlp(i,x,y,z))
end do
mtot=mtot+massa_loc
cupcond=mas_seg(x,y,z)
mas_seg(x,y,z)=0.e0
else if (z==nz .or. (z==1 .and. eqz>r) &
&   .or. eqz > r) then

```

```

!---Siamo sul bordo del Bicchiere-----
!---Applichiamo il metodo 'bounce-back'-

node(1,x,y,z)=n_hlp(2,x,y,z)
node(2,x,y,z)=n_hlp(1,x,y,z)
node(3,x,y,z)=n_hlp(4,x,y,z)
node(4,x,y,z)=n_hlp(3,x,y,z)
node(5,x,y,z)=n_hlp(6,x,y,z)
node(6,x,y,z)=n_hlp(5,x,y,z)
node(7,x,y,z)=n_hlp(10,x,y,z)
node(10,x,y,z)=n_hlp(7,x,y,z)
node(8,x,y,z)=n_hlp(9,x,y,z)
node(9,x,y,z)=n_hlp(8,x,y,z)
node(11,x,y,z)=n_hlp(14,x,y,z)
node(14,x,y,z)=n_hlp(11,x,y,z)
node(12,x,y,z)=n_hlp(13,x,y,z)
node(13,x,y,z)=n_hlp(12,x,y,z)
node(15,x,y,z)=n_hlp(18,x,y,z)
node(18,x,y,z)=n_hlp(15,x,y,z)
node(16,x,y,z)=n_hlp(17,x,y,z)
node(17,x,y,z)=n_hlp(16,x,y,z)
if(r>rbic) then
  cupcond=-10.e0
else
  cupcond=mas_seg(x,y,z)
  massa_loc=sum(n_hlp(:,x,y,z))
  mtot=mtot+massa_loc
end if
mas_seg(x,y,z)=0.e0
vnode(x,y,z,:)=0.e0
else
  cupcond=-10.e0
  mas_seg(x,y,z)=0.e0
  vnode(x,y,z,:)=0.e0
end if
write(10,'(1x,e12.5)') cupcond
end do
end do
write(*,*) 'massa globale = ',mtot
write(30,*) 'massa glob ',mtot
write(*,*) 'punti persi = ',npun_out

end subroutine relax_wr

!---Inizia la 'Subroutine relaxation'-----

```

```

subroutine relaxation(omega)

real(4)  :: omega
integer  :: x,y,z,i,spia
real(4)  :: massa_loc,u_x,u_y,u_z,u_squ
real(4)  :: u_equ(0:18),d_equ(0:18)
real(4)  :: cupcond,c_i,b_i
real(4)  :: m,r,p(3),eqz,vdisc(3)
real(4)  :: p1(3),somma,r1,rbic
integer  :: x1,y1,z1,i1
integer  :: xm,ym,zm,xvm,yvm,zvm
integer  :: npun,npun_out

m=(rbs-rbi)/lz
u_squ=0.e0
do y=ny,1,-1
  p(2)=-mly+(y-1)*deltax
  do x=1,nx
    p(1)=-mlx+(x-1)*deltax
    r=sqrt(p(1)**2+p(2)**2)
    do z=1,nz
      p(3)=(z-1)*deltax
      rbic=p(3)*m+rbi
      eqz=rbic+1.42e0*deltax
      if(z>=2 .and. z<=nz-1 .and. r<=rbic) then
        massa_loc=sum(n_hlp(:,x,y,z))
        !---Siamo nel Bicchiere-----
        !---calcoliamo le velocità-----

        u_x=(n_hlp(1,x,y,z)+n_hlp(13,x,y,z)+ &
          & n_hlp(11,x,y,z)+n_hlp(8,x,y,z)+ &
          & n_hlp(7,x,y,z)-(n_hlp(2,x,y,z)+ &
          & n_hlp(12,x,y,z)+n_hlp(14,x,y,z)+ &
          & n_hlp(10,x,y,z)+n_hlp(9,x,y,z)))/ &
          & massa_loc
        u_y=(n_hlp(3,x,y,z)+n_hlp(7,x,y,z)+ &
          & n_hlp(9,x,y,z)+n_hlp(15,x,y,z)+ &
          & n_hlp(16,x,y,z)-(n_hlp(4,x,y,z)+ &
          & n_hlp(17,x,y,z)+n_hlp(18,x,y,z)+ &
          & n_hlp(10,x,y,z)+n_hlp(8,x,y,z)))/ &
          & massa_loc
        u_z=(n_hlp(5,x,y,z)+n_hlp(12,x,y,z)+ &
          & n_hlp(11,x,y,z)+n_hlp(15,x,y,z)+ &
          & n_hlp(17,x,y,z)-(n_hlp(6,x,y,z)+ &
          & n_hlp(13,x,y,z)+n_hlp(14,x,y,z)+ &
          & n_hlp(16,x,y,z)+n_hlp(18,x,y,z)))/ &

```

```

&   massa_loc
if(x>=nxyi .and. x<=nxys .and.           &
&   y>=nxyi .and. y<=nxys .and.           &
&   z>=nzi .and. z<=nz  .and.           &
&   r*r<=rdisc2 ) then
  spia=0
  !---siamo nel dischetto-----
  !---assegnamo le velocita'-----

  if (r>tol) then
    vdisc=vbdry(p)
  else
    vdisc=0.e0
  end if
  u_x=(vdisc(1)-u_x)/omega+u_x
  u_y=(vdisc(2)-u_y)/omega+u_y
  u_z=(vdisc(3)-u_z)/omega+u_z
  vnode(x,y,z,:)=dx_dt*vdisc(:)
else
  vnode(x,y,z,1)=u_x*dx_dt
  vnode(x,y,z,2)=u_y*dx_dt
  vnode(x,y,z,3)=u_z*dx_dt
end if
u_squ=u_x**2+u_y**2+u_z**2
do i=0,18
  c_i=c(i,1)*u_x+c(i,2)*u_y+c(i,3)*u_z
  b_i=(a1(i)+b1(i)*c_i+c1(i)*           &
&   c_i**2+d1(i)*u_squ)
  u_equ(i)=massa_loc*b_i
  node(i,x,y,z)=n_hlp(i,x,y,z)+           &
&   omega*(u_equ(i)-n_hlp(i,x,y,z))
end do
else if (z==nz .or. (z==1 .and. eqz>r) &
&   .or. eqz > r) then
  !---Siamo sul bordo del Bicchiere-----
  !---appliciamo il metodo 'bounce-back'-

  node(1,x,y,z)=n_hlp(2,x,y,z)
  node(2,x,y,z)=n_hlp(1,x,y,z)
  node(3,x,y,z)=n_hlp(4,x,y,z)
  node(4,x,y,z)=n_hlp(3,x,y,z)
  node(5,x,y,z)=n_hlp(6,x,y,z)
  node(6,x,y,z)=n_hlp(5,x,y,z)
  node(7,x,y,z)=n_hlp(10,x,y,z)
  node(10,x,y,z)=n_hlp(7,x,y,z)
  node(8,x,y,z)=n_hlp(9,x,y,z)
  node(9,x,y,z)=n_hlp(8,x,y,z)

```

```

        node(11,x,y,z)=n_hlp(14,x,y,z)
        node(14,x,y,z)=n_hlp(11,x,y,z)
        node(12,x,y,z)=n_hlp(13,x,y,z)
        node(13,x,y,z)=n_hlp(12,x,y,z)
        node(15,x,y,z)=n_hlp(18,x,y,z)
        node(18,x,y,z)=n_hlp(15,x,y,z)
        node(16,x,y,z)=n_hlp(17,x,y,z)
        node(17,x,y,z)=n_hlp(16,x,y,z)
        vnode(x,y,z,:)=0.e0
    else
        vnode(x,y,z,:)=0.e0
    end if
end do
end do
end do

end subroutine relaxation

!---Inizia la 'Subroutine aggiorna_pos'-----

subroutine aggiorna_pos()

    integer :: npun
    integer :: xinf,yinf,zinf,i,j,k,h
    real(4) :: cxinf,cyinf,czinf,deltatsim
    real(4) :: v_old(3),r,r_n,rbic,m,f_ijk,c

    deltatsim=0.01e0
    m=(rbs-rbi)/lz
    do npun=1,npunti_seg
        v_old=vel3v(npun,:)
        r=sqrt(posizione(npun,1)**2+posizione(npun,2)**2)
        rbic=posizione(npun,3)*m+rbi
        vel3v(npun,:)=0.e0
        if (posizione(npun,3)>=hzi &
            & .and. posizione(npun,3)<=hzs &
            & .and. r<=rdisc) then !siamo nel dischetto
            if (r>tol) then

                vel3v(npun,:)=dx_dt*VBNDRY(posizione(npun,:))
            else
                vel3v(npun,:)=0.e0
            end if
        else if (posizione(npun,3)>=0.e0 &
            & .and. posizione(npun,3)<=lz &
            & .and. r<=rbic) then

```

```

!---fuori dal dischetto-----
!---ma dentro il bicchiere-----

xinf=min(floor((posizione(npun,1)+mlx)/
&      deltax)+1,nx-1)
yinf=min(floor((posizione(npun,2)+mly)/
&      deltax)+1,ny-1)
zinf=min(floor(posizione(npun,3)/
&      deltax)+1,nz-1)
cxinf=-mlx+(xinf-1)*deltax
cyinf=-mly+(yinf-1)*deltax
czinf=(zinf-1)*deltax
do i=0,1
  do j=0,1

    do k=0,1
      f_ijk=(va(i,j,k,1)+vb(i,j,k,1)*
&      (posizione(npun,1)-cxinf)/
&      deltax)*
&      (va(i,j,k,2)+vb(i,j,k,2)*
&      (posizione(npun,2)-cyinf)/
&      deltax)*
&      (va(i,j,k,3)+vb(i,j,k,3)*
&      (posizione(npun,3)-czinf)/
&      deltax)
      vel3v(npun,:)=vel3v(npun,:)+f_ijk*
&      vnode(xinf+i,yinf+j,zinf+k,:)
    end do
  end do
end do
posizione(npun,:)=posizione(npun,)+
& deltat*sim/2.d0*(3.d0*vel3v(npun,:)- v_old)

!---cambiamo i punti finiti fuori-----
end do

!---punti per la miscelazione-----
do h=1,nzone
do npun=1,npzona(h)
  v_old=vel3vpt(npun,h,:)
  r=sqrt(pospt(npun,h,1)**2+pospt(npun,h,2)**2)
  rbic=pospt(npun,h,3)*m+rbi
  vel3vpt(npun,h,:)=0.e0
  if (pospt(npun,h,3)>=hzi .and. pospt(npun,h,3)<=hzs &

```

```

        & .and. r<=rdisc) then !siamo nel dischetto
    if (r>tol) then
        vel3vpt(npun,h,:)=dx_dt*VBNDRY(pospt(npun,h,:))
    else
        vel3vpt(npun,h,:)=0.e0
    end if
else if (pospt(npun,h,3)>=0.e0                                &
        & .and. pospt(npun,h,3)<=lz                          &
        & .and. r<=rbic) then
    !---fuori dal dischetto-----
    !---ma dentro il bicchiere-----

    xinf=min(floor((pospt(npun,h,1)+mlx)/                    &
        &          deltax)+1,nx-1)
    yinf=min(floor((pospt(npun,h,2)+mly)/                    &
        &          deltax)+1,ny-1)
    zinf=min(floor(pospt(npun,h,3)/                          &
        &          deltax)+1,nz-1)
    cxinf=-mlx+(xinf-1)*deltax
    cyinf=-mly+(yinf-1)*deltax
    czinf=(zinf-1)*deltax
    do i=0,1
        do j=0,1
            do k=0,1
                f_ijk=(va(i,j,k,1)+vb(i,j,k,1)*              &
                    &          (pospt(npun,h,1)-cxinf)/deltax)*    &
                    &          (va(i,j,k,2)+vb(i,j,k,2)*              &
                    &          (pospt(npun,h,2)-cyinf)/deltax)*              &
                    &          (va(i,j,k,3)+vb(i,j,k,3)*              &
                    &          (pospt(npun,h,3)-czinf)/deltax)
                vel3vpt(npun,h,:)=vel3vpt(npun,h,:)+        &
                    &          f_ijk*vnode(xinf+i,yinf+j,zinf+k,:)
            end do
        end do
    end do
end if

pospt(npun,h,:)=pospt(npun,h,:)+                            &
&          deltatsim/2.d0*( 3.d0*vel3vpt                    &
&          (npun,h,:) - v_old )

    !---cambiamo i punti finiti fuori-----

end do
end do

end subroutine aggiorna_pos

```

```

!---Inizia la 'Subroutine tind'-----
function tind(p,vet,lv)
!---restituisce l'indice i tale che-----
!---vet(i)<=p<vet(i+1)-----
integer          :: tind
integer, intent(in)  :: lv
real(4), intent(in)  :: p
real(4), dimension(lv+1) :: vet
integer           :: i,j,h
  if (lv<=1) then
    write(*,*) '---errore nel dimensionamento---'
    stop
  end if

  if (p<vet(1) .or. p>=vet(lv+1)) then
    tind=-1
  else if (p<vet(2)) then
    tind=1
  else if (p>=vet(lv)) then
    tind=lv
  else
    i=1
    j=lv
    do
      h=(i+j)/2
      if (p>=vet(h)) then
        if (p<vet(h+1)) then
          tind=h
          exit
        else
          i=h
        end if
      else
        j=h
      end if
    end do
  end if
end function tind
!---Finisce il 'Module' nel quale-----
!---sono definite le 'Function' e-----
!---le 'Subroutine' utilizzate-----
!---nel 'Programma Principale'-----
end module PROCEDURE

```



```

!-----
!---Inizia il 'Programma Principale'-----
!-----

program MAIN

  use PROCEDURE
  implicit none

  integer :: x,y,z,cont,cont1
  integer :: i,j,k,k1,h,l
  integer :: i_stampa,i_mis,p_stampa,p_mis
  real(4) :: p(3),r,v(3)
  real(4) :: massa_loc,ms_loc,m,eqz,mtot
  integer,dimension(7) :: itime
  real(4) :: omega,rbic,rmax,
  real(4) :: vmedia,velmedia
  integer :: t_max,ifile,t1,t2
  integer :: liv_rif,dimensione
  integer :: nscrt,nscr(12),n_file
  integer :: tempop,ix,iy,iz,nscrn,nscrn1

  !---parametri per la miscelazione-----

  nzone=12 !---numero di zone seguite-----
  nliv=72  !---numero di livelli su cui-----
           !---calcoliamo i punti che ci sono-
  nscxy=10 !---numero di scatole-----
           !---nella direzione x o y-----
  nscz=24  !---numero di scatole-----
           !---nella direzione z-----

  pgr=acos(-1.e0)
  lx=1.32e0 !in decimetri
  ly=lx
  mlx=lx/2.e0
  mly=ly/2.e0
  write(*,*) '--0-bicchiere dritto--'
  write(*,*) '--1-bicchiere rovesciato--'
  read(*,*) i
  if (i==0) then
    rbi=0.45e0
    rbs=0.66e0
    liv_rif=nliv+1
  else if (i==1) then

```

```

    liv_rif=1
    rbi=0.66e0
    rbs=0.45e0
else
    write(*,*) '---errore nel dato---'
    STOP
end if
nx=45      !---numero di suddivisioni dispari---
ny=nx
nz=57      !---numero di suddivisioni-----
           !---verticali dispari-----
nzi=9      !---limite inferiore-----
           !---del dischetto in z-----
nzs=nzi+3 !---limite superiore-----
           !---del dischetto in z-----
nxyi=14    !---limite inferiore-----
           !---del dischetto in x o y-----
nxys=32    !---limite superiore-----
           !---del dischetto in x o y-----

deltax=lx/(nx-1)
deltaxsim=2.e0*deltax
hzi=(nzi-1)*deltax
hzs=(nzs-1)*deltax
lz=(nz-1)*deltax
m=(rbs-rbi)/lz
mbordo=(rbs-rbi)/lz

!---deltat=0.5e-04-----
!---è quello-----
!---usato in precedenza-----

deltat=2.5e-04
i_stampa=1
i_mis=1
p_stampa=nint(0.02e0/deltat)
p_mis=nint(0.01e0/deltat)
write(*,*) 'p_stampa=',p_stampa,      &
&          'p_mis=',p_mis
dx_dt=deltax/deltat
cz_disc=(nzs+nzi-2)*deltax/2.e0
write(*,*) 'delatx=',deltax,'lz=',lz,&
&          'centro disch=',cz_disc
rdisc2=(0.275e0)**2
rdisc=0.275e0
rsfera=0.05625

```

```

tol=1.e-7
density=0.1
c_s=deltax/deltat/sqrt(3.e0)
!-----
!---Calcolo del 'Parametro di Rilassamento'---
!-----

      omega=1.e0/(1.e0/(deltat*c_s**2*250.e0)+0.5e0)

!-----
!---Stampa del valore del-----
!---'Numero di Reynolds'-----
!---dipendente dal parametro 'Omega'-----
!-----

write(*,*) 'Re=',1/(deltat*c_s**2*(1/omega-0.5))
write(*,*) 'ni=',deltat*c_s**2*(1/omega-0.5)
write(*,*) 'omega (per Reinolds 250)=' ,omega
velmedia=vmedia()
write(*,*) 'velmedia=',velmedia
write(*,*) 'velocita massima sul dischetto=', &
&          vmaxdisc,'< cs=',c_s

!-----
!---Calcolo dei 'Vettori del Lattice'-----
!-----

c=0.e0
c(1,1)=1.e0;c(2,1)=-1.e0;c(3,2)=1.e0;
c(4,2)=-1.e0;c(5,3)=1.e0;c(6,3)=-1.e0;
c(7,1)=1.e0;c(7,2)=1.e0;c(8,1)=1.e0;
c(8,2)=-1.e0;c(9,1)=-1.e0;c(9,2)=1.e0;
c(10,1)=-1.e0;c(10,2)=-1.e0;c(11,1)=1.e0;
c(11,3)=1.e0;c(12,1)=-1.e0;c(12,3)=1.e0;
c(13,1)=1.e0;c(13,3)=-1.e0;c(14,1)=-1.e0;
c(14,3)=-1.e0;c(15,2)=1.e0;c(15,3)=1.e0;
c(16,2)=1.e0;c(16,3)=-1.e0;c(17,2)=-1.e0;
c(17,3)=1.e0;c(18,2)=-1.e0;c(18,3)=-1.e0;

!-----
!---Calcolo dei 'Pesi W(i)'-----
!-----

a1(0)=1.e0/3.e0
a1(1:6)=1.e0/18.e0
a1(7:18)=1.e0/36.e0

```

```

b1(0)=0.e0
b1(1:6)=1.e0/6.e0
b1(7:18)=1.e0/12.e0
c1(0)=0.e0
c1(1:6)=1.e0/4.e0
c1(7:18)=1.e0/8.e0
d1(0)=-0.5e0
d1(1:6)=-1.e0/12.e0
d1(7:18)=-1.e0/24.e0

do x=0,1
  do y=0,1
    do z=0,1
      va(x,y,z,1)=1.e0-x
      vb(x,y,z,1)=2.e0*x-1.e0
      va(x,y,z,2)=1.e0-y
      vb(x,y,z,2)=2.e0*y-1.e0
      va(x,y,z,3)=1.e0-z
      vb(x,y,z,3)=2.e0*z-1.e0
    end do
  end do
end do

!-----
!---Apriamo i File che-----
!---conterranno i Risultati-----
!---per il calcolo del-----
!---'Coefficiente di Diffusione'-----
!-----

open(70,file='rpundric01.dat')
open(71,file='rpundric02.dat')
open(72,file='rpundric03.dat')
open(73,file='rpundric04.dat')
open(74,file='rpundric05.dat')
open(75,file='rpundric06.dat')
open(76,file='rpundric07.dat')
open(77,file='rpundric08.dat')
open(78,file='rpundric09.dat')
open(79,file='rpundric10.dat')
open(80,file='rpundric11.dat')
open(81,file='rpundric12.dat')
open(82,file='rpundric13.dat')
open(83,file='rpundric14.dat')
open(84,file='rpundric2-3.dat')
open(21,file='rpundriczdr.dat')

```

```

open(22,file='rpundnpundr.dat')
open(51,file='rpundnpundr01.dat')
open(52,file='rpundnpundr02.dat')
open(53,file='rpundnpundr03.dat')
open(54,file='rpundnpundr04.dat')
open(55,file='rpundnpundr05.dat')
open(56,file='rpundnpundr06.dat')
open(57,file='rpundnpundr07.dat')
open(58,file='rpundnpundr08.dat')
open(59,file='rpundnpundr09.dat')
open(60,file='rpundnpundr10.dat')
open(61,file='rpundnpundr11.dat')
open(62,file='rpundnpundr12.dat')
open(30,file='rpndmis7.inf')
write(30,*) 'DRITTO'

!-----
!---Inizia la Richiesta dei Dati-----
!---dall'esterno-----
!-----

write(*,*) 'introduci tmax per 14 s =',14.e0/deltat
read(*,*) t_max
allocate(node(0:18,nx,ny,nz),n_hlp(0:18,nx,ny,nz),&
& mas_seg(nx,ny,nz),vnode(nx,ny,nz,3))

!*****Inizia il calcolo del tempo*****

call X05AAF(itime)
t1=itime(6)+60*(itime(5)+60*itime(4))

!*****PER LA MISCELAZIONE*****
!---costruiamo VETZ-VETXY-VOL-----

allocate(vetz(nliv+1),vetxy(nscxy+1),
& invvol(nliv),stat=i)
if (i/=0) STOP 'allocazione vetz'
do l=1,nliv+1
!---VETZ-vettore delle altezze dei livelli--

    vetz(l)=lz/150+lz/(150*2)+(l-1)*lz/150*2
    write(30,*) 'vetz=',vetz(l)
end do
do l=1,nliv
!---VOL-vettore dei invvolumi dei livelli--

```

```

      invvol(1)=3.d0/(pgr*( vetz(1+1)-vetz(1))*      &
&      ((vetz(1+1)*mbordo+rbi)**2+      &
&      (vetz(1)*mbordo+rbi)**2+      &
&      (vetz(1+1)*mbordo+rbi)*      &
&      (vetz(1)*mbordo+rbi)))
      write(30,*) 'inverso volume vol',invvol(1)
end do

!---calcolo eps differenza-----
!---tra i due raggi di tre livelli-----

      eps=abs(vetz(4)*mbordo-vetz(1)*mbordo)
      write(30,*) 'eps=',eps

!---riempio i punti di inizio-----
!---dei cubi lungo x y-----

do i=1,nscxy+1
      vetxy(i)=- (vetz(liv_rif)*mbordo+rbi)+2*eps+      &
&      (i-1)*(2*(vetz(liv_rif)*      &
&      mbordo+rbi)-4*eps)/nscxy
      write(30,*) 'vetxy=',vetxy(i)
end do

!---calcolo il numero di scatole-----
!---contenute nel bicchiere-----

allocate(nsc(nzone),inbic(nscxy,nscxy,nscz),stat=i)
if (i/=0) STOP 'allocazione nsc'

nsc=0      !---numero di scatole-----
           !---su ognuna delle 12 zone-----
nsct=0      !---numero di scatole totali-----
           !---all'interno del bicchiere-----
inbic=.FALSE.
do i=1,nscxy
  do j=1,nscxy
    do h=1,nscz
      rmax=0.d0
      do k=0,1
        !---calcolo la max distanza-----
        !---dei vertici del cubo dall'asse-----
        p(1)=- (vetz(liv_rif)*m+rbi)+2*eps+      &
&      (i-1+k)*(2*(vetz(liv_rif)*      &
&      m+rbi)-4*eps)/nscxy
        do k1=0,1

```

```

        p(2)=-(vetz(liv_rif)*m+rbi)+2*eps+      &
        &      (j-1+k1)*(2*(vetz(liv_rif)*      &
        &      m+rbi)-4*eps)/nscxy
        r= sqrt(p(1)**2+p(2)**2)
        if (r>rmax) rmax=r
    end do
end do
p(3)=max(vetz(3*h+1)*m+rbi, vetz(3*h-2)*m+rbi)
if (rmax<= p(3)+eps) then

    !---tutti i vertici sono dentro-----

        inbic(i,j,h)=.TRUE.
        nsct=nsct+1
        nsc((h-1)/2+1)=nsc((h-1)/2+1)+1
    end if
end do
end do
end do

write(30,*) 'numero di scatole totali=',nsct
do i=1,nzone
    write(30,*) '---numero di scatole sulla zona---',&
    &      i, '=', nsc(i)
end do

!---scelgo i punti da seguire-----

allocate(npzona(nzone),stat=i)
if (i/=0) STOP      &
&      '---allocazione di npzona fallita---'
rmax=max(rbs,rbi)
npsegric=0 !---numero di punti seguiti-----
npzona=0 !---numero di punti seguiti-----
!---nella zona k-----
npunti=0 !---numero di punti-----
!---sul piano k=0,7(serve solo di stampa)
do k=0,7
    do j=0,150
        p(3)=lz/150*j
        rbic=m*p(3)+rbi
        do i=3+6*k,3+6*7+5
            p(1)=rmax/75*i*cos(k*pgr/8)
            p(2)=rmax/75*i*sin(k*pgr/8)
            r=sqrt(p(1)**2+p(2)**2)
            if (r<=rbic) then
                do h=1,12

```

```

        if (p(3)>=vetz(1+(h-1)*6) .and.      &
        & p(3)<vetz(1+h*6)) then
            npunti(k)=npunti(k)+1
            npsegric=npsegric+1
            npzona(h)=npzona(h)+1
        end if
    end do
else
    exit
end if
end do
end do
write(*,*) 'contati i punti dei piani seguiti=', &
& npsegric
write(30,'(a,i8)') 'NPUNTI:',npsegric
write(30,*) 'npunti sugli 8 piani',      &
& (npunti(k),k=0,7)

!---inserisco pospt e vel3pt-----

dimensione=maxval(npzona)
write(*,*) dimensione
allocate(pospt(dimensione,nzone,3),      &
& vel3vpt(dimensione,nzone,3),stat=i)
if (i/=0) STOP      &
& '---allocazione di pospt fallita---'
vel3vpt=0.d0
npsegric=0 !---numero di punti seguiti-----
npzona=0 !---numero di punti seguiti-----
!---nella zona k-----
do k=0,7
    do j=0,150
        p(3)=lz/150*j
        rbic=mbordo*p(3)+rbi
        do i=3+6*k,3+6*7+5
            p(1)=rmax/75*i*cos(k*pgr/8)
            p(2)=rmax/75*i*sin(k*pgr/8)
            r=sqrt(p(1)**2+p(2)**2)
            if (r<=rbic) then
                do h=1,12
                    if (p(3)>=vetz(1+(h-1)*6) .and.      &
                    & p(3)<vetz(1+h*6)) then
                        npsegric=npsegric+1
                        npzona(h)=npzona(h)+1
                        pospt(npzona(h),h,:)=p
                    end if
                end do
            end if
        end do
    end do
end do

```



```

                !---p e' un punto sulla zona h-----
            end if
        end do
    else
        exit
    end if
end do
end do
end do
write(*,*) 'contati i punti dei piani seguiti =', &
&         npsegric
write(30,'(a,i8)') 'NPUNTI:',npsegric

write(30,*) 'npsegric=',npsegric
write(30,*) (('npzona',i,'=',npzona(i)),i=1,nzone)
write(30,*) 'dimensione=',dimensione
write(*,*) 'scelti i punti'

!---Alloco la Scatola-----

allocate(scatola(nscxy,nscxy,nscz),           &
&        scatolat(nscxy,nscxy,nscz),         &
&        scatolan(nscxy,nscxy,nscz),         &
&        scatolan1(nscxy,nscxy,nscz),        &
&        nps_liv(10,nzone,nliv),stat=i)
if (i/=0) STOP                               &
&        '---Allocazione di Scatola Fallita---'
nps_liv=0

!*****
!*****INIZIA LA SIMULAZIONE*****
!*****

!---Contiamo i punti da seguire-----

npunti_seg=0
do x=1,nx/2+1
    p(1)=-mlx+(x-1)*deltaxsim
    do y=1,ny/2+1
        p(2)=-mly+(y-1)*deltaxsim
        r=sqrt(p(1)**2+p(2)**2)
        do z=nzi-4,nzs+4
            p(3)=(z-1)*deltaxsim
            rbic=p(3)*mbordo+rbi
            if(r<=rbic) npunti_seg=npunti_seg+1
        end do
    end do
end do

```

```

    end do
end do
write(*,*) '---punti seguiti---',npunti_seg

!---Allochiamo memoria e-----
!---Inizializziamo vel3v-----

allocate(posizione(npunti_seg,3),           &
&        vel3v(npunti_seg,3),stat=i)
if (i/=0) stop                               &
&    '---MAIN (pos/vel): allocazione fallita!---'
vel3v=0.d0

!---Riempiamo Posizione-----

i=0
do x=1,nx/2+1
  p(1)=-mlx+(x-1)*deltaxsim
  do y=1,ny/2+1
    p(2)=-mly+(y-1)*deltaxsim
    r=sqrt(p(1)**2+p(2)**2)
    do z=nzi-4,nzs+4
      p(3)=(z-1)*deltaxsim
      rbic=p(3)*mbordo+rbi
      if(r<=rbic) then
        i=i+1
        posizione(i,:)=p
      end if
    end do
  end do
end do

!---Inizializziamo 'node,n_hlp,mas_seg'----

call init_density()

!---stampiamo la configurazione iniziale---
!---e annulliamo 'mas_seg'-----

write(*,*) 'iterata=',0
open(10,file='s0-1.out')
mtot=0.e0
do y=ny,1,-1
  p(2)=-mly+(y-1)*deltax
  do x=1,nx
    p(1)=-mlx+(x-1)*deltax

```

```
r=sqrt(p(1)**2+p(2)**2)
do z=1,nz
  p(3)=(z-1)*deltax
  rbic=p(3)*mbordo+rbi
  if (r<=rbic) then
    massa_loc=sum(node(:,x,y,z))
    mtot=mtot+massa_loc
    write(10,'(1x,e12.5)') mas_seg(x,y,z)
    mas_seg(x,y,z)=0.e0
  else
    write(10,'(1x,e12.5)') -10.0e0
    mas_seg(x,y,z)=0.e0
  end if
end do
end do
write(*,*) 'massa globale = ',mtot
```

```
!---Iniziamo ad iterare sul tempo-----
tempop=1
do time=1,t_max
  if(time==p_stampa*50+1) then
    close(10)
    open(10,file='s1-2.out')
  else if(time==2*p_stampa*50+1) then
    close(10)
    open(10,file='s2-3.out')
  else if(time==3*p_stampa*50+1) then
    close(10)
    open(10,file='s3-4.out')
  else if(time==4*p_stampa*50+1) then
    close(10)
    open(10,file='s4-5.out')
  else if(time==5*p_stampa*50+1) then
    close(10)
    open(10,file='s5-6.out')
  else if(time==6*p_stampa*50+1) then
    close(10)
    open(10,file='s6-7.out')
  else if(time==7*p_stampa*50+1) then
    close(10)
    open(10,file='s7-8.out')
  else if(time==8*p_stampa*50+1) then
    close(10)
    open(10,file='s8-9.out')
  else if(time==9*p_stampa*50+1) then
```

```

        close(10)
        open(10,file='s9-10.out')
    else if(time==10*p_stampa*50+1) then
        close(10)
        open(10,file='s10-11.out')
    else if(time==11*p_stampa*50+1) then
        close(10)
        open(10,file='s11-12.out')
    else if(time==12*p_stampa*50+1) then
        close(10)
        open(10,file='s12-13.out')
    else if(time==13*p_stampa*50+1) then
        close(10)
        open(10,file='s13-14.out')
    end if
    tcorr=time*deltat
    call propagate()
    if (time==i_mis*p_mis) then
        i_mis=i_mis+1
        write(*,*) 'iterata=',time
        write(30,*) 'iterata= ',time
        if (time==i_stampa*p_stampa) then
            call relax_wr(omega)
            i_stampa=i_stampa+1
        else
            call relaxation(omega)
        end if
    end if
    !---Miscelazione-----
    npunti_persi=0
    scatolat=.FALSE.
    scatolan=.FALSE.
    scatolan1=.FALSE.
    do k=1,nzone
        scatola=.FALSE.
        do h=1,npzona(k)
            p=pospt(h,k,:)
            r=sqrt(p(1)**2+p(2)**2)
            rbic=p(3)*m+rbi
            if (r<=rbic .and. p(3)>=0.0d0      &
                & .and. p(3)<=lz) then
                ix=tind(p(1),vetxy,10)
                !---la ix della scatola dove sta p---
                iy=tind(p(2),vetxy,10)
                !---la ix della scatola dove sta p---
                iz=tind(p(3),vetz,72)
                !---in che livello sta p-----
                if (iz/=-1) then

```

```

        if (ix /= -1 .and. iy /= -1 ) then
            scatola(ix,iy,((iz-1)/3+1))=.TRUE.
            scatolat(ix,iy,((iz-1)/3+1))=.TRUE.
            if (k==2 .or. k==3) &
            & scatola(ix,iy,((iz-1)/3+1))=.TRUE.
            if (k>=1 .and. k<=4) &
            & scatola1(ix,iy,((iz-1)/3+1))=.TRUE.
        end if
        nps_liv(tempop,k,iz)= &
        & nps_liv(tempop,k,iz)+1
    end if
else
    npunti_persi=npunti_persi+1
end if
end do

nscr=0
nscrt=0
do i=1,10
    do j=1,10
        do h=1,24
            if (inbic(i,j,h) .and. &
            & scatola(i,j,h)) then
                nscr((h-1)/2+1)=nscr((h-1)/2+1)+1
                nscrt=nscrt+1
            end if
        end do
    end do
end do

n_file=69+k

!---nscrt=numero di scatole-----
!---ricoperte da punti partiti-----
!---dalla zona k-----

write(n_file,*) nscrt*1.0/nsct

!---dispersione dei punti-----
!---partiti dalla zona k-----

end do

nscrt=0
nscrn=0
nscrn1=0
do i=1,10

```

```

do j=1,10
  do h=1,24
    if (inbic(i,j,h)) then
      if(scatolat(i,j,h)) nscrt=nscrt+1
      if(scatolan(i,j,h)) nscrn=nscrn+1
      if(scatolan1(i,j,h)) nscrn1=nscrn1+1
    end if
  end do
end do
end do

!---nscrn1=numero di scatole-----
!---ricoperte dai punti partiti-----
!---dalle zone 1-4-----

write(83,*) nscrn1*1.0/nsct

!---dispersione dei punti partiti-----
!---dalle zone 1-4-----
!---nscrn=numero di scatole-----
!---ricoperte dai punti partiti-----
!---dalle zone 2-3-----

write(84,*) nscrn*1.0/nsct

!---dispersione dei punti partiti-----
!---dalle 2-3-----
!---nscrt=numero di scatole-----
!---ricoperte dai punti seguiti-----

write(82,*) nscrt*1.0/nsct

!---dispersione dei punti seguiti-----

if(tempop==10) then
  do k=1,nzone
    k1=k+50
    write(k1,*) (sum(nps_liv(:,k,j))*      &
      &      invol(j)/10,j=1,72)
  end do
  nps_liv=0

  !---ogni 10 iterate prima-----
  !---tempop=1-----

else
  tempop=tempop+1

```

```

        end if

        !---Calcolo del Tempo necessario-----
        !---Alla Simulazione-----

        call X05AAF(itime)
        t2=itime(6)+60*(itime(5)+60*itime(4))
        write(*,*) 'tempo=',t2-t1
        write(*,*) 'punti seguiti per la diff-disp=',&
        &          npsegric,'punti persi=',          &
        &          npunti_persi
        write(30,*) 'diffus-disp',          &
        &          npsegric,npunti_persi

        call aggiorna_pos()
    else
        call relaxation(omega)
    end if
end do
end program MAIN

!-----
!---Finisce il 'Programma Principale'-----
!-----

!---Inizia la 'Function VMEDIA'-----

real(4) function VMEDIA()

use procedure

integer  :: x,y,z,contap
real(4)  :: p(3),r,vel

vmaxdisc=0.e0
tcorr=1.e0
contap=0
vmedia=0.e0
do x=nxyi,nxys
    p(1)=-mlx+(x-1)*deltax
    do y=nxyi,nxys
        p(2)=-mly+(y-1)*deltax
        r=sqrt(p(1)**2+p(2)**2)
        do z=nzi,nzs
            p(3)=(z-1)*deltax
            if( r<=rdisc ) then

```

```
        contap=contap+1
        if(r>tol) then
            vel=sqrt(sum((dx_dt*vbndry(p))**2))
            if(vel>vmaxdisc) vmaxdisc=vel
            vmedia=vmedia+vel
        end if
    end if
end do
end do
end do

vmedia=vmedia/contap

end function VMEDIA
```



# Bibliografia

- [1] Andrew K.Gunstensen, Daniel H.Rothman, S.Zaleski, G.Zanetti  
*Lattice Boltzmann Model of Immiscible Fluids*,  
Physical Review A, 43 (8):4320-4327, (1991).
  
- [2] Augé, Gillon, Hollier-Larousse  
*Idrodinamica*,  
RCS Rizzoli Libri, (1989).
  
- [3] Augé, Gillon, Hollier-Larousse  
*Il Numero di Reynolds*,  
RCS Rizzoli Libri, (1989).
  
- [4] Bastien Chopard e Michel Droz  
*Cellular Automata Modelling of Physical Systems*,  
University of Geneva,  
Cambridge University Press, (1998).
  
- [5] Daniel H. Rothman e Jeffrey M. Keller  
*Immiscible Cellular-Automaton Fluids*,  
Journal of Statistical Physics, 52 (3/4):1119-1127, (1988).

- [6] David P.Lockard, Bart A.Singer, Li-Shi Luo,  
*“Evaluation of the Lattice-Boltzmann Equation Solver PowerFLOW for Aerodynamic Applications”*,  
Langley Research Center Hampton, Virginia, (Ottobre 2000).
- [7] Dieter A.Wolf-Gladrow,  
*“Lattice-Gas Cellular Automata and Lattice Boltzmann Models,”*  
*(An Introduction)*,  
Springer, (2000).
- [8] E. G. Flekkøy,  
*“Lattice Bhatnagar-Gross-Krook Model for Miscible Fluids,”*  
Physical Review E, 47 (6):4247-4257, (1993).
- [9] Hibbard B., Paul B., Battaiola A.,  
*Vis5d , Version 3.0 .*
- [10] James Maxwell Buick  
*“Lattice Boltzmann Methods in Interfacial Wave Modelling”*,  
International Journal for Numerical Methods in Fluids, (1997).
- [11] Jorg R. Weimar  
*Three-dimensional Cellular Automata for Reaction-Diffusion Systems*,  
Institute of Scientific Computing, Technical University Braunschweig,  
Germany.

- 
- [12] N. Egidi, L. Misici e R. Piergallini  
*Laminar Mixing Flow in a Stirred Vessel*,  
Task Quarterly, 5, (No 1), (2001), 71-83.
- [13] Renwei Mei, Wei Shyy, Dazhi Yu e Li-Shi Luo  
*“Lattice Boltzmann Method for 3-d Flows with Curved Boundary”*  
Journal of Computational Physics **161**, 680-699 (2000).
- [14] Shiyi Chen e Gary D.Doolen  
*“Lattice Boltzmann Method for Fluid Flow”*,  
Annual Review Fluid Mechanics , (1998).30:329-64.
- [15] Stefano Zucchelli  
*Il Principio di Massima Entropia*,  
<http://rcs.re.it/rcs/ruffini/compiani/node3.html>, (1998).
- [16] Stephen Wolfram  
*Cellular Automata and Complexity*  
Collected Papers, (1994).