

POLITECNICO DI TORINO

Collegio di Ingegneria Chimica e dei Materiali

**Corso di Laurea Magistrale
in Ingegneria Chimica e dei Processi Sostenibili**

Tesi di Laurea Magistrale

**Dissipative particle dynamics come
strumento per reologia computazionale:
applicazione a fluidi semplici e complessi**



Relatori

Prof. Daniele Marchisio
Prof. Antonio Buffo
Prof. Gianluca Boccardo

Candidato

Nunzia Lauriello

Marzo, 2021

Indice

Introduzione	11
I	13
1 Elementi introduttivi di dissipative particle dynamics	15
1.1 Il modello DPD: un metodo <i>coarse-graining</i>	18
1.2 Equazioni di governo	18
1.3 Aspetti teorici	20
1.3.1 Equazione di Fokker-Plank	20
1.3.2 Comprimibilità isoterma ed equazione di stato	22
1.3.3 Teoria di Flory-Huggins	23
1.4 Schemi di integrazione temporale	24
1.5 Fattori di conversione e unità DPD	26
2 Dissipative particle dynamics per stima di proprietà di trasporto	28
2.1 Funzione peso modificata	30
2.2 Sforzi di taglio	32
2.3 Forza dissipativa rotazionale	33
3 Calcolo delle proprietà di trasporto	35
3.1 Relazioni analitiche tra i parametri DPD e le proprietà di trasporto	35
3.2 Metodi per il calcolo della viscosità nelle simulazioni DPD	36
3.2.1 Equazioni di Green-Kubo	37
3.2.2 Equazioni di Einstein-Helfand	38
3.2.3 Metodo di Müller-Plathe (momentum interchange)	38
3.2.4 Metodo delle condizioni al contorno di Lees-Edwards	39
3.2.5 Metodo Force-driven Poiseuille flow	41
3.3 Calcolo del coefficiente di diffusione	42

4	Introduzione ai fluidi complessi	44
4.1	I polossameri	44
4.2	Formazione ed evoluzione delle micelle	47
4.3	Proprietà reologiche	49
II		51
5	Condizioni operative investigate e Dettagli computazionali	53
5.1	Parametri del modello DPD utilizzati	54
5.2	Dettagli delle simulazioni: esplorazione dello spazio dei parametri	60
5.3	Calcolo del numero di Schmidt in LAMMPS	63
5.3.1	Procedura per il calcolo della viscosità dinamica basata sulla formula di Green-Kubo	63
5.3.2	Il metodo di Einstein-Helfand come riferimento	66
5.3.3	Il metodo di Müller-Plathe come riferimento	68
5.3.4	Calcolo del coefficiente di diffusione	69
5.4	Algoritmo di clustering	69
6	Presentazione e discussione dei risultati	71
6.1	Calcolo delle proprietà di trasporto di un fluido semplice assimilabile all'acqua	71
6.2	Risultati di simulazione della miscela binaria costituita da acqua e Pluronic L64	85
6.2.1	Proprietà strutturali	85
6.2.2	Calcolo della viscosità a scorrimento nullo	90
	Conclusioni	92
A	Descrizione degli input script in LAMMPS	95
B	Algoritmo di clustering	103
	Bibliografia	108

Elenco delle figure

1.1	Dissipative Particle Dynamics: una tecnica meso-scala per colmare il divario tra la micro- e la macro-scala (elaborazione immagine estratta da Ref. [3]).	16
2.1	Andamento della viscosità dinamica e del numero di Schmidt in funzione del raggio di <i>cutoff</i> per $\rho = 4$, $\gamma = 4.5$ e $k_B T = 1$ (elaborazione immagine estratta da Ref. [25]).	29
2.2	Funzione peso generalizzata relativa alla forza dissipativa (immagine adattata da Ref. [25]).	31
2.3	Contributo dissipativo della viscosità dinamica in funzione dell'esponente della funzione peso generalizzata che definisce la dipendenza dalla distanza interparticellare della forza dissipativa (immagine adattata da Ref. [25]).	32
3.1	Schematizzazione dello scambio di quantità di moto artificiale, <i>swapping</i> , tra i <i>beads</i> nel metodo di Müller-Plathe (elaborazione immagine estratta da Ref. [36]).	39
3.2	Flusso fisico di quantità di moto generato, come risposta alla perturbazione esterna imposta, dal sistema (immagine adattata da Ref. [36]).	40
3.3	Rappresentazione schematica del metodo di Poiseuille (immagine adattata da Ref. [35]).	42
4.1	Struttura generale di una molecola di Pluronic [®] (elaborazione immagine estratta da Ref. [45]).	45

4.2	Griglia del Pluronic [®] (elaborazione immagine estratta da Ref. [45]). I tre colori (verde, rosa e arancione) raggruppano le tipologie di Pluronic, in base allo stato fisico di aggregazione a temperatura ambiente, rispettivamente in (<i>liquid</i>), L, (<i>paste</i>), P, e (<i>flake</i>), F. Lungo le linee verticali (dal basso verso l'alto) si ha un incremento del peso molecolare del blocco di PPO contenuto nella catena polimerica. Viceversa, lungo le linee orizzontali (da sinistra a destra) si ha un incremento della percentuale in massa di EO nella catena polimerica. I numeri nei riquadri sono relativi al Pluronic L64.	46
4.3	Rappresentazione schematica delle strutture, micellare (a), cubica (b), esagonale (c) e lamellare, formate dal Pluronic in solvente acquoso (elaborazione immagine estratta da Ref. [52]).	48
4.4	Diagramma di fase della miscela binaria Pluronic L64-acqua (elaborazione immagine estratta da Ref. [46]). L ₁ : fase micellare. E: fase esagonale. D: fase lamellare. L ₂ : fase micellare inversa.	49
5.1	<i>Beads</i> rappresentativi della componente acquosa per diversi numeri di <i>coarse-graining</i>	54
5.2	Formula di struttura della molecola di Pluronic L64 (elaborazione immagine estratta da Ref. [45]).	55
5.3	Modello DPD risultante dalla procedura di <i>coarse-graining</i> di una molecola di Pluronic L64.	56
5.4	Interazioni conservative tra le diverse tipologie di <i>beads</i> costituenti il sistema: A(PEO), B(PPO) e C(H ₂ O).	57
5.5	Esempio di violazione topologica per l'incrocio non fisico dei legami (linea verde e linea nera) tra i <i>beads</i> della catena polimerica rappresentati dai cerchi pieni, noto come <i>bond crossing</i> (elaborazione immagine estratta da Ref. [62]). Il vettore di distanza minima (in rosso) cambia il suo orientamento di 180° tra gli istanti di tempo t e $t + \Delta t$. La configurazione (a) è relativa all'istante di tempo t . La configurazione (b) è relativa all'istante di tempo $t + \Delta t$	58
5.6	Esempio di una tipica conformazione spaziale assunta da una catena polimerica che si ottiene facendo interagire i legami tra i <i>beads</i> non vicinali della macromolecola mediante mSRP, senza l'aggiunta di un potenziale repulsivo angolare (immagine adattata da Ref. [62]).	59

6.1	In alto: Procedura A. Cerchi neri: punti della SACF. Linea rossa: legge di potenza interpolante. In basso: Procedura B. Linea nera: SACF dopo il filtro a media mobile. Linea rossa: legge di potenza interpolante.	72
6.2	Cerchi neri: punti della coda rumorosa della SACF tal quale. Linea rossa: legge di potenza interpolante la coda rumorosa della SACF dopo l'applicazione del filtro a media mobile. . .	73
6.3	Procedura C. Integrale cumulativo di viscosità calcolata con la formula di Green–Kubo. Linea nera: integrale cumulativo di viscosità. Linea rossa: valore medio calcolato tra $t = 5$ e $t = 10$	74
6.4	Effetto del grado di <i>coarse-graining</i> N_m sulla SACF per due differenti valori dell'esponente della funzione peso generalizzata s	78
6.5	Effetto del grado di <i>coarse-graining</i> N_m sul numero di Schmidt (in alto), sulla viscosità dinamica (al centro) e sul coefficiente di diffusione (in basso).	80
6.6	Effetto dell'esponente della funzione peso generalizzata s (in alto), del raggio di <i>cutoff</i> dissipativo r_c^D (al centro) e del coefficiente di forza dissipativa γ (in basso) sulla SACF. . . .	81
6.7	Effetto dell'esponente della funzione peso generalizzata s sui coefficienti di trasporto: viscosità dinamica (a sinistra) e diffusività (a destra).	82
6.8	Effetto dell'esponente del raggio di <i>cutoff</i> r_c^D sui coefficienti di trasporto: viscosità dinamica (a sinistra) e diffusività (a destra).	82
6.9	Effetto del coefficiente di forza dissipativa γ sui coefficienti di trasporto: viscosità dinamica (a sinistra) e diffusività (a destra).	82
6.10	Effetto dell'esponente della funzione peso generalizzata s (in alto), del raggio di <i>cutoff</i> dissipativo r_c^D (al centro) e del coefficiente di forza dissipativa γ (in basso) sul numero di Schmidt.	83
6.11	Funzione di distribuzione radiale (RDF) valutata per i tre differenti casi con numero di Schmidt prossimo a 370, ovvero a quello dell'acqua a $25^\circ C$	85
6.12	<i>Snapshot</i> di una miscela di Pluronic L64 e acqua all'equilibrio, con concentrazione percentuale in massa di copolimero del 5%. I parametri del potenziale armonico impiegati sono: $r_0 = 1$ e $K_b = 265$. I <i>beads</i> di colore rosso rappresentano i gruppi di PPO.	86

- 6.13 *Snapshot* di una miscela di Pluronic L64 e acqua all'equilibrio, con concentrazione percentuale in massa di copolimero del 5%. I parametri del potenziale armonico impiegati sono: $r_0 = 1$ e $K_b = 700$. I *beads* di colore rosso rappresentano i gruppi di PPO. 87
- 6.14 Risultato della simulazione di una miscela di Pluronic L64 e acqua, con concentrazione percentuale in massa di copolimero del 5%. I parametri del potenziale armonico impiegati sono: $r_0 = 1$ e $K_b = 265$. A sinistra la rappresentazione dei vari *clusters* con differenti colori identificati dall'algoritmo di *clustering*; a destra il numero totale di *clusters* (in ordinata) in funzione del numero di aggregazione N (in ascissa). 87
- 6.15 Risultato della simulazione di una miscela di Pluronic L64 e acqua, con concentrazione percentuale in massa di copolimero del 5%. I parametri del potenziale armonico impiegati sono: $r_0 = 1$ e $K_b = 700$. A sinistra la rappresentazione dei vari *clusters* con differenti colori identificati dall'algoritmo di *clustering*; a destra il numero totale di *clusters* (in ordinata) in funzione del numero di aggregazione N (in ascissa). 88
- 6.16 *Cluster mass distribution* (CMD) di una miscela di Pluronic L64 e acqua, con concentrazione percentuale in massa di copolimero del 5%. I parametri del potenziale armonico impiegati sono: $r_0 = 1$ e $K_b = 265$ 88
- 6.17 *Cluster mass distribution* (CMD) di una miscela di Pluronic L64 e acqua, con concentrazione percentuale in massa di copolimero pari a 5%. I parametri del potenziale armonico impiegati sono: $r_0 = 1$ e $K_b = 700$ 89

Elenco delle tabelle

1.1	Fattori di conversione di massa, lunghezza, energia e tempo per tre valori caratteristici del numero di <i>coarse-graining</i> . Il fattore di conversione dell'unità di tempo è ricavato imponendo il <i>matching</i> tra il valore sperimentale del coefficiente di autodiffusione dell'acqua e quello ottenuto dalla simulazione DPD.	27
2.1	Proprietà di trasporto per due diversi valori di s espresse in funzione degli altri parametri DPD.	31
5.1	Gradi di <i>coarse-graining</i> (N_m) adottati per i <i>beads</i> rappresentativi delle molecole d'acqua e corrispondente coefficiente di repulsione (a).	55
5.2	<i>Coarse-grained</i> (CG) <i>beads</i> , volumi molecolari di H ₂ O, EO e PO calcolati in accordo con l'approccio seguito da Durchschlag e Zipper [56] e volumi dei <i>beads</i> corrispondenti espressi in Å ³	56
5.3	Parametri di interazione a_{ij} , riportati in unità DPD, per le tre specie, PEO, PPO e acqua.	57
5.4	Parametri di solubilità χ_{ij} per le tre specie, PEO, PPO e acqua (valori estratti da Ref. [54]).	57
5.5	Parametri del potenziale armonico, espressi in unità DPD, calcolati secondo l'approccio di Anderson e collaboratori [55].	58
5.6	Studio parametrico per valutare l'effetto del grado di <i>coarse-graining</i> sulla riproduzione delle proprietà di trasporto dell'acqua.	61
5.7	Studio parametrico per valutare l'effetto del grado di s, γ, r_c^D sulla riproduzione delle proprietà di trasporto dell'acqua.	62
5.8	Studio parametrico per il calcolo della SACF in LAMMPS.	65
5.9	Relazione tra parametri di correlazione e <i>timestep</i>	66

6.1	Valori di viscosità (in unità DPD) calcolati con il metodo di equilibrio di Green-Kubo seguendo diverse procedure. A^1 : interpolazione con la legge di potenza della coda rumorosa della SACF e calcolo dell'integrale analitico fino al tempo ultimo di simulazione disponibile. A^2 : interpolazione con la legge di potenza della coda rumorosa della SACF e calcolo dell'integrale analitico fino a ∞ . B^1 : interpolazione con la legge di potenza della coda rumorosa della SACF dopo l'applicazione del filtro a media mobile e calcolo dell'integrale analitico fino al tempo ultimo di simulazione disponibile. B^2 : interpolazione con la legge di potenza della coda rumorosa della SACF dopo l'applicazione del filtro a media mobile e calcolo dell'integrale analitico fino a ∞ . $C(media)$: valore medio della curva cumulativa di viscosità calcolato considerando solo le ultime cinque unità di tempo DPD. $C(std)$: deviazione standard. $C(COV)$: coefficiente di variazione.	75
6.2	Valori di viscosità (in unità DPD) calcolati con i metodi di riferimento di equilibrio e non equilibrio, rispettivamente Einstein-Helfand e Müller-Plathe.	75
6.3	Effetto del numero di <i>coarse-graining</i> e dei parametri DPD sulla viscosità, sul coefficiente di diffusione e sul numero di Schmidt. Tutte le grandezze dimensionate sono espresse in unità DPD.	77
6.4	Risultati per tre differenti terne di parametri DPD (s, r_c^D, γ) per le quali si ha un numero di Schmidt prossimo a 370, ovvero a quello dell'acqua a 25°C. Le unità di misura non specificate sono unità DPD.	84
6.5	Numero totale di <i>clusters</i> identificati all'equilibrio dall'algoritmo di <i>clustering</i> per diversi valori dei parametri del potenziale armonico (riportati in unità DPD), descrittivi le interazioni di legame interne alla catena polimerica.	89
6.6	Valori sperimentali di viscosità [72] della miscela costituita da acqua e Pluronic L64. x_1 è la frazione in massa della componente acquosa e $x_2 = 1 - x_1$ è la frazione in massa di copolimero in miscela. η_S è il valore di viscosità dinamica sperimentale espresso in (Pa×s).	90

-
- 6.7 Studio parametrico per valutare l'effetto di s , γ e r_c^D sulla viscosità della miscela binaria costituita da acqua e Pluronic L64. I pedici dei parametri indicano le tipologie di *beads* che interagiscono: quello rappresentativo dell'acqua è designato con pedice 1, quelli rappresentativi delle unità ripetitive della catena di copolimero, PEO e PPO, sono designati con pedice 2. x_2 è la concentrazione di copolimero nella miscela espressa come frazione in massa. η_C è il valore di viscosità dinamica ottenuto dalla simulazione convertito in unità fisiche reali, quindi espresso in (Pa×s). 91

Introduzione

Questo lavoro di tesi si propone di investigare l'uso della Dissipative Particle Dynamics (DPD), una tecnica di simulazione mesoscala, come strumento per la reologia computazionale, con particolare attenzione all'effetto di alcuni parametri del modello sulle proprietà reologiche predette, e la sua applicazione a fluidi semplici e complessi. I fluidi strutturati sono ampiamente diffusi nelle industrie alimentari, cosmetiche, farmaceutiche e includono i liquidi e i fusi polimerici, le sospensioni di particelle colloidali, le soluzioni di micelle e le schiume liquide. Le loro proprietà reologiche sono estremamente importanti in quanto definiscono le loro possibili applicazioni e giocano un ruolo cruciale nella definizione di regole di progettazione e di scale-up dei processi e delle apparecchiature in cui sono trattati. Hanno proprietà intermedie tra i liquidi e i solidi ordinari. Essi possono creare una vasta gamma di microstrutture, che possono evolvere nel tempo assumendo morfologie completamente differenti. Il loro comportamento reologico è complesso e fortemente influenzato dalla forma e dalle dimensioni delle microstrutture create, le quali sono caratterizzate da scale spaziali maggiori rispetto alla scala atomica. I metodi di simulazione basati sulla Dinamica Molecolare convenzionale non sono quindi adatti per la descrizione di fluidi strutturati, in quanto si basano su un approccio completamente atomistico che comporterebbe elevati costi computazionali. Per ovviare tale limitazione, la Dissipative Particle Dynamics (DPD) è impiegata in questo lavoro di tesi. La DPD è considerata una delle tecniche di simulazione più flessibili: i fluidi strutturati sono simulati a scale spaziali e temporali fisicamente interessanti. Questo metodo rientra tra i modelli *coarse-grained*, nei quali gruppi di atomi e molecole, i cosiddetti *beads*, sono trattati come entità singole, ognuna rappresentativa di una componente molecolare del sistema complesso. È, pertanto, possibile simulare il comportamento idrodinamico di fluidi strutturati con un numero di particelle decisamente minore e riducendo il costo computazionale. L'obiettivo di questo lavoro di tesi è quello di definire una procedura di calcolo della viscosità affidabile e sviluppare un modello DPD, attraverso l'individuazione di parametri *ad hoc*, in grado di riprodurre correttamente le proprietà di trasporto sia di un fluido semplice, come l'acqua, che di un fluido strutturato, come la miscela costituita da acqua e Pluronic L64. Il presente lavoro si suddivide in due parti. Nella Parte I si forniscono le nozioni teoriche generali.

Il Capitolo 1 introduce la DPD descrivendo il modello secondo la sua formulazione originale, esponendone i fondamenti teorici, il modello matematico e gli algoritmi per l'integrazione numerica delle sue equazioni. Il Capitolo 2 presenta le modifiche introdotte al modello DPD al fine di renderlo adeguato per la stima delle proprietà di trasporto dei fluidi simulati. Il Capitolo 3 descrive le tecniche più usate per il calcolo delle proprietà di trasporto, quali la viscosità e il coefficiente di diffusione, di un fluido DPD e presenta le relazioni analitiche tra queste e i parametri del modello. Il Capitolo 4 introduce i fluidi complessi, facendo riferimento principalmente ai polissamperi, copolimeri anfifilici a blocchi, e al loro comportamento reologico in presenza di un solvente selettivo come l'acqua. Nella Parte II, si forniscono le informazioni relative ai dettagli computazionali e alle condizioni operative investigate e si presentano e commentano i risultati ottenuti dalle simulazioni. Il Capitolo 5 descrive lo sviluppo del modello *coarse-grained* e i dettagli di simulazione. Inoltre, spiega a grandi linee l'implementazione delle procedure per il calcolo delle proprietà di trasporto e il funzionamento dell'algoritmo di *clustering*, impiegato per l'analisi quantitativa dei risultati di simulazione. Nel Capitolo 6 si presentano e discutono: la procedura sviluppata per il calcolo della viscosità; i risultati ottenuti dalle simulazioni relativi alle proprietà di trasporto di un fluido semplice, rappresentativo dell'acqua, e al comportamento strutturale e reologico della miscela binaria costituita da acqua e Pluronic L64 in funzione dei parametri di modello utilizzati. In Appendice A e B si riportano alcuni dei codici impiegati, e in parte sviluppati, in questo lavoro di tesi e si forniscono le linee guida per l'implementazione del modello DPD.

Parte I

Capitolo 1

Elementi introduttivi di dissipative particle dynamics

La modellazione computazionale negli ultimi decenni è diventata un supplemento sempre più prezioso alla sperimentazione e alla teoria in virtù di un notevole incremento della potenza di calcolo e della migliorata efficienza dei metodi di simulazione. In particolare, una delle sfide dell'ingegneria chimica moderna è la modellazione di fluidi complessi e materia soffice in generale. Molti dei fluidi reali che si trovano nell'industria alimentare, farmaceutica, cosmetica e biochimica sono, infatti, fluidi complessi. Essi possono creare una molteplicità di microstrutture che evolvono nel tempo assumendo forme e dimensioni completamente diverse. Tipicamente sono caratterizzate da una scala di lunghezza che si colloca tra quella atomistica e quella macroscopica rendendo rilevante la compresenza di scale spaziali e temporali differenti. Molti dei fenomeni interessanti dal punto di vista tecnologico che si verificano nei fluidi complessi avvengono alla mesoscala. Quest'ultima può essere approssimativamente definita come scale spazio-temporali comprese rispettivamente tra $10 - 10^4$ nm e $1 - 10^6$ ns [1]. Le tecniche di simulazione convenzionali sono impraticabili. L'approccio atomistico basato sulla dinamica molecolare (MD) risolve numericamente le equazioni relative al moto degli atomi e delle molecole coinvolte e consente di catturare tutti i dettagli microscopici del sistema. In particolare, nella MD classica vengono trascurati i gradi di libertà elettronici e le interazioni fra atomi sono descritte con campi di forza (in inglese *force field*) specifici. Vengono invece descritti tutti i gradi di libertà vibrazionali, traslazionali e rotazionali, risultando in tempi di calcolo molto elevati. Quindi, questo approccio applicato alla mesoscala risulterebbe in un eccessivo costo computazionale. All'estremo opposto, le simulazioni su scala macroscopica trattano la materia come un corpo continuo descrivendone il comportamento tramite le equazioni idrodinamiche di Navier-Stokes. Si basano essenzialmente su due concetti chiave: il limite del continuo, ogni elemento di volume contiene un numero enorme di

atomi, e l'ipotesi di equilibrio locale, questi volumi sono abbastanza grandi da riprodurre le proprietà termodinamiche dell'intero sistema. Le grandezze fisiche cambiano poco passando da un elemento di volume al suo vicino, pertanto il comportamento del sistema può essere descritto alle scale più grandi mediante equazioni differenziali che, in molte situazioni, ammettono soluzione analitica. Tuttavia, per fenomeni che si manifestano su una scala intermedia, la natura della materia, trascurabile a livello macroscopico, si manifesta. Per colmare il divario tra le simulazioni atomistiche e quelle su larga scala sono necessarie tecniche di simulazione intermedie volte ad una scala di lunghezza maggiore di quella atomistica, ma inferiore a quella macroscopica [2]. D'altra parte c'è la possibilità di utilizzare la teoria del

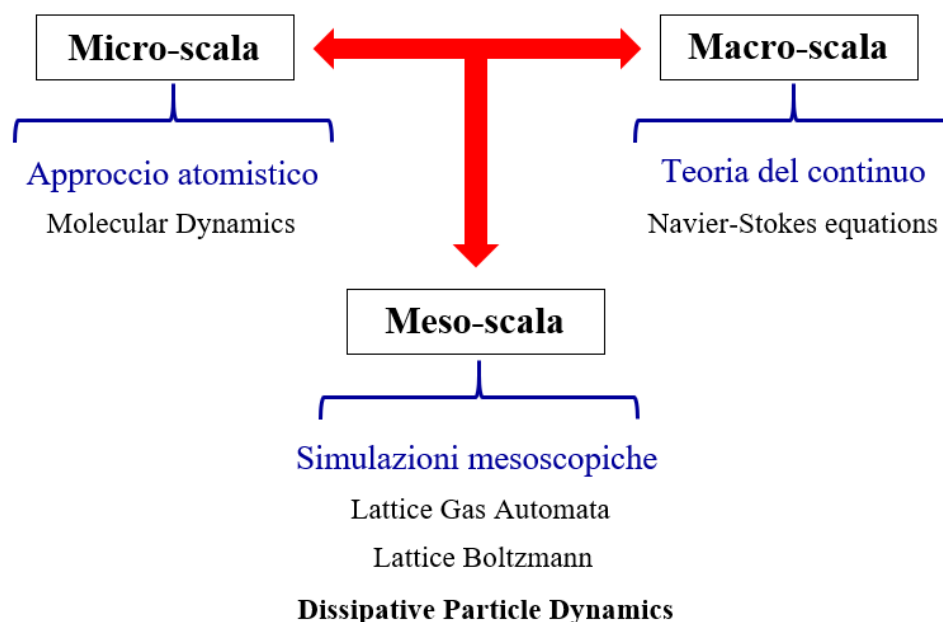


Figura 1.1: Dissipative Particle Dynamics: una tecnica meso-scala per colmare il divario tra la micro- e la macro-scala (elaborazione immagine estratta da Ref. [3]).

continuo fino alla mesoscala, accoppiando la classica idrodinamica a termini fluttuanti per tener conto della natura discreta della materia su queste scale spaziali più piccole [4]. Le equazioni risultanti dell'idrodinamica fluttuante sono note come equazioni di Landau-Lifshitz-Navier-Stokes (LLNS). Anche se l'uso dell'idrodinamica fluttuante può essere appropriato alla mesoscala, ci sono molti sistemi con un grande numero di dettagli microscopici, come proteine, membrane, sistemi polimerici per i quali la descrizione idrodinamica continua non è comunque applicabile. A seconda della natura del problema, sono stati ideati, sviluppati e perfezionati nel corso degli anni diversi metodi di simulazione mesoscala. Nell'ambito della simulazione dei fluidi complessi

sono state impiegate tecniche *grid-based* come Lattice Gas Automata (LGA) e Lattice Boltzmann (LB). Nonostante l'efficienza dal punto di vista computazionale e la capacità di descrivere le proprietà fisiche del sistema, non si prestano bene alla riproduzione delle proprietà strutturali e conformazionali influenzate da alcuni dettagli molecolari. Presentano, inoltre, limitazioni e restrizioni dovute al reticolo come l'induzione di dinamiche spurie e problemi di incompatibilità con alcuni metodi di imposizione delle condizioni al contorno. Per ovviare i problemi intrinseci dei modelli *grid-based* tra cui le questioni appena illustrate, sono desiderabili tecniche *off-lattice* che permettano di mantenere una certa specificità chimica nella modellazione del sistema. La strategia vincente in quest'ottica è quella di usare descrizioni *coarse-grained* nelle quali gli atomi o molecole sono raggruppati nei cosiddetti *beads*, trattati nel processo di simulazione come singole unità. In questo modo si riduce notevolmente il numero di particelle simulate con innegabili vantaggi sul costo computazionale della simulazione. Sebbene i dettagli su come seguire questo approccio modellistico siano specifici di ogni sistema e siano tuttora oggetto di attenzione di un cospicuo numero di ricercatori, c'è una procedura di *coarse-graining* ben definita che è nota con il nome di meccanica statistica di non equilibrio, Teoria di Mori-Zwanzig o teoria di *coarse-graining* [5]. Simulare tutti i dettagli molecolari, oltre ad essere dispendioso dal punto di vista computazionale, è in alcuni casi anche inutile. In particolare, c'è una forte tendenza allo sviluppo di modelli a grana grossa per i solventi, come l'acqua, mantenendo, invece, un numero sufficiente di dettagli molecolari per il soluto in modo da conservare la specificità del prodotto chimico. Tra le tecniche *coarse-graining* la Dissipative Particle Dynamics (DPD) si distingue per la semplicità dell'algoritmo e l'enorme versatilità. Variando, infatti, la forza conservativa tra le particelle si può modellare il comportamento di svariati fluidi complessi come polimeri, anfifilici e tensioattivi, membrane biologiche, sangue, colloidali, olii industriali e soluzioni binarie immiscibili. Il modello DPD è stato originariamente introdotto per simulare il comportamento idrodinamico complesso da Hoogerbrugge e Koelman nel 1992 [6], combinando gli aspetti positivi della MD e della tecnica *grid-based* LGA. Si presentava più efficiente dal punto di vista computazionale della MD e al contempo più flessibile degli schemi LGA. Successivamente è stato rivisto da Español e Warren nel 1995 affermandosi come un modello basato sulla meccanica statistica [7]. Dalla sua introduzione in Europa alla fine del ventesimo secolo, il metodo è stato concettualmente perfezionato e impiegato nello studio di una sempre più vasta gamma di sistemi e applicazioni. Risulta essere, infatti, una delle migliori e più promettenti tecniche mesoscala nella simulazione di fluidi complessi, e di materia soffice in generale, in condizioni statiche e dinamiche, su scale spaziali e temporali fisicamente interessanti che sarebbero altrimenti inaccessibili attraverso simulazioni microscopiche.

1.1 Il modello DPD: un metodo *coarse-graining*

La Dissipative Particle Dynamics è un metodo mesoscopico basato su particelle in cui ogni “particella fluida” chiamata *bead* rappresenta un aggregato di atomi o molecole. Il numero di molecole che costituiscono una particella DPD è tipicamente designato come N_m , il cosiddetto parametro di *coarse-graining*. Questa tecnica permette di simulare sistemi complessi su scale temporali più ampie, eliminando alcuni gradi di libertà aggiuntivi che non sono ritenuti fondamentali per i fenomeni di interesse. Una questione cruciale è proprio individuare il limite superiore del parametro di *coarse-graining* con il quale le proprietà fisiche sono modellate correttamente [8]. Questo parametro ha un impatto significativo sulla velocità e, quindi, sul costo computazionale della simulazione. Secondo una stima effettuata da Groot e Rabone [9], l’aumento di velocità di una simulazione DPD rispetto ad una simulazione MD è, per un dato volume, all’incirca $1000N_m^{8/3}$, ovvero intorno a 2×10^4 per $N_m = 3$ e all’incirca 5×10^5 per $N_m = 10$.

Le *beads* interagiscono tra loro attraverso un potenziale repulsivo soffice, così definito perché all’atto della collisione permette la loro compenetrazione, con conseguente dissipazione di energia. Il potenziale DPD non tende ad infinito quando la distanza tra le particelle si annulla. La definizione di un potenziale repulsivo soffice facilita l’accesso a scale spaziali e temporali maggiori perché consente l’uso di un passo di integrazione temporale più ampio [10, 11]. Nei modelli DPD alla forza di interazione tra le particelle puramente repulsiva derivante dal potenziale a sfere soffici si aggiungono una forza dissipativa o forza di *drag* e una forza stocastica o forza *random*. La forza dissipativa e quella stocastica dipendono dalla differenza di velocità tra le particelle e agendo simultaneamente garantiscono il controllo della temperatura del sistema fungendo da termostato. Hanno, infatti, un effetto opposto dal punto di vista dell’energia del sistema: la forza dissipativa rappresenta l’effetto della viscosità e rallenta le particelle sottraendo energia; la forza stocastica tiene conto dei gradi di libertà che sono stati rimossi dal processo di *coarse-graining* e aggiunge in media energia termica e vibrazionale al sistema.

1.2 Equazioni di governo

Analogamente alla dinamica molecolare classica, l’evoluzione temporale di ogni *bead* è descritta dalle equazioni del moto di Newton:

$$\frac{d\mathbf{r}_i}{dt} = \mathbf{v}_i, \quad \frac{d\mathbf{v}_i}{dt} = \frac{\mathbf{f}_i}{m_i}, \quad (1.1)$$

dove \mathbf{r}_i , \mathbf{v}_i sono rispettivamente i vettori posizione e velocità della particella *i-esima* e \mathbf{f}_i è la forza che agisce sulla particella *i-esima*. La forza che agisce

sulla particella i -esima è la somma delle interazioni con tutte le altre particelle j . Considerando un sistema costituito da particelle interagenti con massa uguale e unitaria si può scrivere come:

$$\mathbf{f}_i = \sum_{j \neq i} (\mathbf{F}_{ij}^C + \mathbf{F}_{ij}^D + \mathbf{F}_{ij}^R). \quad (1.2)$$

dove \mathbf{F}_{ij}^C è la forza conservativa, \mathbf{F}_{ij}^D è la forza dissipativa e \mathbf{F}_{ij}^R è la forza stocastica. Poiché in questo caso particolare una delle forze ha natura stocastica le equazioni vengono a volte chiamate equazioni di Langevin (invece che di Newton). Le tre componenti possono essere scritte individualmente come:

$$\mathbf{F}_{ij}^C = w^C(r_{ij}) \hat{\mathbf{r}}_{ij}, \quad (1.3)$$

$$\mathbf{F}_{ij}^D = -\gamma w^D(r_{ij}) (\hat{\mathbf{r}}_{ij} \cdot \mathbf{v}_{ij}) \hat{\mathbf{r}}_{ij}, \quad (1.4)$$

$$\mathbf{F}_{ij}^R = \sigma w^R(r_{ij}) \zeta_{ij} \Delta t^{-1/2} \hat{\mathbf{r}}_{ij}, \quad (1.5)$$

dove $\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j$ è la distanza tra le particelle i e j ; $r_{ij} = |\mathbf{r}_{ij}|$ e $\hat{\mathbf{r}}_{ij} = \mathbf{r}_{ij}/|\mathbf{r}_{ij}|$ è il vettore unitario la cui direzione è la medesima della retta congiungente i centri di massa delle particelle i e j ; $\mathbf{v}_{ij} = \mathbf{v}_i - \mathbf{v}_j$ è la differenza di velocità tra le particelle i e j . w^C , w^D e w^R sono le funzioni peso che definiscono la dipendenza dalla distanza interparticellare r_{ij} delle forze rispettivamente conservativa, dissipativa e stocastica. Il coefficiente di attrito per la forza dissipativa è designato con γ e l'ampiezza del rumore per le forze stocastica è indicata con σ . Il fattore $\Delta t^{-1/2}$ che appare nell'espressione per la forza stocastica nell' Eq. (1.5) può essere derivato interpretando tale forza come un processo di Wiener, noto anche come *moto Browniano standard*. ζ_{ij} è una variabile aleatoria che segue una statistica di tipo Gaussiano (il valore medio è nullo e la varianza è unitaria), indipendente per ciascuna coppia di particelle interagenti e per ogni scelta del passo temporale, Δt . Il requisito $\zeta_{ij} = \zeta_{ji}$ impone la conservazione della quantità di moto. Nella formulazione originale del modello ciascuna delle tre componenti della forza interparticellare è additiva a coppie, agisce unicamente lungo la direzione della retta congiungente i centri di massa dei *beads*, conserva la quantità di moto e ha valore non nullo solo entro una sfera di interazione definita dal raggio di *cutoff*. Quest'ultimo nel modello convenzionale è il medesimo per ciascuna delle tre forze. La funzione peso relativa alla forza conservativa ha la seguente forma funzionale:

$$w_{ij}^C = \begin{cases} a_{ij} \left(1 - \frac{r_{ij}}{r_c}\right) \hat{\mathbf{r}}_{ij}, & r_{ij} < r_c \\ 0 & r_{ij} \geq r_c \end{cases} \quad (1.6)$$

in cui $a_{ij} > 0$ è il coefficiente di repulsione ed il suo valore rappresenta la massima repulsione agente tra le particelle i e j . Definire la funzione peso conservativa e il parametro di repulsione è uno degli aspetti più importanti

delle simulazioni DPD, infatti il parametro a_{ij} determina la compressibilità del sistema. Le funzioni peso della forza dissipativa e della forza stocastica forniscono l'intervallo di interazione per le rispettive forze e sono legate da una relazione necessaria ad assicurare che il sistema, nel limite di un passo temporale infinitesimo, raggiunga l'equilibrio Gibbsiano, ricavata da Español e Warren [7]. Ottenere dalle simulazioni una distribuzione di equilibrio corrispondente alla distribuzione di Boltzmann di un insieme canonico o di Gibbs è fondamentale per potersi ricondurre, nel sistema in esame, alle classiche relazioni della termodinamica. Una delle due funzioni peso che compaiono nell'Eq. (1.5) e nell'Eq. (1.4) può essere, dunque, scelta arbitrariamente e questa scelta fissa l'altra:

$$w_D(r) = [w_R(r)]^2, \quad (1.7)$$

$$\sigma^2 = \frac{2\gamma k_B T}{m}, \quad (1.8)$$

dove k_B è la costante di Boltzmann e T è la temperatura di equilibrio. Questa condizione fa sì che le equazioni DPD agiscano come un termostato e poiché l'algoritmo dipende dalle velocità relative e le interazioni tra le particelle sono simmetriche, è un termostato invariante Galileano che preserva l'idrodinamica [12]. La forma funzionale convenzionale per le funzioni peso dissipativa e stocastica è la seguente:

$$w^D(r_{ij}) = [w^R(r_{ij})]^2 = \begin{cases} \left(1 - \frac{r_{ij}}{r_c}\right)^2 \hat{\mathbf{r}}_{ij}, & r_{ij} < r_c \\ 0 & r_{ij} \geq r_c \end{cases} \quad (1.9)$$

1.3 Aspetti teorici

1.3.1 Equazione di Fokker-Planck

Il modello DPD presentato da Hoogerbrugge and Koelman [6] ha delle analogie con la Dinamica Browniana convenzionale. Tuttavia, mentre nella Dinamica Browniana si conservano unicamente il numero totale di particelle e la massa risultando in un comportamento macroscopico puramente diffusivo, un elemento distintivo della DPD è la conservazione della quantità di moto totale. Ne deriva che il comportamento macroscopico non è diffusivo ma idrodinamico: all'equazione di conservazione della massa si affianca l'equazione di trasporto della quantità di moto. L'energia, invece, non viene conservata e non c'è un'equazione di trasporto per l'energia. Si assume che il modello descriva una situazione isoterma. Tuttavia, non vi sono espressioni legate alla temperatura e ai parametri del modello, come ad esempio il coefficiente della forza dissipativa e l'ampiezza del rumore della forza stocastica. Un fondamento teorico alla DPD è stato dato successivamente, mediante la derivazione dell'equazione di Fokker-Planck [7] che dimostra la connessione tra la DPD e la meccanica del continuo e rappresenta il formalismo

essenziale per la derivazione delle equazioni idrodinamiche. L'equazione di Fokker-Planck, analoga all'equazione di Liouville per la dinamica molecolare, governa la distribuzione delle variabili microscopiche come le posizioni e le quantità di moto delle particelle. È correlata alla DPD dalle corrispondenti equazioni di Langevin, che sono le equazioni differenziali stocastiche che descrivono la dinamica di particelle sottoposte a forze conservative, dissipative e casuali. Partendo dalle equazioni microscopiche di Fokker-Planck e utilizzando la tecnica standard con gli operatori di proiezione, Espanol ha ottenuto le equazioni idrodinamiche macroscopiche DPD [13]. La funzione di distribuzione del sistema $f(\mathbf{r}, \mathbf{p}; t)$ fornisce la possibilità di trovare il sistema al tempo t in un qualsiasi punto dello spazio delle fasi. L'equazione di Fokker-Planck descrive l'evoluzione temporale di questa distribuzione e presenta la seguente forma:

$$\frac{\partial f}{\partial t} = \mathcal{L}^C f + \mathcal{L}^D \rho, \quad (1.10)$$

dove gli operatori di evoluzione temporale \mathcal{L}^C e \mathcal{L}^D sono stati definiti nel modo seguente

$$\mathcal{L}^C = - \left[\sum_i \frac{\mathbf{p}_i}{m_i} \frac{\partial}{\partial \mathbf{r}_i} + \sum_{i,j \neq i} \mathbf{F}_{ij}^C \frac{\partial}{\partial \mathbf{p}_i} \right], \quad (1.11)$$

$$\mathcal{L}^D = \sum_{i,j \neq i} \mathbf{r}_{ij} \frac{\partial}{\partial \mathbf{p}_i} \left[\gamma w_D(\mathbf{r}_{ij} \cdot \mathbf{v}_{ij}) + \frac{\sigma^2}{2} w_R^2 \mathbf{r}_{ij} \left(\frac{\partial}{\partial \mathbf{p}_i} - \frac{\partial}{\partial \mathbf{p}_j} \right) \right], \quad (1.12)$$

in cui \mathcal{L}^C è l'operatore di Liouville di un sistema Hamiltoniano in cui le particelle interagiscono attraverso la forza conservativa e \mathcal{L}^D , tiene conto degli effetti delle forze stocastica e dissipativa. La soluzione di equilibrio non si ottiene risolvendo l'equazione di Fokker-Planck direttamente, ma è un risultato naturale della dimostrazione del teorema-H e della dimostrazione che la distribuzione di Gibbs-Boltzmann è la soluzione stazionaria dell'equazione di Fokker-Planck [14, 15]. La dimostrazione del teorema-H è il risultato fondamentale in meccanica statistica per un sistema dinamico e mostra che le particelle DPD tendono a migrare verso lo stato di equilibrio, che fornisce come soluzione di equilibrio la distribuzione di Gibbs-Boltzmann:

$$f^{eq}(\mathbf{r}_i, \mathbf{p}_i) = \frac{1}{Z} \exp \left\{ - \frac{V(\{\mathbf{r}\})}{k_B T} - \sum_i \frac{\mathbf{p}_i^2}{2m_i k_B T} \right\}, \quad (1.13)$$

in cui \mathbf{r} è il vettore costituito dai vettori posizione di tutte le particelle, Z è la funzione di partizione, V è il potenziale del sistema che deriva dalla forza conservativa F^C , k_B è la costante di Boltzmann e T è la temperatura

di equilibrio del sistema. Nel caso specifico del modello DPD le condizioni riportate nelle equazioni (1.7) e (1.8) sono il requisito fondamentale per dimostrare il teorema-H e che la distribuzione di Gibbs-Boltzmann è una soluzione stazionaria dell'equazione di Fokker-Planck. Queste relazioni sono analoghe al teorema di fluttuazione-dissipazione della dinamica Browniana convenzionale. Il teorema-H garantisce che la soluzione sia stabile e che tutti gli stati nello spazio delle fasi portino all'equilibrio.

1.3.2 Comprimibilità isoterma ed equazione di stato

Affinché la modellazione DPD di un liquido arbitrario ne riproduca correttamente lo stato termodinamico, si devono descrivere adeguatamente le fluttuazioni al suo interno. Queste ultime sono determinate essenzialmente dalla comprimibilità del sistema. La comprimibilità isoterma di un fluido DPD è così definita:

$$k_T = -\frac{1}{V} \frac{\partial V}{\partial p} \Big|_T = \frac{1}{\rho} \frac{\partial \rho}{\partial p} \Big|_T. \quad (1.14)$$

in cui ρ denota la densità numerica delle particelle. È conveniente definire il seguente parametro adimensionale:

$$k^{-1} = \frac{1}{k_B T \rho k_T} = \frac{1}{k_B T} \frac{\partial p}{\partial \rho}. \quad (1.15)$$

Sulla base della teoria delle perturbazioni dei liquidi sviluppata da Andersen, Weeks e Chandler [16], Groot e Warren [2] hanno imposto che la comprimibilità isoterma (adimensionale) del fluido DPD deve rispettare la seguente relazione di *scaling*:

$$\kappa^{-1} = \frac{1}{k_B T} \frac{\partial p}{\partial \rho} \Big|_{\text{sim}} = \frac{1}{\rho k_B T \kappa_T} = \frac{1}{k_B T} \frac{\partial p}{\partial n} \Big|_{\text{exp}}, \quad (1.16)$$

in cui n denota la densità molecolare numerica del sistema fisico. L'equazione appena riportata, Eq.(1.16), è valida, però, solo per grado di *coarse-graining* pari a uno. Ovvero, se un *bead* corrisponde ad una molecola di solvente. Generalizzando l'espressione per un grado di *coarse-graining* arbitrario si ottiene la seguente condizione:

$$\frac{1}{k_B T} \left(\frac{\partial p}{\partial \rho} \right)_{\text{sim}} = \frac{1}{k_B T} \left(\frac{\partial n}{\partial \rho} \right) \cdot \left(\frac{\partial p}{\partial n} \right)_{\text{exp}} = \frac{N_m}{k_B T} \left(\frac{\partial p}{\partial n} \right)_{\text{exp}}. \quad (1.17)$$

Groot e Warren [2] attraverso una serie di simulazioni numeriche per un sistema DPD monocomponente hanno trovato la seguente equazione di stato del viriale che lega la pressione p alla densità numerica delle particelle della simulazione ρ :

$$p = \rho k_B T + \frac{1}{3V} \left\langle \sum_{j>i} \mathbf{r}_{ij} \mathbf{F}_{ij}^C \right\rangle = \rho k_B T + \frac{2\pi}{3} \rho^2 \int_0^{r_c} r a \left(1 - \frac{r}{r_c} \right) g(r) r^2 dr, \quad (1.18)$$

dove $g(r)$ denota la funzione di distribuzione radiale, che descrive la probabilità di trovare due particelle ad una distanza r . Per valori di $\rho > 2$ può essere semplificata continuando a riprodurre bene i risultati numerici ottenuti:

$$p = \rho k_B T + 0.1 a_{ii} \rho^2 . \quad (1.19)$$

Sostituendo nell'Eq. (1.17) l'equazione di stato (1.19), si ottiene

$$\kappa^{-1} = \frac{1}{N_m} + \frac{0.2 a_{ii} \rho}{k_B T} . \quad (1.20)$$

da cui si ricava l'espressione del parametro di repulsione a_{ii} tra *beads* della stessa specie di seguito riportata:

$$a_{ii} = k_B T \left(\frac{\kappa^{-1} N_m - 1}{0.2 \rho} \right) . \quad (1.21)$$

Tale parametro viene, quindi, derivato dalla condizione di uguaglianza tra la comprimibilità del liquido da studiare e quella del modello DPD simulato.

1.3.3 Teoria di Flory-Huggins

Il modello *lattice fluid* di Flory-Huggins [17] segue un approccio meccanico statistico per riprodurre il comportamento dei polimeri in soluzione acquosa. Nella teoria dei polimeri di Flory-Huggins l'energia interna è descritta come l'energia libera di miscelazione di eccesso, dovuta alla non idealità del sistema. Quindi per un sistema costituito da due componenti, l'energia libera di Helmholtz per unità di reticolo è:

$$\frac{F}{k_B T} = \frac{\Phi_A}{N_A} \ln \Phi_A + \frac{\Phi_B}{N_B} \ln \Phi_B + \chi \Phi_A \Phi_B , \quad (1.22)$$

in cui Φ_A e Φ_B sono le frazioni volumiche di A e B , N_A e N_B sono i numeri di segmenti per le molecole di A e B . In questa teoria le interazioni molecolari tra i diversi segmenti del reticolo sono caratterizzate dai parametri χ . Per considerare le interazioni tra componenti diversi in soluzione, in modo da estendere le simulazioni DPD anche a sistemi multicomponente, Groot e Warren [2] hanno correlato i parametri DPD ai parametri di solubilità χ della teoria di Flory-Huggins, con la procedura di seguito brevemente esposta. La densità di energia libera corrisponde alla pressione di un singolo componente, esprimibile con l'Eq. (1.19):

$$\frac{f_v}{k_B T} = \rho \ln \rho - \rho + \frac{\alpha a \rho^2}{k_B T} , \quad (1.23)$$

che per un sistema costituito da due componenti diventa:

$$\frac{f_v}{k_B T} = \frac{\rho_A}{N_A} \ln \rho_A + \frac{\rho_B}{N_B} \ln \rho_B - \frac{\rho_A}{N_A} - \frac{\rho_B}{N_B} + \frac{\alpha(a_{AA}\rho_A^2 + 2a_{AB}\rho_A\rho_B + a_{BB}\rho_B^2)}{k_B T}. \quad (1.24)$$

Scegliendo $a_{AA} = a_{BB}$ e definendo $x = \frac{\rho_A}{\rho_A + \rho_B}$, l'analogia con l'Eq. (1.22) si ha con la seguente definizione:

$$\chi = \frac{2\alpha(a_{AB} - a_{AA})(\rho_A + \rho_B)}{k_B T}. \quad (1.25)$$

Le simulazioni numeriche confermano la linearità tra χ e Δa che emerge dall'Eq.(1.25) ma portano ad una correzione del coefficiente di proporzionalità che non risulta linearmente legato alla densità. Per $\chi > 3$ vale la seguente relazione per $\rho = 3$:

$$\chi = (0.231 \pm 0.001)\Delta a. \quad (1.26)$$

Essendo Δa l'incremento di repulsione tra *beads* diversi rispetto a *beads* dello stesso tipo, il coefficiente della forza conservativa repulsiva che agisce tra le due specie, i e j , è espresso mediante la forma seguente [9, 18]:

$$a_{ij} \approx a_{ii} + \frac{\chi_{ij}}{(0.231 \pm 0.001)}. \quad (1.27)$$

Dunque, i parametri di repulsione tra le *beads* di specie diverse, sono ricavati applicando la condizione di corretta descrizione delle mutue solubilità nel modello DPD.

1.4 Schemi di integrazione temporale

L'integrazione temporale delle equazioni del modello DPD ha un grado di difficoltà aggiuntivo rispetto a quella della dinamica molecolare convenzionale per la presenza delle equazioni stocastiche. Non è garantita la reversibilità temporale dell'algoritmo di integrazione. Inoltre, la forza dissipativa dipende dalla velocità. Per il calcolo delle velocità, tuttavia, per impiegare un algoritmo del secondo ordine è necessario conoscere le forze al tempo t e al tempo $t + \Delta t$. In altre parole esiste una complicazione dovuta all'accoppiamento non lineare tra forza e velocità. Il metodo più semplice per ovviare tale problema consiste nell'utilizzare lo schema di Eulero attraverso il quale le posizioni e le velocità delle particelle vengono calcolate a partire dai rispettivi valori nello step precedente:

$$\begin{aligned} \mathbf{r}_i(t + \Delta t) &= \mathbf{r}_i(t) + \mathbf{v}_i(t)\Delta t, \\ \mathbf{v}_i(t + \Delta t) &= \mathbf{v}_i(t) + \mathbf{f}_i(t)\Delta t, \\ \mathbf{f}_i(t + \Delta t) &= \mathbf{f}_i(\mathbf{r}_i(t + \Delta t), \mathbf{v}(t + \Delta t)). \end{aligned} \quad (1.28)$$

Tuttavia, tale algoritmo calcola traiettorie non reversibili nel tempo. Inoltre, può portare ad un accumulo di energia nel sistema limitando così il controllo della temperatura. Anche l'algoritmo di Verlet può essere utilizzato per integrare le equazioni DPD [7, 2]. Tra gli algoritmi più efficienti è emerso nel corso degli anni, realizzando un buon compromesso tra accuratezza e costo computazionale, l'integratore di Groot/Warren [2] che è una versione modificata del *velocity*-Verlet :

$$\begin{aligned}
\mathbf{r}_i(t + \Delta t) &= \mathbf{r}_i(t) + \Delta t \mathbf{v}_i(t) + \frac{1}{2}(\Delta t)^2 \mathbf{f}_i(t), \\
\tilde{\mathbf{v}}_i(t + \Delta t) &= \mathbf{v}_i(t) + \lambda \Delta t \mathbf{f}_i(t), \\
\mathbf{f}_i(t + \Delta t) &= \mathbf{f}_i[\mathbf{r}_i(t + \Delta t), \tilde{\mathbf{v}}_i(t + \lambda \Delta t)], \\
\mathbf{v}_i(t + \Delta t) &= \mathbf{v}_i(t) + \frac{1}{2} \Delta t [\mathbf{f}_i(t) + \mathbf{f}_i(t + \Delta t)]. \quad (1.29)
\end{aligned}$$

Questo schema di integrazione è stato ideato per la prima volta da Groot e Warren nel 1997 [2] e successivamente impiegato da molti ricercatori nell'area delle simulazioni DPD. Dal punto di vista delle velocità si tratta di un algoritmo *predictor-corrector*. Infatti, le velocità al tempo $t + \Delta t$ vengono predette per mezzo della seconda equazione dello schema iterativo (*predictor*) partendo dalla velocità al tempo t . Il valore previsto della velocità, indicato con $\tilde{\mathbf{v}}_i$, viene utilizzato per prevedere la forza. L'ultima equazione (*corrector*) calcola il valore corretto della velocità. Il costo computazionale non aumenta rispetto all'algoritmo di Verlet classico in quanto il valore della forza viene aggiornato una sola volta per iterazione. Data la natura stocastica del processo, l'ordine dell'algoritmo diventa indistinto, e il fattore variabile λ , introdotto empiricamente, tiene conto di alcuni effetti dovuti alle interazioni stocastiche. Scegliendo opportunamente la coppia di valori per λ e per il passo temporale, si riesce ad ottenere un *optimum* tra stabilità del sistema e velocità di simulazione.

Sono stati sviluppati anche metodi alle differenze finite, basati su analogie con integratori di ordine superiore per modelli privi di equazioni stocastiche, tra questi l'algoritmo *leap-frog*. Tuttavia, è necessario prestare attenzione nell'implementazione per tener conto della natura stocastica delle equazioni [19].

Contemporaneamente sono stati sviluppati altri schemi di integrazione complessi come l'algoritmo di Verlet *self-consistent* [20] e il metodo di *splitting* di Shardlow [21]. Il primo è un'estensione del *velocity*-Verlet modificato: è *autoconsistente* in quanto ripete il calcolo della forza fino ad ottenere un valore stabile per la velocità. È reversibile nel tempo e garantisce un ottimo controllo della temperatura, ma il costo computazionale aumenta rispetto all'algoritmo *predictor-corrector* in quanto il calcolo della forza si ripete più volte. Lo schema di integrazione di Shardlow, sebbene debolmente convergente, si fonda sull'idea innovativa di splittare il campo di forze nella somma di due termini: quello conservativo e quello dissipativo-stocastico. Il siste-

ma conservativo è risolto impiegando il regolare *velocity*-Verlet modificato, mentre le equazioni relative alle forze stocastiche e dissipative sono integrate implicitamente in modo da conservare la quantità di moto nel processo integrativo.

Oltre alla difficoltà di trovare un algoritmo di integrazione temporale appropriato derivante dalla natura stocastica del modello DPD, spesso sorgono problemi specifici a seconda del sistema fisico simulato che richiedono lo sviluppo di algoritmi *ad hoc* per accelerare le simulazioni. Ad esempio, è stato introdotto uno schema di integrazione a più fasi temporali basato sull'algoritmo *velocity*-Verlet modificato, in cui le particelle di solvente vengono aggiornate a frequenze inferiori rispetto a quelle del soluto [22].

1.5 Fattori di conversione e unità DPD

Il modello computazionale DPD è basato su parametri e variabili che vengono espressi in termini di unità ridotte. Nelle simulazioni vengono definite le unità adimensionali per la lunghezza, il raggio di *cutoff* r_c , per la massa, la massa di un *bead* (tutti i *beads* hanno la stessa massa), e per l'energia del sistema, $\mathcal{E} = k_B T$. A questo punto tutti i parametri rimanenti sono ricavati a partire da questi. L'unità di tempo è definita come:

$$\tau = r_c \sqrt{\frac{m}{k_B T}}. \quad (1.30)$$

La conversione tra le unità adimensionali DPD e le unità fisiche reali è un aspetto tutt'altro che banale. Per le simulazioni di non-equilibrio la questione non è ancora del tutto risolta. Invece, per le simulazioni di equilibrio il livello di *coarse-graining*, cioè il numero N_m di molecole incluse all'interno di un *bead*, può essere considerato il fattore di rinormalizzazione per tale conversione. Le scale spaziali e temporali cambiano a seconda del valore di N_m . A titolo d'esempio si riporta la procedura per la conversione delle unità DPD fisiche reali spaziali e temporali, prendendo come riferimento l'acqua, il componente più semplice e diffuso nei sistemi studiati con DPD [9]. Se il grado di *coarse-graining* è N_m , ovvero un *bead* è costituito N_m molecole d'acqua, allora un cubo di volume r_c^3 contiene ρN_m molecole d'acqua, dove ρ è la densità numerica del sistema, cioè il numero di *beads* per unità di volume. Essendo noti il valore di densità dell'acqua, 1000 kg m^{-3} , e il suo peso molecolare si può calcolare il volume di una molecola d'acqua a temperatura ambiente pari a 30 \AA^3 . Noto, dunque, il volume fisico di questo cubo equivalente, $30\rho N_m \text{ \AA}^3$, l'unità di lunghezza in unità reale è:

$$r_c = 3.107(\rho N_m)^{1/3} (\text{\AA}). \quad (1.31)$$

Questo è quindi il fattore per cui una lunghezza in unità DPD deve essere moltiplicata per trasformarla in unità reali. La massa di una par-

ticella può essere calcolata moltiplicando la massa di una molecola d'acqua ($m_w = 2.992 \times 10^{-26}$ kg) per il numero di molecole d'acqua contenute all'interno di un *bead*:

$$m = N_m m_w (\text{kg}) .$$

Infine, l'unità di energia è calcolata a partire dalla temperatura di riferimento del sistema pari a 298 K come segue:

$$\mathcal{E} = k_B T = 4.112 \times 10^{-21} (\text{J}) ,$$

dove k_B è la costante di Boltzmann pari a 1.38×10^{-23} J/K.

Avendo definito i fattori di conversione per la lunghezza, la massa e l'energia si possono definire quelli relativi alle altre grandezze fisiche del sistema. Un criterio molto utilizzato per la conversione dell'unità di tempo è basato sul *matching* tra il valore sperimentale del coefficiente di autodiffusione dell'acqua e il valore corrispondente ottenuto dalla simulazione DPD [9]. L'imposizione dell'uguaglianza tra i due fornisce la seguente espressione:

$$\tau = \frac{N_m \mathcal{D}_{sim} r_c^2}{\mathcal{D}_{water}} , \quad (1.32)$$

dove $\mathcal{D}_{water} = (2.43 \pm 0.01) \times 10^{-5}$ cm²/s è il coefficiente di autodiffusione dell'acqua calcolato sperimentalmente [23].

Un metodo analogo è stato seguito da Groot [24] che ha, però, utilizzato per il *matching* il coefficiente di diffusione di una micella tensioattiva.

Entrambi i criteri dimostrano che il passo temporale impiegabile nelle simulazioni DPD è di almeno tre ordini di grandezza maggiore rispetto a quello che richiederebbe una simulazione di dinamica molecolare, dove tipicamente i valori di *timestep* utilizzati sono dell'ordine di qualche fs.

Tabella 1.1: Fattori di conversione di massa, lunghezza, energia e tempo per tre valori caratteristici del numero di *coarse-graining*. Il fattore di conversione dell'unità di tempo è ricavato imponendo il *matching* tra il valore sperimentale del coefficiente di autodiffusione dell'acqua e quello ottenuto dalla simulazione DPD.

N_m	m (Kg)	r_c (m)	\mathcal{E} (J)	τ (s)
1	2.9992×10^{-26}	4.4811×10^{-10}	4.112×10^{-21}	1.3221×10^{-12}
3	8.9760×10^{-26}	6.4628×10^{-10}	4.112×10^{-21}	8.2505×10^{-12}
10	2.9992×10^{-25}	9.6542×10^{-10}	4.112×10^{-21}	6.1368×10^{-11}

Capitolo 2

Dissipative particle dynamics per stima di proprietà di trasporto

Per quanto riguarda la dinamica di un fluido un problema significativo intrinseco del metodo DPD convenzionale è rappresentato dal rapporto tra la scala temporale su cui si propagano le interazioni idrodinamiche e quella caratteristica della diffusione. Una quantità chiave in questo contesto è il numero di Schmidt, $Sc = \nu/D$, dove ν è la viscosità cinematica e D è il coefficiente di diffusione. È un parametro macroscopico adimensionale caratterizzante il fluido e interpretabile come il rapporto tra il tempo in cui le molecole diffondono attraverso una data distanza e il tempo in cui le interazioni idrodinamiche raggiungono lo stato stazionario sulla stessa distanza. Equivalentemente rappresenta il rapporto tra la velocità di trasferimento della quantità di moto e quella di materia. In un liquido ordinario come l'acqua, si ha all'incirca $Sc = 1000$, mentre un semplice fluido DPD assimilabile all'acqua presenta $Sc \approx 1$. Tale differenza di tre ordini di grandezza è imputabile alla natura del potenziale a sfere soffici con cui interagiscono le particelle DPD. Infatti, mentre nel potenziale a sfere rigide si ha un effetto cosiddetto di "ingabbiamento", per il quale un atomo subisce molte collisioni prima di essere effettivamente trasportato, nel potenziale a sfere soffici questo effetto non si verifica e, di conseguenza, la mobilità delle particelle aumenta di un fattore circa pari a 1000 [2].

L'interazione soffice tra le particelle non è la sola spiegazione; i valori bassi di viscosità che si ottengono da una simulazione DPD standard sono anche dovuti all'assenza di dissipazione di taglio, la cosiddetta *shear dissipation*. Nella formulazione originale della DPD le particelle sono soggette solo a forze che agiscono nella direzione parallela alla retta congiungente i centri di massa dei *beads*, *central shear forces*.

Negli anni sono state introdotte varie modifiche al modello originale finalizza-

te a risolvere tale problema. Un primo approccio, è stato quello di continuare ad utilizzare il modello DPD studiando l'influenza di alcuni parametri come il coefficiente di dissipazione γ e il raggio di *cutoff* globale, ovverosia il medesimo per tutte le forze, r_c sul numero di Schmidt [25]. L'aumento del raggio di *cutoff* incrementa drasticamente le forze di interazione che agiscono sulla particella e di conseguenza il numero di Sc come illustrato in Fig. 2.1, dove si evidenzia una relazione del tipo $Sc \propto r_c^8$. Tuttavia, allo stesso tempo si ha

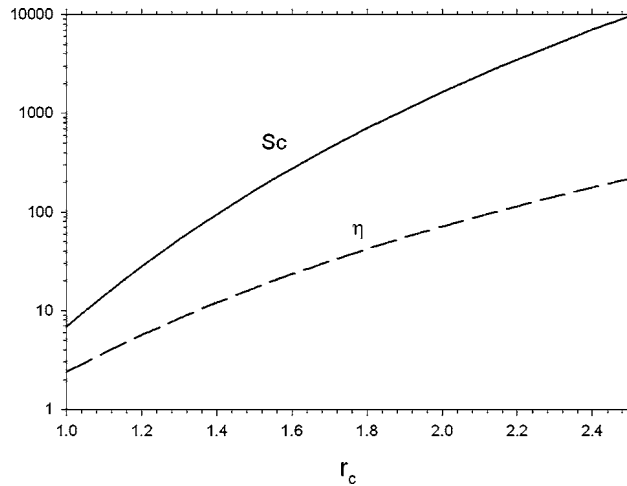


Figura 2.1: Andamento della viscosità dinamica e del numero di Schmidt in funzione del raggio di *cutoff* per $\rho = 4$, $\gamma = 4.5$ e $k_B T = 1$ (elaborazione immagine estratta da Ref. [25]).

un'impennata del costo computazionale che approssimativamente dipende dal cubo del raggio di *cutoff*. Anche γ risulta essere influente sul numero di Sc con una proporzionalità quadratica, $Sc \propto \gamma^2$. Tuttavia, valori elevati di questo parametro potrebbero creare problemi dal punto di vista del controllo termico del sistema e la necessità di ridurre il valore di *timestep* utilizzato nella simulazione, comportando quindi uno svantaggio computazionale.

Successivamente è stata proposta l'introduzione di una funzione peso generalizzata che sostituisce quella quadratica convenzionale [25]. In questo caso si va a studiare l'effetto sul numero di Sc del parametro s , esponente che compare nella funzione peso della forza dissipativa. Per il teorema di fluttuazione-dissipazione cambia automaticamente anche l'esponente della funzione peso relativa alla forza stocastica legato al precedente tramite il fattore moltiplicativo pari a 0.5. La variazione di questo parametro non inficia la velocità della simulazione.

Español e collaboratori avevano individuato già in passato tale carenza del modello DPD [26, 27, 28] e avevano proposto un nuovo modello, noto come Fluid Particle Model (FPM) [27], incorporando due componenti di forza non centrali addizionali nella forza dissipativa, accoppiate alle rispettive compo-

nenti nella forza random per mezzo del teorema di fluttuazione-dissipazione. Il metodo FPM può essere considerato una generalizzazione del metodo DPD che riproduce anche il momento torcente delle particelle e, simulando anche le velocità angolari, conserva sia la quantità di moto che il momento angolare. Quindi, una singola particella nel metodo FPM è soggetta ad una forza di trascinalamento e a un momento torcente. I risultati delle simulazioni suggeriscono che queste forze non sono correttamente bilanciate e non è possibile ottenere il comportamento idrodinamico corretto di una singola particella. È stata, dunque, proposta una nuova formulazione delle equazioni FPM che deviano di poco da quelle della DPD standard. Il metodo risultante [29] descrive correttamente l'idrodinamica della particella e gode della semplicità computazionale e di implementazione dell'approccio DPD.

2.1 Funzione peso modificata

Un semplice approccio adottato per migliorare la modellazione del comportamento dinamico di un fluido DPD è stato introdurre una funzione peso generalizzata [25] per la forza stocastica e dissipativa:

$$w^D(r_{ij}) = [w^R(r_{ij})]^2 = \begin{cases} \left(1 - \frac{r_{ij}}{r_c}\right)^s \hat{\mathbf{r}}_{ij}, & r_{ij} < r_c \\ 0 & r_{ij} \geq r_c \end{cases} \quad (2.1)$$

dove r_c e s sono rispettivamente il raggio di *cutoff* globale e l'esponente della funzione peso della forza dissipativa. Quando $r_c = 1$ e $s = 2$ si ritrova la funzione peso quadratica relativa alla forza dissipativa usata nei modelli DPD convenzionali. Adottata la funzione peso riportata in Eq.(2.1) ed esprimendo le proprietà dinamiche del sistema in funzione dei parametri DPD, si può visualizzare l'effetto del parametro di interesse s su di esse. Seguendo il metodo approssimato proposto da Groot e Warren [2] e riportato nella sezione successiva Sez.(3.1) in riferimento alla funzione peso convenzionale, è possibile ricavare la relazione tra la viscosità dissipativa e il parametro s . Le assunzioni semplificative fatte nella derivazione, come assumere la funzione di distribuzione radiale costante e unitaria, $g(r) = 1.0$, e quindi densità ρ del sistema uniforme, non comportano deviazioni significative rispetto ai risultati derivati utilizzando tecniche più sofisticate [15]. La viscosità di un semplice sistema DPD è costituita da due parti: il termine cinetico, spesso trascurabile, e il termine dissipativo, che è dominante e denotato da η^D . La relazione che esprime tale coefficiente macroscopico di trasporto in funzione dei parametri DPD, nel caso della funzione peso generalizzata proposta, riportata in Eq. (2.1), è:

$$\eta^D = \frac{2\pi\gamma\rho^2r_c^5}{15} \left(\frac{1}{s+1} - \frac{4}{s+2} + \frac{6}{s+3} - \frac{4}{s+4} + \frac{1}{s+5} \right). \quad (2.2)$$

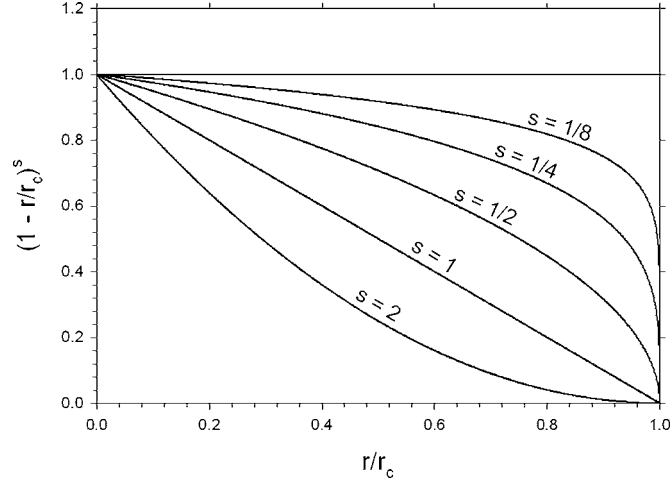


Figura 2.2: Funzione peso generalizzata relativa alla forza dissipativa (immagine adattata da Ref. [25]).

La funzione di ponderazione nel caso in cui $s \leq 2$ comporta un'interazione dissipativa interparticellare più forte rispetto a quella che si ha utilizzando la funzione peso quadratica standard per una data configurazione di particelle. La funzione peso per valori maggiori di r/r_c aumenta drasticamente al diminuire di s . La derivata della funzione di ponderazione in corrispondenza di $r/r_c = 1$ nei casi in cui $s \leq 1$ presenta una discontinuità e la funzione peso stessa è discontinua in $r/r_c = 1$ quando $s = 0$. Non è chiaro quale sia l'effetto della discontinuità dei gradienti della forza dissipativa e stocastica in $r/r_c = 1$ sul sistema. Sicuramente, però, non è desiderabile avere un repentino cambio di forze in prossimità di $r/r_c = 1$. Una scelta ottimale in quest'ottica può essere $s = 1/2$. Le proprietà di trasporto per questo valore di s , in funzione degli altri parametri DPD, sono riportate in Tab. 2.1. Si

Tabella 2.1: Proprietà di trasporto per due diversi valori di s espresse in funzione degli altri parametri DPD.

<i>Proprietà</i>	<i>Convenzionale</i> ($s = 2$)	<i>Modified</i> ($s = \frac{1}{2}$)
Coefficiente di diffusione, \mathcal{D}	$\frac{45k_B T}{2\pi\gamma\rho r_c^3}$	$\frac{315k_B T}{64\pi\rho r_c^3}$
Viscosità, η	$\frac{\rho\mathcal{D}}{2} + \frac{2\pi\gamma\rho^2 r_c^5}{1575}$	$\frac{\rho\mathcal{D}}{2} + \frac{512\pi\gamma\rho^2 r_c^5}{51975}$
Numero di Schmidt, Sc	$\frac{1}{2} + \frac{(2\pi\gamma\rho r_c^4)^2}{70875k_B T}$	$\frac{1}{2} + \frac{(2\pi\gamma\rho r_c^4)^2}{1999k_B T}$

nota che la componente dissipativa del numero di Schmidt per il sistema

modificato è 35.5 volte maggiore di quella per il sistema convenzionale. Al diminuire di s si ha, infatti, una diminuzione del coefficiente diffusivo e un'incremento della viscosità dissipativa, come si può notare dal grafico riportato in Fig. 2.3.

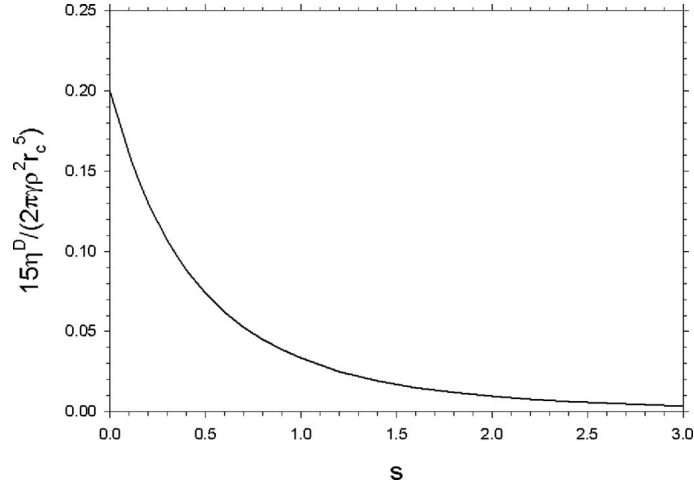


Figura 2.3: Contributo dissipativo della viscosità dinamica in funzione dell'esponente della funzione peso generalizzata che definisce la dipendenza dalla distanza interparticellare della forza dissipativa (immagine adattata da Ref. [25]).

2.2 Sforzi di taglio

Per avere una corretta descrizione del momento torcente è necessaria un'estensione della DPD standard, che considera unicamente le forze agenti nella direzione della retta congiungente i centri di massa dei *beads*, *central forces* designate con in pedice \parallel , con componenti di forza perpendicolari, *shear forces* designate con in pedice \perp . L'estensione del modello prevede l'inclusione della componente di taglio della forza dissipativa, *shear dissipation*. Le forze dissipativa e stocastica presentano, dunque, rispettivamente le seguenti espressioni:

$$\mathbf{F}_{ij}^D = -\gamma_{\parallel} w_{\parallel}(r_{ij})(\hat{\mathbf{r}}_{ij} \cdot \mathbf{v}_{ij})\hat{\mathbf{r}}_{ij} - \gamma_{\perp} w_{\perp}(r_{ij})(\mathbf{I} - \hat{\mathbf{r}}_{ij}\hat{\mathbf{r}}_{ij}^T)\mathbf{v}_{ij}, \quad (2.3)$$

$$\mathbf{F}_{ij}^R = \sigma_{\parallel} w_{\parallel}(r_{ij})\zeta_{ij}\Delta t^{-1/2}\hat{\mathbf{r}}_{ij} + \sigma_{\perp} w_{\perp}(r_{ij})(\mathbf{I} - \hat{\mathbf{r}}_{ij}\hat{\mathbf{r}}_{ij}^T)\zeta_{ij}\Delta t^{-1/2}, \quad (2.4)$$

Questa estensione non comporta un aumento rilevante del costo computazionale. Essa inoltre rappresenta un buon compromesso tra la necessità di simulare in maniera efficiente sistemi complessi e riprodurre correttamente le

proprietà dinamiche del sistema. La DPD convenzionale si ritrova ponendo γ_{\perp} pari a zero.

2.3 Forza dissipativa rotazionale

Un modello che fornisce una descrizione più completa per ciascuna particella si ottiene simulando sia le velocità traslazionali che quelle angolari. Quindi, oltre all'equazione di conservazione della quantità di moto, si ha anche quella di conservazione del momento angolare. Tra i termini di forza che agiscono su ciascuna particella del sistema compare anche il termine di forza dissipativa rotazionale, il cosiddetto *rotational dumping*. Tale modello deriva da una riformulazione delle equazioni del modello FPM sviluppato da Español e collaboratori [27] rispetto al quale ha il vantaggio di consentire una corretta descrizione dell'idrodinamica del sistema, pur essendo meno sofisticato e presentando un algoritmo di più semplice implementazione. Tuttavia, se paragonato all'estensione del modello DDP includente la *shear dissipation* presentato nella Sez. 2.2, è sicuramente più dispendioso dal punto di vista computazionale.

Considerando un sistema costituito da N particelle che hanno posizione \mathbf{r}_i , velocità traslazionale \mathbf{v}_i e velocità angolare $\boldsymbol{\Omega}_i$, ogni particella i -esima è soggetta alla forza e al momento torcente dati da:

$$\mathbf{f}_i = \sum \mathbf{F}_{ij} , \quad (2.5)$$

$$\mathbf{t}_i = \sum \frac{\mathbf{r}_{ij}}{2} \times \mathbf{F}_{ij} , \quad (2.6)$$

in cui $\mathbf{r}_{ij} = \mathbf{r}_i - \mathbf{r}_j$ e $\mathbf{v}_{ij} = \mathbf{v}_i - \mathbf{v}_j$ sono rispettivamente la distanza relativa e la velocità relativa tra la particella i -esima e la particella j -esima. La forza esercitata dalla particella j sulla particella i è costituita dai seguenti contributi:

$$\mathbf{F}_{ij}^C = \mathbf{F}_{ij}^C + \mathbf{F}_{ij}^T + \mathbf{F}_{ij}^R + \tilde{\mathbf{F}}_{ij} , \quad (2.7)$$

in cui il primo termine rappresenta la forza conservativa, il secondo la forza dissipativa traslazionale, il terzo la forza dissipativa rotazionale e il quarto la forza stocastica. La forza conservativa è data da:

$$\mathbf{F}_{ij} = \begin{cases} a_{ij} \left(1 - \frac{r_{ij}}{r_c}\right) \hat{\mathbf{r}}_{ij}, & r_{ij} < r_c \\ 0 & r_{ij} \geq r_c \end{cases} \quad (2.8)$$

in cui a_{ij} determina l'intensità della repulsione, analogamente al modello DPD originariamente formulato. La forza di dissipazione traslazionale è la somma della componente parallela alla retta congiungente i centri di massa delle particelle, cosiddetta *central dissipation* designata con l'apice C , e

della componente di taglio, *shear dissipation*, designata con apice S . La sua espressione è riportata di seguito:

$$\mathbf{F}_{ij}^T = -\gamma_{ij}^C f^2(r_{ij})(\mathbf{v}_{ij} \cdot \hat{\mathbf{r}}_{ij})\hat{\mathbf{r}}_{ij} - \gamma_{ij}^S f^2(r_{ij}) [\mathbf{v}_{ij} - (\mathbf{v}_{ij} \cdot \hat{\mathbf{r}}_{ij})\hat{\mathbf{r}}_{ij}] , \quad (2.9)$$

in cui $f(r_{ij})$ è la funzione peso adimensionale generalizzata. La forza dissipativa rotazionale è esprimibile come segue:

$$\mathbf{F}_{ij}^R = -\gamma_{ij}^S f^2(r_{ij}) \left[\frac{\mathbf{r}_{ij}}{2} \times (\boldsymbol{\Omega}_i + \boldsymbol{\Omega}_j) \right] . \quad (2.10)$$

Infine, la forza stocastica è descritta dalla seguente espressione:

$$\tilde{\mathbf{F}} = f(r_{ij}) \left[\frac{1}{\sqrt{3}} \sigma_{ij}^C \text{tr}(\mathbf{W}_{ij}) \mathbf{1} + \sqrt{2} \sigma_{ij}^S \mathbf{W}_{ij}^A \right] \cdot \hat{\mathbf{r}}_{ij} , \quad (2.11)$$

dove $\sigma_{ij}^{C,S} = \sqrt{\frac{2k_B T \gamma_{ij}^{C,S}}{\Delta t}}$ è rappresentativo dell'ampiezza del rumore della forza stocastica, Δt è il *timestep*, e \mathbf{W}_{ij} è la matrice $d \times d$ di numeri casuali che seguono una statistica Gaussiana per la dimensione spaziale d . La matrice $\mathbf{W}_{ij}^{A,\mu\nu} = \frac{1}{2}(\mathbf{W}_{ij}^{\mu\nu} - \mathbf{W}_{ji}^{\mu\nu})$ è la matrice antisimmetrica di \mathbf{W}_{ij} . La DPD convenzionale si ritrova ponendo γ^S pari a zero.

Capitolo 3

Calcolo delle proprietà di trasporto

Al fine di predire correttamente il comportamento dinamico di fluidi semplici e complessi (come ad esempio della materia soffice) in varie condizioni mediante simulazioni DPD, è essenziale indagare l'influenza dei parametri del modello sui coefficienti di trasporto sia attraverso derivazioni analitiche che simulazioni numeriche con i vari metodi implementabili per modelli DPD.

La viscosità di un fluido DPD è stata ricavata indipendentemente in diversi lavori presenti in letteratura. L'approccio più sofisticato è stato seguito da Marsh e collaboratori [15], i quali hanno sviluppato la teoria cinetica per la DPD con lo scopo di correlare i coefficienti di trasporto, la viscosità e il coefficiente di diffusione, ai parametri del modello DPD. Groot e Warren, invece, hanno seguito una procedura semplificata [2] con una serie di assunzioni ottenendo risultati equivalenti. Le relazioni analitiche tra i parametri DPD e i coefficienti di trasporto sono importanti in quanto costituiscono un termine di paragone almeno qualitativo per i risultati numerici ottenuti dalle simulazioni numeriche.

3.1 Relazioni analitiche tra i parametri DPD e le proprietà di trasporto

L'espressione ricavata analiticamente per la viscosità di un fluido DPD è la seguente:

$$\eta = \eta_K + \eta_D = \frac{45k_B T}{4\pi\gamma r_c^3} + \frac{2\pi\gamma\rho^2 r_c^5}{1575}, \quad (3.1)$$

in cui i due termini hanno origine rispettivamente dal contributo cinetico e dissipativo del tensore degli sforzi. Il primo è dovuto al flusso di quantità di moto generato dal moto diffusivo delle particelle attraverso le linee di flusso. La viscosità cinetica è, infatti, legata al coefficiente di diffusione.

Il contributo dissipativo degli sforzi è, invece, legato alla forza di attrito agente tra *beads* che si muovono su diverse linee di flusso. L'Eq. (3.1) non fornisce una buona stima della viscosità quando le interazioni tra le particelle di tipo conservativo sono significative. Infatti, è stata ricavata trascurando il contributo della forza conservativa [2]. L'espressione per il coefficiente di diffusione è la seguente:

$$D = \frac{45k_B T}{2\pi\gamma\rho r_c^3}. \quad (3.2)$$

che risulta essere uguale al termine cinetico della viscosità.

3.2 Metodi per il calcolo della viscosità nelle simulazioni DPD

La viscosità di un fluido è il parametro principale nel determinare le caratteristiche di flusso. Fluidi reali, per i quali il comportamento reologico è di interesse intrinseco, possono essere simulati con modelli come la MD. La DPD rientra sempre in una classe di modelli basati su particelle sviluppati recentemente per riprodurre correttamente il comportamento idrodinamico nel tentativo di superare i limiti di scala temporale connessi alla MD. In questi modelli la viscosità deve essere generalmente calcolata perché mancano espressioni teoriche sufficientemente accurate. Esistono diversi metodi numerici disponibili per calcolare la viscosità, tutti con i relativi vantaggi e svantaggi. Una macrodivisione può essere fatta tra metodi di equilibrio e di non-equilibrio [30].

Per impiegare i primi non è necessario creare uno stato di non equilibrio nel sistema. Al contrario il sistema simulato deve essere “equilibrato”: la viscosità è determinata valutando le fluttuazioni all'equilibrio e usando le relazioni di Green-Kubo o Einstein-Helfand [31, 32, 33]. In questo caso, le condizioni al contorno periodiche semplici sono adeguate. Inoltre, la velocità di taglio è per definizione zero (*zero shear rate*) quindi si è automaticamente nel regime di risposta lineare. Sebbene questo approccio sia molto appetibile per la motivazioni appena spiegate, le grandi fluttuazioni degli sforzi all'equilibrio comportano uno scarso rapporto segnale/rumore.

Nelle simulazioni di non equilibrio cosiddette NEMD (*non equilibrium dynamic simulation*) viene applicata un'appropriata perturbazione: sufficientemente piccola da essere nel regime di risposta lineare. La viscosità è correlata alle proprietà dello stato stazionario di non equilibrio o al decadimento allo stato equilibrato. In NEMD conviene simulare uno stato stazionario di non equilibrio con proprietà statistiche indipendenti dal tempo. Questo è facilmente realizzabile nel caso di flusso di taglio stazionario: si possono usare le condizioni al contorno di Lees-Edwards, il flusso di Poiseuille oppure lo scambio di quantità di moto [34, 35]. Quest'ultimo è un approccio inverso rispetto ai tipici metodi NEMD, cosiddetto RNEMD (*reverse non equilibrium*

simulation), in quanto consiste nell'invertire causa ed effetto: imponendo il flusso di quantità di moto si calcola il gradiente di velocità [36]. Risulta vantaggioso quando il flusso di quantità di moto converge lentamente.

3.2.1 Equazioni di Green-Kubo

Green e Kubo hanno dimostrato [31, 32] che i coefficienti di trasporto come la conducibilità termica e la viscosità possono essere calcolati dalle funzioni di autocorrelazione dei corrispondenti tensori in condizioni di equilibrio termodinamico. La motivazione fisica è data dall'ipotesi della regressione di Onsager [37, 38] secondo la quale le fluttuazioni sono presenti in tutti i sistemi all'equilibrio. La dissipazione delle fluttuazioni è correlata al rilassamento verso l'equilibrio del sistema una volta che è stato perturbato da una forza esterna. I due fenomeni avvengono sulla stessa scala temporale: secondo il teorema di fluttuazione-dissipazione il decadimento delle fluttuazioni spontanee equivale, in media, al rilassamento del sistema stesso verso lo stato di equilibrio a seguito di un disturbo esterno [39]. Di conseguenza la viscosità a scorrimento nullo è correlabile ad un'opportuna funzione di correlazione che misura il decadimento di tali fluttuazioni spontanee. Per quanto riguarda la viscosità dinamica, la relazione di Green-Kubo permette di calcolarla integrando la funzione di autocorrelazione degli sforzi di taglio, SACF (stress autocorrelation function) come segue:

$$\eta = \frac{V}{k_B T} \int_0^\infty dt \langle \sigma_{\alpha\beta}(t_0) \sigma_{\alpha\beta}(t_0 + t) \rangle, \quad (\alpha \neq \beta), \quad (3.3)$$

dove V è il volume del sistema, ρ è la densità e $\sigma_{\alpha\beta}$ denota la componente del tensore degli sforzi e k_B è la costante di Boltzmann. Nelle simulazioni basate su particelle il tensore degli sforzi può essere valutato dalla formula Irving-Kirkwood riportata di seguito:

$$\sigma_{\alpha\beta}(t) = \frac{1}{V} \sum_i \sum_{j>i} \mathbf{r}_{ij\alpha}(t) \mathbf{F}_{ij\beta}(t) + \frac{1}{V} \sum_i m_i \mathbf{v}_{i\alpha}(t) \mathbf{v}_{i\beta}(t), \quad (3.4)$$

dove V è il volume costante del sistema e m_i è la massa della particella i -esima. Gli indici α e β fanno riferimento alle direzioni spaziali x , y e z (α e $\beta = x, y, z$) e designano, dunque, le componenti vettoriali o tensoriali invece, i e j indicano le particelle. Il termine $\mathbf{F}_{ij\beta}$ include le forze conservativa, dissipativa e stocastica applicate dalla particella j sulla particella i -esima ($F_{ij\beta} = F_{ij\beta}^C + F_{ij\beta}^D + F_{ij\beta}^R$). I termini $\mathbf{v}_{i\alpha}$ e $\mathbf{r}_{ij\alpha}$ rappresentano la componente α rispettivamente del vettore velocità della particella i -esima e del vettore distanza tra la particella j e la particella i . Per calcolare la viscosità deve essere calcolata l'area sottesa al profilo di autocorrelazione attraverso i metodi di integrazione numerica. Il grande vantaggio di questo metodo è quello di poter valutare i coefficienti di trasporto senza dover creare uno

stato di non-equilibrio, aspetto fondamentale nella simulazione di fluidi complessi. Infatti, nell'ottica di calcolare la viscosità del sistema in una precisa configurazione, è bene evitare la possibile deformazione delle microstrutture introducendo stati di non equilibrio.

3.2.2 Equazioni di Einstein-Helfand

Un altro approccio per valutare la viscosità in condizioni di equilibrio è stato proposto da Helfand [33] ed è analogo all'approccio di Einstein seguito per valutare il coefficiente di diffusione:

$$\eta = \lim_{t \rightarrow \infty} \frac{V}{2k_b T t} \langle [G_{\alpha\beta}(t) + G_{\alpha\beta}(0)]^2 \rangle, \quad (\alpha \neq \beta), \quad (3.5)$$

dove $J_{\alpha\beta}$ è il momento di Helfand ed è espresso come segue:

$$G_{\alpha\beta}(t) = \sum_i m r_{i\alpha}(t) v_{i\beta}(t). \quad (3.6)$$

e dal momento che questo metodo è equivalente a quello di Green-Kubo, l'Eq. (3.6) è esprimibile in funzione del tensore degli sforzi:

$$G_{\alpha\beta}(t) = G_{\alpha\beta}(0) + \int_0^t \sigma_{\alpha\beta}(\tau) d\tau. \quad (3.7)$$

3.2.3 Metodo di Müller-Plathe (momentum interchange)

Si differenzia dagli altri metodi RNEMD [36] per la semplicità del modo in cui viene implementato il flusso di quantità di moto stazionario. Gli eventi di scambio di quantità di moto delle particelle, sebbene non fisici, conservano sia la quantità di moto che l'energia cinetica se le masse delle particelle sono uguali. Inoltre poiché le posizioni delle particelle non vengono cambiate si conservano l'energia potenziale e, quindi, l'energia totale del sistema. Inoltre il profilo di velocità, è una proprietà robusta e rapidamente convergente. Il flusso di quantità di moto viene imposto al sistema in maniera non fisica. La *box* di simulazione viene suddivisa in lastre, cosiddetti *slabs*, lungo la coordinata z come si evince dalla Fig. 3.1. Gli atomi appartenenti allo *slab* in corrispondenza della coordinata $z = 0$ (e nella sua immagine periodica) sono spinti nella direzione $+x$, quelli appartenenti allo *slab* in corrispondenza della coordinata $z = L_z/2$ nella direzione $-x$. L'algoritmo di scambio della quantità di moto funziona come segue: individua nello *slab* che si muove in direzione $+x$ l'atomo con la quantità di moto maggiore in direzione $-x$ (ovverosia l'atomo con più piccola quantità di moto); nell'altro *slab* che si muove in direzione $-x$ sceglie l'atomo con la più grande quantità di moto in direzione $+x$; scambia le quantità di moto dei due atomi. La quantità di moto Δp_x trasferita dallo *slab* alla coordinata $z = L_z/2$ a quella

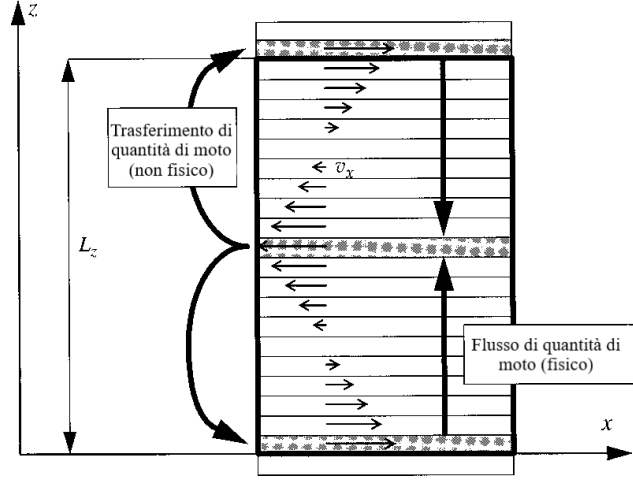


Figura 3.1: Schematizzazione dello scambio di quantità di moto artificiale, *swapping*, tra i *beads* nel metodo di Müller-Plathe (elaborazione immagine estratta da Ref. [36]).

in corrispondenza di $z = 0$ è precisamente nota. Se gli scambi di quantità di moto sono ripetuti periodicamente, la quantità di moto totale trasferita nel corso della simulazione P_x è la somma dei Δp_x .

Il sistema risponde a questa situazione di non equilibrio lasciando che il flusso di quantità di moto fisico avvenga nella direzione opposta come si osserva in Fig. 3.1. Dunque, il flusso fisico di quantità di moto viene calcolato con la seguente equazione:

$$j(p_x) = \frac{p_x}{2tA}, \quad (3.8)$$

dove t è la durata della simulazione e $A = L_x L_y$. Il fattore 2 deriva dalla periodicità del sistema. Il flusso di quantità di moto $j_z(p_x)$ attraversa una superficie perpendicolare alla direzione del flusso (z) di area A nell'unità di tempo t . Il flusso fisico di quantità di moto genera un profilo di velocità nel fluido. La velocità v_x nella direzione x in ogni *slab* è calcolata come media delle velocità $v_{x,i}$ di tutti gli atomi i nello *slab*. Se il flusso di quantità di moto non è eccessivo il profilo di velocità è approssimativamente lineare e la viscosità è stimabile dalla seguente equazione:

$$j_z = \eta \frac{\partial v_x}{\partial z}, \quad (3.9)$$

dove η , il coefficiente di proporzionalità tra il flusso di quantità di moto e il gradiente di velocità, è la viscosità dinamica.

3.2.4 Metodo delle condizioni al contorno di Lees-Edwards

Il metodo è stato sviluppato [34] per investigare il comportamento di un liquido sottoposto all'azione di una forza di taglio di notevole intensità. É

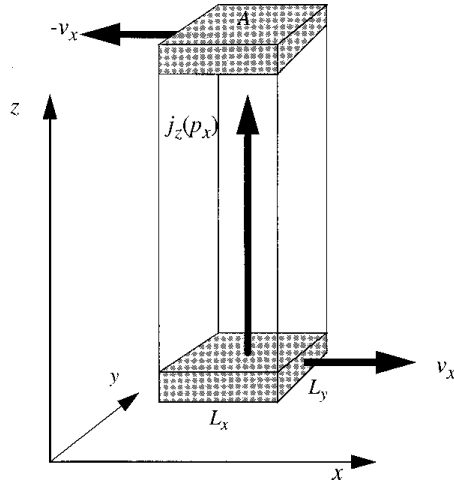


Figura 3.2: Flusso fisico di quantità di moto generato, come risposta alla perturbazione esterna imposta, dal sistema (immagine adattata da Ref. [36]).

stato introdotto come novità rispetto ai metodi fino ad allora utilizzati per il calcolo dei coefficienti di trasporto con l'idea di poter applicare uno sforzo di taglio arbitrario. In linea di principio differenti valori di sforzo di taglio possono essere ottenuti attribuendo differenti velocità ai *beads* vicini al confine superiore e quello inferiore della *box* di simulazione. Il massimo valore di velocità nella parte superiore della *box* è uguale a $\frac{1}{2}\dot{\gamma}l$, dove $\dot{\gamma}$ è il valore di velocità di taglio, nota anche come velocità di deformazione o *shear rate*, imposta al sistema e l è la lunghezza della *box*. Se la conservazione del momento è rispettata, si ottiene un profilo di velocità lineare lungo la *box*. La viscosità può, quindi, essere ottenuta mediante la seguente relazione:

$$\eta = -\frac{P_{xy}}{\dot{\gamma}}, \quad (3.10)$$

dove $\dot{\gamma}$ è la velocità di taglio o velocità di deformazione imposta, mentre p_{xy} è la componente non nulla di velocità del tensore degli sforzi, non appartenente alla diagonale. Tuttavia, un limite al metodo è rappresentato dal range limitato di valori di *shear rate* effettivamente applicabili senza ottenere anomalie nei risultati di viscosità ottenuti rispetto alle previsioni teoriche [40]. Quando i valori di *shear rate* sono superiori all'unità, la viscosità del sistema inizia ad aumentare, a causa di artefatti legati alle condizioni al contorno di Lees-Edwards. Allo stesso tempo, l'intensità dello sforzo di taglio deve indurre velocità sufficientemente alte per evitare che il suo effetto venga mascherato dal moto termico delle *beads*. Un'ultima osservazione riguarda la conversione dalle unità DPD e alle unità fisiche reali. Per confrontare i risultati delle simulazioni DPD con quelli sperimentali, è necessario definire un *benchmark* di conversione, un insieme di valori che rappresentano quantità fisiche. La conversione delle unità adimensionali DPD in unità fisiche reali,

secondo l'insieme dei fattori di conversione ricavati in condizioni di equilibrio, potrebbero produrre valori irrealistici per le quantità non in equilibrio, come, ad esempio, la velocità di taglio effettivamente applicata.

3.2.5 Metodo Force-driven Poiseuille flow

Il flusso di Poiseuille [35] si ottiene applicando una forza a ciascuna particella. Ad esempio se la gravità agisce nella direzione z e il fluido scorre tra due piastre collocate sul piano xy , il campo di flusso risultante presenta un profilo degli sforzi di taglio lineare e un profilo di velocità parabolico:

$$\tau_{xz} = \rho g_z \left(x - \frac{D}{2}\right), \quad (3.11)$$

$$v_z = \frac{\rho g_z}{2\eta} (xD - x^2), \quad (3.12)$$

in cui τ_{xz} è lo sforzo di taglio, v_z è la velocità nella medesima direzione della forza, η è la viscosità dinamica, ρ è la densità massica, g_z è la costante gravitazionale, x identifica la posizione tra le due piastre e D è la distanza tra le piastre. L'integrazione dell'Eq.(3.12) rispetto ad x sulla distanza complessiva tra le piastre fornisce la velocità media del sistema:

$$\langle v_z \rangle = \frac{1}{D} \int_0^D v_z(x) dx = \frac{\rho g_z D^2}{12\eta}. \quad (3.13)$$

Si noti che questa derivazione presuppone la densità costante nel dominio di simulazione e la viscosità costante rispetto alle velocità di taglio variabili. La velocità di taglio va da zero, in corrispondenza del quale la velocità assume il suo valore massimo, ad un valore massimo dove la velocità è nulla. Questo metodo non è quindi adatto per la determinazione della viscosità variabile in funzione della velocità di taglio, ma è limitato a fluidi in regime newtoniano. Le condizioni al contorno periodiche non sono adeguate perché non garantiscono velocità nulla ai bordi del dominio di simulazione. Per aggirare questo problema sono state proposte condizioni al contorno stocastiche [41] e condizioni cosiddette di "rimbalzo" [42]. Tuttavia, in entrambi i casi dall'analisi del profilo di densità risultano anomalie in corrispondenza dei bordi della *box* dovute ad artefatti del metodo di simulazione. Una soluzione più semplice a questo problema è rappresentata dalla suddivisione del dominio di simulazione in due parti e dall'applicazione di forze di medesima intensità ma agenti in direzioni opposte ai due sotto-domini. In questo caso, le condizioni al contorno periodiche sono adeguate all'ottenimento di campi di flusso parabolici, come illustrato in Fig. 3.3. Uno dei vantaggi principali di questo metodo è la minore sensibilità al rumore del valore di velocità media del sistema rispetto a quello dello sforzo di taglio medio. La spiegazione di questa differenza risiede nel fatto che lo sforzo di taglio è correlato alla derivata del campo di velocità. Un altro vantaggio è rappresentato dalla semplicità di implementazione delle forze.

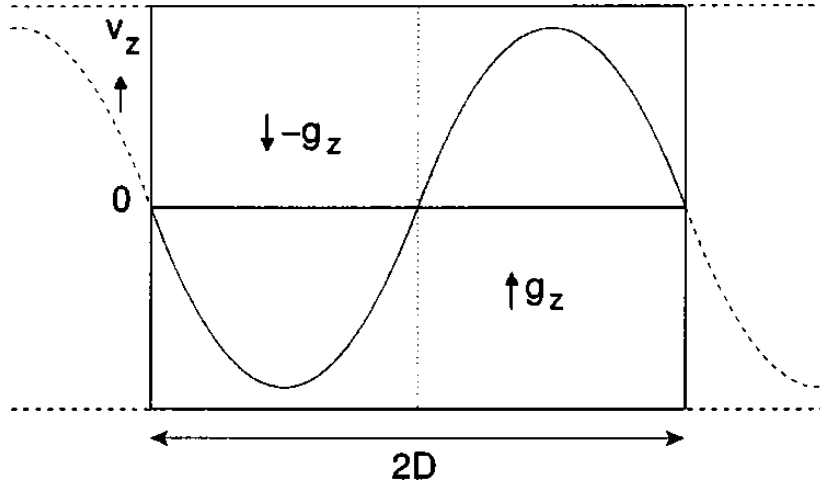


Figura 3.3: Rappresentazione schematica del metodo di Poiseuille (immagine adattata da Ref. [35]).

3.3 Calcolo del coefficiente di diffusione

Il coefficiente di diffusione nelle simulazioni DPD può essere calcolato con la relazione di Einstein o di Green-Kubo. La diffusione è il processo attraverso il quale un profilo non uniforme di concentrazione viene appiattito in assenza di flusso. La diffusione è causata dal moto molecolare delle particelle nel fluido [43]. La legge macroscopica che descrive la diffusione è la legge di Fick di seguito riportata:

$$\mathbf{J} = -D\nabla c, \quad (3.14)$$

secondo la quale il flusso \mathbf{J} della specie diffondente è proporzionale al gradiente di concentrazione di quella specie. L'evoluzione del profilo di concentrazione nel tempo si ottiene inserendo la legge costitutiva di Fick nell'equazione che esprime la conservazione della quantità totale della specie diffondente:

$$\frac{\partial c(\mathbf{r}, t)}{\partial t} = D\nabla^2 c(\mathbf{r}, t). \quad (3.15)$$

Nel caso di concentrazione normalizzata all'unità,

$$\int c(\mathbf{r}, t) d\mathbf{r} = 1 \quad (3.16)$$

la quantità così definita:

$$\langle r^2(t) \rangle \equiv \int c(\mathbf{r}, t) \mathbf{r}^2 d\mathbf{r}^2 \quad (3.17)$$

è la distanza media percorsa dalle particelle, nota anche come *mean square displacement*. L'evoluzione temporale di questa quantità si ricava multipli-

cando per \mathbf{r}^2 ambo i membri dell'Eq.(3.15) ed integrando nello spazio:

$$\frac{\partial}{\partial t} \int c(\mathbf{r}, t) \mathbf{r}^2 \mathbf{r}^2 d\mathbf{r} = \mathcal{D} \int \nabla^2 c(\mathbf{r}, t) \mathbf{r}^2 d\mathbf{r} \quad (3.18)$$

ovvero

$$\frac{\partial \langle r^2(t) \rangle}{\partial t} = \mathcal{D} \int \nabla^2 c(\mathbf{r}, t) r^2 d\mathbf{r} \quad (3.19)$$

da cui si ottiene:

$$\frac{\partial \langle r^2(t) \rangle}{\partial t} = 2d\mathcal{D} . \quad (3.20)$$

Nel caso tridimensionale tale derivazione fornisce la seguente relazione:

$$D = \lim_{t \rightarrow \infty} \frac{1}{6t} \langle |\mathbf{r}(t_0 + t) - \mathbf{r}(t_0)|^2 \rangle , \quad (3.21)$$

che mette in relazione il coefficiente macroscopico di trasporto \mathcal{D} con $\langle r^2(t) \rangle$, la *mean square displacement* che ha un'interpretazione microscopica: la distanza media percorsa dalle molecole nell'intervallo di tempo. In una simulazione numerica per ogni particella i si può misurare la distanza percorsa nel tempo t , come:

$$\langle \Delta r(t)^2 \rangle = \frac{1}{N} \sum_{i=1}^N \Delta r_i(t)^2 . \quad (3.22)$$

Il coefficiente di diffusione può essere espresso direttamente in termini di velocità delle particelle, infatti lo spostamento della particella è l'integrale della sua velocità:

$$\langle \Delta r(t) \rangle = \int_0^t \mathbf{v}(t) dt . \quad (3.23)$$

Il metodo di Green-Kubo correla, infatti, il coefficiente di diffusione alla funzione di autocorrelazione delle velocità (VAFC) risultando in:

$$D = \frac{1}{3} \int_{t_0}^{\infty} dt \langle \mathbf{v}(t_0) \cdot \mathbf{v}(t_0 + t) \rangle . \quad (3.24)$$

La VAFC è una proprietà di equilibrio del sistema che descrive le correlazioni tra le velocità al tempo t_0 e al tempo $t_0 + t$ lungo la traiettoria di equilibrio. I due metodi, dal punto di vista teorico, sono equivalenti e quindi la scelta dell'uno o dell'altro non dovrebbe comportare grandi differenze nei risultati.

Capitolo 4

Introduzione ai fluidi complessi

La DPD negli ultimi decenni è stata perfezionata sia dal punto di vista concettuale, con la proposta di varie estensioni rispetto al modello originariamente sviluppato, che di efficienza computazionale, con il *testing* di diversi algoritmi di integrazione delle equazioni, in quanto risulta essere molto promettente per la simulazione di fluidi di notevole interesse tecnologico. Si tratta di una classe di materiali, nota con il nome di fluidi complessi, che, in virtù della grande versatilità applicativa in vari settori industriali, quali alimentare, cosmetico, farmaceutico e cura della persona, attira sempre più l'interesse scientifico. Rappresentano spesso, inoltre, intermedi dell'industria manifatturiera delle fibre, dei films, del packaging e così via. Rientrano in questa classe di materiali i liquidi e i fusi polimerici, le sospensioni di particelle colloidali, le soluzioni di micelle e le schiume liquide [44]. Si distinguono dai semplici solidi cristallini e dai liquidi ordinari in quanto presentano scale spaziali molecolari o strutturali superiori a quella atomistica. Infatti, possono creare una molteplicità di microstrutture che possono evolvere nel tempo assumendo forma e dimensioni completamente diverse. Hanno proprietà meccaniche intermedie tra i liquidi ordinari e i solidi ordinari. Il comportamento reologico è fortemente influenzato dalle microstrutture formate. È fondamentale riuscirne a modellare la reologia in quanto la viscosità dei fluidi complessi è spesso il deterrente per il loro impiego in una particolare applicazione.

4.1 I polossameri

I Polossameri sono stati sviluppati dalla BASF a partire dal 1973, sono noti con il nome commerciale di Pluronic[®]. Si tratta di copolimeri anfifilici a blocchi non ionici. Sono, infatti, costituiti da un blocco centrale di poli(etilen ossido) e due blocchi laterali di poli(propilen ossido). La sintesi dei polossameri avviene in presenza di catalizzatori, idrossido di sodio e potassio, per reazione di condensazione, con apertura di anello, di monomeri di propilene

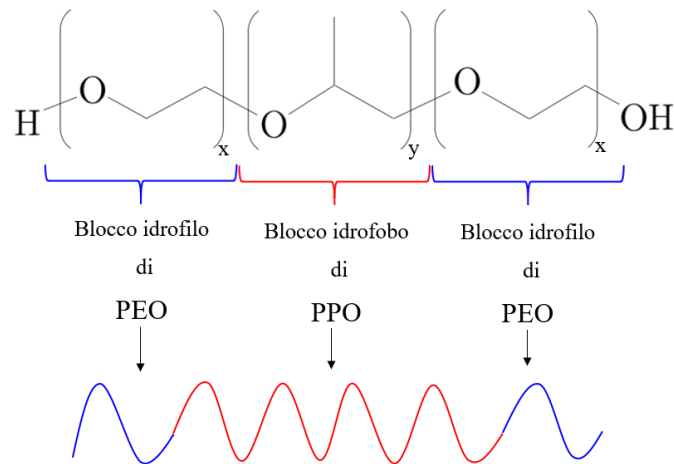


Figura 4.1: Struttura generale di una molecola di Pluronic[®] (elaborazione immagine estratta da Ref. [45]).

ossido (PO) e di etilene ossido (EO). La possibilità di progettare e realizzare, nel processo di sintesi, blocchi di PEO e PPO di varia lunghezza viene sfruttata per ottenere una gamma di prodotti con diverse proprietà anfifiliche, conferendo a questa classe di polimeri una grande versatilità applicativa in ambito industriale. Sono noti per le proprietà di detergenza, stabilizzazione delle dispersioni e delle schiume, emulsificazione e lubrificazione. Sono impiegati nella formulazione di cosmetici e inchiostri. In ambito farmaceutico hanno diverse funzioni: facilitano la solubilizzazione di farmaci, garantiscono il rilascio controllato del farmaco e sono curativi per le ferite da ustione. Nell'industria biotecnologica sono utilizzati per la protezione dei microrganismi dagli stress meccanici introdotti dall'agitatore nel bioreattore [46]. La tipologia di Pluronic è identificata con un codice alfanumerico in cui la lettera rappresenta lo stato fisico di aggregazione (P=paste, L=liquid, F=flake), il primo numero costituito da una o due cifre moltiplicato per 300 fornisce il peso molecolare del blocco di PPO e, infine, l'ultimo numero moltiplicato per 10 indica la percentuale in massa di PEO all'interno della molecola. Un quadro completo dei vari copolimeri in commercio è riportato nella cosiddetta Pluronic grid riportata in Fig. 4.2: lungo le linee verticali varia il peso molecolare del blocco di PPO, mentre lungo le linee orizzontali varia il rapporto PPO/PEO in termini di composizione. Ad esempio, il codice L64 indica un Pluronic allo stato liquido, con un peso molecolare di circa 1800 g/mol di PPO e contenente il 40% in massa di PEO.

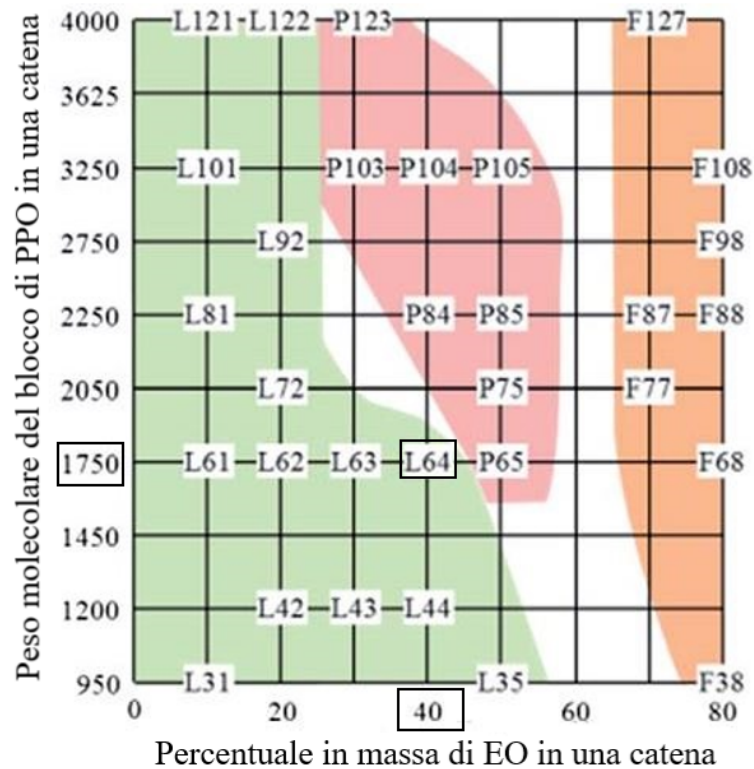


Figura 4.2: Griglia del Pluronic® (elaborazione immagine estratta da Ref. [45]). I tre colori (verde, rosa e arancione) raggruppano le tipologie di Pluronic, in base allo stato fisico di aggregazione a temperatura ambiente, rispettivamente in (*liquid*), L, (*paste*), P, e (*flake*), F. Lungo le linee verticali (dal basso verso l'alto) si ha un incremento del peso molecolare del blocco di PPO contenuto nella catena polimerica. Viceversa, lungo le linee orizzontali (da sinistra a destra) si ha un incremento della percentuale in massa di EO nella catena polimerica. I numeri nei riquadri sono relativi al Pluronic L64.

4.2 Formazione ed evoluzione delle micelle

Le due tipologie di blocchi presenti all'interno del copolimero sono chimicamente incompatibili tra di loro: le catene laterali di Polietilenossido (PEO) sono leggermente idrofile, mentre la catena centrale di Polipropilenossido (PPO) è leggermente idrofoba. Le macromolecole di Pluronic sono, dunque, anfifiliche; ovvero le due tipologie di blocchi costituenti il copolimero hanno affinità verso due ambienti tra loro non compatibili. Questa caratteristica comporta la disposizione delle macromolecole secondo un orientamento preferenziale in presenza di un solvente selettivo verso una delle due parti [47]. In soluzione acquosa, le macromolecole si aggregano spontaneamente andando a formare una varietà di microstrutture differenti a seconda della temperatura, della concentrazione e del rapporto in peso tra i gruppi PPO e PEO. A basse concentrazioni e temperature i polossameri si trovano omogeneamente disciolti in soluzione come monomeri. Infatti, entrambi i tipi di catene del copolimero sono in grado di formare legami a idrogeno con le molecole di solvente presentando un leggero carattere idrofilo. Tuttavia, all'aumentare della temperatura il PPO tende a diventare meno idrofilo più rapidamente del PEO e il copolimero assume le proprietà di tensioattivo. Per individuare l'inizio del fenomeno di aggregazione e formazione delle micelle i due parametri chiave sono la concentrazione micellare critica (CMC) e la temperatura micellare critica (CMT). Essi possono essere definiti per le diverse tipologie di Pluronic: la quantità di catene di PPO è determinante nel fenomeno di micellizzazione e influenza questi due parametri. La CMC e CMT sono inferiori per copolimeri più idrofobi, con una maggiore quantità di PPO, e superiori per copolimeri meno idrofobici, minore quantità di PPO [48]. Le micelle presentano un nucleo costituito dalle unità PPO più idrofobe e una corona esterna formata da quelle PEO più idrofile. Ad esempio, per il Pluronic L64 la CMC in soluzione acquosa misurata sperimentalmente è pari a 0.015 [49]. Questo tipo di organizzazione avviene spontaneamente al fine di minimizzare il contatto tra la parte idrofoba del polimero e l'ambiente idrofilo rappresentato dal solvente acquoso. In altre parole, la formazione di micelle è governata da fattori energetici ed entropici: minimizza l'energia libera all'interno del sistema con un guadagno di entropia [46]. La morfologia delle micelle è determinata dal rapporto tra la quantità di gruppi idrofili e quella di gruppi idrofobi. In caso di prevalenza dei primi le micelle si presentano sferiche, in caso contrario, con una maggiore quantità di catene idrofobiche, hanno forme sferoidali e asimmetriche. All'aumentare della concentrazione di copolimero in soluzione acquosa, le molecole di Pluronic iniziano a formare varie fasi liquide cristalline cosiddette *liotropiche*, illustrate in Fig. 4.3. Le micelle aumentando in numero e dimensione si trovano sempre più ravvicinate consentendo ai gruppi di PEO di interagire tra di loro attraverso forze intermolecolari deboli. Si crea in questo modo un reticolo cristallino costituito da una cella elementare cubica a corpo centrato (CCC) o cubica a

facce centrate (CFC), in cui ciascun vertice è rappresentato da una micella sferica intrappolata dalle corone micellari adiacenti. A livello macroscopico, la soluzione non si presenta più come un liquido, ma come una fase semisolida cosiddetta *fase di gel*. Questo fenomeno è noto come *gelazione termoreversibile* e comporta un drastico aumento di viscosità [50]. A concentrazioni intermedie le micelle sferoidali possono allungarsi assumendo una forma cilindrica: le catene idrofobiche sono disposte nella parte centrale del cilindro, mentre quelle idrofile si dispongono sulla parte esterna. Le micelle cilindriche quando si trovano a distanza ravvicinata interagiscono tra di esse tramite le parti idrofile e si collocano ai nodi di un reticolo cristallino ordinato con celle elementari di forma esagonale. In corrispondenza di concentrazioni ancora più elevate, tali micelle a forma di cilindro (*rod-like*) si accrescono fino ad aggregarsi per riduzione della loro distanza. Si viene a formare in questo modo una struttura particolare, cosiddetta *fase lamellare*, caratterizzata da strati sottili paralleli di molecole di copolimero alternati a strati di solvente acquoso. Infine, per concentrazioni molto elevate di copolimero, si originano le cosiddette *micelle inverse*. In questo caso è il solvente a costituire la fase dispersa all'interno della matrice polimerica che rappresenta, invece, la fase continua [51]. Per prevedere l'evoluzione delle microstrutture per i vari

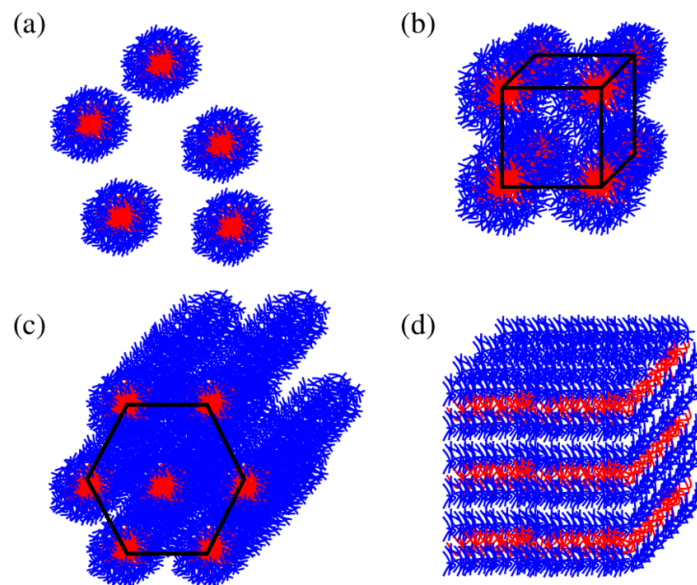


Figura 4.3: Rappresentazione schematica delle strutture, micellare (a), cubica (b), esagonale (c) e lamellare, formate dal Pluronic in solvente acquoso (elaborazione immagine estratta da Ref. [52]).

sistemi costituiti da Pluronic e acqua sono fondamentali i digrammi di fase, nei quali si rappresentano le fasi che si formano in funzione della concen-

trazione di copolimero, sull'asse delle ascisse, e della temperatura sull'asse delle ordinate. A titolo d'esempio si riporta in Fig. 4.4 il diagramma di fase della miscela binaria Pluronic L64-acqua. Il diagramma di fase riportato in

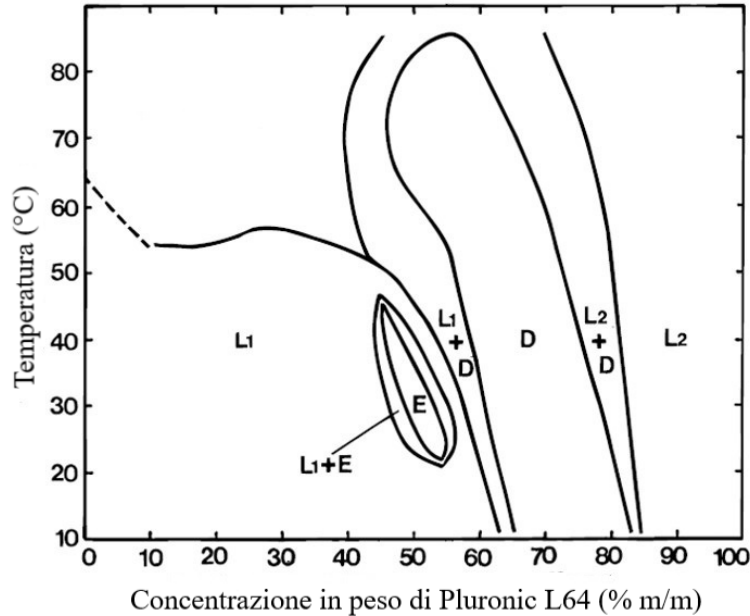


Figura 4.4: Diagramma di fase della miscela binaria Pluronic L64-acqua (elaborazione immagine estratta da Ref. [46]). L₁: fase micellare. E: fase esagonale. D: fase lamellare. L₂: fase micellare inversa.

Fig. 4.4 è stato ricavato utilizzando la spettroscopia di risonanza magnetica nucleare al deuterio (²H-NMR) e la tecnica di diffusione a piccolo angolo di raggi X (SAXS) [51]. Si tratta degli strumenti sperimentali maggiormente impiegati per determinare rispettivamente i confini (²H-NMR) e le caratteristiche delle microstrutture che si formano in soluzioni ascrivibili alle varie fasi che si formano (SAXS). La ²H-NMR è un metodo spettroscopico basato sulle proprietà magnetiche del deuterio, un isotopo dell'idrogeno. La SAXS misura la variazione d'angolo tra la direzione del fascio di raggi X incidente contro il reticolo delle microstrutture rispetto alla direzione in cui lo stesso fascio viene diffuso [50].

4.3 Proprietà reologiche

La viscosità è una misura della propensione di un fluido a trasmettere la quantità di moto in una direzione perpendicolare a quella del flusso:

$$\tau_{xy} = -\mu \frac{dv_x}{dy} . \quad (4.1)$$

Questa espressione è detta *legge di Newton della viscosità*. Tuttavia, l'Eq. (4.1) è valida solo per i fluidi *Newtoniani* per i quali la viscosità rimane costante al variare della velocità di taglio. Nel caso di fluidi non Newtoniani, non è valida [53]. Le proprietà reologiche di soluzioni acquose di copolimeri a blocchi sono straordinariamente complesse in quanto dipendono fortemente dalla forma e dalla dimensione delle microstrutture che si formano. I Polosameri in soluzione acquosa incrementando la concentrazione oltre la CMC evolvono dal punto di vista strutturale attraverso vari microdomini: sferico o cilindrico disordinato; sferico ordinato con disposizione delle micelle in corrispondenza dei nodi di un reticolo costituito da cella elementare CCC o CBC; cilindrico ordinato con disposizione delle micelle in corrispondenza dei nodi di un reticolo costituito da celle elementari esagonali; lamellare. Superata una certa concentrazione si passa di nuovo ad un microdominio disordinato con formazione di micelle inverse. La formazione di reticoli ordinati che si allineano al flusso se sottoposti a sforzo di taglio influisce molto sulle proprietà reologiche. Quando la concentrazione di copolimero è bassa o si hanno poche micelle in soluzione (microdominio sferico disordinato) la miscela binaria acqua-copolimero presenta comportamento Newtoniano, ovvero si comporta come l'acqua. A concentrazioni maggiori di copolimero, in corrispondenza della formazione di micelle cilindriche organizzate in strutture esagonali, il comportamento del fluido è fortemente non-Newtoniano ed, in particolare, pseudo-plastico, la viscosità apparente diminuisce al crescere dello *shear rate*. Una spiegazione microscopica di tale comportamento reologico è data dallo *shear alignment*: le micelle cilindriche si allineano nella direzione del flusso, in modo che i piani più densamente impaccati dei cilindri siano paralleli alle superfici di taglio. Anche per quanto riguarda il microdominio lamellare, si hanno fenomeni di allineamento delle normali alle lamelle nella direzione del gradiente o della vorticità, a seconda delle condizioni di taglio, che vanno ad influenzare il comportamento reologico del sistema [44]. Quando la frazione volumica di copolimero è vicina all'unità e si formano le cosiddette micelle inverse, il fluido presenta di nuovo un comportamento Newtoniano.

Parte II

Capitolo 5

Condizioni operative investigate e Dettagli computazionali

In questo capitolo si riportano i dettagli computazionali, i parametri del modello utilizzati e le condizioni operative relative alla simulazione di un fluido DPD semplice, assimilabile all'acqua, e alla simulazione di un fluido complesso quale ad esempio la miscela composta da acqua e Pluronic.

La prima parte del lavoro di simulazione ha un duplice obiettivo. Innanzitutto, quello di mettere a punto una corretta procedura di calcolo della viscosità dinamica basata sulla formula di Green-Kubo [31, 32] sia individuando i valori dei parametri di correlazione e del *timestep* che permettano di calcolare correttamente la SACF, sia integrando in maniera adeguata la coda rumorosa della SACF. Infatti, entrambi i fattori incidono sul risultato finale di viscosità calcolato dalla simulazione. La convalida delle scelte effettuate è fatta sulla base del confronto con il calcolo realizzato con un altro metodo di equilibrio, la formulazione di Einstein del metodo di Green-Kubo [33] e con un metodo di non equilibrio, la tecnica RNEMD di Müller-Plathe [36]. Quest'ultimo è assunto come metodo di riferimento, o di *benchmark*, in quanto ritenuto più accurato e affidabile del metodo di Green-Kubo. Infatti, la correttezza dei parametri scelti è immediatamente verificabile dall'osservazione del profilo di velocità che si sviluppa lungo la *box* di simulazione. Essi devono essere tali da introdurre un flusso di quantità di moto nè troppo alto nè troppo basso in modo da avere un profilo di velocità lineare. Tuttavia, tale metodo ha il limite di essere valido solo per fluidi Newtoniani. Inoltre, essendo un metodo di non equilibrio è necessario introdurre una perturbazione esterna che potrebbe condizionare l'evoluzione delle microstrutture che caratterizzano il sistema costituito da acqua e Pluronic, influenzandone, dunque, anche le proprietà reologiche. È per tali motivi desiderabile avere una procedura ben definita di individuazione dei parametri di correlazione, di calcolo del-

la viscosità dinamica basata sulla formula di Green-Kubo e di valutazione della relativa incertezza. Il metodo basato sulla formula di Green-Kubo ha tutti i vantaggi di un metodo di equilibrio, quale ad esempio quello di non introdurre una perturbazione che potrebbe avere effetti sull'evoluzione delle microstrutture e sulle proprietà reologiche. Il secondo obiettivo è quello di rendere le simulazioni realistiche individuando i parametri DPD che consentano una corretta modellazione dell'acqua e una corretta predizione delle sue proprietà trasporto, con particolare attenzione ai coefficienti di viscosità e di autodiffusione.

La seconda parte del lavoro di simulazione ha come oggetto l'estensione dello studio parametrico, volto a rendere le simulazioni più realistiche, alla miscela costituita da acqua e Pluronic a varie concentrazioni.

Le simulazioni sono state condotte utilizzando il codice di dinamica molecolare *open source* LAMMPS. La procedura di implementazione del modello DPD si articola in diverse parti e due esempi di script di input completi, illustrati a grandi linee, sono riportati in Appendice A.

5.1 Parametri del modello DPD utilizzati

La modellazione DPD del fluido da simulare deve riprodurre al meglio le caratteristiche fisiche. Le simulazioni sono state condotte per due tipologie di fluidi: un fluido semplice assimilabile all'acqua e un fluido complesso, una miscela binaria composta da acqua e Pluronic L64.

Il fluido DPD semplice è costituito da un'unica specie di *beads* rappresentativi di un numero definito di molecole d'acqua a seconda del valore di N_m , o grado di *coarse-graining*, scelto, come schematizzato in Fig. 5.1. Quest'ul-

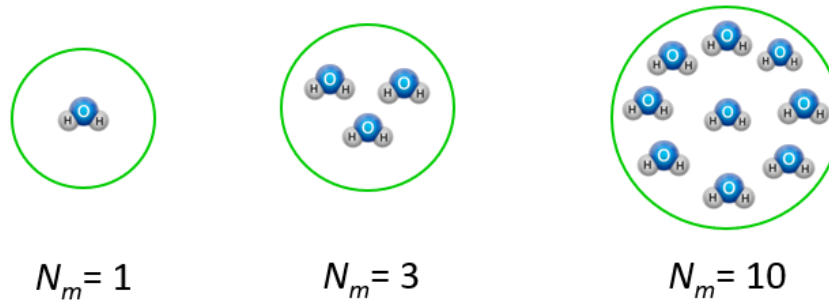


Figura 5.1: *Beads* rappresentativi della componente acquosa per diversi numeri di *coarse-graining*.

timo influenza il valore del coefficiente di repulsione della forza conservativa agente tra *beads* rappresentativi dell'acqua, valutato dall'Eq. (1.21), ricavata nella Sez. 1.3.2, imponendo l'uguaglianza tra la comprimibilità isoterma adimensionale del fluido DPD simulato e il valore di comprimibilità isoterma

proprio dell'acqua a temperatura ambiente ($T = 25^\circ\text{C}$), dunque $k^{-1} = 16$. I coefficienti di repulsione della forza conservativa riportati in Tab. 5.1 so-

Tabella 5.1: Gradi di *coarse-graining* (N_m) adottati per i *beads* rappresentativi delle molecole d'acqua e corrispondente coefficiente di repulsione (a).

N_m	a
1	25
3	76
10	265

no calcolati per $\rho = 3$, valore di densità numerica impostato in tutte le simulazioni. Una molecola di Pluronic L64 presenta la formula di struttura illustrata in Fig.5.2 Attraverso la procedura di *coarse-graining*, in accordo

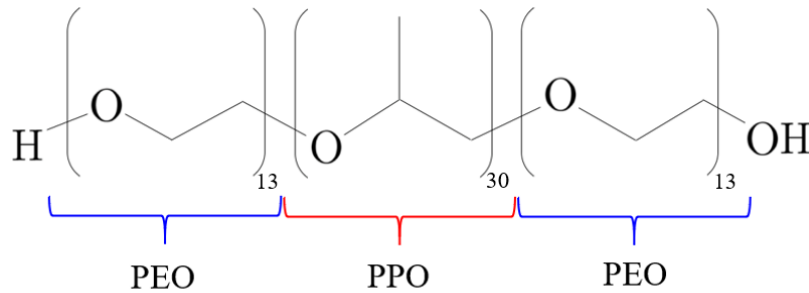


Figura 5.2: Formula di struttura della molecola di Pluronic L64 (elaborazione immagine estratta da Ref. [45]).

con l'approccio mesoscopico della modellazione DPD, diventa una molecola costituita da 15 *beads* come illustrato in Fig. 5.3 . Essa è schematicamente rappresentabile come segue: $A_3B_9A_3$, dove A e B indicano rispettivamente i *beads* di EO e PO. Questo vuol dire che un *bead* di EO è costituito da 4.3 unità ripetitive di etilene ossido e, allo stesso modo, un *bead* di PO contiene 3.3 unità ripetitive di propilene ossido. Le molecole d'acqua sono rappresentate da una terza tipologia di *bead* (C) corrispondente a dieci molecole di acqua. I fattori di rinormalizzazione sono scelti in modo tale che tutti i *beads* abbiano i medesimi massa e volume [55]. I coefficienti di repulsione tra *beads* di diverso tipo nel modello DPD sono stati valutati con l'Eq. (1.27), ottenuta dalla teoria di Flory-Huggins spiegata in Sez. 1.3.3. Sono riportati in Tab 5.3 e una schematizzazione delle interazioni conservative tra le diverse tipologie di *beads* è presentata in Fig. 5.4

I legami chimici tra i monomeri costituenti la molecola polimerica sono stati rappresentati con il modello *bead-and-spring* secondo il quale, i *beads* vicini costituenti la catena polimerica interagiscono mediante il potenziale

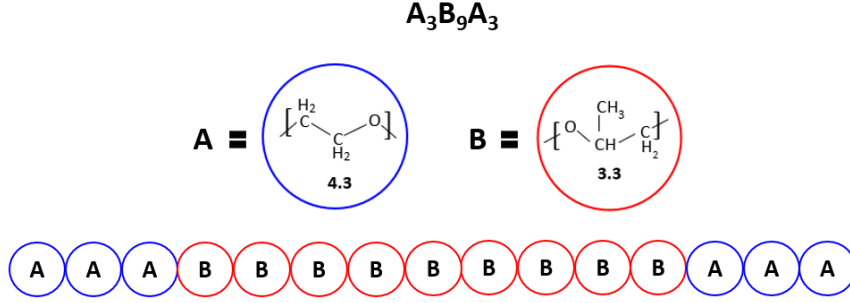


Figura 5.3: Modello DPD risultante dalla procedura di *coarse-graining* di una molecola di Pluronic L64.

Tabella 5.2: *Coarse-grained* (CG) *beads*, volumi molecolari di H₂O, EO e PO calcolati in accordo con l'approccio seguito da Durchschlag e Zipper [56] e volumi dei *beads* corrispondenti espressi in Å³.

CG <i>bead</i>	Volume molecola (Å ³)	Volume <i>bead</i> (Å ³)
[H ₂ O] ₁₀	30	300
[CH ₂ CH ₂ O] _{4.3}	64.6	278
[CH ₃ CHCH ₂ O] _{3.3}	96.5	318

armonico, descritto dall'equazione:

$$E_{harm} = K_b(r_{ij} - r_0)^2 . \quad (5.1)$$

La scelta della distanza nominale di equilibrio r_0 e della costante armonica K_b per ogni coppia di *bead* deve essere tale da non violare la topologia delle catene polimeriche, aspetto fondamentale nella modellazione delle proprietà reologiche. Sono stati testati attraverso varie simulazione differenti valori di tali parametri del potenziale armonico. I primi valori di tentativo sono stati ricavati mediante il calcolo basato sull'approccio adottato da Anderson e collaboratori [55]. Quest'ultimo prevede la valutazione delle distanze nominali per ogni coppia di *beads* mediante la seguente relazione:

$$r_0 = 0.1(n_i - n_j) - 0.01 , \quad (5.2)$$

in cui n_i e n_j sono gli atomi con peso molecolare maggiore inclusi nei i e j .

Dall'analisi qualitativa della forma e delle dimensioni delle microstrutture formatesi a diverse concentrazioni di Pluronic L64, questa scelta non si è rivelata adeguata ai fini della riproduzione della struttura interna del fluido. Nello specifico, non avviene la formazione delle micelle a concentrazioni superiori alla CMC come teoricamente atteso. Un secondo approccio seguito è

Tabella 5.3: Parametri di interazione a_{ij} , riportati in unità DPD, per le tre specie, PEO, PPO e acqua.

	A (PEO)	B (PPO)	C (H ₂ O)
A (PEO)	265.0	278.0	266.0
B (PPO)	278.0	265.0	274.0
C (H ₂ O)	266.0	274.0	265.0

Tabella 5.4: Parametri di solubilità χ_{ij} per le tre specie, PEO, PPO e acqua (valori estratti da Ref. [54]).

	PEO	PPO	H ₂ O
PEO	-	3.0	0.3
PPO	3.0	-	2.1
H ₂ O	0.3	2.1	-

stato quello di impostare la distanza nominale di equilibrio unitaria, in unità DPD, valore adottato in diversi studi presenti in letteratura, e studiare l'effetto di diversi valori della costante armonica K_b sull'evoluzione delle microstrutture formate dal fluido a varie concentrazioni. Valori eccessivamente alti di K_b riducono le distanze tra i *beads* interni alla catena polimerica e contribuiscono alla rigidità della catena. È fondamentale preservare le due caratteristiche fondamentali della modellazione DPD, che consentono simulazioni su scala mesoscopica: le interazioni repulsive tra i *beads* sono soffici e la catena polimerica a cui appartengono è flessibile. Sempre da un'analisi qualitativa si è rilevata ottimale la scelta di porre la costante di forza elastica pari al coefficiente di repulsione tra *beads* dello stesso tipo: è un *optimum* che, bilanciando l'effetto delle interazioni repulsive che tendono a separare i *beads* facenti parte della catena polimerica e l'intensità della forza di le-

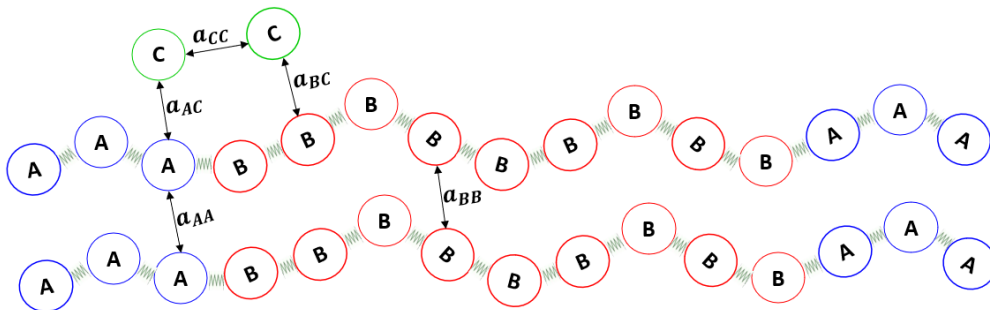


Figura 5.4: Interazioni conservative tra le diverse tipologie di *beads* costituenti il sistema: A(PEO), B(PPO) e C(H₂O).

Tabella 5.5: Parametri del potenziale armonico, espressi in unità DPD, calcolati secondo l'approccio di Anderson e collaboratori [55].

K_b	A(EO)	B(PO)
A(EO)	645.0	524.0
B(PO)	524.0	403.0

r_0	A(EO)	B(PO)
A(EO)	0.59	0.69
B(PO)	0.69	0.79

game, consente di mantenere la configurazione di catena polimerica senza renderla estremamente rigida. Per concentrazioni di Pluronic L64 tali per cui si ha la formazione di una *fase di gel*, è necessario introdurre il potenziale repulsivo segmentale (SRP) che impedisce l'incrocio non fisico dei legami dei *beads* di catena, altrimenti consentito dalle interazioni repulsive soffici del modello DPD. Le proprietà reologiche dipendono dagli *entanglements* fisici caratteristici del gel. Nel contesto della simulazione DPD bisogna isolare e considerare solo il contributo di questi ultimi per il calcolo della viscosità, per non avere incrementi non fisici di quest'ultima [57, 58, 59, 60, 61]. La

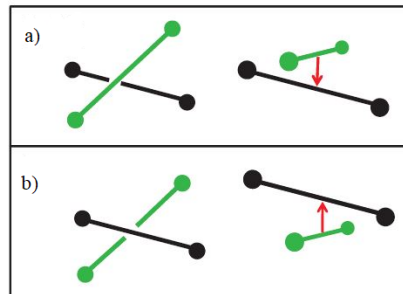


Figura 5.5: Esempio di violazione topologica per l'incrocio non fisico dei legami (linea verde e linea nera) tra i *beads* della catena polimerica rappresentati dai cerchi pieni, noto come *bond crossing* (elaborazione immagine estratta da Ref. [62]). Il vettore di distanza minima (in rosso) cambia il suo orientamento di 180° tra gli istanti di tempo t e $t + \Delta t$. La configurazione (a) è relativa all'istante di tempo t . La configurazione (b) è relativa all'istante di tempo $t + \Delta t$.

forza di interazione di tipo repulsivo tra legami non vicinali ha una forma

analoga alla forza conservativa tramite cui interagiscono i *beads*:

$$\mathbf{F}_{kl}^{SRP} = b \left(1 - \frac{d_{kl}}{d_c} \right) \hat{\mathbf{d}}_{kl}, \quad (5.3)$$

in cui \mathbf{F}_{kl}^{SRP} è la forza agente tra i legami k e l separati dalla distanza d_{kl} ; b e d_c sono rispettivamente la costante della forza e la distanza di *cutoff bond-bond*. La distanza tra i due legami può essere calcolata come distanza minima, *minimum bond-bond distance* o come distanza tra i punti medi dei legami, *midpoint-to-midpoint distance*. In quest'ultimo caso il potenziale repulsivo tra i legami viene designato con la sigla mSRP (modified segmental repulsive potential) in quanto deriva da un'ottimizzazione del precedente [62]. Infatti, un effetto non desiderato indotto dall'SRP è quello di incrementare l'interazione repulsiva tra i *beads* e, quindi, di introdurre un volume escluso intorno al legame. Questo compromette la corretta riproduzione della topologia del sistema e di conseguenza influisce negativamente sulla formazione delle microstrutture e sulle proprietà reologiche. Il modello mSRP minimizza questo effetto negativo, tuttavia provoca la formazione di angoli acuti tra *beads* consecutivi. Per ovviare a tale inconveniente è opportu-

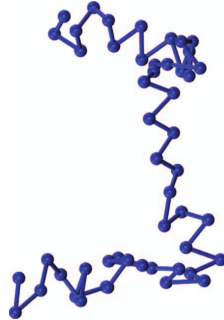


Figura 5.6: Esempio di una tipica conformazione spaziale assunta da una catena polimerica che si ottiene facendo interagire i legami tra i *beads* non vicini della macromolecola mediante mSRP, senza l'aggiunta di un potenziale repulsivo angolare (immagine adattata da Ref. [62]).

no aggiungere l'interazione tra legami consecutivi mediante un potenziale repulsivo angolare:

$$U_{bend} = K_{\theta}(1 + \cos \theta), \quad (5.4)$$

in cui K_{θ} è la costante di forza e θ è l'angolo formato da tre *beads* consecutivi di catena. Da uno studio, volto a minimizzare le violazioni topologiche, a riprodurre al meglio le proprietà strutturali e termodinamiche del fluido simulato, è emerso che il modello mSRP funziona bene con i seguenti parametri: $b = 100$, $d_c = 0.8$ e $k_{\theta} = 2.0$ [62].

5.2 Dettagli delle simulazioni: esplorazione dello spazio dei parametri

In tutte le simulazioni sono posti pari a 1, in unità DPD, la massa dei *beads*, il raggio di *cutoff*, r_c , che definisce la sfera d'azione della forza conservativa e l'energia del sistema, $k_B T$. La densità numerica del sistema, ρ , il numero di *beads* per unità di volume è pari a 3. Il valore è stato scelto come compromesso tra il minimo richiesto per soddisfare l' Eq. (1.19) di *scaling* ed esigenze di carattere computazionale. Infatti, poiché la quantità di interazioni per ogni particella del sistema aumenta linearmente con la densità, il tempo di CPU necessario per *timestep* e per unità di volume aumenta con una relazione quadratica rispetto alla densità. Per le simulazioni di un fluido costituito esclusivamente da acqua il dominio computazionale è stato definito come una *box* cubica tridimensionale con lato di lunghezza $L = 15$ nel caso di impiego di metodi di equilibrio e come un parallelepipedo di dimensione $15 \times 15 \times 45$ per la tecnica RNEMD di Müller-Plathe, per visualizzare il gradiente di velocità che si sviluppa nella direzione di trasferimento della quantità di moto. La durata delle simulazioni è di 5.000.000 *timesteps*, senza considerare quella necessaria ad equilibrare il sistema di 200.000 *timesteps*. Lo stile che descrive le interazioni di coppia tra i *beads* è specificato dal comando *pair_style dpdext*, che implementa il modello esteso descritto nella Sez. 2.2, includente, a differenza del classico stile *dpd*, la componente di taglio della forza di dissipazione e la corrispondente componente della forza stocastica. I coefficienti da definire in questo caso, tramite il comando *pair_coeff*, sono sette. I primi due si riferiscono alla tipologia di *beads* per i quali valgono i coefficienti di interazione riportati: in questo caso sono dello stesso tipo, cioè rappresentativi dell'acqua, indicati con 1.

Codice 5.1: dpdext pair style

```

1 pair_style dpdext ${T} ${rC} 34387
2 pair_coeff 1 1 ${gamma_par} ${gamma_perp} ${s_par} ${s_perp} ${rcD}

```

Codice 5.2: dpd pair style

```

1 pair_style dpdext ${T} ${rC} 34387
2 pair_coeff 1 1 ${gamma_par} ${s_par}

```

In tutte le simulazioni si pongono $\gamma_{\parallel} = \gamma_{\perp} \equiv \gamma$ e $s_{\parallel} = s_{\perp} \equiv s$. Il parametro s specificato è l'esponente che compare nella funzione peso della forza stocastica, legata a quella della forza dissipativa dall'Eq. (1.7). Lo stile *dpdext*, a differenza del classico *dpd*, permette di definire distintamente due raggi di *cutoff*: r_c definisce la sfera d'azione delle forze conservative e r_c^D quella delle forze dissipativa e stocastica. Questa possibilità è vantaggiosa dal punto di vista della riproduzione delle proprietà di trasporto, in quanto consente di modificare r_c^D molto influente sui valori di viscosità e diffusività, senza alterare lo stato fisico del sistema, dipendente invece da r_c . Nel caso di mancata

Tabella 5.6: Studio parametrico per valutare l'effetto del grado di *coarse-graining* sulla riproduzione delle proprietà di trasporto dell'acqua.

<i>timestep</i>	N_m	s	γ	r_c^D
0.01	1	0.125	4.5	1.0
0.01	1	0.25	4.5	1.0
0.01	1	0.5	4.5	1.0
0.01	1	1	4.5	1.0
0.01	3	0.125	4.5	1.0
0.01	3	0.25	4.5	1.0
0.01	3	0.5	4.5	1.0
0.01	3	1	4.5	1.0
0.01	10	0.125	4.5	1.0
0.01	10	0.25	4.5	1.0
0.01	10	0.5	4.5	1.0
0.01	10	1	4.5	1.0

definizione del secondo, che è opzionale, la simulazione assegna ad entrambi il valore di r_c che diventa il raggio di *cutoff* globale. La viscosità è stata calcolata con la procedura basata sulla formula di Green-Kubo, riportata nella Sez. 3.2.1 e il coefficiente di diffusione valutando il MSD, come descritto nella Sez. 3.3. I dettagli di simulazione relativi all'implementazione, alla scelta dei parametri e alle procedure di calcolo sui metodi implementati in LAMMPS per la valutazione dei coefficienti di trasporto sono forniti in Sez. 5.3. Sono state condotte una serie di simulazioni per valutare l'effetto del grado di *coarse-graining*, per modellare l'acqua raggruppandone le molecole in *beads* in accordo con l'approccio mesoscopico, sulla riproduzione delle sue proprietà di trasporto, ovvero la viscosità e il coefficiente di autodiffusione. Con lo stesso fine, una volta identificato $N_m = 10$ come valore ottimale e, dunque, $a = 265$, è stato esplorato lo spazio dei parametri s, γ e r_c^D . Il primo studio è stato condotto per tre valori di N_m e quattro terne di parametri s, γ e r_c^D , come illustrato in Tab. 5.6.

La seconda analisi parametrica è stata effettuata esplorando un certo numero di valori per ognuno dei tre parametri tenendo fissi gli altri due per valutare la sensibilità del numero di Sc rispetto ad ognuno di essi, come riassunto in Tab. 5.7.

Il *timestep* impiegato è pari a 0.01 in tutte le simulazioni relative alla variazione di γ e s . Per quelle relative alla variazione di r_c^D , invece, per alcuni valori è necessario un *timestep* inferiore, di 0.005, per evitare perdita di atomi dal dominio di simulazione. Le tre terne di parametri (s, γ, r_c^D) individuate come rappresentative dello stesso fluido, assimilabile all'acqua, sono state adottate per calcolare la viscosità dinamica del fluido con la formula di Einstein-Helfand, riportata in Sez. 3.2.2, e il metodo di Müller-Plathe,

Tabella 5.7: Studio parametrico per valutare l'effetto del grado di s, γ, r_c^D sulla riproduzione delle proprietà di trasporto dell'acqua.

<i>timestep</i>	N_m	s	γ	r_c^D
0.01	10	0.26	4.5	1.0
0.01	10	0.28	4.5	1.0
0.01	10	0.30	4.5	1.0
0.01	10	0.32	4.5	1.0
0.01	10	0.40	4.5	1.0
0.01	10	0.42	4.5	1.0
0.01	10	0.43	4.5	1.0
0.01	10	0.432	4.5	1.0
0.01	10	0.434	4.5	1.0
0.01	10	0.46	4.5	1.0
0.01	10	0.48	4.5	1.0
0.01	10	0.5	4.5	1.0
0.01	10	0.533	4.5	1.0
0.01	10	0.633	4.5	1.0
0.01	10	0.733	4.5	1.0
0.01	10	1	4.5	1.0
0.01	10	1	9	1.0
0.01	10	1	18	1.0
0.01	10	1	22	1.0
0.01	10	1	24	1.0
0.01	10	1	25	1.0
0.01	10	1	30	1.0
0.01	10	1	37	1.0
0.01	10	1	72	1.0
0.005	10	1	4.5	1.2
0.005	10	1	4.5	1.4
0.005	10	1	4.5	1.6
0.005	10	1	4.5	1.8

descritto in Sez. 3.2.3, per avere un'ulteriore conferma della correttezza della scelta dei parametri DPD e di simulazione. I dettagli di simulazione per la scelta dei parametri, l'implementazione della formulazione di Einstein del metodo di Green-Kubo e per l'imposizione del flusso di quantità di moto nella tecnica RNEMD di Müller-Plathe sono illustrati in Sez. 5.3. Per le simulazioni del fluido complesso costituito da acqua e Pluronic L64 condotte in condizioni di equilibrio è stato scelto un dominio computazionale più grande, un cubo con lato di lunghezza $L = 30$. Il sistema è stato simulato per varie concentrazioni di Pluronic L64.

5.3 Calcolo del numero di Schmidt in LAMMPS

5.3.1 Procedura per il calcolo della viscosità dinamica basata sulla formula di Green-Kubo

In una simulazione completamente equilibrata, in cui il flusso di quantità di moto attraverso la *box* di simulazione è allo stato stazionario, l'implementazione del metodo per il calcolo della viscosità basata sulla formula di Green-Kubo in LAMMPS [64] avviene attraverso i seguenti comandi:

Codice 5.3: Green-Kubo viscosity calculation

```

1 reset_timestep 0
2 variable pxy equal pxy
3 variable pxz equal pxz
4 variable pyz equal pyz
5 variable V equal vol
6 variable K equal 1/({kb}*T)*V*{Nev}*{dt}
7 fix SS all ave/correlate $Nev $Nrep $Nfreq &
8 v_pxy v_pxz v_pyz type auto
9 file time_cor.txt ave running
10
11 variable v11 equal trap(f_SS[3])*{K}
12 variable v22 equal trap(f_SS[4])*{K}
13 variable v33 equal trap(f_SS[5])*{K}
14
15 variable v equal (v_v11+v_v22+v_v33)/3.0
16 print "viscosity :_v_(DPD_units)"

```

Le variabili pxy, pxz e pyz sono le tre componenti fuori dalla diagonale del tensore degli sforzi di Irving-Kirkwood, riportato nell'Eq. (3.4). Le variabili v11, v22 e v33 rappresentano i coefficienti di viscosità calcolati considerando le rispettive componenti del tensore degli sforzi. Assumendo il sistema omogeneo e isotropo, il valore di viscosità finale è il risultato della media aritmetica delle tre componenti xy , xz e yz per avere una maggiore rappresentatività statistica [63]. Per ciascuna componente del tensore degli sforzi, la relativa viscosità dinamica è calcolata utilizzando la formula di Green-Kubo riportata nell'Eq. 3.3. Dunque, per ottenere il coefficiente di viscosità è necessario calcolare l'area sottesa dai profili di autocorrelazione (SACF). La funzione *trap()* effettua l'integrazione numerica utilizzando la regola dei trapezi

sul vettore globale f_SS , prodotto dall'operazione *ave/correlate*. LAMMPS, mediante quest'ultima, utilizza un algoritmo approssimativo per valutare la funzione di autocorrelazione delle tensioni e non la sua definizione che comporterebbe un elevato costo computazionale. Innanzitutto, utilizza l'ipotesi ergodica per valutare la media dell'insieme come integrale del tempo, e per questo motivo il tempo di simulazione finale deve essere sufficientemente ampio per effettuare una stima corretta. Inoltre, utilizza solo alcuni dati nella valutazione della funzione di autocorrelazione. Le variabili da definire nel codice per il calcolo della funzione di autocorrelazione sono, dunque, Nev , $Nrep$ e $Nfreq$. I valori di input vengono campionati ogni Nev *timesteps* da un tempo iniziale pari all'ultimo *output time* investigato ad un tempo finale corrispondente a $Nfreq$ passi di integrazione temporale successivi all'istante di tempo di partenza. Per la serie di campioni, il valore di correlazione $C_{\alpha\beta}$, è calcolato come:

$$C_{\alpha\beta}(t) = \left\langle \sum_{\alpha \neq \beta} P_{\alpha\beta}(t) P_{\alpha\beta}(t + \delta) \right\rangle. \quad (5.5)$$

La media in Eq. (5.5) è calcolata su ogni coppia di valori di input separati da un passo temporale δ . Il massimo valore di δ utilizzato equivale all'istante di tempo pari a $((Nrep - 1) \times Nev)$ passi di integrazione temporale. Pertanto, la quantità di dati di correlazione calcolati risulta pari a $Nrep$:

$$C_{\alpha\beta}(0), C_{\alpha\beta}(Nev), C_{\alpha\beta}(2 \times Nev), \dots, C_{\alpha\beta}((Nrep - 1) \times Nev).$$

Le medie di autocorrelazione vengono calcolate ogni $Nfreq$ *timesteps* [65]. Trattandosi di una procedura approssimata, la scelta di questi tre parametri di correlazione è fondamentale nella valutazione della SACF e, in generale, nel calcolo della viscosità. La correttezza e l'attendibilità del valore di viscosità risultante dipendono dal *timestep* impiegato, dal tipo di integratore, dalla giusta scelta dei parametri citati per il calcolo della SACF e dal metodo di integrazione numerica di quest'ultima. In questo lavoro di tesi è stato condotto uno studio per valutare l'effetto combinato dei parametri di correlazione e del *timestep* sull'andamento della SACF. Il valore di $Nrep$ deve essere tale da garantire che le fluttuazioni della SACF diventino in media zero. L'elevato coefficiente di diffusione nelle simulazioni DPD richiede che il campionamento dei valori di input per il calcolo della SACF sia frequente, quindi una scelta conservativa è porre Nev pari a uno. Per quanto riguarda l'influenza del *timestep* sulla SACF, per passi temporali più piccoli, una migliore risoluzione temporale, porta a fluttuazioni maggiori, invece, passi temporali più grandi portano ad uno smorzamento delle fluttuazioni. Il *timestep* deve però essere sufficientemente piccolo da mantenere la temperatura di equilibrio all'interno del sistema nel corso della simulazione, quindi il valore scelto risulta da un compromesso. In particolare, la deviazione standard dalla temperatura media impostata all'inizio deve essere inferiore al

3%. Questo studio preliminare finalizzato ad individuare la giusta combinazione di parametri, *timestep*, *Nev*, *Nrep*, *Nfreq*, è stato condotto attraverso una serie di simulazioni effettuate per un semplice fluido DPD con i parametri standard del modello, ovvero $a = 25$, valore derivante dall'aver scelto $N_m = 1$, $\gamma = 4.5$, $s = 1$. Le interazioni tra i *beads*, costituiti da una sola molecola d'acqua, in quest'analisi sono quelle proprie della versione originale del modello DPD, ossia determinate dalle sole componenti parallele delle forze conservativa, dissipativa e stocastica, e implementate in LAMMPS tramite il comando *pair_style dpd*. L'esplorazione dello spazio dei parametri

Tabella 5.8: Studio parametrico per il calcolo della SACF in LAMMPS.

<i>timestep</i>	<i>Nev</i>	<i>Nrep</i>	<i>Nfreq</i>
0.001	1	20	1000
0.002	1	20	1000
0.003	1	20	1000
0.004	1	20	1000
0.005	1	20	1000
0.01	1	20	1000
0.02	1	20	1000
0.03	1	20	1000
0.04	1	20	1000
0.04	1	10	1000
0.04	1	20	1000
0.04	1	30	1000
0.04	1	40	1000
0.04	1	50	1000
0.04	1	60	1000
0.04	1	100	1000
0.04	1	200	1000
0.04	1	300	1000
0.04	1	400	1000
0.04	1	500	1000
0.04	1	600	1000
0.04	1	700	1000
0.04	1	800	1000
0.04	1	20	200
0.04	1	20	400
0.04	1	20	800
0.04	1	20	1000
0.01	1	1000	1000

di correlazione e del *timestep* ha condotto alla scelta della combinazione ot-

timale $N_{ev} = 1$, $N_{rep} = 1000$, $N_{freq} = 1000$ e $dt = 0.01$. È emerso, inoltre, che la scelta del *timestep* e quella di N_{rep} sono legate. La relazione ottimale individuata tra i due parametri deve essere preservata anche per valori diversi, ai fini di ottenere gli stessi risultati di viscosità. Se dovesse rendersi necessario l'impiego di un passo temporale minore, come nel caso di valori di raggio di *cutoff* maggiori di uno, il valore di N_{rep} deve essere opportunamente aumentato. Di conseguenza anche il parametro N_{freq} deve essere variato per rispettare la relazione $N_{freq} = N_{ev} \times N_{rep}$. A titolo d'esempio, in Tab. 5.9 si riportano le due combinazioni di parametri impiegate in questo lavoro. Diversi studi presenti in letteratura hanno dimostrato che la SACF

Tabella 5.9: Relazione tra parametri di correlazione e *timestep*.

<i>timestep</i>	N_{ev}	N_{rep}	N_{freq}
0.001	1	1000	1000
0.005	1	2000	2000

calcolata nelle simulazioni DPD presenta fluttuazioni molto rapide e, anche nel limite di tempi lunghi, non si stabilizza necessariamente a zero. Per tale motivo l'integrazione numerica del profilo di autocorrelazione effettuata in LAMMPS con la regola dei trapezi, restituisce risultati che oscillano intorno ad un valore medio. Per ridurre l'incertezza che ne consegue, nel calcolo della viscosità con la formula di Green-Kubo, sono state seguite diverse procedure. La prima (Procedura A) si basa sull'integrazione numerica del profilo di autocorrelazione tal quale fino al punto in cui la SACF raggiunge l'1% del valore iniziale. La restante parte della funzione di autocorrelazione, caratterizzata da oscillazioni di ampiezza variabile, viene approssimata con la legge di potenza, $c(t) = A \times t^{-B}$. Sono state testate altre funzioni di interpolazione come un esponenziale decrescente, ma la legge di potenza risulta la migliore. Lo stesso approccio è stato adottato (Procedura B) applicando prima un filtro, *moving average filter* per ridurre il rumore della SACF. La risultante funzione di potenza viene integrata analiticamente fino all'infinito o fino al tempo finale della SACF. La terza procedura (Procedura C) si basa sul calcolo dell'integrale cumulativo della viscosità dall'Eq. (3.3). La viscosità risultante è la media dei valori cumulativi su un certo intervallo di tempo. In particolare, in questo lavoro è stata calcolata tra $N_{rep}/2$ *timesteps* e N_{rep} *timesteps*, per escludere il periodo iniziale in cui cresce rapidamente.

5.3.2 Il metodo di Einstein-Helfand come riferimento

Codice 5.4: Einstein-Helfand viscosity calculation

```

1 reset_timestep 0
2 variable pxy equal pxy
3 variable pxx equal pxx-press

```

```

4 variable pxz equal pxz
5
6 fix avstress all ave/time ${Nev} ${Nrep} ${Nfreq} v_pxy v_pxx v_pxz
7     ave one & file profile.einstein.3d
8
9 #Diagonal components of SS are larger by factor 4/3
10 variable scale equal vol/(2.0*$T*dt*${Nfreq})
11 variable diagfac equal 4/3
12
13 variable deltasqxy equal (f_avstress[1]*${Nfreq}*dt)^2
14 variable deltasqxx equal (f_avstress[2]*${Nfreq}*dt)^2/${diagfac}
15 variable deltasqxz equal (f_avstress[3]*${Nfreq}*dt)^2
16
17
18 fix avdeltasq all ave/time ${Nfreq} 1 ${Nfreq} v_deltasqxy
19     v_deltasqxx v_deltasqxz ave running
20
21 variable vxy equal f_avdeltasq[1]*${scale}
22 variable vxx equal f_avdeltasq[2]*${scale}
23 variable vxz equal f_avdeltasq[3]*${scale}
24 variable etaxy equal v_vxy
25 variable etaxx equal v_vxx
26 variable etaxz equal v_vxz
27 variable eta equal (${etaxy}+${etaxx}+${etaxz})/3
28
29 print "running_average_viscosity : _${eta}"

```

Il confronto tra il valore di viscosità ottenuto con il metodo basato sulla formula di Green-Kubo, esposto nella Sez. 5.3.1, e quello ottenuto con la formulazione di Einstein permette la valutazione dell'attendibilità dei risultati. Il calcolo del momento di Helfand, espresso in funzione del tensore degli sforzi, secondo l'Eq. (3.6), è implementato in LAMMPS attraverso il comando *fix avstress*. Gli argomenti da specificare sono *Nev*, *Nrep* e *Nfreq* che definiscono in termini di numero di *timesteps* rispettivamente la frequenza temporale di campionamento dei valori di input, il numero complessivo di dati campionati e il periodo su cui viene calcolata la media. I valori scelti per i tre parametri sono i medesimi di quelli impostati per il calcolo della SACF nel metodo di Green-Kubo. L'opzione *ave one* indica che le medie calcolate ogni *Nfreq* passi temporali sono indipendenti tra di loro, non sono cumulative: vengono emesse così come sono senza un'ulteriore operazione di media. Si calcola a questo punto il MSD per le tre direzioni spaziali, *deltasqxy*, *deltasqxx* e *deltasqxz*. Attraverso il comando *fix avdeltasq*, le tre componenti sono mediate nel tempo e, successivamente, convertite nei tre contributi di viscosità applicando la relazione riportata in Eq.3.5. La viscosità finale è data dalla media delle tre componenti. Si nota che in questo metodo di calcolo si utilizzano tre componenti diverse del tensore degli sforzi (*pxy*, *pxx*—*press*, *pxz*) rispetto al caso precedente (*pxy*, *pxz*, *pyz*). Questo è possibile sulla base della proprietà di simmetria del tensore degli sforzi viscosi ed è vantaggioso in quanto permette di aumentare l'affidabilità statistica dei risultati. In questo caso la componente del tensore degli sforzi appartenente alla diagonale deve essere corretta di un fattore 4/3 [66].

5.3.3 Il metodo di Müller-Plathe come riferimento

Codice 5.5: Muller-Plathe viscosity calculation

```

1 fix          MP all viscosity ${Nvisc} vdim pdim ${Nbin} swap Nswap
2             vtarget V
3 variable flux equal f_MP/(2.0*((step-${nrelax})*${dt})*(lx+lz)+1.0e-10)
4 compute     layers all chunk/atom bin/ld y center ${binW} units box
5 fix        velocity all ave/chunk $s $p $d layers vx file
6           VelocityProfile.dat
7 fix        1 all nve
8 fix        2 all temp/berendsen

```

Il comando *fix* MP richiama l'algoritmo di Müller-Plathe [67], il cui funzionamento è stato illustrato nella Sez. 3.2.3, per eseguire con una certa frequenza scambi di quantità di moto tra due *beads* siti in due diverse regioni della *box* di simulazione. Le variabili da definire sono, in questo caso, Nbin, Nvisc e Nswap. Inoltre, bisogna specificare vdim e pdim, che indicano rispettivamente la componente della quantità di moto da scambiare e la direzione del trasferimento della quantità di moto. Il valore di Nbin, che deve essere un numero pari, rappresenta il numero di *slabs*, qui chiamati layers, in cui viene divisa la *box* di simulazione nella direzione pdim. Lo scambio di quantità di moto avviene tra un numero di coppie di *beads* definito dal parametro Nswap, con una frequenza indicata dal valore di Nvisc, che indica quanti *timesteps* intercorrono tra due scambi successivi. Quindi, all'interno di ciascuna coppia, costituita da un *bead* appartenente al *layer* 1 avente velocità positiva prossima al valore target V e un *bead* del *layer* intermedio, definito come lo strato $Nbin/2 + 1$, avente velocità negativa prossima al valore target V, avviene lo scambio della componente vdim di quantità di moto. A seguito di quest'ultimo i *beads* ripristinano la loro velocità tipicamente in direzioni opposte. Nel tempo questo induce un profilo di velocità di taglio nel sistema che può essere calcolato con i comandi *compute layers* e *fix velocity* che riportano il profilo di velocità nel file VelocityProfile.dat. Si dovrebbe sviluppare un profilo di velocità simmetrico perché i due layers sono separati dalla stessa distanza in entrambe le direzioni in senso periodico. Al fine di ottenere un profilo di velocità lineare, è necessaria una scelta appropriata di Nswap e Nvisc, parametri che consentono di regolare l'intensità del flusso di quantità di moto imposto. Uno scambio eccessivo di quantità di moto induce un profilo di velocità non lineare. Il rapporto tra il flusso di quantità di moto scambiata totale, rappresentato dalla variabile flux, e la pendenza del profilo di velocità di taglio è proporzionale alla viscosità del fluido. Il flusso di quantità di moto effettivo è la metà rispetto a quello calcolato da LAMMPS. Infatti, essendo il sistema simulato con condizioni al contorno periodiche, il flusso va in due direzioni. Per simulazioni di non equilibrio che utilizzano l'algoritmo di Müller-Plathe per il calcolo della viscosità, nel caso di incremento delle interazioni dissipative, si ha una deviazione della temperatura dal valore di equilibrio desiderato e risulta necessario aggiungere un

termostato esterno che ripristini l'isoterma del sistema simulato. Il termostato *temp/berendsen* modifica le velocità per effettuare la termostatazione. Non effettua l'integrazione temporale, pertanto è necessario accoppiarlo all'integratore NVE per aggiornare effettivamente le posizioni delle particelle utilizzando le velocità modificate. I parametri *Tstart* e *Tstop* rappresentano rispettivamente le temperature desiderate all'inizio e alla fine della simulazione. Il parametro *Tdamp* rappresenta il periodo espresso in termini di numero di *timesteps* con cui la temperatura viene riportata al valore di equilibrio. Tale metodo di calcolo richiede valori di *timesteps* ridotti. I parametri usati nelle simulazioni sono $N_{\text{bin}} = 30$, $N_{\text{visc}} = 25$ e $dt = 0.001$ o $dt = 0.0001$ per valori di r_c^D rispettivamente uguali o maggiore di uno.

5.3.4 Calcolo del coefficiente di diffusione

Per il calcolo del coefficiente di diffusione è stata implementata la relazione di Einstein, riportata nell'Eq. 3.21, più semplice dal punto di vista del calcolo computazionale rispetto al metodo di Green-Kubo. Entrambi i metodi sono spiegati in Sez. 3.3.

Codice 5.6: Diffusivity calculation

```

1 compute      msd all msd com yes
2 fix          9 all vector  $\${N_{\text{ev}}}$  c_msd[4]
3 variable     D equal slope(f_9)/6/dt

```

Il comando *compute msd* calcola lo spostamento quadratico medio delle particelle, tenendo conto di tutti gli effetti dovuti alle condizioni al contorno periodiche [68]. Viene generato un vettore di quattro unità, in cui i primi tre elementi sono gli spostamenti quadratici medi rispettivamente nelle tre direzioni spaziali x , y , z sommati e mediati su tutti i *beads*. La quarta componente è lo spostamento quadratico medio totale. La posizione di riferimento del *bead* da cui viene calcolato lo spostamento è quella occupata da essa nel momento in cui viene emesso il comando. L'opzione *com* impostata su *yes* assicura che l'effetto di qualsiasi deriva del centro di massa del *bead* venga sottratto prima di calcolare lo spostamento di ciascuna di essi. I valori istantanei dell'MSD vengono salvati in un vettore attraverso il comando *fix vector* ogni N_{ev} *timesteps*. Tali valori sono interpolati da una retta la cui pendenza è calcolata dalla funzione *slope*, da cui si estrae il coefficiente di diffusione ad essa proporzionale mediante l'Eq. (3.21).

5.4 Algoritmo di clustering

L'analisi quantitativa dei risultati di simulazione della miscela costituita da Pluronic L64 e acqua è stata effettuata mediante un algoritmo di *clustering*, che consente di identificare la morfologia delle microstrutture nonché di calcolarne le dimensioni. È scritto con il linguaggio di programmazione Python

(Python Software Foundation. Disponibile al sito <http://www.python.org>) e in Appendice B vengono riportati per esteso i codici da cui è composto: uno script (*test.py*) contenente tutte le operazioni che richiama delle funzioni, definite in un secondo script (*fnc.py*). L'algoritmo si basa sul metodo di *clustering* noto come DBSCAN [69] (*Density Based Spatial Clustering of Applications with Noise*), che identifica come *clusters* le regioni a densità di punti superiore rispetto alla regione esterna. Per la definizione di *cluster* DBSCAN usa la nozione di *density-connectivity* e necessita di soli due parametri di input: ϵ e *MinPts*. L'algoritmo identifica un punto appartenente ad un *cluster* se il suo ϵ -vicinato contiene almeno un numero minimo di punti (*MinPts*). Un punto (o una particella) che si trova al centro di una sfera di raggio ϵ al cui interno vi è un numero di punti maggiore o uguale alla soglia richiesta è definito *punto centrale (core)*. Un *cluster* C è definito da un gruppo di punti *density-connected*. Un punto P all'interno di un *cluster* C si dice *direttamente raggiungibile (directly density-reachable)* da un punto Q contenuto anch'esso in C se si trova all'interno dell' ϵ -vicinato di Q e se allo stesso tempo tale intorno di Q contiene un numero di punti almeno pari alla soglia minima (*MinPts*). Il codice utilizza il file di traiettoria prodotto da LAMMPS (file di output *.xyz*) memorizzando le coordinate e il rispettivo *timestep* in un oggetto denominato *Times* in cui vi saranno per ciascun *timestep* le coordinate di ogni *bead*. È necessario specificare l'intervallo di tempo in cui eseguire l'algoritmo di *clustering* ed è possibile impostare un valore di *frameskip*, in modo tale da utilizzare effettivamente solo alcuni dei *timesteps* memorizzati per il calcolo. Bisogna specificare le condizioni al contorno assegnando lo stato 1 al flag *pb*: l'algoritmo di *clustering* utilizza le distanze relative tra le particelle, calcolate specificando in *fnc.py* un'opportuna matrice delle distanze (*distance_matrix*). Per ogni *timestep* viene registrato il numero di *cluster* e di *noise* identificati. Ogni *cluster* è rappresentato con un colore diverso. La scelta dei parametri ϵ e *MinPts* è stata ottimizzata empiricamente, per minimizzare il numero di *noise*. L'algoritmo genera una serie di grafici che consentono di valutare l'evoluzione delle microstrutture nel tempo di simulazione, fino al raggiungimento della condizione di equilibrio termodinamico. È interessante, da questo punto di vista, il grafico riportante il numero totale dei *clusters* individuati rispetto al tempo di simulazione. Altrettanto significativa è la distribuzione dimensionale dei *clusters*, nota come *cluster mass distribution*, che fornisce informazioni morfologiche importanti. Infatti, a partire da essa si possono ricavare sia il valore della CMC che i parametri di forma delle micelle formatesi.

Capitolo 6

Presentazione e discussione dei risultati

6.1 Calcolo delle proprietà di trasporto di un fluido semplice assimilabile all'acqua

Uno degli aspetti cruciali durante il calcolo della viscosità, basato sulla formula di Green-Kubo, è l'integrazione temporale della SACF. La questione è strettamente legata all'andamento di quest'ultima. Come si evince da diversi lavori presenti in letteratura [70] e come si può notare in Fig. 6.1, mentre la prima parte della curva ha un andamento chiaro, nettamente decrescente, la parte finale è costituita da valori, anche negativi, che oscillano intorno allo zero con ampiezza variabile. L'integrale calcolato in LAMMPS con la semplice regola dei trapezi risente di questo rumore e influenza il valore finale di viscosità predetto, che non risulta essere descrittivo unicamente delle proprietà reologiche del sistema fisico simulato ma è affetto da un'incertezza legata al rumore della coda della SACF. Al fine di quantificarla e minimizzarla, in questo lavoro di tesi è stato messo a punto un metodo di calcolo adeguato della viscosità a partire dalla SACF generata in LAMMPS. Un primo approccio (Procedura A) è stato quello di seguire quanto proposto da Jung e Schmidt [71] che prevede l'impiego della regola dei trapezi per l'integrazione numerica della funzione di autocorrelazione tal quale dal tempo iniziale fino al tempo corrispondente al raggiungimento dell'1% del valore iniziale. La restante parte della curva, quella rumorosa, viene approssimata con la legge di potenza $c(t) = At^{-B}$ e integrata analiticamente.

Un altro metodo investigato (Procedura B) è stato quello di seguire una procedura analoga alla precedente, ma con l'applicazione preliminare di un filtro a media mobile (*moving average filter*) alla SACF per attenuare le oscillazioni. I filtri sono, infatti, funzioni che sostituiscono una serie temporale con un'altra. Mediante un'appropriata selezione del filtro si possono rimuovere o sostituire alcune componenti della serie originale in modo da ridurre

le oscillazioni. Il filtro a media mobile, qui utilizzato, è un filtro lineare e richiede la definizione di un intervallo su cui effettuare l'operazione di media (*window length*). A valle di una serie di test preliminari, è risultata ottimale la scelta di un intervallo pari a 20 *timesteps*. Anche in questo caso il contributo all'integrale relativo alla prima parte della SACF, fino all'1% del valore iniziale, è calcolato numericamente con la regola dei trapezi; invece, quello relativo alla restante parte, interpolata con la legge di potenza, è calcolato analiticamente. Il *fitting* con la legge di potenza della coda rumorosa della SACF, tal quale o dopo l'applicazione del filtro a media mobile, è illustrato in Fig. 6.1.

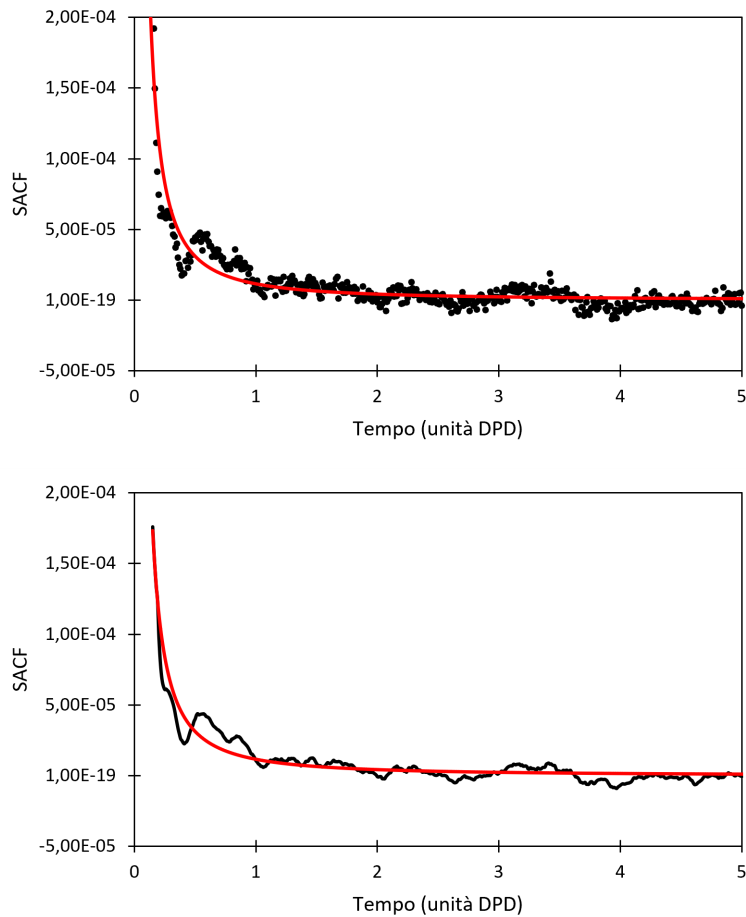


Figura 6.1: In alto: Procedura A. Cerchi neri: punti della SACF. Linea rossa: legge di potenza interpolante. In basso: Procedura B. Linea nera: SACF dopo il filtro a media mobile. Linea rossa: legge di potenza interpolante.

Le due procedure esposte hanno un problema comune: la legge di potenza

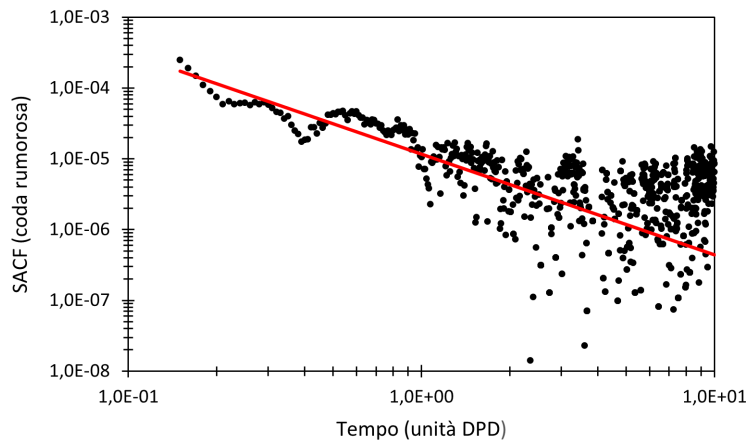


Figura 6.2: Cerchi neri: punti della coda rumorosa della SACF tal quale. Linea rossa: legge di potenza interpolante la coda rumorosa della SACF dopo l'applicazione del filtro a media mobile.

tende asintoticamente allo zero in maniera più lenta rispetto ai valori effettivi della SACF. Quest'ultima inizia ad oscillare tra valori positivi e negativi quando la legge di potenza ancora non raggiunge il valore asintotico. Da questa differenza di comportamento tra le due curve nasce l'incertezza sulla scelta dell'estremo superiore di integrazione della funzione interpolante. Sia nella Procedura A che nella Procedura B sono state seguite due linee: un primo approccio è stato quello di valutare l'integrale analitico tra il tempo per il quale il valore della SACF è pari all'1% di quello iniziale e il tempo finale della SACF disponibile dalla simulazione, dipendente dal parametro N_{rep} e dal *timestep* (designato in Tab. 6.1 con l'apice 1); un secondo approccio è stato quello di calcolare l'integrale considerando come estremo superiore infinito (designato in Tab. 6.1 con l'apice 2). Tale scelta non risulta irrilevante ai fini del risultato; al contrario, influenza in maniera significativa il valore finale di viscosità predetto. Inoltre, l'incertezza del valore di viscosità legato al rumore della coda non è ancora quantificato impiegando queste procedure. Per ovviare tali problemi, è stato seguito un terzo approccio basato sul calcolo dell'integrale cumulativo di viscosità in funzione del tempo (Procedura C). L'idea su cui si fonda questo metodo è quella di valutare i contributi cumulativi, progressivi nel tempo, dell'integrale della SACF alla viscosità ed effettuare un'operazione di media sui valori oscillanti intorno al valore asintotico teorico. Infatti, usando l'Eq. (3.3) ed estendendo progressivamente l'intervallo di integrazione fino al tempo finale della SACF il risultato atteso, dal punto di vista teorico, è una curva monotona crescente che tende asintoticamente al valore della viscosità del sistema. Dalle simulazioni DPD, per via del rumore stocastico intrinseco, si ottiene tipicamente l'andamento illustra-

to in Fig. 6.3. Il valore di viscosità cresce, come atteso, in maniera monotona fino ad un certo punto, approssimativamente corrispondente a tre unità di tempo DPD; poi, però, inizia ad oscillare intorno ad un valore medio. Tale andamento deriva dal corrispondente rumore della SACF. È stato, dunque, deciso di valutare la viscosità come il valore medio della curva considerando solo le ultime cinque unità di tempo DPD. Il vantaggio di questo approccio è rappresentato dalla possibilità di poter quantificare l'incertezza del valore di viscosità associata alle oscillazioni calcolando la deviazione standard dal valore medio.

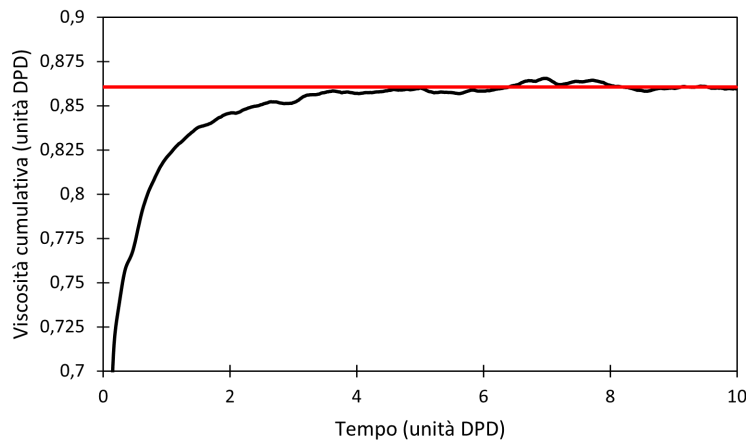


Figura 6.3: Procedura C. Integrale cumulativo di viscosità calcolata con la formula di Green-Kubo. Linea nera: integrale cumulativo di viscosità. Linea rossa: valore medio calcolato tra $t = 5$ e $t = 10$.

In Tabella 6.1 sono riportati i valori di viscosità ottenuti utilizzando la formula di Green-Kubo (G-K) secondo le diverse procedure esposte e in Tabella 6.2 quelli calcolati con i metodi di Müller-Plathe (M-P) e Einstein-Helfand (E-H). Questi ultimi sono presi come riferimento. Le simulazioni sono state condotte implementando il modello DPD standard, non includente le forze di taglio tra *beads*, quindi $\gamma_{\perp} = 0$, $s = 1$, $r_c^D = 1$ e $N_m = 1$. I risultati presenti in Tabella 6.1 fanno riferimento a differenti valori di γ_{\parallel} . Si può notare l'effetto della variazione di un singolo parametro del modello DPD sulla viscosità del sistema simulato. Come si evince dal confronto tra le Tabelle 6.1 e 6.2, eseguendo il calcolo della viscosità in accordo con le procedure basate sul *fitting* della coda rumorosa della SACF con la legge di potenza si ottengono risultati che si scostano molto dai valori di riferimento. In particolare la Procedura B comporta differenze significative. Invece, la Procedura C fornisce risultati molto prossimi ai valori obiettivo calcolati con M-P e E-H. Inoltre, in Tabella 6.1 si riportano la deviazione standard (std) e il coefficiente di variazione (COV) che quantificano l'effetto delle oscillazioni

Tabella 6.1: Valori di viscosità (in unità DPD) calcolati con il metodo di equilibrio di Green-Kubo seguendo diverse procedure. A^1 : interpolazione con la legge di potenza della coda rumorosa della SACF e calcolo dell'integrale analitico fino al tempo ultimo di simulazione disponibile. A^2 : interpolazione con la legge di potenza della coda rumorosa della SACF e calcolo dell'integrale analitico fino a ∞ . B^1 : interpolazione con la legge di potenza della coda rumorosa della SACF dopo l'applicazione del filtro a media mobile e calcolo dell'integrale analitico fino al tempo ultimo di simulazione disponibile. B^2 : interpolazione con la legge di potenza della coda rumorosa della SACF dopo l'applicazione del filtro a media mobile e calcolo dell'integrale analitico fino a ∞ . $C(media)$: valore medio della curva cumulativa di viscosità calcolato considerando solo le ultime cinque unità di tempo DPD. $C(std)$: deviazione standard. $C(COV)$: coefficiente di variazione.

γ_{\parallel}	A^1	A^2	B^1	B^2	$C(media)$	$C(std)$	$C(COV)$
4.5	0.8535	0.8846	0.8706	0.9399	0.860	0.002	0.002
9	0.9699	1.0018	0.9892	1.0613	0.957	0.004	0.004
15	1.1128	1.1468	1.1308	1.2065	1.065	0.002	0.004

Tabella 6.2: Valori di viscosità (in unità DPD) calcolati con i metodi di riferimento di equilibrio e non equilibrio, rispettivamente Einstein-Helfand e Müller-Plathe.

γ_{\parallel}	$E - H$	$M - P$
4.5	0.847	0.860
9	0.923	0.930
15	1.071	1.055

intorno al valore medio di viscosità (media) derivanti dalla natura stocastica della tecnica DPD.

Trovata una procedura affidabile per il calcolo della viscosità basata sulla formula di Green-Kubo, è stato studiato l'effetto di alcuni parametri del modello DPD sull'andamento della SACF, sulle proprietà di trasporto, viscosità e coefficiente di diffusione, e sul numero di Schmidt di un fluido semplice DPD rappresentante l'acqua a $25^{\circ}C$. I parametri da investigare sono il numero di *coarse-graining* N_m , gli esponenti della funzione peso generalizzata, s_{\parallel} e s_{\perp} e i coefficienti dissipativi, γ_{\parallel} e γ_{\perp} . In questo lavoro per tutte le simulazioni sono stati assunti $\gamma_{\parallel} = \gamma_{\perp} = \gamma$ e $s_{\parallel} = s_{\perp} = s$. La semplice estensione del modello DPD standard con l'introduzione della *shear dissipation*, non accompagnata da un'opportuna scelta dei parametri sopracitati, non comporta un significativo incremento del numero di Schmidt, come si nota nella prima riga della tabella 6.3. Quindi, è stato innanzitutto stu-

diato il legame tra le proprietà di trasporto del fluido simulato e il numero di *coarse-graining* utilizzato per la modellazione del sistema in accordo con l'approccio mesoscopico.

Da questa analisi è emerso che il numero di Schmidt non aumenta in maniera significativa riducendo l'esponente della funzione peso generalizzata, a meno che non si scelga un valore di *coarse-graining* sufficientemente alto. La ragione di ciò risiede nel comportamento della SACF per i vari valori di N_m osservabile in Fig. 6.4. Il valore di viscosità calcolato con la formula di Green-Kubo è correlato, in accordo con l'Eq. (3.3) all'area sottesa al profilo di autocorrelazione, la quale aumenta all'aumentare di N_m . In altre parole, la SACF tende allo zero più lentamente all'aumentare del numero di molecole contenute all'interno di un *bead*. Una considerazione immediata che deriva, inoltre, dall'osservazione dell'andamento della SACF per i tre valori di N_m riguarda l'impatto del coefficiente della forza conservativa sulla viscosità risultante. Esso, infatti, dipende dal numero di *coarse-graining* secondo l'Eq. (1.21). La scelta di $N_m = 10$ ben si accorda con questioni inerenti la struttura locale del fluido DPD simulato. Quest'ultima è caratterizzabile mediante la funzione di distribuzione radiale, la cui forma dipende dal numero di *coarse-graining* [8]. Nel caso di $N_m = 1$ ($a = 25$), valore tipicamente adottato in diversi lavori di letteratura [2], la funzione di distribuzione radiale è circa costante e unitaria indicando che lo stato del sistema è assimilabile alla fase gassosa e non a quella liquida. Viceversa, per $N_m > 10$ ($a > 265$) la funzione di distribuzione radiale presenta picchi secondari indicativi della presenza di una struttura parzialmente cristallina, dunque non liquida. Per $N_m = 10$ ($a = 265$) la funzione di distribuzione radiale è indicativa dello stato liquido [8]. Dunque, la scelta di impiegare un numero di *coarse-graining* pari a dieci non altera lo stato fisico del fluido DPD simulato e allo stesso tempo comporta un incremento significativo del numero di Schmidt, rispetto ai parametri convenzionali.

Inoltre, confrontando le SACFs per due diversi valori di s a parità di N_m in Fig. 6.6 si nota che per $s = 0.125$ si ha un incremento delle oscillazioni intorno allo zero. L'andamento della SACF per valori di s inferiori al valore standard, $s = 1$, conferma l'importanza di utilizzare una procedura di calcolo adeguata, che permetta di quantificare l'incertezza legata al rumore stocastico, nella valutazione delle proprietà reologiche di un fluido DPD.

Le proprietà di trasporto sono, dunque, influenzate dal numero di *coarse-graining* e di conseguenza anche dal coefficiente della forza conservativa. Infatti, all'aumentare di N_m i *beads* in una simulazione DPD tendono a muoversi più lentamente e ad effettuare più scambi di quantità di moto interagendo con i *beads* circostanti. Questo comportamento induce un aumento di viscosità e una diminuzione del coefficiente di diffusione con conseguente incremento del numero di Schmidt, come si nota in Fig. 6.5. Per ottenere numeri di Schmidt realistici è stato scelto, in seguito a questo primo studio, un numero di *coarse-graining* pari a 10, utilizzato in tutta l'analisi successiva

Tabella 6.3: Effetto del numero di *coarse-graining* e dei parametri DPD sulla viscosità, sul coefficiente di diffusione e sul numero di Schmidt. Tutte le grandezze dimensionate sono espresse in unità DPD.

N_m	s	γ	r_c^D	ν	\mathcal{D}	Sc
1	1	4.5	1	0.430	0.173	2.489
1	0.5	4.5	1	1.078	0.075	14.373
1	0.25	4.5	1	2.446	0.038	64.377
1	0.125	4.5	1	3.687	0.026	141.82
3	1	4.5	1	0.748	0.099	7.560
3	0.5	4.5	1	1.61	0.047	34.291
3	0.25	4.5	1	2.91	0.026	112.06
3	0.125	4.5	4.5	4.919	0.017	289.33
10	1	4.5	1	3.116	0.029	107.45
10	0.733	4.5	1	3.949	0.024	166.07
10	0.633	4.5	1	4.245	0.021	203.12
10	0.533	4.5	1	5.013	0.018	276.35
10	0.5	4.5	1	5.021	0.017	295.36
10	0.48	4.5	1	5.055	0.017	300.38
10	0.46	4.5	1	5.619	0.016	349.02
10	0.434	4.5	1	5.890	0.016	368.54
10	0.432	4.5	1	5.950	0.015	384.92
10	0.43	4.5	1	6.053	0.015	404.09
10	0.42	4.5	1	6.106	0.014	423.13
10	0.4	4.5	1	6.253	0.014	444.77
10	0.32	4.5	1	7.757	0.012	636.88
10	0.3	4.5	1	7.758	0.011	681.18
10	0.28	4.5	1	7.834	0.010	725.40
10	0.26	4.5	1	8.135	0.010	813.46
10	0.25	4.5	1	8.743	0.010	892.11
10	0.125	4.5	1	12.13	0.007	1784.1
10	1	9	1	4.289	0.024	177.00
10	1	18	1	5.414	0.017	319.96
10	1	20.2	1	5.830	0.016	367.70
10	1	22	1	5.950	0.015	397.25
10	1	24	1	6.361	0.014	453.40
10	1	25	1	6.387	0.014	458.62
10	1	30	1	7.862	0.012	650.81
10	1	36	1	7.999	0.010	767.75
10	1	72	1	15.33	0.005	2797.2
10	1	4.5	1.2	4.472	0.021	209.19
10	1	4.5	1.38	5.870	0.016	366.87
10	1	4.5	1.4	5.990	0.015	395.63
10	1	4.5	1.6	14.32	0.007	1962.1

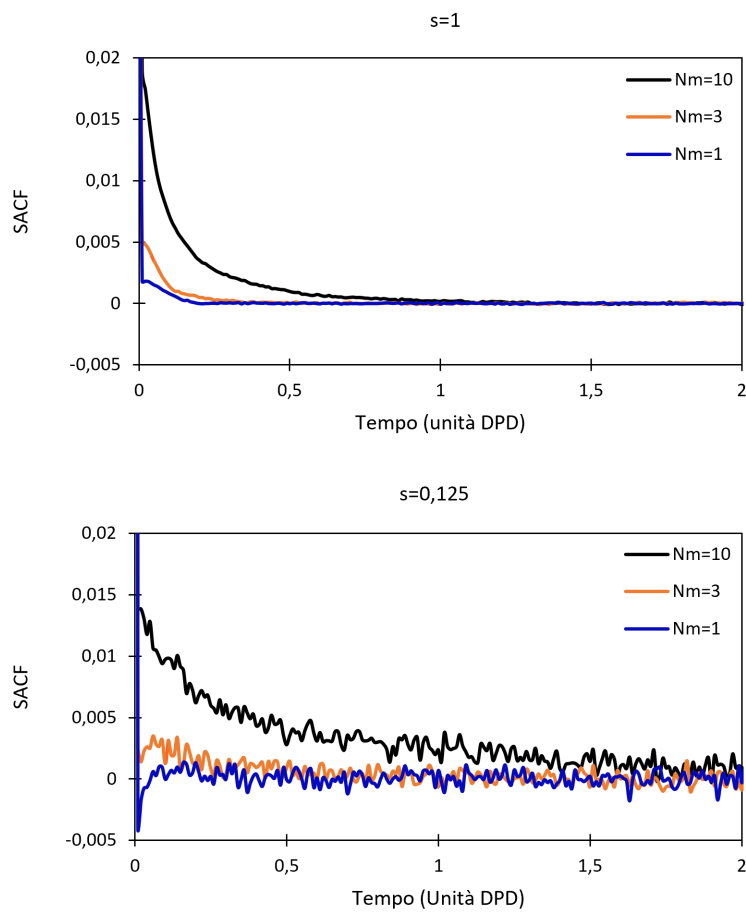


Figura 6.4: Effetto del grado di *coarse-graining* N_m sulla SACF per due differenti valori dell'esponente della funzione peso generalizzata s .

finalizzata allo studio del legame tra le proprietà di trasporto e i parametri del modello DPD, con particolare attenzione al valore dell'esponente s , della distanza di *cutoff* r_c^D e del coefficiente γ .

La Fig. 6.6, invece, mostra la dipendenza della SACF dai parametri DPD, (s, r_c^D, γ) . L'area sottesa al profilo di autocorrelazione aumenta al diminuire dell'esponente della funzione peso generalizzata, s , all'aumentare del raggio di *cutoff* dissipativo, r_c^D e del coefficiente dissipativo, γ . Queste variazioni, infatti, inducono tutte un incremento della dissipazione di quantità di moto dei *beads* risultando in una maggiore viscosità. Inoltre, allo stesso tempo inducono un aumento dell'ampiezza delle oscillazioni della SACF intorno allo zero. Ancora una volta emerge l'importanza di effettuare il calcolo della viscosità mediante una procedura ben definita che tenga conto dell'effetto dovuto al rumore e lo quantifichi. Infatti, se da un lato la diminuzione di s , l'incremento di r_c^D e di γ consentono di ottenere valori di viscosità maggiori e, dunque, una descrizione più realistica delle proprietà reologiche del fluido simulato dall'altro amplificano il rumore stocastico intrinseco delle simulazioni DPD. Tali variazioni, comportano, allo stesso tempo, una diminuzione della diffusività e, di conseguenza, un aumento significativo del numero di Schmidt. In questo lavoro di tesi, è stato studiato l'effetto dei singoli parametri del modello DPD sulla viscosità, sul coefficiente di diffusione e di conseguenza sul numero di Schmidt come illustrato nelle Figure 6.7, 6.8, 6.9 e 6.10.

Esplorando lo spazio dei parametri DPD e valutando la dipendenza delle proprietà di trasporto del fluido DPD da essi sono state individuate tre terne di valori (s, r_c^D, γ) che forniscono un numero di Schmidt prossimo a 370, che è approssimativamente il numero di Schmidt dell'acqua liquida a $25^\circ C$. In Tabella 6.4 sono riportati i dettagli di queste tre diverse simulazioni. I valori di viscosità ottenuti mediante la formula di Green-Kubo e seguendo la Procedura C sono gli stessi nei tre casi, dimostrando che i le tre terne sono descrittive delle proprietà reologiche dello stesso fluido. Inoltre, dal confronto con quelli di riferimento calcolati con i metodi di Müller-Plathe e di Einstein-Helfand si conferma di nuovo la validità della procedura basata sulla relazione di Green-Kubo nella valutazione della viscosità. Un aspetto importante da sottolineare riguarda la conversione della viscosità e del coefficiente di diffusione dalle unità DPD a quelle reali, riportata in Tabella 6.4: i valori dei coefficienti di trasporto ottenuti dalle simulazioni sono realistici per l'acqua a $25^\circ C$, in quanto prossimi ai valori sperimentali $0.9 \times 10^{-6} \text{ m}^2 \text{ s}^{-1}$ per la viscosità e $2.4 \times 10^{-9} \text{ m}^2 \text{ s}^{-1}$ per il coefficiente di diffusione. Dunque, il modello DPD esteso, includente la *shear dissipation*, con un'opportuna scelta dei parametri (N_m, s, r_c^D, γ) riproduce in maniera appropriata il trasporto di quantità di moto e di materia dell'acqua liquida, con almeno tre diverse terne di parametri per un valore di N_m pari a 10. Un altro elemento importante contenuto in Tab. 6.4 è il cosiddetto *effective friction coefficient*, che è una misura della dissipazione complessiva alla quale i *beads* sono soggetti.

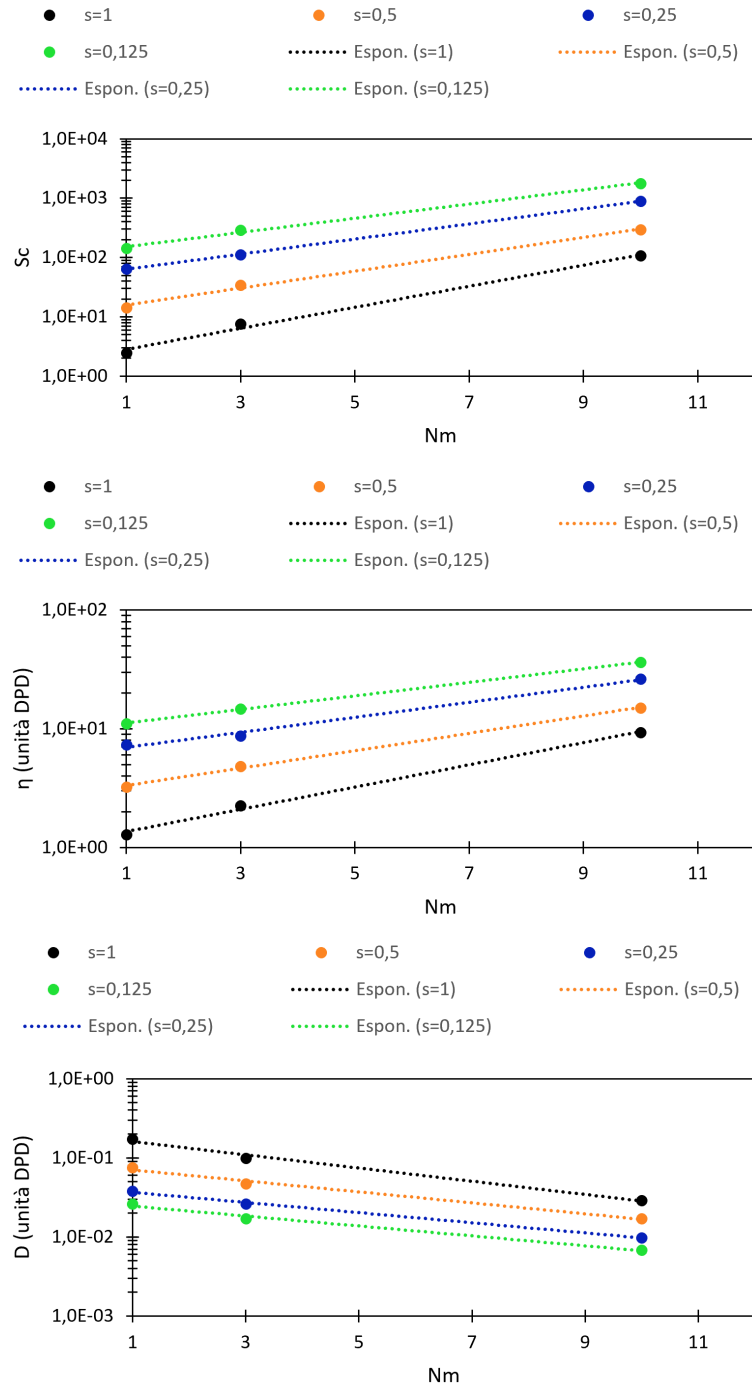


Figura 6.5: Effetto del grado di *coarse-graining* N_m sul numero di Schmidt (in alto), sulla viscosità dinamica (al centro) e sul coefficiente di diffusione (in basso).

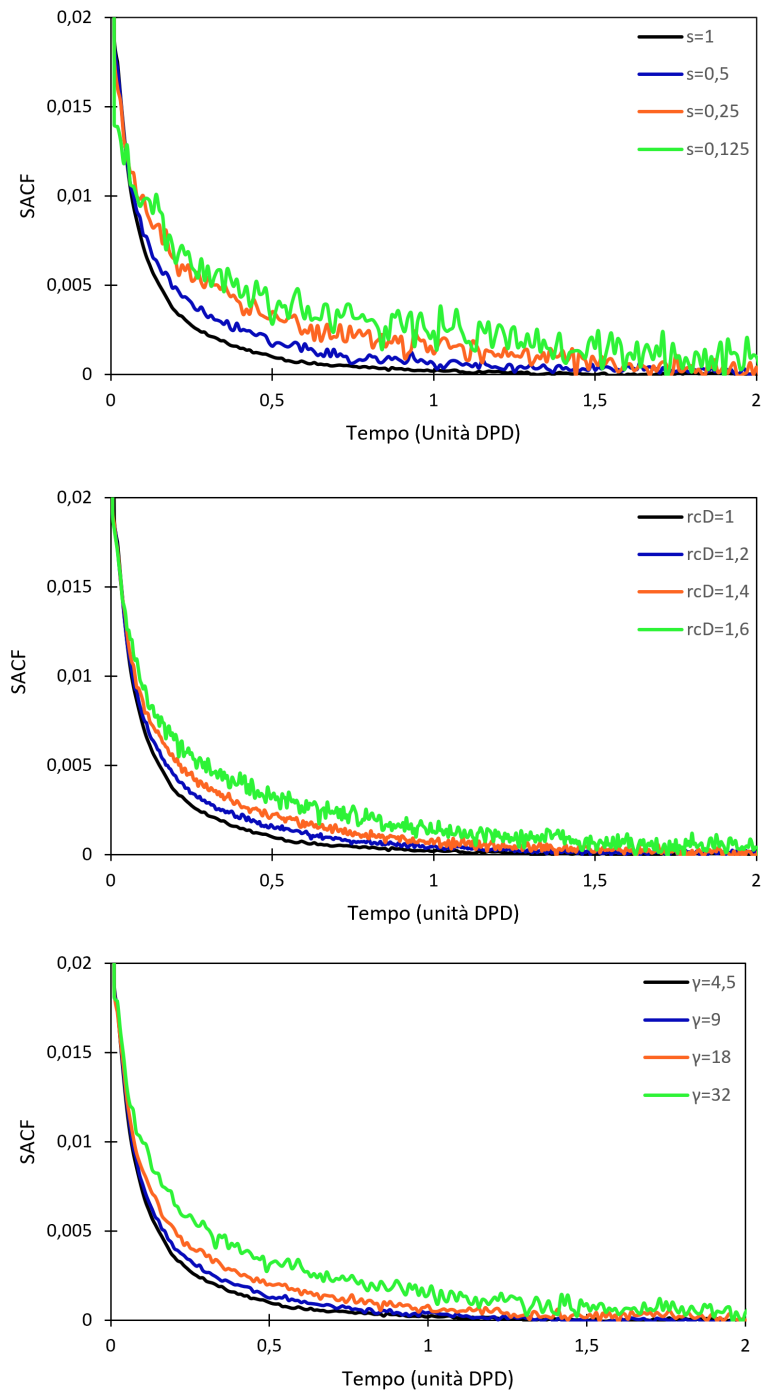


Figura 6.6: Effetto dell'esponente della funzione peso generalizzata s (in alto), del raggio di *cutoff* dissipativo r_c^D (al centro) e del coefficiente di forza dissipativa γ (in basso) sulla SACF.

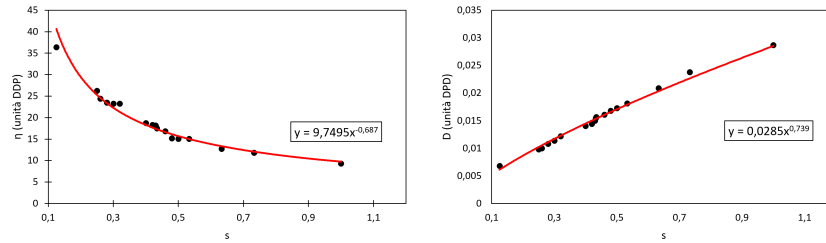


Figura 6.7: Effetto dell'esponente della funzione peso generalizzata s sui coefficienti di trasporto: viscosità dinamica (a sinistra) e diffusività (a destra).

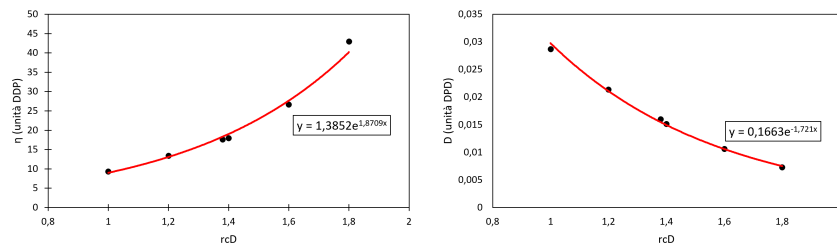


Figura 6.8: Effetto dell'esponente del raggio di *cutoff* r_c^D sui coefficienti di trasporto: viscosità dinamica (a sinistra) e diffusività (a destra).

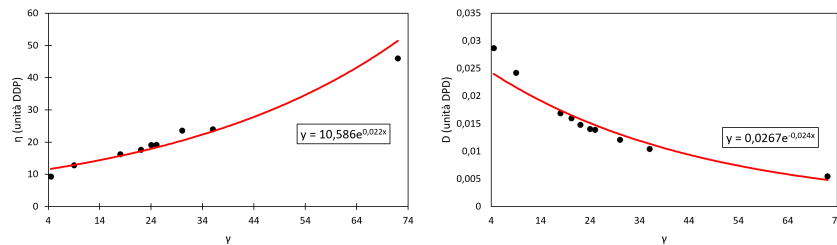


Figura 6.9: Effetto del coefficiente di forza dissipativa γ sui coefficienti di trasporto: viscosità dinamica (a sinistra) e diffusività (a destra).

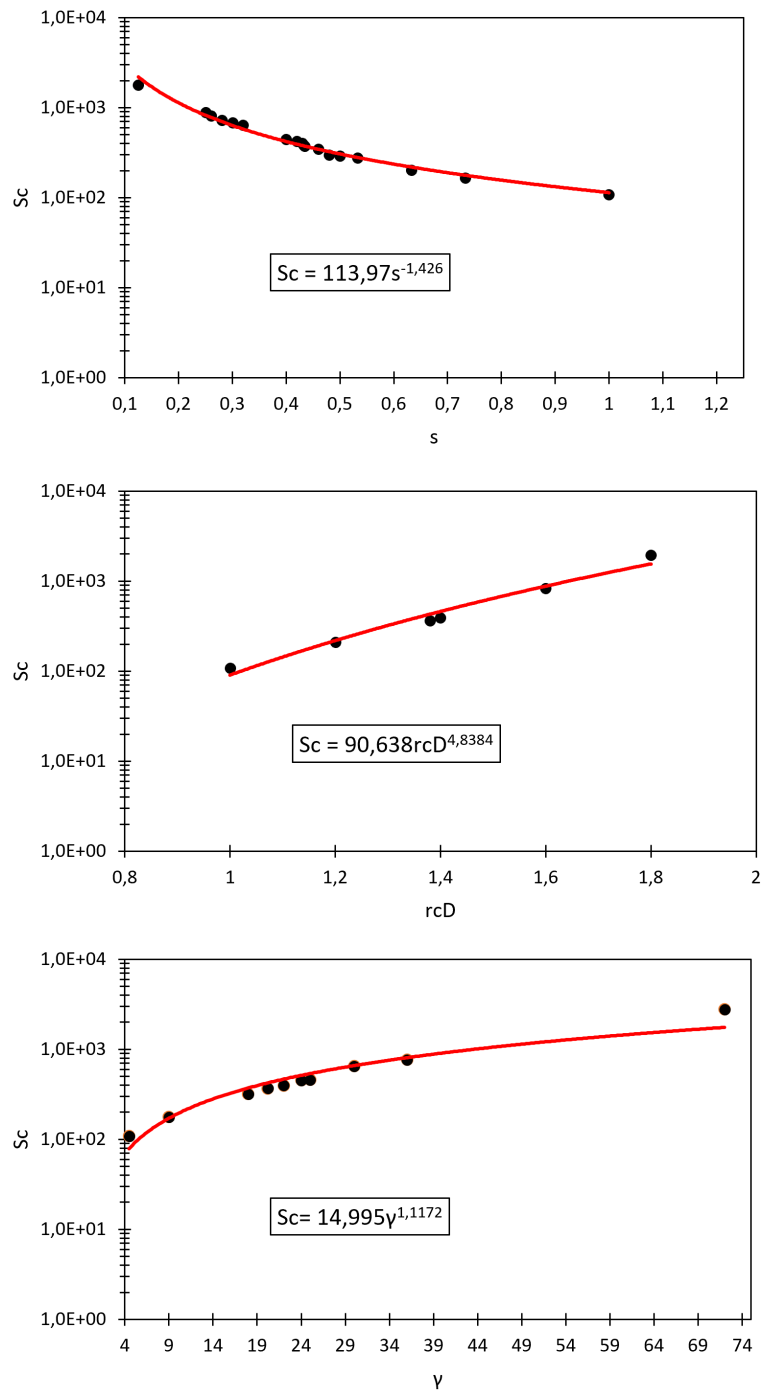


Figura 6.10: Effetto dell'esponente della funzione peso generalizzata s (in alto), del raggio di *cutoff* dissipativo r_c^D (al centro) e del coefficiente di forza dissipativa γ (in basso) sul numero di Schmidt.

Tale parametro, infatti, ha lo scopo di combinare in una singola quantità gli effetti di s, r_c^D, γ e implicitamente N_m . I coefficienti di dissipazione effettivi per i tre casi sono simili: le tre terne di parametri introducono lo stesso aumento di dissipazione di quantità di moto dei *beads* nel sistema rispetto al modello DPD standard. Inoltre, si nota che le proprietà di trasporto sono le medesime a parità di γ^{eff} . È, dunque, evidente il legame tra i parametri del modello DPD e il numero di Schmidt. Il coefficiente di dissipazione effettivo è definito nel seguente modo:

$$\gamma^{eff} = \int_0^{r_c^D} \gamma w(r_{ij}) g(r_{ij}) 4\pi r^2 dr_{ij} , \quad (6.1)$$

dove $g(r_{ij})$ è la funzione di distribuzione radiale. La forma della funzione di distribuzione radiale è determinata unicamente dal potenziale di interazione conservativo, che è il medesimo in tutte le simulazioni. È indipendente dal coefficiente di dissipazione effettivo. Se il contributo della forza conservativa fosse nullo, la funzione di distribuzione radiale sarebbe una costante, pari ad uno. In Fig. 6.11 è possibile vedere che le funzioni di distribuzione radiale calcolate dalle simulazioni con le tre differenti terne di parametri si sovrappongono, come atteso, dal momento che il contributo della forza conservativa alle interazioni tra i *beads* del sistema simulato è il medesimo.

Tabella 6.4: Risultati per tre differenti terne di parametri DPD (s, r_c^D, γ) per le quali si ha un numero di Schmidt prossimo a 370, ovvero a quello dell'acqua a 25°C. Le unità di misura non specificate sono unità DPD.

	Terna 1	Terna 2	Terna 3
s	0.434	1	1
r_c^D	1	1.38	1
γ	4.5	4.5	20.2
Δt	0.01	0.005	0.01
N_m	10	10	10
ρ	3	3	3
ν (G-K)	5.89	5.87	5.88
ν (E-H)	5.74	5.53	5.74
ν (M-P)	5.76	5.26	4.67
ν (m ² /s)	0.895×10^{-6}	0.892×10^{-6}	0.893×10^{-6}
\mathcal{D}	0.016	0.016	0.016
\mathcal{D} (m ² /s)	2.43×10^{-9}	2.43×10^{-9}	2.43×10^{-9}
Sc	368.54	367.22	367.5
γ^{eff}	4.36	3.90	4.00

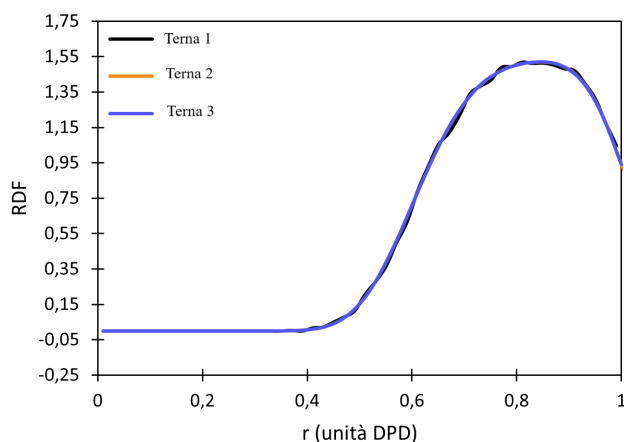


Figura 6.11: Funzione di distribuzione radiale (RDF) valutata per i tre differenti casi con numero di Schmidt prossimo a 370, ovvero a quello dell'acqua a 25°C .

6.2 Risultati di simulazione della miscela binaria costituita da acqua e Pluronic L64

6.2.1 Proprietà strutturali

Le proprietà strutturali del sistema sono determinate essenzialmente dalle interazioni conservative derivanti dal potenziale repulsivo a sfere soffici e dalle interazioni di legame descritte mediante il potenziale armonico. Di conseguenza, al fine di predire correttamente l'evoluzione delle microstrutture mediante una simulazione DPD è fondamentale, innanzitutto, la scelta dei parametri repulsivi, i quali sono correlati al numero di *coarse-graining*. È, altresì, fondamentale riprodurre e mantenere una corretta topologia delle catene polimeriche attraverso la scelta adeguata dei parametri del potenziale armonico, ovvero sia la distanza nominale di equilibrio r_0 e la costante di forza K_b . Questi ultimi due sono stati individuati grazie ad uno studio preliminare basato sul confronto tra le caratteristiche morfologiche del sistema simulato all'equilibrio e le predizioni teoriche. In particolare, è stato simulato un fluido costituito da acqua e Pluronic L64 al 5% in massa utilizzando il modello DPD secondo la sua formulazione originale. È stato valutato l'effetto di due diversi valori della costante di forza K_b sulla distribuzione dimensionale dei *clusters* (CMD) all'equilibrio, riportata per i due diversi casi analizzati in Fig. 6.16 e in Fig. 6.17. Tali grafici riportano il valore di concentrazione, designata con X_N , delle molecole di copolimero contenute in un aggregato di dimensione N in funzione del numero di aggregazione, indicato con N , di quell'aggregato. La dimensione degli aggregati, nota come numero di ag-

gregazione N , dunque, rappresenta il numero di molecole di Pluronic che costituiscono il *cluster* ed è così definita:

$$N = \frac{N_a}{N_c}, \quad (6.2)$$

dove N_a è il numero di *beads* di Pluronic costituenti l'aggregato, o *cluster*, di dimensione N e N_c è il numero di *beads* di Pluronic appartenenti ad una catena polimerica. La corrispondente concentrazione è definita come segue:

$$X_N = \frac{N_p \times N}{N_{tot}} \times \frac{\sum_i^n F_i}{n}, \quad (6.3)$$

N_{tot} è il numero totale di molecole presenti nel sistema e F_i è la frequenza, calcolata come media sul numero complessivo di *frame* analizzati, con cui appare l'aggregato di dimensione N . Inoltre, in Fig. 6.14 e in Fig. 6.15, si possono osservare i *clusters*, distinguibili grazie alla diversa colorazione, identificati dall'algoritmo di *clustering*, il cui funzionamento è stato illustrato in Sez. 5.4, nella *box* di simulazione all'equilibrio. Tali rappresentazioni, che consentono un'analisi puramente qualitativa, sono supportate dai grafici che riportano il numero totale di *clusters* di dimensione N in funzione del corrispondente numero di aggregazione N .

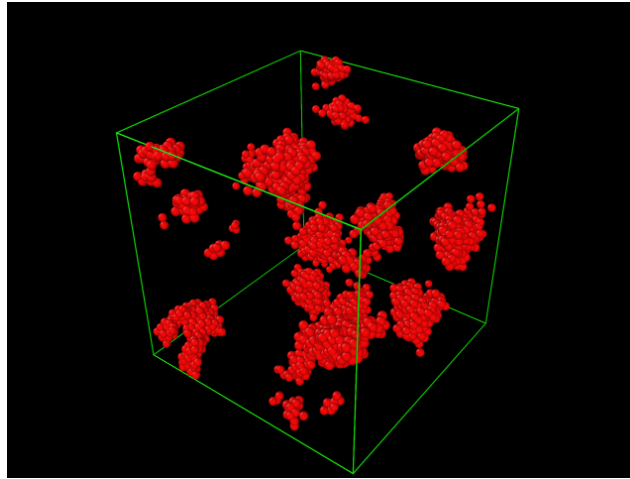


Figura 6.12: *Snapshot* di una miscela di Pluronic L64 e acqua all'equilibrio, con concentrazione percentuale in massa di copolimero del 5%. I parametri del potenziale armonico impiegati sono: $r_0 = 1$ e $K_b = 265$. I *beads* di colore rosso rappresentano i gruppi di PPO.

I risultati della simulazione condotta impostando il valore della costante di forza K_b pari a 265 sono maggiormente concordi con le previsioni teoriche. Infatti, la CMC, espressa come frazione in massa, del Pluronic L64 ricavata sperimentalmente è pari a 0.015 [49]. Per la concentrazione di copolimero nel

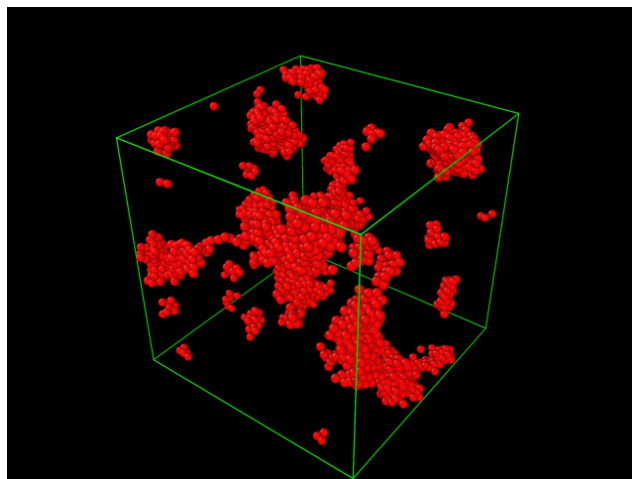


Figura 6.13: *Snapshot* di una miscela di Pluronic L64 e acqua all'equilibrio, con concentrazione percentuale in massa di copolimero del 5%. I parametri del potenziale armonico impiegati sono: $r_0 = 1$ e $K_b = 700$. I *beads* di colore rosso rappresentano i gruppi di PPO.

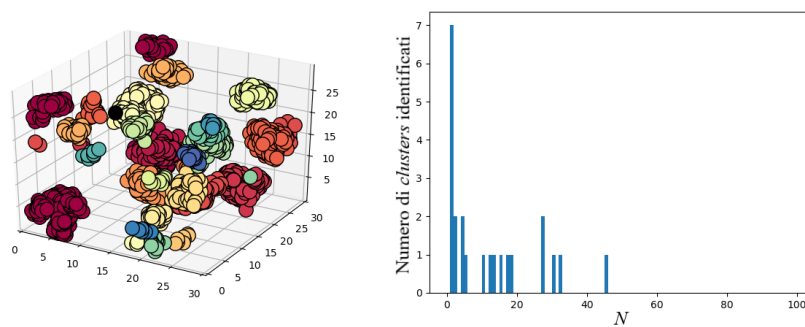


Figura 6.14: Risultato della simulazione di una miscela di Pluronic L64 e acqua, con concentrazione percentuale in massa di copolimero del 5%. I parametri del potenziale armonico impiegati sono: $r_0 = 1$ e $K_b = 265$. A sinistra la rappresentazione dei vari *clusters* con differenti colori identificati dall'algoritmo di *clustering*; a destra il numero totale di *clusters* (in ordinata) in funzione del numero di aggregazione N (in ascissa).

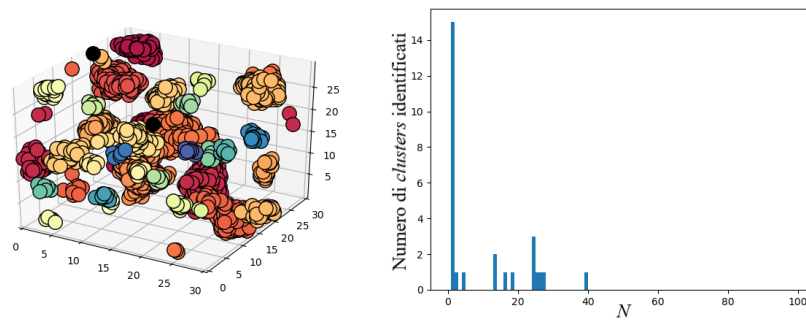


Figura 6.15: Risultato della simulazione di una miscela di Pluronic L64 e acqua, con concentrazione percentuale in massa di copolimero del 5%. I parametri del potenziale armonico impiegati sono: $r_0 = 1$ e $K_b = 700$. A sinistra la rappresentazione dei vari *clusters* con differenti colori identificati dall'algorithm di *clustering*; a destra il numero totale di *clusters* (in ordinata) in funzione del numero di aggregazione N (in ascissa).

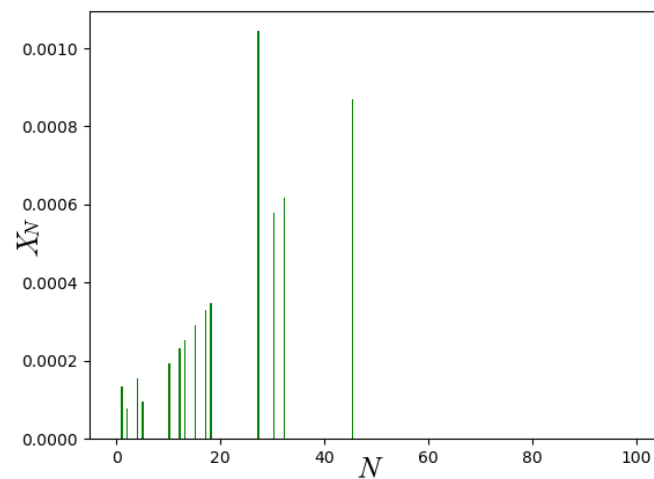


Figura 6.16: *Cluster mass distribution* (CMD) di una miscela di Pluronic L64 e acqua, con concentrazione percentuale in massa di copolimero del 5%. I parametri del potenziale armonico impiegati sono: $r_0 = 1$ e $K_b = 265$.

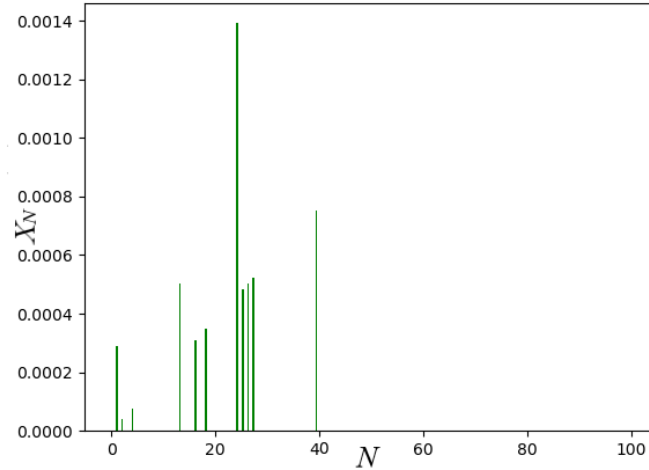


Figura 6.17: *Cluster mass distribution* (CMD) di una miscela di Pluronic L64 e acqua, con concentrazione percentuale in massa di copolimero pari a 5%. I parametri del potenziale armonico impiegati sono: $r_0 = 1$ e $K_b = 700$.

Tabella 6.5: Numero totale di *clusters* identificati all'equilibrio dall'algoritmo di *clustering* per diversi valori dei parametri del potenziale armonico (riportati in unità DPD), descrittivi le interazioni di legame interne alla catena polimerica.

r_0	K_b	Numero totale di <i>clusters</i>
1	265	23
1	700	28

sistema simulato, ovvero 0.05 (espressa come frazione in massa), ci si aspetta la formazione di micelle sferiche, come si può dedurre anche dal diagramma di fase della miscela binaria costituita da acqua e Pluronic L64, riportato in Fig. 4.4 in Sez. 4.2. Impostando un valore maggiore di costante di forza k_B , pari a 700, si osserva la formazione di un numero maggiore di *clusters*, complessivamente pari a 28 come si evince dalla Tab. 6.5, ma il processo di aggregazione non è tale da portare alla formazione di micelle sferiche. Il numero di *clusters*, identificati dall'algoritmo, costituiti da una singola catena polimerica, $N = 1$, sono più del doppio rispetto all'altro caso. Un valore di costante di forza eccessivamente alto, rispetto al valore del coefficiente della forza di repulsione agente tra i *beads*, rende le catene polimeriche troppo rigide influenzando negativamente il processo di aggregazione e la formazione

di micelle. In base alle motivazioni esposte, per i parametri del potenziale armonico in tutte le simulazioni successive sono stati impiegati i seguenti valori: $r_0 = 1$ e $K_b = 265$.

6.2.2 Calcolo della viscosità a scorrimento nullo

Dallo studio condotto sul fluido semplice, rappresentativo dell'acqua, è emerso, come illustrato nella Sez. 6.1, che le proprietà di trasporto sono influenzate sia dal numero di *coarse-graining*, al quale è legato il coefficiente della forza repulsiva, che dai seguenti parametri: l'esponente della funzione peso generalizzata s , il raggio di *cutoff* dissipativo r_c^D e il coefficiente di forza dissipativa γ . Le proprietà strutturali non sono influenzate dalla variazione di questi parametri la cui scelta è unicamente finalizzata alla corretta riproduzione delle proprietà di trasporto, tra cui la viscosità. Si riportano in Tab. 6.7 i valori di viscosità ottenuti dalle simulazioni DPD esplorando lo spazio dei parametri, s, r_c^D e γ , e in Tab. 6.6 i corrispondenti valori sperimentali, trovati in letteratura [72]. Per quanto riguarda i parametri s e r_c^D ,

Tabella 6.6: Valori sperimentali di viscosità [72] della miscela costituita da acqua e Pluronic L64. x_1 è la frazione in massa della componente acquosa e $x_2 = 1 - x_1$ è la frazione in massa di copolimero in miscela. η_S è il valore di viscosità dinamica sperimentale espresso in (Pa×s).

x_1	x_1	η_S
0.95	0.05	1.78×10^{-3}
0.75	0.25	1.08×10^{-2}
0.65	0.35	2.57×10^{-2}

si assumono uguali i valori descrittivi delle interazioni dei *beads* rappresentativi delle unità ripetitive della catena (designati con pedice 2) di copolimero tra di loro e con quelli rappresentativi dell'acqua (designati con pedice 1) ($s_{22} = s_{12} = s_{22,12}$; $r_{c,22}^D = r_{c,12}^D = r_{c,22,12}^D$). Invece, per quanto riguarda il coefficiente dissipativo γ , il valore di γ_{12} , noto come *cross γ -parameter*, è calcolato come segue:

$$\gamma_{12} = x_2\gamma_{11} + x_1\gamma_{22}, \quad (6.4)$$

ovvero come combinazione lineare di γ_{11} , coefficiente di dissipazione relativo all'interazione di due *beads* di acqua, e γ_{22} , coefficiente di dissipazione relativo all'interazione di due *beads* di Pluronic; x_1 è la frazione in massa della componente acquosa in miscela e $x_2 = 1 - x_1$ è la frazione in massa del copolimero in miscela. Tale legge è sviluppata in analogia al modello di Darken per il calcolo dei coefficienti di mutua diffusione di due componenti in miscela in funzione dei rispettivi coefficienti di autodiffusione[73]. Secondo quanto si evince dalla prima riga della Tab. 6.7 e come atteso, impiegando i valori standard della DPD ($s = 1, \gamma = 4.5$ e $r_c^D = 1$), per descrivere le

Tabella 6.7: Studio parametrico per valutare l'effetto di s , γ e r_c^D sulla viscosità della miscela binaria costituita da acqua e Pluronic L64. I pedici dei parametri indicano le tipologie di *beads* che interagiscono: quello rappresentativo dell'acqua è designato con pedice 1, quelli rappresentativi delle unità ripetitive della catena di copolimero, PEO e PPO, sono designati con pedice 2. x_2 è la concentrazione di copolimero nella miscela espressa come frazione in massa. η_C è il valore di viscosità dinamica ottenuto dalla simulazione convertito in unità fisiche reali, quindi espresso in (Pa \times s).

x_2	s_{11}	$s_{22,12}$	γ_{11}	γ_{22}	γ_{12}	$r_{c,11}^D$	$r_{c,22,12}^D$	η_C
0.05	1	1	4.5	4.5	4.5	1	1	5.34×10^{-4}
0.05	0.434	0.434	4.5	4.5	4.5	1	1	1.02×10^{-3}
0.25	0.434	0.434	4.5	4.5	4.5	1	1	1.59×10^{-3}
0.25	0.434	0.1085	4.5	4.5	4.5	1	1	2.24×10^{-3}
0.25	0.434	0.000434	4.5	4.5	4.5	1	1	2.24×10^{-3}
0.35	0.434	0.000434	4.5	4.5	4.5	1	1	3.63×10^{-3}
0.35	1	1	20.2	40.4	33.33	1	1	2.63×10^{-3}
0.35	1	1	20.2	323.2	217.15	1	1	3.41×10^{-3}
0.35	1	1	4.5	4.5	4.5	1.38	2.76	9.51×10^{-3}
0.35	1	0.5	4.5	4.5	4.5	1.38	2.76	2.64×10^{-2}
0.35	0.434	0.434	4.5	240	157.58	1	1	2.72×10^{-2}
0.35	1	1	4.5	240	157.58	1.38	1.38	1.53×10^{-2}
0.35	1	1	4.5	260	170.58	1.38	1.38	1.57×10^{-2}

interazioni tra tutte le tipologie di *beads* presenti nel sistema, il valore di viscosità ottenuto dalla simulazione si discosta molto dal valore sperimentale atteso. Il risultato non migliora di molto adottando i valori della Terna 1, identificata nella Sez. 6.1, per descrivere le interazioni tra tutte le tipologie di *beads* presenti nel sistema, come si evince dal confronto tra i valore di viscosità sperimentali attesi e quelli calcolati per concentrazioni di copolimero del 5% e del 25%. Si deduce che i parametri di modello relativi alle interazioni che coinvolgono i *beads* di Pluronic devono avere un valore diverso rispetto a quelli descrittivi delle interazioni tra i *beads* di acqua. Per quest'ultima si adotta una delle tre Terne individuate nella Sez. 6.1. Riducendo il valore di $s_{22,12}$ di un fattore pari a $1/8$ o $1/1000$ rispetto a s_{11} , il valore di viscosità non aumenta in maniera significativa. La stessa considerazione vale per i risultati ottenuti incrementando γ_{22} di un fattore 2 o 16 rispetto a γ_{11} . Un incremento maggiore si ottiene raddoppiando il valore di $r_{c,22,12}^D$ rispetto a $r_{c,11}^D$, ma non sufficiente alla corretta predizione del valore sperimentale. Solo variando contemporaneamente due parametri relativi alle interazioni del Pluronic, si registra un significativo aumento di viscosità. Come si evince dall'ultima riga della Tab 6.7 dimezzando il valore di $s_{22,12}$ rispetto a s_{11} e, allo stesso tempo, raddoppiando il valore di $r_{c,22,12}^D$ rispetto a $r_{c,11}^D$ il valore di viscosità calcolato è molto simile a quello sperimentale. Tuttavia, è auspicabile utilizzare gli stessi valori di s e r_c^D per descrivere tutte le interazioni, non avendo a disposizione relazioni che permettano di calcolare il valore di s_{12} e $r_{c,12}^D$. Un approccio promettente, seguito nelle ultime tre righe della Tab. 6.7, consiste nell'adottare per la descrizione delle interazioni tra *beads* di acqua i parametri della Terna 1 o della Terna 2, assumendo valori uguali di s e r_c^D per le interazioni coinvolgenti i *beads* di Pluronic, e nel modulare il valore di γ_{22} al fine di ottenere il valore di viscosità predetto sperimentalmente. La scelta tra le Terne è dettata da considerazioni inerenti il costo computazionale della simulazione. Il vantaggio di quest'ultimo approccio è rappresentato dalla possibilità di stimare il valore di γ_{12} , o *cross γ -parameter*, in accordo con il modello di Darken, a partire dal valore di γ_{11} , derivante dallo studio precedente sul fluido semplice, e dal valore imposto di γ_{22} . L'obiettivo è individuare il valore di quest'ultimo parametro che consenta la corretta riproduzione della viscosità della miscela simulata.

Conclusioni

In questo lavoro di tesi è stata utilizzata la tecnica mesoscala DPD come strumento computazionale predittivo delle proprietà reologiche dei fluidi. In particolare, di un fluido semplice rappresentante l'acqua e di un fluido complesso costituito da acqua e Pluronic L64. La prima parte del lavoro è stata finalizzata alla definizione di una procedura di calcolo della viscosità affidabile basata sulla formula di Green-Kubo per sfruttare i vantaggi propri di un metodo di equilibrio e all'individuazione dei parametri del modello DPD descrittivi delle reali proprietà di trasporto del fluido simulato. Le principali fasi sono state:

- individuare sperimentalmente i parametri di simulazione e di correlazione per il calcolo della SACF in LAMMPS;
- mettere a punto una corretta procedura di calcolo della viscosità basata sulla formula di Green-Kubo;
- studiare il legame tra il numero di *coarse-graining* e le proprietà di trasporto del sistema simulato;
- valutare l'effetto di alcuni parametri del modello DPD sulle proprietà di trasporto del sistema simulato;
- individuare tre terne di parametri in grado di riprodurre le proprietà di trasporto dell'acqua liquida a temperatura ambiente.

La seconda parte del lavoro è stata incentrata, invece, sulla simulazione del sistema complesso costituito da acqua e Pluronic L64 a diverse concentrazioni. In particolare è stato:

- condotto uno studio preliminare per la scelta dei parametri del potenziale armonico, descrittivo delle interazioni di legame tra i *beads* della catena polimerica, adeguati ad una corretta riproduzione delle catene polimeriche e predizione dell'evoluzione delle microstrutture da esse formate;
- esplorato lo spazio dei parametri DPD per individuare quelli adatti alla riproduzione delle reali proprietà reologiche del fluido.

Si può concludere che scegliendo opportunamente i parametri di simulazione per il calcolo della SACF e seguendo la procedura individuata (Procedura C), la formula di Green-Kubo permette di ottenere valori di viscosità comparabili con quelli prodotti con i metodi di riferimento (ovvero di Müller-Plathe e Einstein-Helfand). I vantaggi sono principalmente due:

- tale metodo non soffre delle limitazioni proprie dei metodi di non equilibrio;
- la procedura individuata permette di superare la principale limitazione del calcolo basato sulla formula di Green-Kubo legata alle oscillazioni della SACF derivanti dalla natura stocastica intrinseca delle simulazioni DPD. Infatti, oltre a fornire il valore medio della viscosità permette di quantificare l'incertezza associata alla procedura numerica completa che risulta essere minore o dello stesso ordine di grandezza di quella propria delle misure sperimentali.

Il modello DPD esteso, includente la *shear dissipation*, si presta bene alla simulazione di fluidi ad elevato numero di Schmidt utilizzando nella modellazione un numero di *coarse-graining* sufficientemente alto. Le proprietà di trasporto del fluido simulato risultano essere legate ai parametri del modello DPD, quali l'esponente della funzione peso generalizzata s , il raggio di *cutoff* dissipativo r_c^D e il coefficiente della forza dissipativa γ . In particolare, la giusta scelta di parametri (s, r_c^D, γ) , il cui effetto combinato di incremento delle interazioni dissipative tra i *beads* del sistema è quantificabile tramite il coefficiente dissipativo effettivo γ^{eff} , permette la corretta riproduzione delle proprietà di trasporto dell'acqua a temperatura ambiente. Sono state individuate tre terne che descrivono il comportamento reologico dello stesso fluido. La definizione di terna ottimale deriva, dunque, da considerazioni di tipo esclusivamente computazionale: aumentare il raggio di *cutoff* dissipativo è meno conveniente rispetto alla modulazione degli altri due parametri, ovvero l'esponente della funzione peso generalizzata e il coefficiente della forza dissipativa. Infine, si può anche concludere che sviluppando opportunamente il modello *coarse-grained*, descrivendo correttamente la topologia della catena polimerica e individuando i parametri DPD adeguati, è possibile impiegare il modello DPD esteso per la predizione delle proprietà reologiche di un fluido strutturato, come la miscela binaria costituita da acqua e Pluronic L64.

Appendice A

Descrizione degli input script in LAMMPS

In questo capitolo vengono illustrati due esempi di script utilizzati rispettivamente per simulare un fluido semplice, assimilabile all'acqua, e un fluido complesso, ovvero una miscela binaria costituita da acqua e Pluronic L64. L'obiettivo è quello di fornire le linee guida per simulare un sistema di particelle DPD attraverso il codice *open source* di dinamica molecolare LAMMPS. Nei due esempi di script il calcolo della viscosità è eseguito tramite la formula di Green-Kubo.

Codice A.1: **in.water**

```
1 # LAMMPS input script specifies a simple DPD simulation about water
2   to evaluate transport properties
3
4 #Initialization
5 units      lj
6 variable ndim    equal 3
7
8 # Box size
9 variable xsize  equal 15
10 variable ysize  equal 15
11 variable zsize  equal 15
12
13 # DPD parameters
14 variable rho    equal 3
15 variable kb     equal 1
16 variable T      equal 1/{kb}
17 variable rc     equal 1.0
18 variable rcD    equal 1.0
19 variable a      equal 265
20 variable dt     equal 0.01
21 variable gammapar equal 4.5
22 variable gammaper equal 4.5
23 variable spar   equal 1
24 variable sper   equal 1
25 variable nBeads equal ({xsize}*{ysize}*{zsize})*{rho}
26
27 # Simulation parameters
28 timestep        {dt}
```

```

91
92 run                ${nrun}
93
94 variable v         equal (v_v11+v_v22+v_v33)/3.0
95 print             "viscosity :_ $v_(DPD_units) "
96
97 # Restart File
98 write_restart     restart.lam
99
100 # DONE

```

La prima parte dello script, la fase di inizializzazione, consiste nella definizione delle unità di misura delle grandezze fisiche e nell'assegnazione di un valore numerico a tutte le variabili che restano costanti nel corso della simulazione. Il comando *units* specifica lo stile per le unità di misura, in particolare *lj* (Lennard-Jones) indica che tutte le grandezze sono adimensionali, in accordo con la formulazione del modello DPD. Le variabili sono definite da un nome arbitrario, seguito da un valore numerico. Vengono impostate le tre dimensioni spaziali *xsize*, *ysize* e *zsize* pari a 15 unità DPD. Si definiscono i parametri del modello DPD esteso, ovvero la densità numerica del sistema, ρ , l'energia $k_B T$, il raggio di *cutoff* r_c che definisce la sfera d'interazione della forza conservativa, il raggio di cutoff relativo all'azione delle forze dissipativa e stocastica r_c^D , il coefficiente della forza repulsiva a , i coefficienti delle componenti parallela e perpendicolare della forza viscosa, rispettivamente γ_{\parallel} e γ_{\perp} , gli esponenti della funzione peso relativa alla forza stocastica per le componenti parallela e perpendicolare, rispettivamente s_{\parallel} e s_{\perp} . In questa fase bisogna anche definire le variabili di simulazione: il passo temporale, *dt*, la dimensionalità del sistema, *ndim*, la durata della fase di equilibratura del sistema *neql* e la durata della simulazione *nrun*.

Nella seconda fase, dopo la definizione del dominio computazionale, si configura il sistema. Il comando *region* definisce una regione dello spazio a forma di prisma delimitata dalle coordinate spaziali $(x, y, z) = (0, 0, 0)$ e $(\{xsize\}, \{ysize\}, \{zsize\})$. Il comando *create_box* converte la regione in una *box* di simulazione. Il comando *lattice none 1.0* fornisce un set di punti con specifica spaziatura, pari a 1 unità DPD. Il comando *create_atoms* genera all'interno della *box* in maniera casuale (*random*) le particelle che rappresentano l'acqua. Il comando *atom_style* definisce lo stile di interazione tra le particelle, nello specifico *dpdext*. Le condizioni al contorno sono state impostate periodiche tramite il comando *boundary p p p*. Il comando *comm_modify* definisce i parametri che gestiscono lo scambio di informazioni riguardanti gli atomi tra processori effettuata ogni passo temporale; l'opzione *vel yes* attiva la memorizzazione e la comunicazione delle velocità tra i vari processori, oltre alle posizioni. Essa è importante per la dipendenza della forza dissipativa dalle velocità relative delle particelle. A questo punto è necessario specificare i parametri per il calcolo delle interazioni tra coppie di *beads* attraverso i comandi *pair_style* e *pair_coeff*.

Nell'ultima parte dello script si impostano le caratteristiche della simula-

zione. Attraverso i comandi *fix* e *run* si esegue per il numero di *timesteps* indicati da \$nrun un'integrazione di tipo NVE delle equazioni del moto, per aggiornare la posizione e la velocità delle particelle. L'integratore scelto garantisce la conservazione del numero delle particelle (N), il volume (V) e l'energia interna (E) per tutta la durata della simulazione. I comandi *thermo* e *thermo_style* calcolano e stampano le informazioni termodinamiche del sistema come temperatura, energia, pressione e volume ogni 1000 *timesteps*. Il comando *dump* memorizza le coordinate delle particelle nel file video.lammpstrj con una frequenza temporale definita, in questo caso ogni 100 *timesteps*. Questo file di traiettoria può essere letto da programmi di visualizzazione molecolare che producono immagini e video del sistema delimitato dalla *box* di simulazione come VMD (Visual Molecular Dynamics) e OVITO (Open Visualization Tool). Nella fase di post-processing si possono implementare calcoli di vario tipo, di elaborazione dei dati ottenuti per il calcolo di grandezze d'interesse. In questo script sono riportati a titolo di esempio la valutazione della viscosità secondo la formula di Green-Kubo e del coefficiente di diffusione mediante il calcolo dell'MSD.

Codice A.2: **molecule.txt**

```

1 # Topology file of Non-ionic Tri-block Co-polymer: Pluronic L64 A(3)-B(9)-A(3)
2 15 atoms
3 14 bonds
4
5 Coords
6
7 1 0.00 0 0
8 2 0.05 0 0
9 3 0.10 0 0
10 4 0.15 0 0
11 5 0.20 0 0
12 6 0.25 0 0
13 7 0.30 0 0
14 8 0.35 0 0
15 9 0.40 0 0
16 10 0.45 0 0
17 11 0.50 0 0
18 12 0.55 0 0
19 13 0.60 0 0
20 14 0.65 0 0
21 15 0.70 0 0
22
23
24
25 Types
26
27 1 1
28 2 1
29 3 1
30 4 2
31 5 2
32 6 2
33 7 2
34 8 2
35 9 2
36 10 2

```

37 11 2
38 12 2
39 13 1
40 14 1
41 15 1

42
43

44
45 Bonds

46
47 1 1 1 2
48 2 1 2 3
49 3 1 3 4
50 4 1 4 5
51 5 1 5 6
52 6 1 6 7
53 7 1 7 8
54 8 1 8 9
55 9 1 9 10
56 10 1 10 11
57 11 1 11 12
58 12 1 12 13
59 13 1 13 14
60 14 1 14 15

61
62

63 Special Bond Counts

64
65 1 1 0 0
66 2 2 0 0
67 3 2 0 0
68 4 2 0 0
69 5 2 0 0
70 6 2 0 0
71 7 2 0 0
72 8 2 0 0
73 9 2 0 0
74 10 2 0 0
75 11 2 0 0
76 12 2 0 0
77 13 2 0 0
78 14 2 0 0
79 15 1 0 0

80
81

82 Special Bonds

83
84 1 2
85 2 1 3
86 3 2 4
87 4 3 5
88 5 4 6
89 6 5 7
90 7 6 8
91 8 7 9
92 9 8 10
93 10 9 11
94 11 10 12
95 12 11 13
96 13 12 14
97 14 13 15
98 15 14

```

62 bond_style      harmonic
63 bond_coeff      1 265.0 1
64
65 pair_style       hybrid dpdext ${T}  ${rc} 928948 srp ${rc_srp} * mid exclude yes
66 pair_coeff       1 1 dpdext 265.0  ${gamma_wat}  ${gamma_wat}  ${s_wat}  ${s_wat}
67 pair_coeff       1 2 dpdext 266.0  ${gamma_pol}  ${gamma_pol}  ${s_pol}  ${s_pol}
68 pair_coeff       2 2 dpdext 265.0  ${gamma_pol}  ${gamma_pol}  ${s_pol}  ${s_pol}
69 pair_coeff       1 3 dpdext 274.0  ${gamma_pol}  ${gamma_pol}  ${s_pol}  ${s_pol}
70 pair_coeff       2 3 dpdext 278.0  ${gamma_pol}  ${gamma_pol}  ${s_pol}  ${s_pol}
71 pair_coeff       3 3 dpdext 265.0  ${gamma_pol}  ${gamma_pol}  ${s_pol}  ${s_pol}
72
73 pair_coeff       1 4 none
74 pair_coeff       2 4 none
75 pair_coeff       3 4 none
76
77 pair_coeff       4 4 srp  ${C_srp}  ${rc_srp}
78
79 # Change neighbor list parameters to avoid dangerous builds
80 neighbor          1.0 bin
81 neigh_modify      delay 0 every 1 check yes
82
83 # Equilibration stage
84 fix                1 all nve
85 thermo             ${Nfreq}
86 run                ${neql}
87
88 # Post-processing:
89 # Post-processing correlation function parameters
90 variable Ncv       equal 1          # correlation length
91 variable Nrep      equal 1000       # sample repeat
92 variable Nfreq     equal 1000       # dump interval
93
94 # Green-Kubo viscosity calculation settings
95 reset_timestep     0
96 variable pxy       equal pxy
97 variable pxz       equal pxz
98 variable pyz       equal pyz
99 variable V         equal vol
100 variable K         equal 1/(${kb}*${T})*$V*${Ncv}*${dt}
101
102 fix                SS all ave/correlate $Ncv $Nrep $Nfreq &
103                   v_pxy v_pxz v_pyz type auto file time_cor.txt ave running
104
105 variable v11       equal trap(f_SS[3])*${K}
106 variable v22       equal trap(f_SS[4])*${K}
107 variable v33       equal trap(f_SS[5])*${K}
108
109 thermo_style       custom step temp press v_v11 v_v22 v_v33
110
111 # Dump Atom Positions for Visualization
112 dump                dumpnew all atom 1000 video.lammpstrj
113 dump                1 hydrophobic xyz 1000 cluster.xyz
114
115 run                ${nrun}
116
117 variable            v equal (v_v11+v_v22+v_v33)/3
118 print              "viscosity:_${v_}(DPD_units)"
119
120 # Restart File
121 write_restart       restart.lam
122
123 # DONE

```

Per quanto riguarda la simulazione della miscela acqua-Pluronic L64, l'implementazione del modello DPD corrispondente segue le stesse fasi illustrate nel caso precedente. È, tuttavia, necessaria la scrittura preliminare del file di topologia descrittivo della catena di Pluronic L64 (*molecole.txt*), in cui vengono riportati il numero di *beads* e il numero di legami tra di essi, le coordinate di ogni particella, il tipo di ogni legame e, infine, come sono legate le molecole tra di loro. Nello specifico una catena di Pluronic L64 è costituita da 15 *beads* in serie, tenuti insieme da 14 legami. Attraverso il comando *create_atoms*, contenuto nello script, vengono generati all'interno della box, in posizione casuale i *beads* rappresentativi dell'acqua e del polimero. Questi ultimi sono organizzati secondo la descrizione riportata nel file di topologia "*molecole.txt*" e nello script vengono identificati con il nome di *polymer*. È necessario descrivere i legami tra i *beads* costituenti le catene. Il comando *bond_style harmonic* definisce un legame di tipo armonico tra le particelle e il comando *bond_coeff* assegna un determinato valore ai reattivi parametri, ovvero la distanza nominale di equilibrio e la costante elastica. A seconda della concentrazione polimerica del sistema simulato, può essere necessaria l'implementazione in LAMMPS del potenziale repulsivo agente tra i legami non vicinali della catena, mediante lo stile *srp*. I parametri da definire sono il coefficiente della forza repulsiva e la distanza di *cutoff* che definisce la regione d'azione della forza, rispettivamente *C_srp* e *rc_srp*. L'opzione *mid* specifica che la distanza tra i legami è calcolata facendo riferimento ai punti medi di essi. L'opzione *Newton on on* comporta un modesto risparmio computazionale. Infatti, l'impostazione del primo *flag* prevede che se due atomi interagenti si trovano su processori diversi, solo un processore calcola l'interazione e la comunica all'altro, in virtù della terza legge di Newton. Il secondo *flag* attiva la stessa opzione in riferimento alle interazioni di legame.

Appendice B

Algoritmo di clustering

Codice B.1: **fnc.py**

```
1  #Class: TimeStep
2  #Comment: PPO atoms positions
3  #Param:
4  # @number: the "label of the atoms"
5  # @xCoord: x position of one atom
6  # @yCoord: y position of one atom
7  # @zCoord: z position of one atom
8  # @beadHisto: number of beads per cluster
9  # @chainHisto: number of chains per cluster
10 class TimeStep(object):
11     number = []
12     xCoord = []
13     yCoord = []
14     zCoord = []
15     beadHisto = []
16     chainHisto = []
17     F = []
18
19 #Function:
20 #Comment: ## Object Initialization
21 def __init__(self, number, xCoord, yCoord, zCoord, beadHisto, chainHisto, F):
22     self.number = number
23     self.xCoord = xCoord
24     self.yCoord = yCoord
25     self.zCoord = zCoord
26     self.beadHisto = beadHisto
27     self.chainHisto = chainHisto
28     self.F = F
29
30 #Function:
31 #Param:
32 # @self: reference object
33 # @atoms: label of the atom
34 # @xCoord: x coordinate
35 # @yCoord: y coordinate
36 # @zCoord: z coordinate
37 def add_atom(self, atoms, mxCoord, myCoord, mzCoord):
38     self.number.append(atoms)
39     self.xCoord.append(float(mxCoord))
40     self.yCoord.append(float(myCoord))
41     self.zCoord.append(float(mzCoord))
```

```

42 #Function:
43 #Param:
44 # @histo: list containing all the cluster sizes
45 # @size: number of clusters
46 #Return:
47 # @matrix: the matrix counting the occurrences
48 def AverageManager(histo ,size ,matrix):
49     columns = 2
50     for cluster_size in histo:
51         for mini_row in range(0,5000):
52             if (matrix[mini_row][0] != cluster_size) and
53                 (matrix[mini_row][0] == 0) and
54                 (matrix3mini_row][1] != 1):
55                 matrix[mini_row][0] = cluster_size
56                 matrix[mini_row][1] += 1
57                 break
58             elif (matrix[mini_row][0] == cluster_size):
59                 matrix[mini_row][1] += 1;
60                 break
61         return (matrix)
62
63 def sortingHistograms(normalized_histogram_x , normalized_histogram_y):
64     ordered_normalized_histogram_y = []
65     ordered_normalized_histogram_x = sorted(normalized_histogram_x)
66     for x in range(0, len(ordered_normalized_histogram_x), 1):
67         for y in range(0, len(normalized_histogram_x), 1):
68             if (normalized_histogram_x[y] == ordered_normalized_histogram_x[x]):
69                 ordered_normalized_histogram_y.append(normalized_histogram_y[y])
70     return ordered_normalized_histogram_y
71
72 def rebuildingHistograms(ordered_normalized_histogram_x ,
73 ordered_normalized_histogram_y):
74     reBUILT_Histo = []
75     for x_element in range(0, len(ordered_normalized_histogram_x), 1):
76         bin_size = int(round(ordered_normalized_histogram_y
77 [x_element] * 10))
78         for y_element in range(0, bin_size , 1):
79             reBUILT_Histo.append(ordered_normalized_histogram_x[x_element])
80     return reBUILT_Histo
81
82 def pbc_distance_matrix(coordinates , box):
83     import numpy as np
84     xdistance = 0.0
85     ydistance = 0.0
86     zdistance = 0.0
87     distance = 0.0
88     matrix_dimension = coordinates.shape[0]
89     sq_dist = 0.0
90     j=0
91     distance_matrix = [[0 for x in range(matrix_dimension)] for y
92 in range (matrix_dimension)]
93     for i in range (0, matrix_dimension , 1):
94         j=0
95         if i != j:
96             while j < i:
97                 xdistance = coordinates[i][0] - coordinates[j][0]
98                 if (abs(xdistance) > 0.5*box):
99                     xdistance -=box
100                 ydistance = coordinates[i][1] - coordinates[j][1]
101                 if (abs(ydistance) > 0.5*box):
102                     ydistance -=box
103                 zdistance = coordinates[i][2] - coordinates[j][2]

```

```

104 if (abs(zdistance) > 0.5*box):
105     zdistance -=box
106     sq_dist = xdistance**2 + ydistance**2 + zdistance**2
107     distance = np.sqrt(sq_dist)
108     distance_matrix[i][j] = distance
109     distance_matrix[j][i] = distance
110     j+=1
111 return distance_matrix

```

Codice B.2: test.py

```

1  #!/usr/bin/env python3
2  #!/usr/bin/python3
3  # Pluronic L64/Water mixture
4  # Evaluation of number/shape/dimension of PPO (L64) clusters
5  # for different timesteps.
6
7  from fnc import *
8
9  ## Variables Initialization
10 ppo = 9 # beads in a PPO chain chain
11 box = 30 # box size
12 Pconc = 0.1 # numerical Pluronic concentration
13 Wconc = 1-Pconc # numerical water concentration
14 n = 10 # water CG mapping number
15 frameskip = 2
16 flag_pbc = 1 # flag for periodic boundary conditions
17 flag_hist = 1 # flag for calculations
18 rho = 3 # system number density
19 frame_coll = 100 # step for the collection of results
20 interval = 250000 # how often bead positions appear
21 Pchain = 15 # beads in a Pluronic chain
22 timecounter = []
23 CheckCluster = []
24 check_status = 0
25
26 TotBeads = box*box*box*rho # beads in the system
27 PBeads = TotBeads*Pconc*ppo/Pchain # PPO beads in the system
28 WBeads = TotBeads*Wconc # water beads in the system
29 PMols = TotBeads*Pconc/Pchain # Pluronic molecules in the system
30 WMols = TotBeads*Wconc*n # water molecules in the system
31 TotMols = WMols + PBeads # total molecules in the system
32
33 ini = 20000000 # initial time step to display the results
34 fin = 40000000 # final time step to display the results
35 first = ini/frame_coll # first set of positions
36 last = fin/frame_coll + 1 # last set of positions
37
38 import sklearn
39 import numpy as np
40 from sklearn import cluster
41 from sklearn.cluster import DBSCAN
42 from sklearn import metrics
43 from sklearn.datasets.samples_generator import make_blobs
44 from sklearn.preprocessing import StandardScaler
45 import warnings
46 warnings.filterwarnings("ignore")
47 import matplotlib.pyplot as plt
48 from mpl_toolkits.mplot3d import Axes3D
49 import errno
50 import subprocess
51 import os

```

```

52 import math as mth
53 from math import sqrt
54 import traceback
55 import sys
56 import pylab as pl
57 from scipy.spatial.distance import pdist, squareform
58 import scipy
59 from collections import Counter
60 sys.tracebacklimit = None
61 filename = "cluster.xyz"
62 times = [TimeStep([],[],[],[],[],[],[])] for _ in range(50000)]
63 print ("I_am_collecting_coordinates_...")
64 coordinates = []
65
66 ## File opening (cluster.xyz) and storage of input files in vectors
67 xyz = open(filename, "r")
68 coordinate_counter = 0
69 for line in xyz:
70     try:
71         atom, x, y, z = line.split()
72         index = int (coordinate_counter / int(PBeads))
73         times[index].add_atom(atom, x, y, z)
74         coordinate_counter += 1
75     except ValueError:
76         pass
77 xyz.close()
78 box_norm = 0
79 clusterSize = []
80
81 ## Compute clustering algorithm (DBSCAN)
82 Ftot=np.zeros(100)
83 print ("I_am_creating_clusters_and_plotting_...")
84 for j in range (int(ini),int(fin),frameskip):
85     coordinates = np.column_stack((times[j].xCoord,times[j].yCoord,
86     times[j].zCoord))
87     coordinates_norm = coordinates/box
88     distance_matrix = [[0 for x in range(len(coordinates[0]))] for
89     y in range (len(coordinates[0]))]
90     if flag_pbc == 1:
91         box_norm = box / box
92         distance_matrix = pbc_distance_matrix(coordinates, box)
93         db = DBSCAN(eps = 1.2, min_samples=5,
94         metric='precomputed').fit(distance_matrix)
95         core_samples_mask = np.zeros_like(db.labels_, dtype=bool)
96         core_samples_mask[db.core_sample_indices_] = True
97         labels = db.labels_
98     else:
99         box_norm = box / box
100         db = DBSCAN(eps=2.2, metric='euclidean',
101         algorithm='auto', p=None).fit(coordinates)
102         core_samples_mask = np.zeros_like(db.labels_, dtype=bool)
103         core_samples_mask[db.core_sample_indices_] = True
104         labels = db.labels_
105
106     current_clusters = len(set(labels))
107     n_clusters_ = len(set(labels)) - (1 if -1 in labels else 0)
108     n_noise_ = list(labels).count(-1)
109     print('Timestep'+str((1+j)*frame_coll))
110     print('Estimated_number_of_clusters:%d' % n_clusters_)
111     print('Estimated_number_of_noise_points:%d' % n_noise_)
112
113

```

```

114 ## Results
115 import matplotlib.pyplot as plt
116 fig = plt.figure(figsize=(15,5))
117 if flag_hist == 1:
118 ax = fig.add_subplot(121, projection='3d')
119 bx = fig.add_subplot(122)
120 else:
121 ax = fig.add_subplot(111, projection='3d')
122 unique_labels = set(labels)
123 colors = plt.cm.Spectral(np.linspace(0, 1, len(unique_labels)))
124 for k, col in zip(unique_labels, colors):
125 if k == -1:
126 col = 'k'
127 class_member_mask = (labels == k)
128 xy = coordinates[class_member_mask]
129 times[j].beadHisto.append(len(xy[:,0]))
130 numberChain=len((xy[:,0])) / ppo
131 if times[j].beadHisto !=0:
132 approx=round(numberChain)
133 if numberChain > (approx + 0.4) and
134 numberChain < (approx + 0.5):
135 times[j].chainHisto.append(approx+1)
136 print('WARNING: ')
137 print('Required_adjustment_of_number_of_beads_per_chain ')
138 print('Correct_values_of_number_of_chain(s)_per_cluster ')
139 print(times[j].chainHisto)
140 else:
141 times[j].chainHisto.append(approx)
142 ax.plot(xy[:, 0], xy[:, 1], xy[:,2], 'o', markerfacecolor=col,
143 markeredgecolor='k', markersize=14)
144 check_status = check_status + (len(xy[:,0]))
145
146 if flag_hist == 1:
147 print('Number_of_beads_per_cluster ')
148 print(times[j].beadHisto)
149 if n_noise_ >= 1:
150 del times[j].chainHisto[n_clusters_]
151 print('Number_of_chain(s)_per_cluster_with_noise_>=1 ')
152 print(times[j].chainHisto)
153 else:
154 print('Number_of_chain(s)_per_cluster_with_noise_=_0 ')
155 print(times[j].chainHisto)
156
157 asd=bx.hist(times[j].chainHisto, bins=100,
158 range=(0,100),rwidth=1,histtype='bar')
159 asdlist=list(asd)
160 e=asdlist[0]
161 print('Frequency_of_appearance_of_N-dimensional_clusters ')
162 print(e)
163 a0=np.arange(100.0)
164 a1=Pchain*e*1*a0/TotBeads
165 a2=sum(a1)
166 print('Number_of_PPO_beads_in_the_system ')
167 a3=sum(times[j].beadHisto)
168 print(a3)
169 print('Number_of_polymer_chain(s)_in_the_system ')
170 a4=sum(times[j].chainHisto)
171 print(a4)
172 print('Molar_concentration_of_polymer_in_the_system ')
173 print(a2)
174 a5=np.arange(2)
175 a5.resize((100,))

```

```
176 if a3 == PMols + 1:
177     e=e-a5
178     print('One_monomer_is_eliminated:')
179     print(e)
180     else:
181         e=e
182         times[j].F.append(e)
183         Ftot=Ftot+times[j].F
184
185     CheckCluster.append(n_clusters_)
186     timecounter.append(j*frame_coll)
187     ax.set_ylim((0,box))
188     ax.set_xlim((0,box))
189     plt.title('Estimated_number_of_clusters:%d' % n_clusters_)
190     plt.savefig('timestep_'+str((1+j)*frame_coll)+'.png')
191     plt.close()
192     if (j >= (index-frameskip)):
193         fig_2 = plt.figure()
194         cx = fig_2.add_subplot(111)
195         cx.plot(timecounter, CheckCluster, 'o', markersize=6, ls='--')
196         plt.title('#Cluster_Vs_Time')
197         plt.xlabel('Timestep')
198         plt.ylabel('#Cluster')
199         plt.savefig('cluster_'+str(j*frame_coll)+'.png')
200         plt.close()
201
202     ## cluster mass distribution calculations
203     FFtot=Ftot[0]
204     NStep=1+((fin-ini)/frameskip/frame_coll)
205     var=Pchain/TotMols/NStep
206     N=np.arange(100.0)
207     XN=var*N
208     XXN=XN*FFtot
209     print('N')
210     print(N)
211     print('XN')
212     print(XXN)
213     dim=np.linspace(0., 100., 100)
214     plt.bar(dim, XXN, align='center', width=0.35, color='green')
215     plt.xlabel('dimension_(N)');
216     plt.ylabel('concentration_(XN)');
217     plt.title('cluster_mass_distribution')
218     plt.savefig('cmd')
219     plt.show()
220     plt.close()
221     print('End')
```

Bibliografía

- [1] P. Español and P.B. Warren. Perspective: Dissipative particle dynamics. *Journal of Chemical Physics*, 146:150901, (2017).
- [2] R.D. Groot and P.B. Warren. Dissipative particle dynamics: Bridging the gap between atomistic and mesoscopic simulation. *Imperial College Press*, 107:4423–4435, (1997).
- [3] E. Moeendarbary, T. Y. NG and M. Zangeneh. Dissipative Particle Dynamics: introduction, methodology and complex fluid applications - A Review. *Journal of chemical physics*, 1(4):737–763, (2009).
- [4] L.D. Landau and E.M. Lifshitz. *Fluid Mechanics*. Pergamon, 1959.
- [5] C. Hijón, P. Español, E. Vanden-Eijnden and R. Delgado-Buscalioni. Mori–zwanzig formalism as a practical computational tool. *Faraday Discuss.*, 144:301–322, (2010).
- [6] P.J. Hoogerbrugge and J.M.V.A. Koelman. Simulating microscopic hydrodynamic phenomena with dissipative particle dynamics. *Europhys. Lett.*, 19:155–160, (1992).
- [7] P. Español and P. Warren. Stochastic mechanics of dissipative particle dynamics. *Reviews in computational chemistry*, 30:191–196, (1995).
- [8] I.V. Pivkin and G.E. Karniadakis. Coarse-graining limits in open and wall-bounded dissipative particle dynamics systems. *Journal of Chemical Physics*, 124, (2006).
- [9] R.D. Groot and K.L. Rabone. Mesoscopic simulation of cell membrane damage, morphology change and rupture by nonionic surfactants. *Biophysical journal*, 81:725–736, (2001).
- [10] D.J. Diestler, S.H.L. Klapp and M. Schoen. Why are effective potentials “soft”. *Journal of Physics Condensed Matter*, 16:7331–7352, (2004).
- [11] G. Besold, J. M. Polson, I. Vattulainen and M. Karttunen. Integration schemes for dissipative particle dynamics simulations: From softly interacting systems towards hybrid models. *Journal of Chemical Physics*, 116:3967–3979, (2002).

-
- [12] M.P. Allen and F. Schmid. A thermostat for molecular dynamics of complex fluids. *Molecular simulation*, 33:21–26, (2007).
- [13] P. Español. Hydrodynamics from dissipative particle dynamics. *Physical Review*, 52:1743, (1995).
- [14] P.V. Coveney and P. Español. Dissipative particle dynamics for interacting multicomponent systems. *Journal of Physics A: Mathematical and General*, 30:779–784, (1997).
- [15] C.A. Marsh, G. Backx and M. H. Ernst. Static and dynamic properties of dissipative particle dynamics. *Phys. Rev. E*, 56:1676–1691, (1997).
- [16] H.C. Andersen, J.D. Weeks and D. Chandler. Relationship between the hard-sphere fluid and fluids with realistic repulsive forces. *Physical Review A*, 4:1597–1607, (1971).
- [17] P.J. Flory. *Principles of Chemistry Cap.12*. Cornell University Press, Itacha, NY, (1953).
- [18] C. Wijmans, B. Smit and R. Groot. Phase behavior of monomeric mixtures and polymer solutions with soft interaction potentials. *Journal of Chemical Physics*, 114:7644–7654, (2001).
- [19] K.E. Novik and P.V. Convey. Finite-difference methods for simulation models incorporating nonconservative forces. *Journal of Chemical Physics*, 109:7667–7677, (1998).
- [20] M.H.J. Hagen, I.Pagonabarraga and D.Frenkel. Self-consistent dissipative particle dynamics algorithm. *Europhys. Lett.*, 42:377–382, (1998).
- [21] T.Shardlow. Splitting for dissipative particle dynamics. *SIAM Journal on Scientific Computing*, 24:1267–1282, (2003).
- [22] G. Besold, O. G. Mouritsen and A. F. Jakobsen. Multiple time step update schemes for dissipative particle dynamics. *Journal of Chemical Physics*, 124:094104, (2006).
- [23] J.R. Partington, R.F. Hudson and K.W. Bagnall. Self-diffusion of aliphatic alcohols. *Nature*, 169:583–584, (1952).
- [24] R. Groot. Mesoscopic simulation of polymer-surfactant aggregation. *Langmuir*, 16:7493–7502, (2000).
- [25] S. Chen X.Wu X. Fan, N. Phan-Thien and T. Yong Ng. Simulating flow of dna suspension using dissipative particle dynamics. *Physics of Fluids*, 18:063102, (2006).

- [26] P. Español. Fluid particle dynamics: a synthesis of dissipative particle dynamics and smoothed particle dynamics. *Europhys.Lett.*, 39(6):605–610, (1997).
- [27] P. Español. Fluid particle model. *Physical Review*, 57:2930, (1998).
- [28] P. Español and M. Revenga. Smoothed dissipative particle dynamics. *Europhys.Lett.*, 67:026705, (2003).
- [29] I. V. Pivkin and G.E. Karniadakis W. Pan. Single-particle hydrodynamics in dpd: a new formulation. *Nature Materials*, 84:10012, (2008).
- [30] G. Jung and F. Schimid. Computing bulk and shear viscosities from simulations of fluids with dissipative and stochastic interactions. *Journal of Chemical Physics*, 144:204104, (2016).
- [31] M.S. Green. Markoff random processes and the statistical mechanics of time dependent phenomena. *Journal of Chemical Physics*, 20:1281–1295, (1952).
- [32] R. Kubo. Statistical-mechanical theory of irreversible processes. General theory and simple applications to magnetic and conduction problems. *J. Phys. Soc. Jpn.*, 12:570–586, (1957).
- [33] E. Helfand. Transport coefficients from dissipation in a canonical ensemble. *Physical Review*, 119:1, (1960).
- [34] A. W. Lees and A.F. Edwards. The computer study of transport processes under extreme conditions. *Journal of Chemical Physics*, 5, (1972).
- [35] H.C. J. Hoefsloot J.A. Backer, C.P. Lowe and P.D. Iedema. Poiseuille flow to measure the viscosity of particle model fluids. *Journal of Chemical Physics*, 122:154503, (2005).
- [36] F. Müller-Plathe. Reversing the perturbation in nonequilibrium molecular dynamics: An easy way to calculate the shear viscosity of fluids. *Physical Review E.*, 59:5, (1999).
- [37] C. Wesp, F. Reining A. El, Z. Xu, I. Bouras and C. Greiner. Calculation of shear viscosity using green-kubo relations within a parton cascade. *Phys. Rev. C*, 81:026707–11, (2011).
- [38] L. Onsager. Reciprocal relations in irreversible processes. *Physical Review*, 37:405–426, (1931).

- [39] A. Baldassarri, A. Puglisi, D. Villamaina and A. Vulpiani. Relazioni fluttuazione-dissipazione: una finestra sul mondo microscopico. *Lett. Math.*, (2009).
- [40] P. Carbone, P. Asinari, H. Droghetti, I. Pagonabarra and D. Marchisio. Dissipative particle dynamics simulations of tri-block co-polymer and water: Phase diagram validation and microstructure identification. *Journal of Chemical Physics*, 149:184903, (2018).
- [41] E. Allahyarov and G. Gompper. Mesoscopic solvent simulations: Multi-particle collision dynamics of three-dimensional flows. *Physical Review E*, 66:036702, (2002).
- [42] A. Grainer, D. Kauzlaric and J.G. Korvink. Technical proceedings of the 2004 nanotechnology conference and trade show. 2:454–457, (2004).
- [43] D. Frenkel and B. Smit. *Understanding Molecular Simulation, 2nd Edition*. Academic Press, 2001.
- [44] R.G. Larson. *The Structure and Rheology of Complex Fluids*. Oxford University Press, 1999.
- [45] A. Pitto-Barry and N. P. E. Barry. Pluronic block-copolymers in medicine: from chemical and biological versatility to rationalisation and clinical advances. *Polymer Chemistry*, 5(10):3281–3496, (2014).
- [46] P. Alexandridis and T.A. Hatton. Poly(ethylene oxide)-poly(propylene oxide)-poly(ethylene oxide) block copolymer surfactants in aqueous solutions and at interfaces: thermodynamics, structure, dynamics, and modeling. *Colloids and surfaces A*, 96:1–46, (1995).
- [47] P. Alexandrilis and B. Lindman. *Amphiphilic Block Copolymers - Self-Assembly and Applications*. Elsevier, 2000.
- [48] P. Alexandrilis. Amphiphilic copolymers and their applications. *Current Opinion in Colloid and Interface Science*, 1:490–501, (1996).
- [49] P. Alexandridis, J.F. Holzwarth and T.A. Hatton. Micellization of poly(ethylene oxide)-poly(propylene oxide)-poly(ethylene oxide) triblock copolymers in aqueous solutions: thermodynamics of copolymer association. *Macromolecules*, 27(9):2414–2415, (1993).
- [50] G. Wanka, H. Hoffmann and W. Ulbricht. Phase diagrams and aggregation behaviour of poly(oxyethylene)-poly(oxypropylene)-poly(oxyethylene) triblock copolymers in aqueous solutions. *Macromolecules*, 27:4145–4159, (1994).

- [51] P. Alexandrilis, D. Zhou and A. Khan. Lyotropic liquid crystallinity in amphiphilic block copolymers: Temperature effects on phase behavior and structure for poly(ethylene oxide)-b-poly(propylene oxide)-b-poly(ethylene oxide) copolymers of different composition. *Langmuir*, 12:2690–2700, (1996).
- [52] K. Ulrich, P. Galvosas, J. Kärger and F. Grinberg. “Pore-Like” Effects of Super-Molecular Self-Assembly on Molecular Diffusion of Poly(Ethylene Oxide)-Poly(Propylene Oxide)-Poly(Ethylene Oxide) in Water. *Materials*, 5:966–984, (2012).
- [53] R.B. Bird, W.E. Stewart and E.N. Lightfoot. *Transport Phenomena, 2nd Edition*. Wiley, 2002.
- [54] S. L. Guo, T. J. Hou and X. J. Xu. Simulation of the Phase Behavior of the (EO)₁₃(PO)₃₀(EO)₁₃(Pluronic L64)/Water/p-Xylene System Using MesoDyn. *Journal of Chemical Physics*, 106:11397–11403, (2002).
- [55] R.L. Anderson, D.J. Bray, A.S. Ferrante, M.G. Noro, I.Stott and P. B. Warren. Dissipative particle dynamics: Systematic parametrization using water-octanol partition coefficients. *Journal Chemical Physics*, 147:094503–12, (2017).
- [56] P. Durchschlag and H. Zipper. Calculation of the partial volume of organic compounds and polymers. *Prog. Colloid Polym. Sci.*, 94:20–39, (1994).
- [57] Y. Araki and N. Arai. Dissociation effect of non-covalent bond for morphological behavior of triblock copolymers: a dissipative particle dynamics study. *Chemical Engineering Communications*, 208(1):1–13, (2019).
- [58] T. L. Chantawansri, T. W. Sirk and Y. R. Silozberg. Entangled triblock copolymer gel: Morphological and mechanical properties. *Journal of Chemical Physics*, 138:024908, (2013).
- [59] G. Dorenbos. Coarse-grained study of the effect of hydrophobic side chain length on cluster size distributions and water diffusion in (amphiphilic-hydrophobic) multi-block co-polymer membranes. *Journal of Chemical Physics*, 138:024908, (2013).
- [60] R. Ketkaew and Y. Tantirungrotechai. Dissipative Particle Dynamics Study of SWCNT Reinforced Natural Rubber Composite System: An Important Role of Self-Avoiding Model on Mechanical Properties. *Macromolecular Theory and Simulations*, 1700093:1–10, (2018).

- [61] G. S. Jiao, Y. Li, Hu-Jun Quian and Jian Liu and Z-Y. Lu. Disperse cyclic diblock copolymer: another promising candidate for fabricating irregular bicontinuous structure. *Molecular Simulations*, 44(2):137–146, (2017).
- [62] T.W. Sirk, Y.R. Sliozberg, J.K. Brennan, M. Lisal and J.W. Andzelm. An enhanced entangled polymer model for dissipative particle dynamics. *Polymer*, 173:43–57, (2019).
- [63] A. Chaudhri and J.R. Lukes. Velocity and stress autocorrelation decay in isothermal dissipative particle dynamics. *Physical Review E*, 84:054911–17, (2010).
- [64] Calcolo viscosità. https://lammps.sandia.gov/doc/Howto_viscosity.html.
- [65] Calcolo SACF. https://lammps.sandia.gov/doc/fix_ave_correlate.html.
- [66] P. J. Daivis and D. J. Evans. Comparison of constant pressure and constant volume nonequilibrium simulations of sheared model decane. *Journal of Chemical Physics*, 100, (1994).
- [67] Algoritmo di Müller-Plathe. https://lammps.sandia.gov/doc/fix_viscosity.html.
- [68] Calcolo coefficiente di diffusione. https://lammps.sandia.gov/doc/Howto_diffusion.html.
- [69] M. Ester M., H. Kriegel, J. Sander and X. Xu. Comparison of constant pressure and constant volume nonequilibrium simulations of sheared model decane. *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, 226–231, (1996).
- [70] E. Helfand. Transport Coefficients from Dissipation in a Canonical Ensemble. *Physical Review*, 119(1):1–9, (1960).
- [71] G. Jung and F. Schmidt. Computing bulk and shear viscosities from simulations of fluids with dissipative and stochastic interactions. *Journal of Chemical Physics*, 144:204104, (2016).
- [72] R. Pasquino, H. Droghetti, P. Carbone, S. Mirzaagha, N. Grizzuti and D. Marchisio. An experimental rheological phase diagram of a tri-block co-polymer in water validated against dissipative particle dynamics simulations. *Soft Matter*, 6:1-9, (2019).
- [73] R. Sharma and K. Tankeshwar. Mutual diffusion in binary systems. *Journal of Chemical Physics*, 108(6):2601-2608, (1997).

Ringraziamenti

A conclusione di questo lavoro di tesi, desidero dedicare qualche riga alle persone che mi hanno guidato e sostenuto per la realizzazione dello stesso.

Un ringraziamento particolare al Prof. Daniele Marchisio per avermi dato la possibilità di svolgere la tesi in un contesto di ricerca così stimolante e formativo, per la completa disponibilità in tutte le fasi di realizzazione del progetto e per le conoscenze trasmesse. Un ringraziamento speciale anche al Prof. Antonio Buffo e al Prof. Gianluca Boccardo per le preziose indicazioni, per i suggerimenti, per il sostegno costante e la disponibilità durante tutto il corso di preparazione della tesi. Vorrei, inoltre, ringraziare il Prof. Martin Lisal i cui suggerimenti sono stati importantissimi per l'attività di ricerca svolta in questo lavoro di tesi.

Un grazie infinito ai miei genitori, Franco e Graziella, e a mia sorella Lorella per non avermi mai fatto mancare l'affetto, la fiducia e il sostegno in questo percorso di studi e aver sempre appoggiato le mie scelte. Un grazie immenso a miei zii, Sonia e Peppe, per la loro presenza e il loro supporto speciale e insostituibile in ogni momento di questo percorso. Un grazie di cuore a mia nonna Nunzia per la forza d'animo che mi ha sempre trasmesso.

