



**Politecnico
di Torino**

ScuDo

Scuola di Dottorato - Doctoral School

WHAT YOU ARE, TAKES YOU FAR

Doctoral Dissertation

Doctoral Program in Chemical Engineering (37th cycle)

**Advancing Dissipative Particle
Dynamics as a Computational Tool
for Structured Fluids Rheology
From Schmidt Number Calibration to
Temperature-Dependent Transport Properties**

By

Nunzia Lauriello

Supervisor(s):

Prof. Gianluca Boccardo, Supervisor

Prof. Daniele Marchisio, Co-Supervisor

Prof. Antonio Buffo, Co-Supervisor

Doctoral Examination Committee:

Prof. Paola Carbone, University of Manchester Institute of Science and Technology

Prof. Nicodemo Di Pasquale, Università di Bologna

Prof. Salvalaglio Matteo, University College London

Prof. Striolo Alberto, University College London

Politecnico di Torino
2025

Declaration

I hereby declare that, the contents and organization of this dissertation constitute my own original work and does not compromise in any way the rights of third parties, including those relating to the security of personal data.

Nunzia Lauriello
2025

* This dissertation is presented in partial fulfillment of the requirements for **Ph.D. degree** in the Graduate School of Politecnico di Torino (ScuDo).

I would like to dedicate this thesis to my loving parents and my dear sister.

Acknowledgements

I am deeply grateful to all those who supported me throughout the process of completing this thesis.

The work presented in Chapter 6 was carried out during my time at the Institute of Chemical Process Fundamentals of the CAS, where I was hosted by the Research Group of Molecular and Mesoscopic Modelling led by Prof. Martin Lìsal. I would like to thank Prof. Lìsal for his invaluable guidance and support.

The work presented in Chapter 7 was carried out in collaboration with “Università degli studi Federico II”. I am grateful to Prof. Rossana Pasquino, Prof. Nino Grizzuti, and Nicola Antonio di Spirito for their insightful discussions.

The work presented in Chapter 8 was conducted during my visit at the Department of Mechanical Engineering at the University of Wisconsin-Madison, where I was hosted by the Research Group of Prof. Wenxiao Pan. I wish to thank Prof. Pan for his guidance and collaboration.

I am sincerely grateful to my supervisors for their mentorship.

Finally, I would like to thank James Larentzos and Karel Šindelka for their valuable contributions to our discussions.

Computational resources were provided by HPC@POLITO, and we acknowledge the CINECA award under the ISCRA initiative for granting access to high-performance computing resources and technical support. The financial support from ICSC (Centro Nazionale di Ricerca in High Performance Computing, Big Data, and Quantum Computing, funded by the European Union - NextGenerationEU) is also gratefully acknowledged. Part of this research was carried out within the project "Non-equilibrium self-assembly of structured fluids: a multi-scale engineering problem" funded by the European Union under the Next Generation EU initiative within

the PRIN 2022 program (D.D. 104 - 02/02/2022 Ministero dell'Università e della Ricerca).

Abstract

Structured fluids, such as polymer solutions and melts, colloidal suspensions, micellar solutions, and liquid foams, are extensively used in the food, personal care, and pharmaceutical industries. These materials exhibit unique rheological behaviors based on their internal microstructures, which has generated significant commercial and scientific interest. A notable class of amphiphilic materials, known commercially as Pluronics, has gained increased attention in recent years due to its extensive range of over fifty polymer-based polymorphic materials. Pluronics are tri-block copolymers, consisting of two hydrophilic polyethylene oxide (PEO) chains and a central hydrophobic polypropylene oxide (PPO) chain. Their customizable molecular characteristics, including PPO/PEO ratio, block lengths, and molecular weight, enable the creation of tailored macromolecules for specific applications. Depending on factors such as concentration, temperature, and shear stresses, these materials can exist in aqueous solutions as individual macromolecules (unimers) or self-assemble into various microstructures, which influence their ultimate properties. Predicting and controlling these properties is crucial for optimizing performance across diverse applications and for designing and scaling up processing equipment. Computational tools, particularly Dissipative Particle Dynamics (DPD), offer an efficient alternative to labor-intensive experimental methods, enabling rapid and cost-effective exploration of structural changes and their effects. DPD, a mesoscopic technique, bridges the gap between micro-scale and macro-scale models, operating at time and length scales inaccessible to molecular-level methods.

This dissertation explores the use of DPD simulation techniques for studying both static and dynamic properties in simple and structured fluids. The primary goal is to enhance the effectiveness of these techniques for reliable virtual experiments on technologically significant materials, including both simple and structured fluids. Two main systems are considered: liquid water and binary mixtures of Pluronics and water. The standard DPD model, originally designed for simulating molecular

liquids, requires extensions and variations for specific applications. Key challenges include improving accuracy in predicting fluid transport properties and rheology, and developing robust parameterization strategies.

The work is divided into methodological development and application sections. The first section addresses challenges in simulating high Schmidt number fluids and improving the reliability and computational feasibility of predicting transport properties. An "extended" DPD model suitable for high Schmidt number fluids was implemented into the open-source LAMMPS code and extensively tested. This model has been shown to convert gas-like dynamics, described by the standard DPD model, to liquid-like dynamics. The time step size dependency of dynamic properties and the effectiveness of different time integration schemes were investigated. The Shardlow-splitting algorithm, newly applied to the "extended" DPD model, demonstrated superior performance compared to the modified velocity-Verlet algorithm for high Schmidt number fluids. An automated method to compute transport properties from DPD equilibrium simulations was developed, focusing on the reliability and feasibility of the Green-Kubo approach for a wide range of simple DPD systems with varying Schmidt numbers. These findings facilitated a multi-parametric study to systematically explore the relationship between Schmidt number and model parameters. The second section focuses on parameterization challenges in simulating real systems. The initial case study involves modeling liquid water at various temperatures using a new temperature-dependent version of the extended DPD model. This model improves upon the standard version, accurately predicting both equilibrium and dynamic transport properties, and was validated against experimental data across the entire liquid temperature range. The study also examines DPD-based virtual experiments on Pluronic systems in water, comparing results with real-world data to understand the relationship between morphology and properties. Two distinct applications are proposed, each involving a different type of Pluronic. Additionally, a data-driven approach for parameterizing Pluronic systems under various conditions is introduced.

Contents

List of Figures	xiii
List of Tables	xxi
Nomenclature	xxiv
1 Introduction	1
1.1 Structured fluids in chemical engineering	2
1.2 Computational modeling by Dissipative Particle Dynamics	4
1.3 Thesis outline	9
2 Theoretical background	13
2.1 Poloxamers: the interplay between morphology and rheology	13
2.2 Dissipative particle dynamics: a mesoscopic technique	18
2.2.1 The mathematical formulation: governing equations and reduced units	19
2.2.2 Interaction parameters	20
2.2.3 Time integration schemes	24
2.3 Gaining insights from DPD simulations	26
2.3.1 Transport properties evaluation	26
2.3.2 Static properties evaluation	30

2.3.3	Real-space re-normalization of static and transport properties	32
2.4	Key issues in DPD	34
2.4.1	Simulating the dynamics of mesoscopic systems	34
2.4.2	The parameterization problem	35
3	Simulation of high Schmidt number fluids with DPD	37
3.1	An extended DPD model: from gas-like dynamics to liquid like dynamics	41
3.2	Simulation details	42
3.3	Results and discussion	43
3.3.1	Robust viscosity evaluation procedure	43
3.3.2	Schmidt number scaling	47
3.3.3	Tuning water transport properties	51
3.4	Conclusions	54
3.5	Critical Analysis and Next Steps	55
4	Time Integration Schemes for Fluids with High Schmidt Numbers	57
4.1	VV-SSA implementation for the extended DPD model	58
4.2	VV-SSA performance in transport properties calculation	62
4.2.1	Simulation details	62
4.2.2	Results and discussion	62
4.3	Conclusions	66
5	Automated method for transport properties calculation	68
5.1	Description of the method	72
5.2	Computational details	77
5.2.1	Simulation setup details	77
5.2.2	Parametric study	77

5.3	Results and discussion	78
5.3.1	Automated transport properties calculation	78
5.3.2	Schmidt number scaling with effective friction coefficient	83
5.4	Conclusions	85
6	Temperature-Dependent Transport Properties in DPD	87
6.1	Temperature dependent-transport properties in DPD simulations	89
6.2	Interaction parameters	93
6.3	Results and discussion	97
6.4	Conclusions	103
7	Modeling Pluronic systems by DPD	106
7.1	Rheology and Morphology of L64/Water Mixtures	107
7.1.1	Model Description, Methodology and Computational details	108
7.1.2	Results and discussion	111
7.2	Rheology and Morphology of F68/Water Mixtures	114
7.2.1	Model Description, Methodology and Computational details	116
7.2.2	Results and discussion	120
7.2.3	Morphology	120
7.2.4	Rheology	125
7.3	Conclusions	126
8	ML parameterization methodology for Pluronics in DPD	131
8.1	Data-driven approach: methods and computational details	136
8.1.1	DPD simulations: setup and post-processing	136
8.1.2	Dataset creation	139
8.1.3	GPR as surrogate model	140
8.1.4	Grid-search for parameters optimisation	142

8.2	Results and discussion	143
8.2.1	Performance of the GPR	143
8.2.2	Optimization results	153
8.3	Conclusions and future steps	158
8.3.1	Transfer learning	158
9	Conclusions	162
9.1	Overview of the key contributions of this research	162
9.2	Conclusions summary	164
	References	169
	Appendix A Appendix A	175
A.1	Schmidt number scaling with effective friction coefficient	175
	Appendix B Appendix B	178
B.1	Error Estimation in Viscosity Calculations	178
B.2	Assessing the reliability of self-diffusivity calculations	180

List of Figures

1.1	Structured fluids in chemical engineering.	3
1.2	The mesoscale gap. This figure is reproduced from [1].	5
2.1	General structure of a Pluronic® molecule (image from Ref. [2]). . .	13
2.2	Pluronic® grid (image from Ref. [3]). Color code: physical state of copolymers under ambient conditions: green = liquid; blue = paste; pink = flake. Vertical lines (bottom to top) show increasing molecular weight of the PPO block, while horizontal lines (left to right) show increasing EO percentage by mass. The numbers in the boxes refer to Pluronic L64.	14
2.3	Schematic representation of micellar (a), cubic (b), hexagonal (c), and lamellar structures formed by Pluronic in aqueous solvent (image adapted from Ref. [4]).	16
2.4	Phase diagram of the Pluronic L64-water binary mixture (image adapted from Ref. [5]). L ₁ : micellar phase. E: hexagonal phase. D: lamellar phase. L ₂ : inverse micellar phase.	17
2.5	Schematic representation of the RNEMD method. Velocity field is represented by red arrows (image adapted from Ref. [6]).	29
3.1	Two different representations of the tail of the same SACF. Top: procedure A. Black circles: points of the SACF. Red line: power-law fit. Bottom: procedure B. Black line: SACF after a moving average filter. Red line: power-law fit.. . . .	44

3.2	Cumulative integral of viscosity calculated using the Green-Kubo approach. Black line: viscosity cumulative integral. Red line: mean value and coefficient of variation calculated between $t = 5$ and 10.	46
3.3	Effect of the degree of CGing, N_m , on the SACF for two different values of the GWF exponent s	49
3.4	Effects of the (left) dissipative parameter γ , (center) GWF exponent s and (right) dissipative cutoff radius r_c^D on the SACF.	50
3.5	The Schmidt number as a function of (left) dissipative parameter γ , (middle) GWF exponent s and (right) dissipative cutoff radius r_c^D . Dependencies of the Schmidt number on the DPD parameters were fit to the power-law.	50
3.6	The Schmidt number as a function of the degree of CGing, N_m , for different values of the GWF exponent s	50
3.7	Radial distribution function evaluated for the three sets of DPD parameters, calibrated against the Schmidt number of liquid water at 25°C equal to about 370. Black line: Set 1, red line: Set 2, blue line: Set 3.	53
4.1	SACF and CVI for VV at $\gamma_{\text{eff}} = 1.09$	64
4.2	SACF and CVI for VV-SSA at $\gamma_{\text{eff}} = 1.09$	64
4.3	SACF and CVI for VV at $\gamma_{\text{eff}} = 20.87$	65
4.4	SACF and CVI for VV-SSA at $\gamma_{\text{eff}} = 20.87$	65
4.5	Comparison of RDF for VV and VV-SSA integrators at different values of γ_{eff} ; $\gamma_{\text{eff}} = 1.09$ (left), $\gamma_{\text{eff}} = 20.87$ (right).	66
5.1	Workflow of the automated method for viscosity calculation from DPD simulation with the Green-Kubo approach.	75
5.2	N_{rep} -loop: SACF curves produced for different values of N_{rep} (left) and the correspondent CVI curves (right). The indicators in the SACFs are the viscosity UI points for different values of N_{rep} . The N_{rep} -loop stops at $N_{\text{rep}} = 3000$ and identifies as UI point the blue one.	79

5.3	N_{sim} -loop: SACF curves produced for different values of N_{sim} (left) and the correspondent CVI curves (right). The N_{sim} -loop stops at $N_{\text{sim}} = 700$	80
5.4	Trend of the mean viscosity value, η_{avg} , and the corresponding coefficient of variation, COV, increasing N_{sim} . Blue lines indicate the values of η_{fin} and COV_{fin} identified by the algorithm.	80
5.5	Δt -loop: SACF curves produced for the different time step sizes (left) and the correspondent values of η_{int} computed (right). The Δt -loop stops at $\Delta t = 0.003$ and returns the value of η_{fin} which is the blue one.	81
5.6	Time step size effect on the RDF (left) and on ε (right). The blue RDF curve (left) corresponds is measured with the Δt value identified by the Δ -loop and the blue point (right) is the ε associated, which is the lowest one.	82
5.7	Linear fitting on the viscosity values (left), power-law fitting on the diffusivity values (middle) and the polynomial fitting on the Schmidt number values (right) as a function of the effective friction coefficient parameter. The red lines are to fit the simulation results (black symbols).	85
6.1	Theoretically predicted behavior of the kinematic viscosity ν (left), the self-diffusivity $\mathcal{D}$ (center) and the Schmidt number Sc (right) of a DPD fluid over the system temperature θ . The SWF curves refer to standard DPD model including standard dissipative quadratic weighting function. The GWF curves refer to standard DPD model including generalized dissipative weighting function.	91
6.2	Experimental behavior of the kinematic viscosity ν (left), the self-diffusivity $\mathcal{D}$ (center) and the Schmidt number Sc (right) of liquid water over temperature T	92

6.3	Comparison between the behavior of repulsive parameters a over temperature θ obtained by assuming constant isothermal compressibility κ_{T_R} and by considering temperature-dependent isothermal compressibility values κ_T (left). Error estimates on a due to neglecting temperature variation of the isothermal compressibility (right).	94
6.4	Comparison of transport properties at different temperatures between experimental data and simulation results with temperature-dependent conservative and non-conservative interaction parameters. The kinematic viscosity and self-diffusivity are scaled by their reference values $\nu_R = 8.926 \times 10^{-7} \text{m}^2/\text{s}$ and $\mathcal{D}_R = 2.299 \times 10^{-9} \text{m}^2/\text{s}$ at $T_R = 298.15 \text{K}$. Errors in ν and $\mathcal{D}$ are smaller than points in the viscosity plots	95
6.5	The s power coefficient as a function of temperature θ which provides the correct Sc number of liquid water at different real temperatures $T = T_R \times \theta$. The fitting of those data gives s as a function of θ (left). Dissipative weighting functions at different temperatures $T = T_R \times \theta$ (right).	98
6.6	Comparison of the transport properties at different temperatures between experimental data and the results of DPD simulations with the temperature-dependent conservative interaction parameter a and the temperature-dependent non-conservative interaction parameter s . The kinematic viscosity and diffusivity are scaled by their reference values at ν_R and $\mathcal{D}_R$ at $T_R = 25^\circ\text{C}$.	101
6.7	Kinematic viscosity ν obtained from DPD simulations at different temperatures θ splitted into ν^{CC} and $\nu - \nu^{CC}$ contributions (left). Comparison between $\nu - \nu^{CC}$ simulated and the theoretical predictions of Fig. B.1 (right).	102
6.8	Radial distribution function for each temperature investigated $T = T_R \times \theta$ (left). Effective friction coefficient γ_{eff} obtained at different temperatures (right).	103
7.1	Pluronic L64 and water molecules (left); coarse-grained model used in DPD (center); lamellar microstructure from DPD equilibrium simulation (right).	107

7.2	Structural formula of the Pluronic L64 molecule (image adapted from Ref. [7]).	108
7.3	DPD model resulting from the coarse-graining procedure of a Pluronic L64 molecule.	108
7.4	Conservative interactions between the different types of beads in the system: A (PEO), B (PPO), and C (H ₂ O).	110
7.5	Concentration-induced phase transition analyzed through shape descriptors. This figure illustrates the relationship between eigenvalues and varying concentrations, highlighting the detection of phase transitions induced by concentration changes.	112
7.6	Concentration-induced phase transition analyzed through shape descriptors. This figure presents the relationship between the ratios of eigenvalues and varying concentrations, emphasizing the detection of phase transitions induced by changes in concentration.	113
7.7	Complex viscosity, evaluated at 1 rad s ⁻¹ , plotted against Pluronic L64 weight concentration. Predicted microstructures using DPD at various concentrations are illustrated alongside. The figure also includes a comparison with DPD zero-shear viscosity at different concentrations within the micellar range.	114
7.8	Pluronic F68 water solution plotted at various temperatures. A zoomed-in view at 100 rad/s is provided, highlighting the transition from a liquid-like to a solid-like phase with increasing temperature. Additionally, the figure includes a snapshot of the simulated body-centered cubic (BCC) lattice.	115
7.9	Comparison of the computed core radius and the experimental core radius of Pluronic F68 at 25°C.	116
7.10	Comparison of the computed micellar radius and the experimental micellar radius of Pluronic F68 at 25°C.	117

7.11	Example of topological violation for the nonphysical crossing of bonds (green line and black line) between the beads of the polymer chain represented by the filled circles, known as bond crossing (image processing extracted from Ref. [8]). The minimum distance vector (in red) changes its orientation by 180° between the time instants t and $t + \Delta t$. Configuration (a) is relative to the time instant t . Configuration (b) is relative to the time instant $t + \Delta t$	118
7.12	Illustration of the possible mechanisms behind elasticity: ordered domains and entanglements.	121
7.13	Aggregation number as a function of temperature.	121
7.14	Number of aggregates (sub-micellar and micellar) as a function of temperature.	122
7.15	Weight-averaged distribution function at $T=5^\circ\text{C}$	123
7.16	Weight-averaged distribution function at $T=25^\circ\text{C}$	123
7.17	Weight-averaged distribution function at $T=28^\circ\text{C}$	124
7.18	Weight-averaged distribution function at $T=40^\circ\text{C}$	124
7.19	Snapshots at different temperatures.	125
7.20	Experimental measurements of micellar and PPO core radii across a range of temperatures.	126
7.21	Comparison of experimental and computed micellar radius measurements across a range of temperatures.	126
7.22	Comparison of experimental and computed PPO core radius measurements across a range of temperatures.	126
7.23	Comparison of experimental and computed PEO shell radius measurements across a range of temperatures.	126
7.24	Radial distribution functions (RDFs) of PPO-PPO interactions at varying temperatures.	127
7.25	Radial distribution functions (RDFs) of PPO-Water interactions at varying temperatures.	128

7.26	Radial distribution functions (RDFs) of PEO-Water interactions at varying temperatures.	128
7.27	Radial distribution functions (RDFs) of PEO-PEO interactions at varying temperatures.	129
7.28	Experimental measurements of the complex modulus across different temperatures.	130
7.29	Computed values of the complex modulus in DPD units across different temperatures.	130
7.30	Computed values of the elastic modulus in DPD units across different temperatures.	130
7.31	Computed values of the loss modulus in DPD units across different temperatures.	130
8.1	Typical trial-and-error parameters identification workflow.	135
8.2	GPR surrogate model predictions for CMC based on DPD model input parameters a_{AW} , a_{BW}	143
8.3	GPR surrogate model predictions of CMC based on a_{BW} at fixed $a_{AW} = 27.346$	144
8.4	GPR surrogate model predictions of CMC based on a_{AW} at fixed $a_{BW} = 36.046$	145
8.5	Parity diagram of the CMC on the test set for the GPR.	146
8.6	GPR surrogate model predictions for R_g based on DPD model input parameters a_{AW} , a_{BW}	147
8.7	GPR surrogate model predictions of R_g based on a_{BW} at fixed $a_{AW} = 27.346$	148
8.8	GPR surrogate model predictions of R_g based on a_{AW} at fixed $a_{BW} = 36.046$	149
8.9	Parity diagram of the R_g on the test set for the GPR.	149
8.10	GPR surrogate model predictions for A_s based on DPD model input parameters a_{AW} , a_{BW}	150

8.11 GPR surrogate model predictions of A_s based on a_{BW} at fixed $a_{AW} = 27.346$	151
8.12 GPR surrogate model predictions of A_s based on a_{AW} at fixed $a_{BW} = 36.046$	152
8.13 Parity diagram of the A_s on the test set for the GPR.	152
8.14 Adaptive grid search for optimizing L64 parameters, showing results at different step sizes.	154
8.15 Adaptive grid search for optimizing F68 parameters, showing results at different step sizes.	155
8.16 Optimized DPD model parameters for the considered Pluronics systems, defined by PO molecular weight and EO content.	156
8.17 Comparison of experimental targets and computed results obtained through optimized parameters, for the Critical Micellar Concentration (CMC), radii of spherical aggregates (R_g), and aggregation numbers (A_s) of the considered Pluronics at 30°C and low concentrations.	157
B.1 Self-diffusion coefficient of water $\mathcal{D}^{\text{PBC}}$ at ambient temperature as a function of the inverse box length $1/L$ with different number of water beads $N = 5,184 - 192,000$. The black line is a linear fit of the simulated self-diffusion coefficients, $\mathcal{D}^{\text{PBC}}$, to $1/L$. The linear fit provides "true" self-diffusion coefficient $\mathcal{D}^0 = 0.0161$	180

List of Tables

3.1	Viscosity values (in DPD unit) calculated with various approaches. Procedure A: power-law fitting. Procedure B: moving average filter before power-law fitting. The superscripts 1 and 2 correspond respectively to a final integration time of $N_{rep} \times \Delta t$ and infinite. Procedure C: cumulative integral. E-H and RNEMD refers to Einstein-Helfand and Müller-Plathe methods.	45
3.2	Summary of all DPD simulation performed, including the results in terms of viscosity, self-diffusion coefficient and Schmidt number; all properties are in DPD units. All simulations are carried out with $\rho = 3$. Rows highlighted in bold font are close to the Schmidt number of water at 25°C equal to about 370.	48
3.3	The properties of DPD fluid corresponding to the three sets of DPD parameters, calibrated against the Schmidt number of liquid water at 25°C equal to about 370.	52
4.1	Outline of the modified VV integration scheme (see text for acronym) to update momentum and position of each i -particle ($\mathbf{p}_i^s \rightarrow \mathbf{p}_i^{s+1}; \mathbf{r}_i^s \rightarrow \mathbf{r}_i^{s+1}$).	59
4.2	The modified VV integration scheme (see text for acronym) with detailed integration steps listed. [9, 10].	59
4.3	Outline of the VV-SSA integration scheme to update momentum and position of each i -particle ($\mathbf{p}_i^s \rightarrow \mathbf{p}_i^{s+1}; \rightarrow \mathbf{r}_i^s \rightarrow \mathbf{r}_i^{s+1}$) (see text for acronym).	60

4.4	The VV-SSA integration scheme for transverse DPD thermostat (see text for acronym) with detailed integration steps listed [11, 12].	61
4.5	Comparison between VV and VV-SSA performance.	66
5.1	Time step size effect on the system temperature.	82
5.2	Goodness-of-fit measures from the fitting on the effective friction coefficient, the viscosity, the diffusivity the Schmidt number values.	84
6.1	Main differences among the weighting functions labelled by SWF, GWF and GWF^D	90
6.2	Re-elaboration of theoretical expressions for the transport properties from literature to highlight their relationship with system temperature θ . The expressions derived in the case of the standard DPD model including the standard dissipative quadratic weighting function [13] (SWF) are compared with the ones obtained in the case of the standard DPD model including a generalized weighting function [14] (GWF^D).	91
6.3	Target experimental values in real units and scaled with respect to the reference values. The reference values for the temperature T , the kinematic viscosity ν , and the self-diffusivity $\mathcal{D}$ are highlighted.	92
6.4	Targeted system temperature θ , the average kinetic temperature θ_{avg} and associated standard deviation θ_{std} for each temperature T investigated.	98
6.5	Values of the non-conservative parameter, s , providing an excellent match between the experimental Schmidt number, Sc_{exp} , and the DPD Schmidt number, Sc_{DPD} , for each temperature, T , as evident from the relative error ε_{Sc}	100
6.6	Goodness-of-fit measures from the fitting of the s values and the effective friction coefficient values γ_{eff}	103
7.1	Coarse-grained beads, molecular volumes of H_2O , EO, and PO calculated according to the approach by Durchschlag and Zipper [15], and the corresponding bead volumes expressed in $\text{\AA}^3$	109

7.2	Interaction parameters [16], reported in DPD units, for the three species: PEO, PPO, and water.	109
8.1	Coarse-Grained Representation of the analyzed Pluronic systems . .	137
8.2	Conservative interaction parameters for the three species: PEO, PPO, and water.	138
8.3	Physiochemical properties of the considered Pluronics [17]	139
8.4	Experimental values for the Critical Micellar Concentration (CMC), radii of spherical aggregates (R_g), and aggregation numbers (A_s) of the considered Pluronics at 30°C and low concentrations.	142
8.5	Optimized DPD model parameters for the considered Pluronics. . .	156
A.1	Summary of all DPD simulation performed, including the results in terms of viscosity, self-diffusion coefficient and Schmidt number; all properties are in DPD units. All simulations are carried out with $a = 25$ and $\rho = 3$	175
B.1	Error estimates in the viscosity calculations across different temperatures considered in the work. η_{avg} is the mean value of viscosity, η_{std} and COV are the associated standard deviation and coefficient of variation, respectively.	179
B.2	Impact of the finite-size correction on self-diffusivity calculations. .	181
B.3	Impact of the corrected self-diffusivity values on the final Sc number.	181

Nomenclature

Acronyms / Abbreviations

^2H -NMR Deuterium Nuclear Magnetic Resonance Spectroscopy

BCC Body-Centered Cubic

CFD Computational Fluid Dynamics

CG Coarse-Grained

CGing Coarse-Graining

CMC Critical Micellar Concentration

CMT Critical Micellar Temperature

CVI Cumulative Viscosity Integral

DPD Dissipative Particle Dynamics

E-H Einstein-Helfand

EO Ethylene Oxide

EoM Equation of Motion

EoS Equation-of-State

FCC Face-Centered Cubic

FDT Fluctuation-Dissipation Theorem

FG Fine-Grained

G-K	Green-Kubo
GLE	Generalized Langevin Equation
GWF	Generalized Weighting Function
MB-PMF	Many-Body Potential of Mean Force
MD	Molecular Dynamics
MSD	Mean Square Displacement
NEMD	Non-Equilibrium Molecular Dynamics
PEO	Poly(Ethylene Oxide)
PO	Propylene Oxide
PPO	Poly(Propylene Oxide)
RDF	Radial Distribution Function
RNEMD	Müller-Plathe
SACF	Stress Auto-Correlation Function
SAXS	Small-Angle X-ray Scattering
VACF	Velocity Auto-Correlation Function
VV	Velocity-Verlet algorithm
VV-SSA	Shardlow-splitting algorithm

Chapter 1

Introduction

Structured fluids and soft matter systems are fundamental in a wide range of technologies and biological processes. Their unique behaviors and tunable properties make them essential in fields such as drug delivery, food technology, cosmetics, and biomaterials. Given their significance and the intellectual challenges they pose from both theoretical and experimental perspectives, computational modeling research in this area has expanded rapidly in recent decades. The vast variations in relevant time and length scales further complicate computational studies. Atomistic simulations have proven invaluable for investigating soft matter, with applications in drug transport, nanoscale material self-organization, and rheological properties. However, there is a strong demand for developing approaches that can accurately capture these behaviors, which requires significant efforts in the development of Coarse-Grained (CG) techniques, novel simulation methods, and strategies for bridging different regimes. Dissipative Particle Dynamics (DPD) is among the most promising techniques for achieving time and length scales beyond those accessible with atomistic simulations. Although DPD has been used successfully in numerous applications, certain limitations still constrain its use in real-world engineering scenarios. Given these challenges, the use of DPD to simulate the behavior of structured fluids and, more broadly, soft-matter systems represents an exciting and evolving area of research with significant potential for advancing the field and addressing real-world engineering challenges. This work is positioned within this dynamic and expanding domain. This introductory chapter begins with Section 1.1, which explores the aspects of structured fluids within chemical engineering. Section 1.2 then outlines the

key challenges related to the original DPD formulation when applied to structured fluids and real systems. Lastly, Section 1.3 presents the outline of the thesis.

1.1 Structured fluids in chemical engineering

Structured fluids, which include polymer solutions and melts, colloidal suspensions, micellar solutions, and liquid foams, are widely used in the food, personal care, and pharmaceutical industries. They also serve as intermediates in the production of fibers, films, packaging, and other products. These materials exhibit unique properties that defy traditional concepts of solidity and liquidity, behaving like solids or liquids depending on the time scale or the level of applied stress. As a part of the broader class of soft matter systems, structured fluids are finding an increasing number of applications across various industries. The term "soft" highlights that even relatively weak external forces, such as mechanical stresses, are sufficient to induce significant structural deformations and alter the dynamic properties of these materials. This sensitivity provides valuable opportunities to manipulate the system across a wide range of scales, from the nanoscale to the macroscopic level. This unique behavior has generated significant commercial and scientific interest in structured fluids, and more in general in soft matter systems. Block copolymers are a notable example within this class of materials; under appropriate conditions, their distinct blocks can undergo microphase separation, forming microstructured materials with one or more types of domains. Consequently, block copolymers self-assemble into composite materials whose properties can be finely tuned by varying the types of blocks, their volume fractions, molecular weights, and other parameters. Among them a peculiar class of amphiphilic materials, commercially known as Pluronics, during the last years has received increased attention, since it offers a pool of more than fifty polymer-based polymorphic materials. Pluronics are tri-block co-polymers, composed by two lateral hydrophilic chains of Poly(Ethylene Oxide) (PEO) and a central hydrophobic chain of Poly(Propylene Oxide) (PPO). The ability to precisely control the molecular characteristics of Pluronics, such as the PPO/PEO composition ratio, block lengths, and molecular weight, offers a high degree of tunability, allowing for the creation of macromolecules with tailored properties to meet specific application requirements. They are commonly referred to as an alphanumeric name made of up a letter, indicating the physical status, and

two digits, which refer to the molecular mass and composition. Such systems find widespread industrial use along with more specialized applications in pharmaceuticals and bioprocessing. Thanks to their biocompatibility, they are prominently featured in the development of nanocarriers for drug applications. Depending on factors like concentration, temperature, and shear stresses, these materials can exist in aqueous solutions either as individual macromolecules, unimers, or self-assemble into micelles once they surpass the Critical Micellar Concentration (CMC) and the Critical Micellar Temperature (CMT). They can evolve in various microstructures, which in turn govern the ultimate material properties they exhibit. The properties of these materials, including the CMC and viscosity, are pivotal factors in determining their potential applications. The CMC represents the concentration threshold at which self-assembly into micelles occurs, impacting various functionalities such as drug delivery. Meanwhile, their rheological properties often dictate the uses to which they can put; in fact the choice of a particular type of Pluronic for a given application is often driven by the rheological properties that are needed in that application. This is certainly true for foods or personal care products, where the rheological properties can determine the success or failure of the product. Even when rheological properties are not critical in the final product, they often influence the processability of the material and their suitability for various processes, dictating flow behavior and stability in formulations and manufacturing. Thus, predicting and controlling

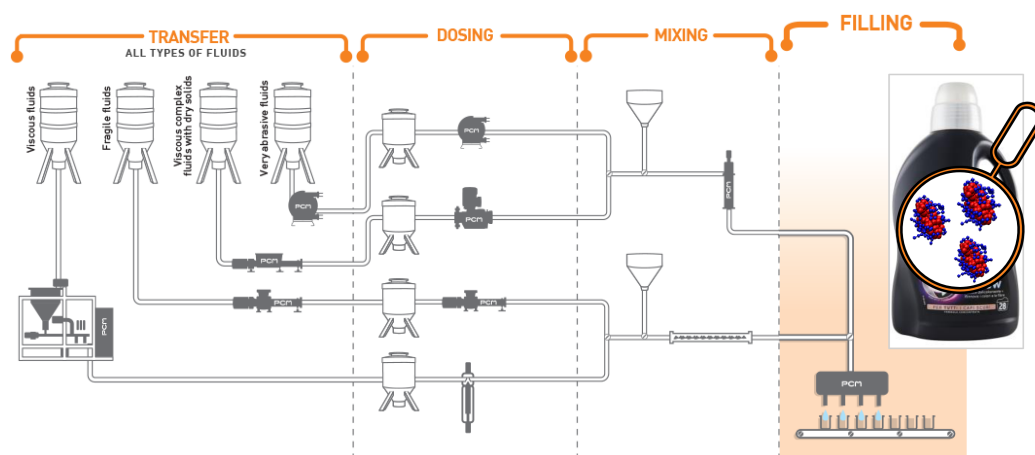


Fig. 1.1 Structured fluids in chemical engineering.

these properties is essential to optimize their performance across a diverse range

of applications. Therefore, rational design and performance optimization of such materials and equipment involved in their processing require the ability to predict their morphology and properties on process-relevant length and time scales in a reasonable time. However, this task is hindered by the time-consuming and costly experimental campaigns necessary to estimate the properties of the material, which are influenced by numerous variables such as composition, concentration, and temperature. Capturing the interaction between morphology and material properties poses a significant theoretical challenge. In this context, computational tools offer an intriguing alternative: they facilitate the mimicking of various scenarios rapidly and cost-effectively, providing a faster understanding on the effects of structural changes.

Given the significance of structured fluids, and soft matter systems in a more general sense, in various fields of technology and biology, research in this area is crucial. Understanding soft matter systems presents considerable intellectual challenges. This highlights the need for the development of valuable computational tools to advance research in these domains.

1.2 Computational modeling by Dissipative Particle Dynamics

The properties of structured fluids and, in general, soft condensed matter are determined by structure and dynamics in an extremely wide range of time and length scales. Therefore, computational modeling of these systems is inherently challenging due to the huge variations in the relevant time and length scale involved. The main challenge lies in the difficulty of bridging these scales to connect chemical structures with their corresponding properties. Although multiscale modeling is a particularly challenging problem for this class of materials, it is the key to success in modeling their complex behavior. Therefore, several computational models have been developed to describe the behavior of these fluids at different levels of representation. At the largest scale of interest Computational Fluid Dynamics (CFD) can predict the flow field in industrial scale equipment. The transport properties are an input information for a CFD simulation, and their prediction requires fluid characterization at lower scales. Because of the characteristic size of the system and the time evolution

of the process under investigation, conventional molecular simulation methods are not suitable for the description of structured fluids; in fact, they rely on a fully atomistic approach, resulting in high computational costs. To bridge the gap between the microscale and the macroscale, a mesoscopic technique is necessary. This approach

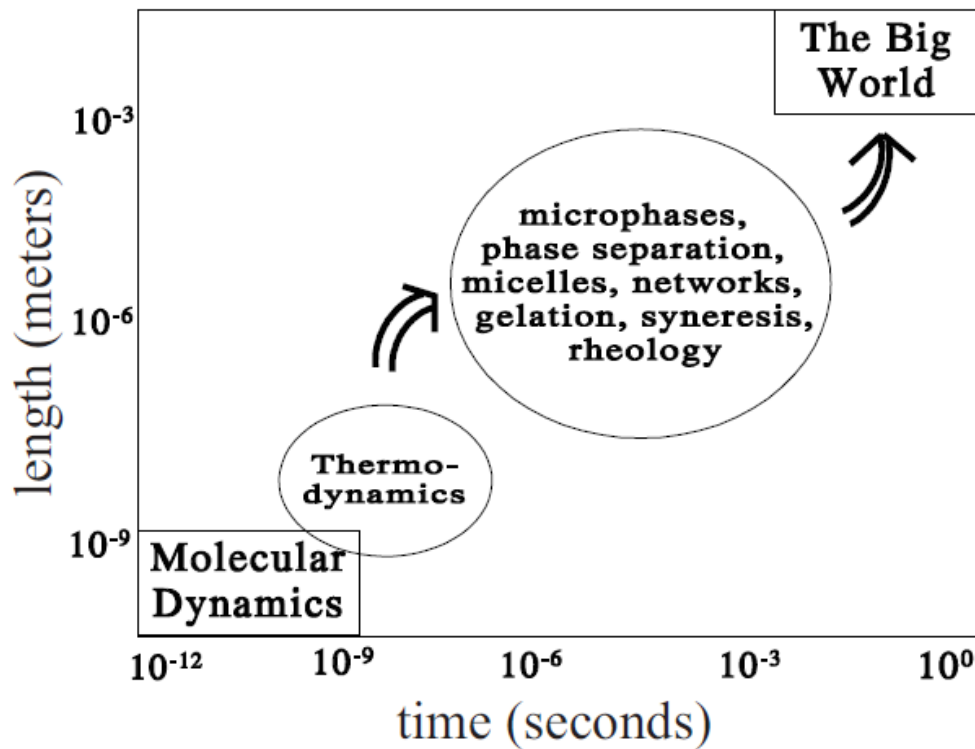


Fig. 1.2 The mesoscale gap. This figure is reproduced from [1].

is crucial for simulating the intermediate regime, which is essential to understand macroscopic phenomena, such as the spontaneous ordering of macromolecules on mesoscopic length and time scales. One of the most successful methods in this area is DPD. DPD is one of the most widely utilized methods for investigating the behavior of these materials and predicting their properties. It is a mesoscopic technique that operates at time and length scales which would be inaccessible using molecular-level methods and are relevant to capture the evolution of the morphologies that dictate the final materials properties. One of the reasons for this is that DPD is a CG model, where groups of atoms and molecules are treated as single entities called beads, interacting via a mesoscopic force field, made of conservative soft-repulsive and non-conservative interactions. The internal degrees-of-freedom of the molecules that a bead is intended to represent are included in an averaged way or effectively

ignored. In this perspective, it is possible to simulate the hydrodynamic behavior of structured fluids with a considerably lower number of particles, thereby reducing the computational effort. The computational advantages offered by DPD make it particularly appealing for simulating systems containing long polymer chains, such as Pluronics, in water.

DPD is one of the most promising simulation techniques for studying the mesoscopic properties of soft matter systems. Although it is a relatively new technique, DPD has already been applied to a wide variety of problems, including complex two-phase flow, such as the rheology of dense colloidal suspensions, the breakup of oil droplets in gravitational and shear fields, spinodal decomposition and domain growth, polymer-polymer surface tension calculations, diblock copolymer melt microphase separation, and studies of shear flow on solid surfaces.

Although the technique has been used successfully in many applications, there are still some gaps in the method that restrict its applicability to real-world engineering problems. In its original, unmodified form, DPD may yield results that are unrealistic and quantitatively inaccurate. This discrepancy with respect to real-world data can be re-conducted to several aspects of the model. In recent years, generalizations and extensions of the original formulation have emerged. The common aim is to address the main deficiencies of the original DPD formulation and to expand the potential applications of the method. This research work can be located in this evolving field. The main issues faced in this research work are briefly introduced in the following and detailed in Chapter 2.

- **The Schmidt number problem.** A well-known problem is linked to the dynamics: the standard DPD framework fails in reproducing the transport properties of real fluids. Several studies claimed that it is not able to accurately describes liquid-like dynamics [18]. The speed by which momentum diffuses is the kinematic viscosity ν , the speed by which particles travel is the diffusion coefficient $\mathcal{D}$. The Schmidt number, $Sc = \nu/\mathcal{D}$, is in the order of 100-1000 for a liquid-like water, whereas it is of the order unity in standard DPD. This effectively means that the diffusion coefficient is overestimated by a factor of 1000 when the viscous time scale is matched. There are two combined effects that lead to this issue, also known as the "Schmidt number problem". Firstly, the origin of this difference can be traced back to the removal of the hard core from the interaction potential. This hard core leads to a caging effect, i.e. an atom

undergoes many collisions before it is actually transported. The soft potential used in DPD removes this caging affect, so that the mobility of particles is increased by a factor of 1000. The second factor contributing to this issue, lies in the empirical form for the non-conservative forces which include only radial frictions, while the underlying microscopic dynamics indicates the presence of non-central shear friction forces which act in a transverse-to-inter-particle connector direction.

- **Temperature-dependence of the DPD model.** Another limitation of the method, which narrows its scope of application in real-world problems, regards the lack of clarity on the introduction of the system temperature variation into the model. Recently a growing and promising trend towards the modification of the conventional DPD model, appropriate for the non-isothermal simulation of conservative interaction force fields, that are both density- and temperature-dependent, is slowly emerging [19–21]. In other cases, the temperature enters into the definition of the model only parametrically without modifying the conventional framework of the DPD model. Different approaches calculate the temperature effect on the DPD conservative repulsive parameters for both simple and complex fluids and successfully validate them against equilibrium properties [13, 22, 23]. Regardless of the approach followed, the application of a DPD model, which incorporates temperature-dependent conservative interactions, allows to describe accurately equilibrium properties when the temperature varies. However, it does not ensure the simultaneous correct prediction of transport properties. Most of DPD studies on transport properties were carried out without varying temperature, although transport properties usually vary strongly with the system temperature and the effect of temperature on the flow dynamics of non-isothermal mesoscopic systems is not negligible.
- **The parametrization problem.** Another significant drawback arises from the absence of a universal, systematic, and well-established approach for identifying repulsive model parameters in multi-component systems. Typically, the parametrization is system-specific and depends on the properties selected as the target. In this context, two key concepts to reflect on are the representability and the transferability. Representability refers to the ability of a CG model

to accurately capture physical properties at the specific thermodynamic state for which it was parameterized. Transferability, on the other hand, relates to the model capability to reliably predict behavior for conditions beyond the parameterization space. Both issues persist regardless of whether the model is mapped onto experimental data (top-down) or onto an atomistic model (bottom-up). The simple form of DPD interactions limits the accuracy and the transferability of the model. As a means of addressing these deficiencies, a new class of DPD variants with more expressions for forces to capture more realistically interactions is emerging, representing an open and active field of research. Bottom-up Coarse-Graining (CGing), which rely on atomistic simulations, also exist and a typical example includes multi-scale CGing (force matching) [24, 25]. Here within the Mori-Zwanzig projection operator method [26, 27], the CG equations of motion (EoM) can be uniquely determined by the underlying microscopic dynamics. They have a form of the generalized Langevin equation and describe the evolution of a subset of the degrees of freedom. Under Markovian approximation, the generalized Langevin equation can be reduced to DPD EoM with conservative, dissipative force and random forces whose expressions can be obtained directly from atomistic simulations. This strategy may be prohibitive from a computational point of view in the case of high molecular weight polymers, where large molecular dynamics trajectories would be required to inform the DPD model into a bottom-up procedure. For many practical purposes involving such systems, it may be useful and computationally advantageous to retain the original simple form of linear repulsive forces and fit the parameters to the specific properties of the system under investigation following a top-down procedure. For such systems, e.g. polymer-based structured fluids, the identification of DPD interaction parameters is largely based on this empirical approach. In fact, their behavior is dictated by a subtle interplay between enthalpic and entropic effects. For such systems, there is still a long way to bridge the gap from the electronic structure of constituent molecular groups and their interactions on the Ångstrom scale, to the effective interactions that are predominantly of entropic origin at mesoscopic scales of 10 or 100 nanometers or more. This simple approach offers a practical advantage, but also poses the challenge of creating an efficient methodology to streamline the trial-and-error process.

- **Accuracy and efficiency in the calculation of physical observable** Eventually, a crucial point concerns the numerical aspects involved in deriving physical observations from DPD simulations and their conversion in real units to allow comparison with experimental data. The numerical methods used to compute physical properties are influenced by statistical uncertainty, stemming from the inherent stochastic noise in DPD simulations. The accuracy of these computations is highly dependent on statistical sampling, and achieving a desired level of accuracy often demands significant computational time. Developing automated routines to manage the trade-off between statistical accuracy and computational feasibility is essential for improving efficiency.

In light of the considerations just made, the use of DPD to simulate the behavior of structured fluids, and more broadly soft matter systems, represents an exciting field. It has significant potential for research and advancement, with the aim of discovering new opportunities to tackle real-world engineering challenges.

1.3 Thesis outline

This paragraph serves as a reading guide for this dissertation. It includes the state-of-the-art overview and objectives, the conceptual framework that guides the various steps of this research, and a concise summary of each chapter.

As previously mentioned, DPD, as originally introduced, is like a minimal version to simulate a molecular fluid. Instead, for specific real-world engineering applications, DPD in its original form, without extensions or modifications, may yield results that are unrealistic and quantitatively inaccurate. At the same time, several studies pointed out that DPD is a very promising technique in the field of structured fluids and more generally in soft matter systems. In light of this, this work aims to augment the DPD as a computational tool to accurately simulate the realistic behavior of both simple and structured fluids at the mesoscale, looking at both their transport and equilibrium properties.

This dissertation aims primarily to bring attention to several cases in which standard DPD fails to accurately predict the properties of the system under investigation. Secondly, it seeks to describe the methodological work conducted, which constitutes a substantial portion of the research. This segment focuses on addressing key issues identified in the DPD to unlock new opportunities to tackle real-world engineering challenges. Three major themes addressed from a methodological perspective can be identified in this work: the development of an extended DPD model suitable for simulating high Schmidt number fluids; the formulation of an automated method to compute transport properties from equilibrium simulations with controlled accuracy; and the creation of a parameterization methodology aimed at capturing realistic equilibrium and transport properties. The physical systems considered in this methodological part are: liquid water and binary mixtures of water and Pluronic. Lastly, this dissertation proposes several studies in which DPD is employed as a complementary tool to real experiments, to gain a deeper understanding of the intricate interplay between morphology and rheology in Pluronic systems.

The dissertation is structured into nine chapters, with a summary of the content presented in the following chapter-by-chapter format.

In this chapter, the first paragraph introduces the industrial and practical context of structured fluids in chemical engineering, establishing the rationale for focusing specifically on Pluronic systems. The second paragraph highlights the challenges of understanding these systems from theoretical and experimental perspectives, making this field an ideal setting for computational modeling. This also explains why DPD is a suitable technique for this purpose.

Chapter 2 provides the theoretical background relevant to the aspects addressed in this research, covering the physical systems investigated, as well as the models and methods employed. The Pluronic class is briefly discussed, with particular emphasis on the key industrial aspects related to the interplay between morphology and rheology. The chapter then describes the DPD model, including its mathematical formulation, governing equations, parameters involved, and solution algorithms. The numerical methods used to compute equilibrium and transport properties from DPD simulations are described, as well as their conversion from DPD units to real-world units. Eventually, the key challenges associated to the use of the DPD model to tackle

real-world engineering problems are discussed and grouped in two macro-issues: the description of the dynamics at the mesoscale and the parameterization problem.

Chapter 3 faces mainly with the issue related to dynamics: the standard DPD model, as originally introduced, intrinsically describes gas-like dynamics. The work demonstrates that an extended version of the DPD model turns gas-like dynamics into liquid-like dynamics. This work demonstrates that an extended version of the DPD model turns gas-like dynamics into liquid-like dynamics. A robust viscosity evaluation procedure is also presented, proving the reliability of equilibrium approaches in comparison to benchmark non-equilibrium methods. Furthermore, the results show that the transport properties of the fluid can be fine-tuned by adjusting some model parameters, enabling the simulation of high Schmidt number fluids. The model can reproduce the proper momentum and mass transport behavior of liquid water. A critical analysis of the conclusions drawn from this work is presented at the end of this chapter. This analysis identifies several key aspects to investigate in order to enhance DPD as a computational tool for predicting realistic transport properties: the role of solution algorithms, the reliability and computational feasibility of the numerical procedures used to compute transport properties, the development of a systematic way to tune transport properties, and the incorporation of temperature effects into the model. The investigation of these aspects constitutes the core of chapters 4, 5, and 6.

Chapter 4 focuses on investigating the role of the time integration scheme in the simulation of fluids with a high number of Schmidt. This contribution presents, for the first time, the application of the Shardlow-splitting algorithm to the transverse DPD thermostat for various Schmidt number fluids, based on systematic numerical experiments. The first part of this chapter details the implementation of the algorithm in the open-source software LAMMPS [28]. The second part demonstrates the superiority of the Shardlow-splitting algorithm compared to the widely used modified velocity-Verlet algorithm (VV) in simulating high Schmidt number fluids.

Chapter 5 focuses on the numerical procedures used to compute transport properties and proposes a systematic way to tune transport properties. In particular, the work analyzes some well-known unsolved methodological issues related to the use of the Green-Kubo equilibrium approach for viscosity calculation. The development of an automated algorithm presented in this work is shown to be crucial for computing transport properties with controlled accuracy and ensuring the computational feasi-

bility of the approach for a large set of systems with increasing Schmidt numbers. The availability of a computational protocol facilitates a systematic multi-parametric study to explore the relationship between the Schmidt number and relevant input model parameters.

Chapter 6 focuses on the incorporation of temperature effects into the DPD model to capture temperature dependent transport properties. This work proposes for the first time a temperature-dependent form of the extended DPD model. A temperature-dependent parameterization methodology for both conservative and non-conservative interactions capable to ensure the correctness of not only equilibrium properties but also dynamic properties. In fact, this methodology has been successfully validated against experimental data on water transport properties across the entire liquid temperature range.

Chapter 7 makes use of some of the findings of previous chapters to carry out virtual experiments of Pluronic systems in water. The aim of the simulations is to provide a deeper understanding of the interplay between morphology and macroscopic rheological properties. Two distinct applications are considered, each involving a different type of Pluronic.

Chapter 8 proposes a data-driven approach for parameterizing Pluronic systems. This work utilizes a data-driven approach to determine the model parameters of Pluronic systems. In particular, a reliable data-driven model is developed to accurately surrogate DPD simulations. This model ability to deliver faster responses facilitates more efficient parameter optimization and enables the calibration of various Pluronic systems against experimental ground truth. Additionally, a transfer learning framework is proposed to make the parameterization process more efficient across different types of Pluronic systems.

Chapter 9 provides an overview of the key contributions of this research and a summary of the conclusions.

Chapter 2

Theoretical background

2.1 Poloxamers: the interplay between morphology and rheology

Developed by BASF in 1973, Poloxamers are commercially known as Pluronics®. These are non-ionic amphiphilic block copolymers, featuring a central PPO block flanked by PEO blocks. These copolymers are synthesized using sodium and potas-

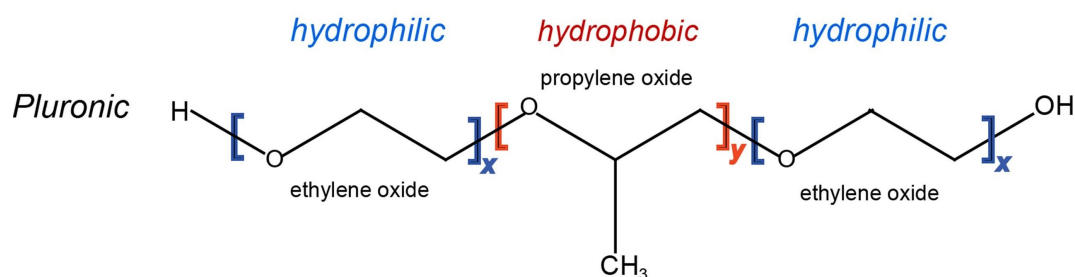


Fig. 2.1 General structure of a Pluronic® molecule (image from Ref. [2]).

sium hydroxide catalysts through a ring-opening polymerization of Propylene Oxide (PO) and Ethylene Oxide (EO) monomers. By adjusting the lengths of the PEO and PPO blocks during synthesis, a variety of Pluronics with diverse amphiphilic properties are created. This versatility makes them useful in many industrial applications, including detergency, stabilization of dispersions and foams, emulsification, and lubrication. They are also used in cosmetics and inks. In pharmaceuticals,

Pluronics help solubilize drugs, control drug release, and aid in the healing of burn wounds. In biotechnology, they protect microorganisms from mechanical stress in bioreactors [5]. The Pluronic grid in Fig. 2.2 offers a detailed overview; Pluronic

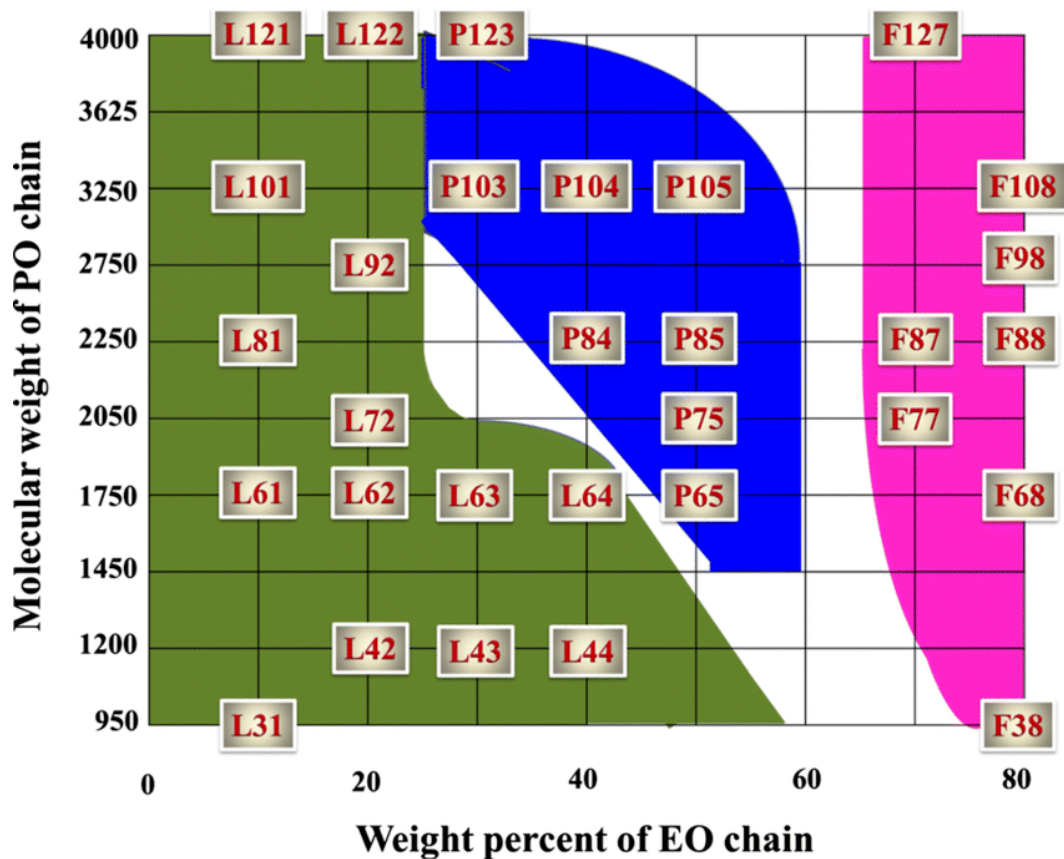


Fig. 2.2 Pluronic® grid (image from Ref. [3]). Color code: physical state of copolymers under ambient conditions: green = liquid; blue = paste; pink = flake. Vertical lines (bottom to top) show increasing molecular weight of the PPO block, while horizontal lines (left to right) show increasing EO percentage by mass. The numbers in the boxes refer to Pluronic L64.

types are designated with an alphanumeric code: the letter indicates the physical state (P=paste, L=liquid, F=flake), the first number (multiplied by 300) indicates the molecular weight of the PPO block, and the last number (multiplied by 10) shows the percentage of PEO by mass. For example, L64 indicates a liquid Pluronic with a PPO molecular weight of about 1800 g/mol and 40% PEO by mass. The two types of blocks in the copolymer are chemically incompatible: the PEO side chains are slightly hydrophilic, while the PPO central chain is slightly hydrophobic. Thus, Pluronic macromolecules are amphiphilic, meaning they have an affinity for two

incompatible environments. This causes the macromolecules to orient themselves preferentially in the presence of a solvent that selectively interacts with one of the blocks [29]. In aqueous solution, these macromolecules spontaneously aggregate to form various microstructures depending on temperature, concentration, and the PPO/PEO weight ratio. At low concentrations and temperatures, poloxamers dissolve homogeneously as monomers, as both copolymer chains can form hydrogen bonds with the solvent, exhibiting slight hydrophilicity. However, as the temperature increases, PPO becomes less hydrophilic faster than PEO, and the copolymer behaves like a surfactant. The key parameters for identifying micelle formation are the CMC and the CMT. These parameters vary with different types of Pluronic and are influenced by the PPO content. More hydrophobic copolymers with higher PPO content have lower CMC and CMT, while less hydrophobic ones have higher values [30]. Micelles have a hydrophobic PPO core and a hydrophilic PEO corona. This organization minimizes the contact between the hydrophobic part of the polymer and the hydrophilic aqueous environment, driven by energetic and entropic factors to reduce the system's free energy and increase entropy [5]. The micelle morphology is determined by the ratio of hydrophilic to hydrophobic groups: a higher hydrophilic content results in spherical micelles, while a higher hydrophobic content leads to spheroidal and asymmetric shapes. At intermediate concentrations, spheroidal micelles elongate into cylindrical shapes with hydrophobic chains at the center and hydrophilic chains on the exterior. These cylindrical micelles interact through their hydrophilic parts, arranging into a crystalline lattice with hexagonal unit cells. At even higher concentrations, these cylindrical micelles aggregate to form a lamellar phase, characterized by alternating thin layers of copolymer and aqueous solvent. As the copolymer concentration in aqueous solution increases, Pluronic molecules form various liquid crystalline phases, known as lyotropic phases, illustrated in Fig. 2.3. The micelles grow in number and size, bringing PEO groups closer together to interact through weak intermolecular forces, forming a crystalline lattice with either Body-Centered Cubic (BCC) or Face-Centered Cubic (FCC) unit cells, appearing macroscopically as a semi-solid gel phase. This process, known as thermoreversible gelation, leads to a significant increase in viscosity [31]. At very high copolymer concentrations, inverse micelles form, where the solvent constitutes the dispersed phase within the continuous polymer matrix [32]. Phase diagrams are crucial for predicting the evolution of microstructures in Pluronic-water systems, illustrating the phases formed as a function of copolymer concentration (x-axis)

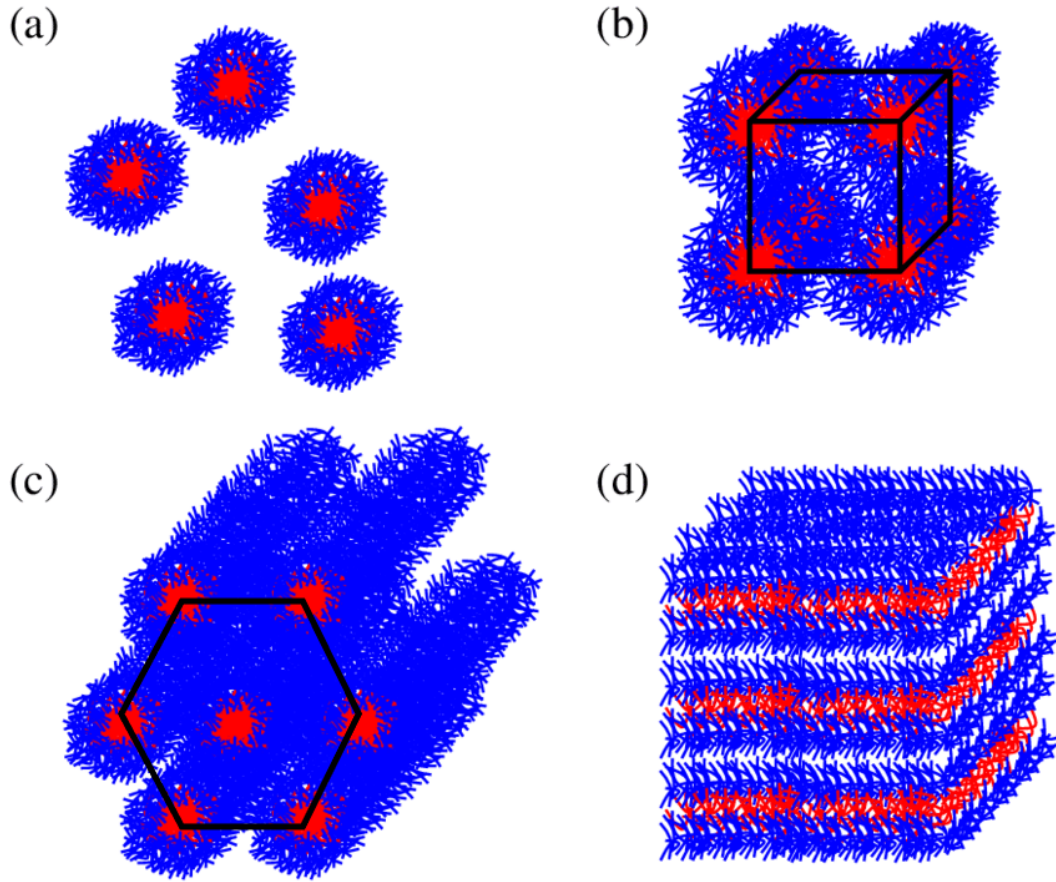


Fig. 2.3 Schematic representation of micellar (a), cubic (b), hexagonal (c), and lamellar structures formed by Pluronic in aqueous solvent (image adapted from Ref. [4]).

and temperature (y-axis). Figure 2.4 shows the phase diagram for the Pluronic L64-water binary mixture. The rheological properties of aqueous solutions of block copolymers are extraordinarily complex because they are highly dependent on the shape and size of the microstructures that form. When poloxamers are in an aqueous solution and their concentration exceeds the CMC, they structurally evolve through various micro-domains: disordered spherical or cylindrical; ordered spherical with micelles positioned at the nodes of a CCC or BCC lattice; ordered cylindrical with micelles positioned at the nodes of a hexagonal lattice; lamellar. Beyond a certain concentration, the solution reverts to a disordered micro-domain with the formation of inverse micelles. The formation of ordered lattices that align with the flow when

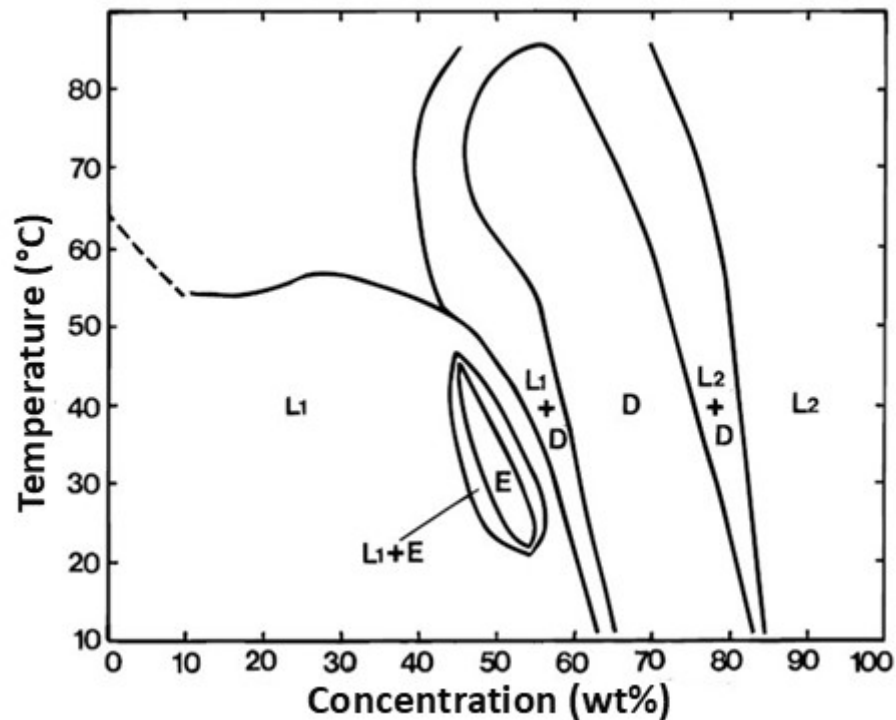


Fig. 2.4 Phase diagram of the Pluronic L64-water binary mixture (image adapted from Ref. [5]). L₁: micellar phase. E: hexagonal phase. D: lamellar phase. L₂: inverse micellar phase.

subjected to shear stress greatly affects the rheological properties. The phase diagram in Fig. 2.4 was obtained using Deuterium Nuclear Magnetic Resonance Spectroscopy (^2H -NMR) and Small-Angle X-ray Scattering (SAXS) techniques [32]. These are the main experimental tools used to determine phase boundaries (^2H -NMR) and characterize microstructures in different phases (SAXS). ^2H -NMR leverages the magnetic properties of deuterium, while SAXS measures the angular variation of X-ray beams scattered by the microstructure lattice [31]. At low copolymer concentrations or with few micelles in solution (disordered spherical micro-domain), the binary mixture of water and copolymer behaves as a Newtonian fluid, similar to water. At higher copolymer concentrations, where cylindrical micelles organized into hexagonal structures form, the fluid exhibits strongly non-Newtonian behavior, specifically pseudo-plastic behavior, where the apparent viscosity decreases with increasing shear rate. This rheological behavior can be microscopically explained by *shear alignment*: cylindrical micelles align in the direction of the flow, so that the densest planes of the cylinders are parallel to the shear surfaces. In the case of

the lamellar micro-domain, alignment phenomena occur where the normals to the lamellae align in the direction of the gradient or vorticity, depending on the shear conditions, influencing the system's rheological behavior [33]. When the volume fraction of the copolymer approaches unity and inverse micelles form, the fluid once again exhibits Newtonian behavior.

2.2 Dissipative particle dynamics: a mesoscopic technique

The DPD capability to simulate molecular motions on scales of length and time far beyond the capabilities of classic all-atom Molecular Dynamics (MD) simulations relies on two key components. Firstly, DPD is a coarse-grain method where groups of atoms or molecules are treated as single entities, so called "beads". The number of molecules that constitute a DPD particle is typically designated as N_m , the so-called coarse-graining factor. This factor significantly impacts the speed and, therefore, the computational cost of the simulation. According to an estimate by Groot and Rabone, the speed increase of a DPD simulation compared to an MD simulation, for a given volume, is approximately $1000N_m^{8/3}$, which is around 2×10^4 for $N_m = 3$ and approximately 5×10^5 for $N_m = 10$. Secondly, these new particles interact with each other through a soft repulsive potential, which allows them to penetrate each other upon collision, leading to energy dissipation. The DPD potential does not become infinite when the distance between particles is zero. The definition of a soft repulsive potential facilitates access to larger spatial and temporal scales because it permits the use of a larger time integration step. In fact, to describe the correct thermodynamics on a larger length-scale than the atomistic one, we only need to ensure that the liquid's compressibility and the solubilities of various components within each other are correctly represented [34]. To accomplish this goal, we have the flexibility to use an effective interaction that is relatively soft in repulsion, as long as the criteria mentioned earlier are met. This approach allows us to bypass the hard core repulsive interactions between atoms. Since these hard core interactions necessitate the use of very small time steps (about 10^{-15} seconds), their removal enables a significant increase in the time step size, often by a factor of four orders of magnitude.

2.2.1 The mathematical formulation: governing equations and reduced units

DPD is a particle-based mesoscopic simulation technique, where groups of atoms are projected into a statistically equivalent ensemble of structureless CG particles, the so-called beads, interacting via a mesoscopic force field. The DPD force field consists of variables in reduced units, the so-called DPD units. Typically in DPD, the mass of a single DPD bead, conservative cut-off radius and thermal energy are taken as, respectively, mass, time and energy units. Thus, other quantities such as the viscosity and diffusivity of the system are not defined explicitly but in terms of the DPD units. The time evolution of each bead can be calculated by the Newton second law as

$$\frac{d\mathbf{r}_i}{dt} = \mathbf{v}_i, \quad \frac{d\mathbf{v}_i}{dt} = \frac{\mathbf{f}_i}{m_i}, \quad (2.1)$$

with $i = 1, \dots, N$; $\mathbf{r}_i$ and $\mathbf{v}_i$ are the position and velocity of the bead i with mass m_i , respectively, and N is the number of DPD beads in the system. In the case of a DPD fluid, the force $\mathbf{f}_i$ acting on the i -th bead is the sum of three pairwise contributions:

$$\mathbf{f}_i = \sum_{j \neq i} (\mathbf{F}_{ij}^C + \mathbf{F}_{ij}^D + \mathbf{F}_{ij}^R). \quad (2.2)$$

In Eq. (2.2), the sum runs over the indices of beads contained in the closest vicinity of bead i within a certain cut-off radius r_c . The conservative contribution, $\mathbf{F}_{ij}^C$, is a soft-repulsive force acting between a pair of beads i and j , and having the following functional form

$$\mathbf{F}_{ij}^C = \begin{cases} a_{ij} \left(1 - \frac{r_{ij}}{r_c^C}\right) \hat{\mathbf{r}}_{ij}, & r_{ij} < r_c^C \\ 0, & r_{ij} > r_c^C \end{cases} \quad (2.3)$$

where a_{ij} denotes the maximum repulsion (the repulsive parameter) between beads i and j , $r_{ij} = |\mathbf{r}_{ij}| = |\mathbf{r}_i - \mathbf{r}_j|$ is the separation distance between a pair of beads, and $\hat{\mathbf{r}}_{ij} = \mathbf{r}_{ij}/r_{ij}$ is the unit vector of the bead-bead separation distance, r_c^C is the cut-off radius for the conservative interactions. Between neighboring beads on a chain, an extra spring force is defined to bind the beads together. In this case, the conservative bonding and non-bonding forces, together with the non-conservative forces then enter into the equations-of-motion. Dissipative and random forces, $\mathbf{F}_{ij}^D$ and $\mathbf{F}_{ij}^R$, respectively, represent the effect of viscosity slowing down the particles motion with respect to each other and of thermal/vibrational energy of the system.

They act together as a thermostat. The random and dissipative forces are coupled via the Fluctuation–Dissipation Theorem (FDT) [35], ensuring sampling from the appropriate probability distribution. In addition, DPD conserves the total momentum. The weight functions for the dissipative force and the stochastic force provide the interaction range for their respective forces and are linked by a relationship necessary to ensure that the system, in the limit of an infinitesimal time step, reaches Gibbs equilibrium, derived from Español and Warren [35]. Obtaining from the simulations an equilibrium distribution corresponding to the Boltzmann distribution of a canonical or Gibbs ensemble is fundamental for relating to the classical relations of thermodynamics in the system under consideration. One of the two weight functions that appear can thus be chosen arbitrarily, and this choice determines the other:

$$w_D(r) = [w_R(r)]^2, \quad (2.4)$$

$$\sigma^2 = \frac{2\gamma k_B T}{m}, \quad (2.5)$$

where k_B is the Boltzmann constant and T is the equilibrium temperature. This condition ensures that the DPD equations act as a thermostat, and since the algorithm depends on relative velocities and the interactions between particles are symmetric, it is a Galilean-invariant thermostat that preserves hydrodynamics [36]. The conventional functional form for the dissipative and stochastic weight functions is as follows:

$$w^D(r_{ij}) = [w^R(r_{ij})]^2 = \begin{cases} \left(1 - \frac{r_{ij}}{r_c}\right)^2 \hat{\mathbf{r}}_{ij}, & r_{ij} < r_c \\ 0 & r_{ij} \geq r_c \end{cases} \quad (2.6)$$

2.2.2 Interaction parameters

Defining the interaction parameters is one of the most important aspects of DPD simulations.

Single-component simple fluids

To accurately represent the thermodynamic state of an arbitrary liquid in DPD simulations, it is essential to adequately capture the internal fluctuations. These fluctuations are primarily influenced by the system's compressibility. To correctly describe the fluctuations in the system, Groot and Warren related the (like) repulsion

parameter, a , to the compressibility of the system in their seminal DPD work [34]. Firstly, Groot and Warren [34], through a series of numerical simulations for a single-component DPD system, found the following virial equation of state that relates the pressure p to the numerical density of the simulation particles ρ :

$$p = \rho k_B T + \frac{1}{3V} \left\langle \sum_{j>i} \mathbf{r}_{ij} \mathbf{F}_{ij}^C \right\rangle = \rho k_B T + \frac{2\pi}{3} \rho^2 \int_0^{r_c} r a \left(1 - \frac{r}{r_c} \right) g(r) r^2 dr, \quad (2.7)$$

$$p = \rho k_B T + 0.1 a_{ii} \rho^2, \quad (2.8)$$

where $g(r)$ denotes the radial distribution function, which describes the probability of finding two particles at a distance r . For values of $\rho > 2$, it can be simplified while still accurately reproducing the numerical results obtained. Secondly, using the equation-of-state (EoS) for the DPD fluid and applying the definition of compressibility, Groot and Warren arrived at

$$a = \frac{\kappa^{-1} - 1}{2\alpha\rho}, \quad (2.9)$$

where κ^{-1} is the reference dimensionless compressibility, ρ is the bead density, and the EoS parameter $\alpha = 0.101 \pm 0.001$ for $\rho > 2$. Groot and Warren employed the experimental $\kappa^{-1} = 16$ for ambient water which gives $a = 75/\rho$. Later, Groot and Rabone [37] modified Eq. (2.9) by including the level of coarse-graining, Φ , (e.g., the number of water molecules per bead) as

$$a = \frac{\Phi \kappa^{-1} - 1}{2\alpha\rho}. \quad (2.10)$$

However, there is a controversy in the literature whether to take into account the level of coarse-graining (e.g., the number of water molecules per bead) when relating the repulsion parameter to the compressibility. Fuchslin et al. [38] realized that this limits use of DPD method since for a large Φ , the system solidifies due to a large a . They showed that Φ does not need to be taken into account when relating a to κ^{-1} if both a and the cut-off radius r_c are properly scaled by Φ ; specifically as $\Phi^{2/3}a$ and $\Phi^{1/3}r_c$. Then in reduced units, the repulsion parameter becomes invariant under Φ (see Eq. (35) in Ref. [38]), and Eq. (2.9) can be used to relate a to κ^{-1} .

Multi-component polymer system

In multi-component systems, it's also necessary to determine the repulsive parameters between different types of beads. For polymer systems, the (unlike) repulsion parameters are typically set by matching the solubility behavior. In polymer chemistry, solubility is commonly expressed through the F-H χ -parameters. This parameter represents the excess free energy of mixing in the F-H model. The F-H lattice fluid model [39] employs a statistical mechanical approach to model the behavior of polymers in aqueous solutions. This is a cell model, where every cell is filled by a fraction ϕ of A molecules and by a fraction $1 - \phi$ of B molecules, i.e. the lattice is completely filled. If A is a polymer occupying N_A cells, and B is a solvent that occupying N_B cells, the Helmholtz free energy per cell (disregarding constant and terms linear in ϕ) can be written as

$$\frac{F}{k_B T} = \frac{\phi \ln \phi}{N_A} + \frac{(1 - \phi) \ln(1 - \phi)}{N_B} + \chi \phi(1 - \phi), \quad (2.11)$$

where ϕ and $(1 - \phi)$ represent the volume fractions of components A and B and N_A and N_B are the number of segments in molecules A and B. In this framework, the interactions between different segments within the lattice are characterized by the parameter χ . To account for the interactions between different components in the solution, enabling the extension of DPD simulations to multi-component systems, Groot and Warren [34] established a relationship between the DPD parameters and the F-H solubility parameter χ , as briefly outlined below. The free energy density corresponds to the pressure of a single component, which can be expressed using Eq. (2.8):

$$\frac{f_v}{k_B T} = \rho \ln \rho - \rho + \frac{\alpha a \rho^2}{k_B T}. \quad (2.12)$$

For a system consisting of two components, this expression becomes:

$$\begin{aligned} \frac{f_v}{k_B T} = & \frac{\rho_A}{N_A} \ln \rho_A + \frac{\rho_B}{N_B} \ln \rho_B \\ & - \frac{\rho_A}{N_A} - \frac{\rho_B}{N_B} + \frac{\alpha(a_{AA}\rho_A^2 + 2a_{AB}\rho_A\rho_B + a_{BB}\rho_B^2)}{k_B T}. \end{aligned} \quad (2.13)$$

By setting $a_{AA} = a_{BB}$ and defining $x = \rho_A + \rho_B$, the analogy with Eq. (2.11) leads to the following definition:

$$\chi = \frac{2\alpha(a_{AB} - a_{AA})(\rho_A + \rho_B)}{k_B T}. \quad (2.14)$$

Numerical simulations validate the linear relationship between χ and Δa as predicted by Eq. (2.14), but they also reveal a correction to the proportionality coefficient, which is not linearly dependent on density. For $\chi > 3$, the following relationship holds:

$$\chi = (0.231 \pm 0.001)\Delta a. \quad (2.15)$$

Given that Δa represents the increase in repulsion between different beads compared to beads of the same type, the coefficient of the repulsive conservative force between two species, i and j , is approximately given by [40, 41]:

$$a_{ij} \approx a_{ii} + \frac{\chi_{ij}}{(0.231 \pm 0.001)}. \quad (2.16)$$

Thus, the repulsion parameters between beads of different species are determined by ensuring that the mutual solubilities are correctly described within the DPD model. The pertinent χ -parameters can be determined by matching the F-H model to relevant experimental solubility data. This approach enables us to compare simulations to experiments directly, or alternatively to extract the simulation parameters from experimental data. The parameters discussed so far pertain to non-bonded conservative interactions. In polymer systems, however, it is essential to also consider conservative parameters related to bonded interactions. In DPD simulations, polymers are typically modeled as flexible, freely jointed chains or semiflexible chains, with DPD beads connected by *springs* that mimic covalent bonds, following the bead-spring model. The bonds between neighboring beads in polymer chains are characterized by a combination of stretching potentials—such as harmonic (Hookean), FENE (finitely extendible non-linear elastic), or Morse potentials—and a soft repulsive potential. For instance, when using the harmonic potential, described by the equation:

$$E_{\text{harm}} = K_b(r_{ij} - r_0)^2, \quad (2.17)$$

two additional parameters must be determined. They are: the harmonic constant, K_b , and the equilibrium distance, r_0 .

2.2.3 Time integration schemes

The time integration of the equations of the DPD model has an additional degree of difficulty compared to that of conventional molecular dynamics due to the presence of stochastic equations. The time reversibility of the integration algorithm is not guaranteed. Moreover, the dissipative force depends on the velocity. For the calculation of velocities, however, to use a second-order algorithm, it is necessary to know the forces at time t and at time $t + \Delta t$. In other words, there is a complication due to the non-linear coupling between force and velocity. A straightforward method to overcome this problem is the Euler scheme, through which the positions and velocities of the particles are calculated based on the respective values in the previous step:

$$\begin{aligned}\mathbf{r}_i(t + \Delta t) &= \mathbf{r}_i(t) + \mathbf{v}_i(t)\Delta t, \\ \mathbf{v}_i(t + \Delta t) &= \mathbf{v}_i(t) + \mathbf{f}_i(t)\Delta t, \\ \mathbf{f}_i(t + \Delta t) &= \mathbf{f}_i(\mathbf{r}_i(t + \Delta t), \mathbf{v}(t + \Delta t)) .\end{aligned}\tag{2.18}$$

However, this algorithm calculates non-time-reversible trajectories. Additionally, it can lead to an accumulation of energy in the system, thus limiting the control of the temperature. The Verlet algorithm can also be used to integrate the DPD equations. [42, 34]. Among the most efficient algorithms that have emerged over the years, achieving a good compromise between accuracy and computational cost, is the Groot/Warren integrator [34] which is a modified version of the *velocity*-Verlet algorithm:

$$\begin{aligned}\mathbf{r}_i(t + \Delta t) &= \mathbf{r}_i(t) + \Delta t \mathbf{v}_i(t) + \frac{1}{2}(\Delta t)^2 \mathbf{f}_i(t), \\ \tilde{\mathbf{v}}_i(t + \Delta t) &= \mathbf{v}_i(t) + \lambda \Delta t \mathbf{f}_i(t), \\ \mathbf{f}_i(t + \Delta t) &= \mathbf{f}_i[\mathbf{r}_i(t + \Delta t), \tilde{\mathbf{v}}_i(t + \lambda \Delta t)], \\ \mathbf{v}_i(t + \Delta t) &= \mathbf{v}_i(t) + \frac{1}{2} \Delta t [\mathbf{f}_i(t) + \mathbf{f}_i(t + \Delta t)] .\end{aligned}\tag{2.19}$$

This integration scheme was first devised by Groot and Warren in 1997 [34] and has subsequently been employed by many researchers in the area of DPD simulations. From the perspective of velocities, it is a *predictor-corrector* algorithm. In fact, the velocities at time $t + \Delta t$ are predicted using the second equation of the iterative scheme (*predictor*), starting from the velocity at time t . The predicted value of the

velocity, denoted as $\tilde{\mathbf{v}}_i$, is used to predict the force. The last equation (*corrector*) calculates the corrected value of the velocity. The computational cost does not increase compared to the classical Verlet algorithm since the force value is updated only once per iteration. Given the stochastic nature of the process, the order of the algorithm becomes indistinct, and the variable factor λ , introduced empirically, accounts for some effects due to stochastic interactions. By appropriately choosing the pair of values for λ and the time step, an *optimum* can be achieved between system stability and simulation speed.

Finite difference methods have also been developed, based on analogies with higher-order integrators for models without stochastic equations, among these the *leap-frog* algorithm. However, it is necessary to be careful in the implementation to account for the stochastic nature of the equations [43].

Simultaneously, other complex integration schemes have been developed, such as the *self-consistent* Verlet algorithm [44] and Shardlow's *splitting* method [45]. The former is an extension of the modified *velocity*-Verlet: it is *self-consistent* because it repeats the force calculation until a stable value for the velocity is obtained. It is time-reversible and ensures excellent temperature control, but the computational cost increases compared to the *predictor-corrector* algorithm since the force calculation is repeated multiple times. Shardlow's integration scheme, although weakly convergent, is based on the innovative idea of splitting the force field into the sum of two terms: the conservative one and the dissipative-stochastic one. The conservative system is solved using the regular modified *velocity*-Verlet, while the equations related to the stochastic and dissipative forces are integrated implicitly to conserve momentum in the integration process.

Besides the difficulty of finding an appropriate time integration algorithm due to the stochastic nature of the DPD model, specific problems often arise depending on the physical system being simulated, requiring the development of *ad hoc* algorithms to accelerate the simulations. For example, a multi-time-step integration scheme based on the modified *velocity*-Verlet algorithm has been introduced, where the solvent particles are updated at lower frequencies compared to the solute particles [46].

2.3 Gaining insights from DPD simulations

2.3.1 Transport properties evaluation

The estimation of transport coefficients in DPD simulations takes place with numerical techniques developed for atomistic simulations. We review them in the following sections.

Self-diffusivity

The self-diffusion coefficient in DPD simulations can be calculated either by the Einstein relation or with the Green-Kubo approach [47]. In the first method, the self-diffusion coefficient, proportional to the mean-square displacement (MSD) of the beads and for the bulk phase, is expressed as follows:

$$\mathcal{D} = \lim_{t \rightarrow +\infty} \frac{\langle [\mathbf{r}(t_0 + t) - \mathbf{r}(t_0)]^2 \rangle}{6t}, \quad (2.20)$$

where $\langle \dots \rangle$ denotes an ensemble average. While, the Green-Kubo approach relates the self-diffusion coefficient to the Velocity Auto-Correlation Function (VACF) and reads as follows:

$$\mathcal{D} = \frac{1}{3} \int_0^{+\infty} \langle \mathbf{v}(t_0) \cdot \mathbf{v}(t_0 + t) \rangle dt. \quad (2.21)$$

It is worth noting that from a theoretical point of view both approaches are equivalent and the choice of one over the other should not give different results. On one side the Einstein relation uses the MSD, which can be easily evaluated from DPD simulations, whereas on the other side the Green-Kubo approach requires the calculation of the time integral of the VACF, which can be tricky from a numerical point of view as seen below. For this reason in this work, we used the Einstein relation to calculate the self-diffusion coefficient.

Viscosity

In principle, we can distinguish between two classes of methods for the calculation of shear viscosity in MD/DPD. The first one, usually denoted as non-equilibrium

molecular dynamics (NEMD), is based on the measurement of the system response after imposing on it either a velocity gradient, $\frac{(\partial u_x)}{\partial z}$, (by Lees-Edwards boundary condition [48] or boundary-driven shear flow) or a momentum flux (RNEMD) [6] and/or a body force (Poiseuille method) [49]. The viscosity is then calculated, $\mu = \eta/\rho$, from the generalized-Newton-viscosity law:

$$j_z(p_x) = -\mu \frac{(\partial u_x)}{\partial z}, \quad (2.22)$$

where $j_z(p_x)$ is the flux of the x -component of the momentum, p_x , in the z -direction. The second class of methods calculates the viscosity by means of an equilibrium simulation, using either the Green-Kubo approach or the Einstein-Helfand relation.

Green-Kubo approach

Similarly to the self-diffusion coefficient calculations, Green-Kubo [50, 51] (G-K) relation allows to calculate the viscosity by integrating the autocorrelation function. In this case, however, the Stress Auto-Correlation Function (SACF) is used:

$$\mu = \frac{V}{k_B T} \int_0^{+\infty} \langle \sigma_{\alpha\beta}(t_0) \sigma_{\alpha\beta}(t_0 + t) \rangle dt, \quad (\alpha \neq \beta), \quad (2.23)$$

where V is the system volume and $\sigma_{\alpha\beta}$ is the off-diagonal component of the stress tensor. In DPD simulations, the stress tensor can be evaluated by the Irving-Kirkwood formula

$$\sigma_{\alpha\beta}(t) = \frac{1}{V} \sum_i \sum_{j < i} r_{ij,\alpha}(t) F_{ij,\beta}(t) + \frac{1}{V} \sum_i m v_{i,\alpha}(t) v_{i,\beta}(t) \quad (2.24)$$

where $F_{ij,\beta} = F_{ij,\beta}^C + F_{ij,\beta}^D + F_{ij,\beta}^R$, and subscripts α and β are reserved for vector or tensor indices while subscripts i and j for bead indices.

Einstein-Helfand approach

Another equilibrium method for calculating viscosity was proposed by Helfand [52] and is analogous to the Einstein (MSD) approach for evaluating the self-diffusion coefficient:

$$\mu = \lim_{t \rightarrow +\infty} \frac{V}{2k_B T} \langle [G_{\alpha\beta}(t) - G_{\alpha\beta}(0)]^2 \rangle \quad (\alpha \neq \beta) \quad (2.25)$$

where $G_{\alpha\beta}$ is the Helfand moment given by:

$$G_{\alpha\beta}(t) = \sum_i m_i r_{i,\alpha}(t) v_{i,\beta}(t). \quad (2.26)$$

Since the Einstein-Helfand method is equivalent to the Green-Kubo approach, it can also be expressed in terms of the Irving-Kirkwood stress tensor as follows:

$$G_{\alpha\beta}(t) = G_{\alpha\beta}(0) + \int_0^t \sigma_{\alpha\beta}(\tau) d\tau. \quad (2.27)$$

Non-Equilibrium method

Non-equilibrium techniques measure the system's transient response to an applied perturbation, closely mimicking real experimental conditions by imposing the corresponding flow parameters within the simulation box. The most widely used "box deforming" technique to produce a shear field is based on the so-called Lees-Edward boundary conditions. In the following, two distinct non-equilibrium methods utilized in this study are described. The Reverse Non-Equilibrium Molecular Dynamics (RNEMD) method, originally introduced by Müller-Plathe [6], is based on imposing a momentum flux on the system and then calculating the velocity gradient in order to measure the viscosity of the fluid. It is conducted in the way that the system (a periodic box) is divided into a finite number of layers perpendicular to the z -direction. Then, a bead with the largest momentum in the x -direction is chosen from the top layer, as well as a bead with the smallest momentum in the x -direction from the middle layer. The momenta of both the beads are swapped. The momentum flux created in this way can be controlled by changing the rate of the momentum swaps. Figure 2.5 shows the schematic representation of the simulation box with the velocity profile imposed by RNEMD method. The momentum flux is then calculated by using the total transferred momentum, P_x , over a time interval, Δt , as follows:

$$j_z(p_x) = \frac{P_x}{2tL_xL_y} \quad (2.28)$$

where L_x and L_y are box lengths in the x - and y -directions, respectively. The factor 2 in the denominator arises because the momentum is transferred in two directions - above and below the middle layer. Due to the momentum flux imposed by the momenta swap, the mean temperature of the domain deviates from the initially

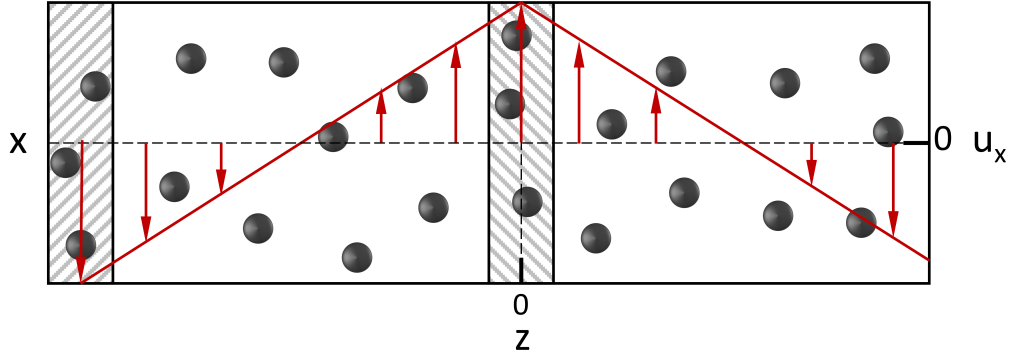


Fig. 2.5 Schematic representation of the RNEMD method. Velocity field is represented by red arrows (image adapted from Ref. [6]).

specified value and, therefore, the use of a smaller time step and of an additional thermostat may be required when RNEMD method is used.

While the RNEMD method is generally suited for Newtonian fluids, to study the viscoelastic properties of polymer systems, a non-equilibrium oscillatory shear technique is required [53]. This technique utilizes the SLLD equations of motion, imposing a time-dependent velocity profile across the domain. The imposed velocity corresponds to a sinusoidal strain of:

$$\varepsilon(\omega) = \varepsilon_0 \sin(\omega t), \quad (2.29)$$

where ε_0 is the amplitude of the oscillation and ω is the angular frequency. At the start of the simulation, an initial mean velocity profile corresponding to the instantaneous deformation rate of the simulation box is imposed on all beads to facilitate the rapid convergence of the flow condition. In general, for a viscoelastic sample, the stress response $\sigma(t)$ oscillates with the same frequency as the strain input:

$$\sigma(\omega) = \sigma_0 \sin(\omega t + \delta). \quad (2.30)$$

There is, however, a phase angle shift δ which varies between 0, representing the purely elastic limit, and $\pi/2$, representing the purely viscous limit. The stress can be further decomposed into two orthogonal functions

$$\sigma(t) = \varepsilon_0 [G'(\omega) \sin(\omega t) + G''(\omega) \cos(\omega t)] \quad (2.31)$$

such that one of them is in sync with the imposed strain and the other has a $\pi/2$ phase lead. Eq. 2.31 can be seen as the trigonometric expansion of Eq. 2.30 using:

$$\sin(\omega t + \delta) = \cos \delta \sin(\omega t) + \sin \delta \cos(\omega t). \quad (2.32)$$

Comparing Eq. 2.30, Eq. 2.31, and Eq. 2.32, the elastic and storage moduli can be defined as follows:

$$G' = \frac{\sigma_0}{\epsilon_0} \cos \delta \quad (2.33)$$

$$G'' = \frac{\sigma_0}{\epsilon_0} \sin \delta. \quad (2.34)$$

2.3.2 Static properties evaluation

A primary output produced by a DPD simulation is a trajectory file that stores information on all particles, such as their positions or velocities, at regular time intervals. Analyzing these trajectory files statistically is essential for gaining valuable insights into aggregation behavior during the simulations. Various tools can be used to detect phenomena such as micelles, aggregates, and other structures formation. However, the core of this type of analysis is a clustering algorithm able to determine if there is any aggregation occurring within the DPD simulation and to calculate various properties of aggregates. To analyze the self-assembly of copolymers, it is essential to first establish a definition of aggregates. A straightforward and commonly used criterion is that two molecules are considered to be part of the same aggregate if they have at least a specified number of contact pairs. A contact pair is defined as a pair of beads from different molecules that are closer than a given distance [54].

Determining the Aggregation Number

The aggregation or association number, denoted as A_s , represents the number of chains present within a single aggregate. The weight-average aggregation number is defined as

$$\langle A_s \rangle_w = \frac{\sum_i m_{A,i}^2}{\sum_i m_{A,i}}, \quad (2.35)$$

where $m_{A,i}$ is the weight of an aggregate i . The weight distribution function of aggregation numbers, F_w , is defined as:

$$F_w(A_s) = \frac{m_{A_s} N_{A_s}}{\sum_i m_{A,i} N_{A,i}}, \quad (2.36)$$

where $m_{A,i}$ represents the weight of an aggregate with an association number $A_s = i$ and $N_{A,i}$ denotes the number of aggregates with that association number. Similarly, the number distribution function of aggregation numbers, F_n , is expressed as:

$$F_n(A_s) = \frac{N_{A_s}}{\sum_i N_{A,i}}. \quad (2.37)$$

Both of these functions are normalized, so the value of F_w and F_n indicates the fraction of aggregates with a specific mass or aggregation number present in the system.

Determining shape descriptors from the gyration tensor

The radius of gyration is defined as the square root of the mean squared distance of beads in a chain or aggregate from its center of mass, r_{CM} , assuming uniform mass distribution among the particles:

$$R_g^2 = \frac{1}{N} \sum_{i=1}^N (\mathbf{r}_i - \mathbf{r}_{CM})^2, \quad (2.38)$$

where N is the number of particles forming the chain or aggregate. The radius of gyration can be calculated using a gyration tensor, $\mathbf{S}$, which describes the second moments of the position vectors of the beads:

$$S_{mn} = \frac{1}{N} \sum_{i=1}^N (r_i^m - r_{CM}^m)(r_i^n - r_{CM}^n), \quad (2.39)$$

where r_i^m stands for the m -th Cartesian coordinate of $\mathbf{r}_i$ and r_{CM}^m is the m -th Cartesian coordinate of $\mathbf{r}_{CM}$.

$$\mathbf{S} = \begin{pmatrix} \lambda_x^2 & 0 & 0 \\ 0 & \lambda_y^2 & 0 \\ 0 & 0 & \lambda_z^2 \end{pmatrix}, \quad (2.40)$$

where λ_x^2, λ_y^2 and λ_z^2 are eigenvalues chosen so that $\lambda_x^2 < \lambda_y^2 < \lambda_z^2$. The eigenvectors indicate the principal axes of the aggregate, while the eigenvalues correspond to the gyration lengths along each principal direction. These eigenvalues can be interpreted as the semi-axes of an equivalent ellipsoid. They are used to calculate various properties, including the radius of gyration and other shape descriptors of the aggregates [55, 56].

The Radius of Gyration is:

$$R_g^2 = \lambda_x^2 + \lambda_y^2 + \lambda_z^2. \quad (2.41)$$

The asphericity is:

$$b = \lambda_z^2 - \frac{1}{2}(\lambda_x^2 + \lambda_y^2)^2. \quad (2.42)$$

The acilindricity is:

$$c = \lambda_y^2 - \lambda_z^2. \quad (2.43)$$

The shape anisotropy is:

$$\kappa^2 = \frac{b^2 + 0.75c^2}{R_g^4}. \quad (2.44)$$

Flatness measures are:

$$f_{zy} = \frac{\lambda_z - \lambda_y}{\lambda_z}; \quad (2.45)$$

$$f_{yx} = \frac{\lambda_y - \lambda_x}{\lambda_y}; \quad (2.46)$$

$$f_{zx} = \frac{\lambda_z - \lambda_x}{\lambda_z}; \quad (2.47)$$

These quantities derived from the eigenvalues can be used to establish a classification criterion for different regions in the phase diagram [57].

2.3.3 Real-space re-normalization of static and transport properties

The DPD computational model is based on parameters and variables expressed in reduced units. In the simulations, dimensionless units are defined for length, the *cutoff* radius r_c , mass of one *bead* (with all *beads* having the same mass), and the system's energy, $\mathcal{E} = k_B T$. From these definitions, all other parameters are derived.

The unit of time is then defined as:

$$\tau = r_c \sqrt{\frac{m}{k_B T}} . \quad (2.48)$$

The conversion between DPD dimensionless units and real physical units is a complex task. For non-equilibrium simulations, this issue remains largely unresolved. However, in equilibrium simulations, the level of *coarse-graining*, defined by the number N_m of molecules within a *bead*, can serve as the re-normalization factor for this conversion. Spatial and temporal scales vary according to the value of N_m . As an example, the procedure for converting DPD units to real physical spatial and temporal units is demonstrated using water—the simplest and most common component in systems studied with DPD—as a reference [40]. If the degree of *coarse-graining* is N_m , meaning that each bead represents water molecules, then a cube with a volume of r_c^3 contains ρN_m water molecules, where ρ is the numerical density of the system, or the number of *beads* per unit volume. Given the known density of water, 1000 kg m^{-3} , and its molecular weight, the volume of a single water molecule at room temperature can be calculated as $30 \text{ \AA}^3$. Thus, knowing the physical volume of this equivalent cube, $30\rho N_m \text{ \AA}^3$, the real unit of length is:

$$r_c = 3.107(\rho N_m)^{1/3} (\text{\AA}) . \quad (2.49)$$

This factor is used to multiply a length in DPD units to convert it into real units. The mass of a particle can be calculated by multiplying the mass of a single water molecule ($m_w = 2.992 \times 10^{-26} \text{ kg}$) by the number of water molecules contained within a *bead*:

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

Finally, the energy unit is calculated based on the reference temperature of the system, equal to 298 K, as follows:

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

where k_B is the Boltzmann constant. where k_B is the Boltzmann constant equal to $1.38 \times 10^{-23} \text{ J/K}$.

Having established the conversion factors for length, mass, and energy, we can define the corresponding factors for other physical quantities in the system. A widely

used criterion for converting the time unit is based on matching the experimental value of the self-diffusion coefficient of water with the value obtained from the DPD simulation. Setting these two equal provides the following expression:

$$\tau = \frac{N_m \mathcal{D}_{sim} r_c^2}{\mathcal{D}_{water}}, \quad (2.50)$$

where $\mathcal{D}_{water} = (2.43 \pm 0.01) \times 10^{-5} \text{ cm}^2/\text{s}$ is the experimental self-diffusion coefficient of water [58].

A similar method was followed by Groot [59], who used the diffusion coefficient of a surfactant micelle for the *matching* process. Both criteria demonstrate that the time step used in DPD simulations can be at least three orders of magnitude larger than that required for molecular dynamics simulations, where typical timestep values are on the order of a few femtoseconds.

2.4 Key issues in DPD

DPD, as originally introduced and described above, is like a minimal version to simulate a molecular liquid. For particular applications, extensions and variations are required.

2.4.1 Simulating the dynamics of mesoscopic systems

A well-known problem is linked to the dynamics. The formal derivation of CG models involves averaging over atomic degrees of freedom that are considered irrelevant to the problem under study, a process that leads to the definition of Many-Body Potential of Mean Force (MB-PMF). However, because of the complexity and high dimensionality of these systems, obtaining the exact MB-PMF from detailed simulation data is typically infeasible, but there are several approaches to derive close approximations based on effective pair potentials. These effective pair potentials enable Coarse-Graining Molecular Dynamics (CG-MD) simulations to mimic the structural and thermodynamic properties of their Fine-Grained (FG) counterparts, thus extending the range of MD simulations to larger length and time scales. CG-MD simulations, based on Hamiltonian dynamics, do not accurately capture the dynamic properties of FG systems. This discrepancy arises because the Degrees-of-Freedoms

(DoFs) that are removed during coarse-graining contribute to friction and random noise, which are not accounted for in CG-MD. Ignoring these dissipative interactions is known to lead to accelerated dynamics and a mismatch in both absolute and relative time scales for various molecular processes. To correct for this, the missing frictional forces and random fluctuations can be incorporated into the CG EoM, resulting in a modified CG-MD EoM that includes a stochastic thermostat. Although, in theory, an exact CG EoM can be derived using the projection operator approach (Mori-Zwanzig theory), in practice, it is necessary to make certain approximations and assumptions to develop manageable models. Within the Mori-Zwanzig projection operator method [26, 27], the CG equations of motion (EoM) have the form of the Generalized Langevin Equations (GLEs) and describe the evolution of a subset of DFs. Under Markovian approximation, the GLEs can be reduced to DPD EoM with conservative, dissipative force and random forces whose expressions can be obtained directly from atomistic simulations. Within the pairwise and Markovian limits, which are always valid for systems interacting with non-linear potentials at sufficiently aggressive coarse-graining, the GLEs dynamics acquire a classical Galilean invariant form of the DPD equations. A significant drawback of the standard DPD method lies in the empirical form for the non-conservative forces which include only radial frictions, while the microscopic GLEs dynamics indicates the presence of non-central shear friction forces which act in a transverse-to-inter-particle connector direction. In summary, the standard DPD framework does not reproduce the transport properties of real fluids. In this context, chapters 3, 4, and 5 address this issue.

2.4.2 The parameterization problem

To extract meaningful insights from molecular simulations, it is essential to model molecular interactions precisely. This is achieved through inter-particle potentials, which mathematically represent the forces between particles and capture specific interactions like bond stretching, bending, twisting, van der Waals forces, and electrostatic attractions or repulsions, usually between particular atoms or groups. The overall behavior in a simulation is determined by a set of these potentials, known as a force field. Force fields are typically refined to match experimental data or high-level theoretical results, such as quantum chemical calculations. This challenge becomes even more pronounced in CG simulations. Despite recent advances, CG modeling still faces significant challenges, particularly with two key issues: representability

and transferability. Representability refers to the ability of a CG model to accurately capture physical properties in the specific thermodynamic state for which it was parameterized. Transferability, on the other hand, is related to the model capability to reliably predict behavior at different state points where no parameterization data is available. For example, there are long-standing ambiguities regarding the transferability of model parameters at different temperatures. These challenges are also evident in systems such as water-Pluronic mixtures, where transferability issues arise across different concentrations and types of Pluronic. Both issues are problematic and often a source of contention, and both exist whether or not the CG model is developed by reference to experimental data, top-down coarse-graining, or by reference to an underlying atomistic model, bottom-up coarse-graining. Using a top-down parameterization approach, the interaction parameters are usually related to thermodynamic properties or fit to experimental observables. Using a bottom-up approach, these mesoparticle interactions can also be parameterized by deriving conservative CG interactions from structure matching, force matching [24, 25], or relative entropy approaches. The top-down approach is often preferred since it does not require large molecular dynamics trajectories to inform the DPD model, reducing the computational effort. Although manual fitting against experimental data might be manageable for simple fluids, it becomes complex and demanding for multi-component systems or those with chain-like molecules. In such cases, the adjustment process must account for various *itype* – *jtype* interactions (e.g., 1-1, 1-2, 2-2 for a binary system), requiring numerous trial-and-error iterations to find suitable parameters. Indeed, by employing a top-down approach, these can be acquired through multiple iterations of DPD simulations and a systematic comparison with experimental data. This extensive search through the parameter space is not only labor-intensive and costly but also demands substantial expertise from researchers. Machine learning techniques could potentially streamline this parameter optimization process by reducing the effort and enhancing accuracy. In this context, Chapter 8 utilizes a data-driven approach to determine the model parameters of Pluronic systems. In particular, a reliable data-driven model is developed to accurately surrogate DPD simulations. This model ability to deliver faster responses facilitates more efficient parameter optimization and enables the calibration of various Pluronic systems against experimental ground truth. Additionally, a transfer learning framework is proposed to make the parameterization process more efficient across different types of Pluronic systems.

Chapter 3

Simulation of high Schmidt number fluids with dissipative particle dynamics: Parameter identification and robust viscosity evaluation

In the original DPD formulation, described in Chapter 2 each bead experiences not only conservative forces, but also dissipative and random forces acting along the separation distance between two beads, satisfying the FDT and working as a thermostat [60]. However, although standard DPD gives good results in equilibrium simulations, the DPD method has been shown to fail to recover the transport properties of real fluids [34]. Two important transport properties are grouped into the Schmidt number, Sc , defined as the ratio between the fluid kinematic viscosity, ν , and the self-diffusion coefficient, $\mathcal{D}$:

$$Sc = \frac{\nu}{\mathcal{D}}. \quad (3.1)$$

The Schmidt number is a measure of the relative importance between fluid particle diffusion and momentum diffusion. For a typical liquid, the Schmidt number is

The content of this chapter has been published in a peer-reviewed journal. The complete citation is provided below: Lauriello et al., *Physics of Fluids*, vol. 33, no. 7, 2021. <https://doi.org/10.1063/5.0055344>.

$O(10^3)$, meaning that the momentum is usually transported more efficiently than fluid particles themselves. In their work, Groot and Warren [34] showed that the original DPD scheme results in Schmidt number of $O(1)$, with a dependence on the various DPD parameters that can be expressed as:

$$Sc \sim \gamma^2 \rho^2 r_c^8, \quad (3.2)$$

where γ is the dissipative parameter, ρ is the density of DPD fluid and r_c is the cutoff radius for interparticle forces. However, even if these parameters strongly affect the Schmidt number simply by increasing them, also the computational costs would increase either directly, or indirectly by having to decrease the time step in order to maintain the simulation stability [34]. Since the pioneering work of Groot and Warren, many researchers investigated the possibility to describe high Schmidt number fluids with DPD. One of the proposed approaches was to modify the weighting function for the dissipative and random forces, applying the so-called Generalized Weighting Function (GWF) [61, 62, 14], which still satisfies the FDT as standard DPD. The effect of this modification was the increase of the fluid viscosity, allowing to investigate higher Sc numbers than standard DPD with a comparable computational effort. Another modification to the weighting function was proposed by Yaghoubi and coworkers [63]. The drawback of these approaches, as in standard DPD, is the lack of shearing dissipation which has a great impact on the resulting viscosity of the simulated DPD fluid. In fact, with the standard DPD method, the dissipative force vanishes, no matter how close the two particles are, whenever the relative velocity between two beads is perpendicular to their distance. To overcome this issue, Junghans and coworkers proposed an extended DPD version in which the shearing dissipation is considered, by evaluating the perpendicular contribution of dissipative and random forces [64]. As explained in their work, this extension can be interpreted as a different thermostat for DPD, a transverse DPD thermostat, that is able to describe the damping perpendicular to the interatomic axis, mimicking the shear of those degrees of freedom that were integrated out in the CGing procedure. This extended DPD makes possible to increase Sc number because the shearing dissipation is considered. In fact, the dynamical properties, both viscosity and diffusion coefficient, are very sensitive to the damping perpendicular to the interatomic axis. Moreover, in the case of transverse DPD the friction coefficient and the cutoff radius strongly affect the Schmidt number. From a qualitative point of view the influence of these parameters is quite similar. Increasing r_c or γ increases

the mean value of the dissipative force applied to the beads, resulting in a higher viscosity and a lower diffusion constant. In a simulation with the transverse DPD thermostat, the effect of these two parameters on the Sc number is more accentuated than in Eq. 3.2, which was derived from Groot and Warren [34] in the case of standard DPD. In the case of the transverse DPD, another relationship should be derived including the perpendicular contribution of dissipative and random forces in the derivation. However, it should be noted that these analytical relationships are derived neglecting contributions from the conservative forces. Therefore, this method results in higher Sc number than the standard DPD while preserving within an acceptable error bar hydrodynamics. In fact, by using the transverse DPD thermostat, Galilei invariance remains valid by construction while hydrodynamics is conserved within acceptable error bar, while the local angular momentum is not conserved, due to the presence of non-central forces acting between particles. Nevertheless, according to simulation results showed in the work of Junghans and coworkers, these angular momentum fluctuations cancel out on average and hence the total angular momentum of the system is almost a conserved quantity [64]. For this reason, in this work the requirement on angular momentum conservation, which is particularly important when non-equilibrium simulations are used to measure the fluid viscosity, can be relaxed since equilibrium simulations are performed. In this sense, a more complete method, called Fluid Particle Model (FPM), was proposed by Español [42], in which both shearing dissipation and conservation of angular momentum are considered. Pan and coworkers [62] proposed a modified version of FPM, which is more similar to the transverse DPD thermostat and standard DPD method, including all the modifications which have an impact on the Sc number of the investigated fluid, such as the GWF and the shearing dissipation, in a framework that preserves the total angular momentum. With this approach, the beads experience not only parallel and shear dissipation, but also a rotational damping contribution which can be properly evaluated only if the total angular momentum is conserved. This additional contribution has an impact on the resulting fluid viscosity, as shown by Krafnik and Garcia [65]. In their work Krafnik and Garcia pointed out the influence of the different DPD parameters on the resulting Sc number of the simulated DPD fluid. However, the major drawback of this approach is the computational time, which is significantly higher than standard or extended DPD due to the evaluation of the rotational damping contribution.

Therefore, in this chapter of the thesis, we investigate in detail for the first time

the performance of the transverse DPD thermostat in simulating high Sc number fluids with equilibrium simulations, aiming at finding the DPD parameters capable of properly recovering the rheological behavior of liquid water at room temperature and pressure. Moreover, special attention was devoted to a peculiar aspect, which is often overlooked, namely the calculation of the fluid viscosity from simulations. In general, the methods for the evaluation of the viscosity from atomistic simulations are divided into two groups: equilibrium and non-equilibrium methods. In the first category are methods such as Green-Kubo and Einstein-Helfand, while in the latter category, reverse non-equilibrium molecular dynamics (RNEMD), the periodic Poiseuille flow method and the periodic Lees-Edwards flow method. These techniques are described in Chapter 2. Although it was shown that non-equilibrium methods for viscosity evaluation are generally more accurate and stable [66], they require subjecting the investigated system to an extreme shear rate, impossible to reach in real systems, while it is not feasible to explore low shear rates, since the effect of the shear is covered by dissipative and random forces. When Newtonian fluids are investigated, non-equilibrium measurement technique can be used, since the shear rate is independent of the shear stress. In contrast, in the case of non-Newtonian fluids, such as polymer or surfactant solutions, the use of non-equilibrium methods would result in viscosity values corresponding to unphysical large shear rates. In addition, applying the non-equilibrium viscosity measurement for self-assembled systems may alter equilibrium morphologies, discriminating the non-equilibrium approaches for the self-assembled systems. Equilibrium methods overcome these issues and allow us to measure the zero-shear viscosity, often presented in experimental studies of non-Newtonian fluids. For this reason, in this chapter, different viscosity estimation methods were tested with the aim of finding a reliable equilibrium algorithm that can be used in further studies on the dynamics of polymer or surfactant solutions. The RNEMD method by Müller-Plathe [6] is used as a benchmark to double-check and verify the evaluation of viscosity by the equilibrium methods. The RNEMD method by Müller-Plathe [6] is used as a benchmark to verify viscosity evaluation by equilibrium methods. Due to the momentum flux imposed by the momenta swap in RNEMD, the mean temperature of the domain deviates from the initial value. Therefore, a smaller time step and an additional thermostat may be required. The Berendsen thermostat is used for this purpose. This method requires specifying three parameters: (i) the number of momenta pairs exchanged, typically set to one; (ii) the number of layers in the domain, which depends on the computational domain size;

and (iii) the momentum swap frequency, which significantly affects the development of the linear velocity profile inside the box.

3.1 An extended DPD model: from gas-like dynamics to liquid like dynamics

As mentioned above, in the standard DPD thermostat, as originally introduced [67], employed in many prior works [68–71], the dissipative and random forces act, similarly as the conservative force, along the bead-bead separation distance. However in this work, the transverse DPD thermostat, proposed by Junghans and coworkers [64], was used. In this approach the standard DPD thermostat is combined with the transverse DPD thermostat acting in the plane perpendicular to the bead-bead separation distance. The expressions for general dissipative and random forces can then be written as

$$\mathbf{F}_{ij}^D = -\gamma_{\parallel} w_{\parallel}^2(r_{ij})(\hat{\mathbf{r}}_{ij} \cdot \mathbf{v}_{ij})\hat{\mathbf{r}}_{ij} - \gamma_{\perp} w_{\perp}^2(r_{ij})(\mathbf{I} - \hat{\mathbf{r}}_{ij}\hat{\mathbf{r}}_{ij}^T)\mathbf{v}_{ij}, \quad (3.3)$$

$$\mathbf{F}_{ij}^R = \sigma_{\parallel} w_{\parallel}(r_{ij}) \frac{\xi_{ij}}{\sqrt{\Delta t}} \hat{\mathbf{r}}_{ij} + \sigma_{\perp} w_{\perp}(r_{ij})(\mathbf{I} - \hat{\mathbf{r}}_{ij}\hat{\mathbf{r}}_{ij}^T) \frac{\boldsymbol{\xi}_{ij}}{\sqrt{\Delta t}}. \quad (3.4)$$

Here $\gamma_{\parallel}$ and $\gamma_{\perp}$ are parallel and perpendicular dissipative parameters, respectively, analogously $\sigma_{\parallel}$ and $\sigma_{\perp}$ are parallel and perpendicular noise parameters, $w_{\parallel}(r_{ij})$ and $w_{\perp}(r_{ij})$ are weighting functions for the parallel and perpendicular parts of $\mathbf{F}_{ij}^D$ and $\mathbf{F}_{ij}^R$, $\mathbf{v}_{ij} = \mathbf{v}_i - \mathbf{v}_j$, ξ_{ij} and $\boldsymbol{\xi}_{ij}$ are scalar and vector random numbers, respectively, with zero mean value and unit variance, $\mathbf{I}$ is the identity second-rank matrix and Δt is the simulation time step. The dissipative and noise parameters are related to each other in order to satisfy the fluctuation-dissipation theorem (FDT):

$$\sigma_{\alpha} = \sqrt{2k_B T \gamma_{\alpha}}, \quad \alpha \in \{\parallel, \perp\}, \quad (3.5)$$

where k_B denotes the Boltzmann constant and T the system temperature. This paper utilizes, instead of the traditional quadratic scheme usually adopted [68], the GWF, introduced by Fan and coworkers [61] as previously mentioned, but with the introduction of a Dissipative cutoff radius, r_c^D . This parameter does not change the range of conservative force and consequently has no influence on the nature of

the phase assumed by the simulated system, described by the Radial Distribution Function (RDF). In fact, the equilibrium behavior of the system is determined solely by conservative forces because the time average of the dissipative and fluctuation forces is zero [68–70]. Conversely, it is important for the description of the transport properties. So we use the following GWF:

$$w_\alpha(r_{ij}) = \left(1 - \frac{r_{ij}}{r_c^D}\right)^{s_\alpha}, \quad \alpha \in \{\parallel, \perp\}, \quad (3.6)$$

and its exponent, $s_\alpha \in (0, 1]$, plays an important role in modeling of dynamic properties of a DPD fluid. In this work, we do not study the exponents separately but we simply assume that $s_\parallel = s_\perp = s$ and consequently $w_\parallel(r_{ij}) = w_\perp(r_{ij}) = w(r_{ij})$.

3.2 Simulation details

All simulations were performed by using the open-source software LAMMPS [28] with our newly developed LAMMPS `pair_style, dpd/ext`, for extended DPD model, including the transverse DPD thermostat and GWF, described in the previous section. It is available on GitHub¹. DPD fluid was represented by single beads with different degrees of CGing, N_m , which affect the evaluation of the repulsion parameter a of the conservative force, i.e.,

$$a = \frac{N_m \kappa^{-1} - 1}{0.2\rho} k_B T, \quad (3.7)$$

where N_m corresponds to the number of real molecules per bead, κ^{-1} is the inverse of the dimensionless isothermal compressibility of DPD fluid (e.g., $\kappa^{-1} = 16$ for water) and ρ is the DPD number density corresponding to the number of beads inside a volume of $(r_c^C)^3$. All the equilibrium simulations were performed in a cubic box of the length equal to 15 DPD length units with periodic boundary in all directions, considering r_c^C as the DPD length unit. In cases of RNEMD simulations, the box was a cuboid of size $15 \times 15 \times 45$ DPD length units to capture the velocity gradient in the direction of transferred momentum. The system bead density and system temperature were the same for all the cases and set to $\rho = 3$ and $k_B T = 1$. DPD equations of motion were integrated by using the velocity-Verlet algorithm, proposed

¹<https://github.com/mulmopro/LAMMPS-DPD-EXT>

by Groot and Warren [34], and used in several prior studies [68, 72], with Δt set such that temperature fluctuations were lower than about 3% of $k_B T$. This results in $\Delta t = 0.01$ in the case of equilibrium simulations (apart from some specific test cases highlighted below), and $\Delta t = 0.001$ for non-equilibrium cases. The simulation time statistics were collected within $5 \cdot 10^6$ time steps. Regarding the calculation of the viscosity with the Green-Kubo approach, it is worth mentioning that LAMMPS uses an approximate algorithm to evaluate SACF. First, the algorithm makes use of the ergodic hypothesis to evaluate the ensemble average as a time integral, and for this reason the final simulation time needs to be sufficiently long to carry out a correct estimation. Then, the SACF is not evaluated according to its definition, since it would be too expensive from the computational point of view. Instead, the SACF averages are computed every N_{freq} time steps and only a limited data set is actually employed in the SACF evaluation, namely the data points correlated every N_{ev} -th time step for $N_{\text{ev}} \times (N_{\text{rep}} - 1)$ time steps. For further details on this algorithm readers are referred to the LAMMPS code [28]. Therefore, the choice of N_{ev} , N_{rep} and N_{freq} is critical in the proper evaluation of SACF and, in general, in the viscosity evaluation. After numerous tests performed, we found that by selecting $N_{\text{ev}} = 1$, $N_{\text{freq}} = 1000$ and $N_{\text{rep}} = 1000$, the best results are obtained in the evaluation of the SACF. In the cases where $\Delta t = 0.005$, N_{freq} and N_{rep} are taken equal to 2000 to be consistent in the evaluation of the SACF.

3.3 Results and discussion

3.3.1 Robust viscosity evaluation procedure

The challenge for the Green-Kubo viscosity calculation stands mainly in the numerical evaluation of the time integral of the SACF from time $t = 0$ up to $+\infty$. The first issue is related to the fact that the SACF is numerically evaluated at each simulation time step, and after a first clear decaying trend the values of the SACF oscillate around zero with a certain noise, also assuming negative values. An example of this well-known behavior [50], is reported in Fig. 3.1. Therefore, the calculation of such a time integral with a simple trapezoidal rule is affected by the tail oscillations. To overcome this issue, we tested several procedures. The first attempt was to use a procedure proposed by Jung and Schmid [73]. The procedure involves numerical

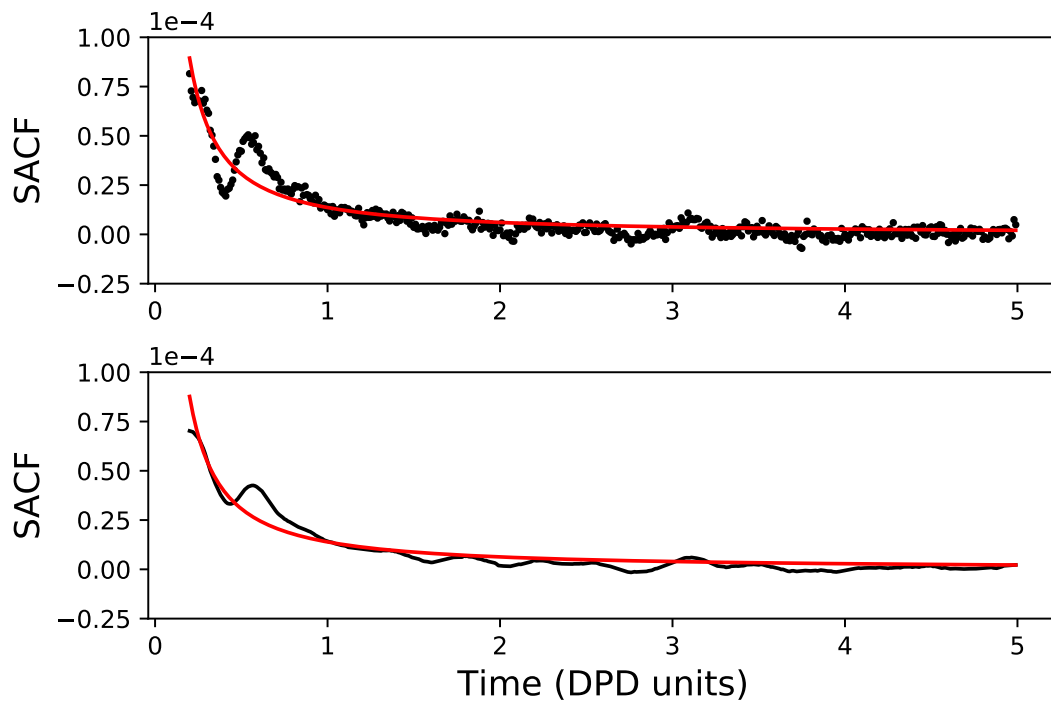


Fig. 3.1 Two different representations of the tail of the same SACF. Top: procedure A. Black circles: points of the SACF. Red line: power-law fit. Bottom: procedure B. Black line: SACF after a moving average filter. Red line: power-law fit..

Table 3.1 Viscosity values (in DPD unit) calculated with various approaches. Procedure A: power-law fitting. Procedure B: moving average filter before power-law fitting. The superscripts 1 and 2 correspond respectively to a final integration time of $N_{rep} \times \Delta t$ and infinite. Procedure C: cumulative integral. E-H and RNEMD refers to Einstein-Helfand and Müller-Plathe methods.

$\gamma_{ }$	A^1	A^2	B^1	B^2	C	E-H	RNEMD
4.5	0.8535	0.8846	0.8706	0.9399	0.860 ± 0.002	0.847	0.860
9	0.9699	1.0018	0.9892	1.0613	0.957 ± 0.004	0.923	0.930
15	1.1128	1.1468	1.1308	1.2065	1.065 ± 0.002	1.071	1.055

integration using the trapezoidal rule until the SACF values reached about 1% of their initial values. Then, the tail is fitted using a power-law, i.e.: At^{-B} , and integrated analytically. This first will be denoted as "Procedure A" in the following. An alternative is to use a moving average filter to smooth out the tail oscillations. This involves the choice of the moving average span and then performing the power-law fitting. This is denoted as "Procedure B". Also in procedure B the analyzed tail starts at about 1% of the initial values of SACF. After preliminary tests on the value of the moving average span, the best result was obtained with an interval of 20 time steps. Figure 3.1 shows examples of Procedure A and Procedure B for a generic DPD simulation. However, a second issue associated with the power-law fitting procedures must be addressed. In fact, the fitted power law always approaches zero slower than the actual numerical value of the SACF, which oscillates between positive and negative values when the power-law fit has not reached its asymptotic value. For this reason, we tried to use two approaches for the analytical integration of the fitted power-law. In the first approach, the final integration time is the final SACF available from the simulation post-processing, depending on the N_{rep} and Δt , and is labelled with number 1 in Tab. 3.1. In the second approach, the analytical integration is carried out up to $+\infty$ and is labelled with the number 2 in Tab. 3.1. As it can be seen in Tab. 3.1, the second approach has a significant impact on final viscosity values, which may increase of about 3-8% if the analytical integration up to infinity is carried out. Moreover, the effect of the tail noise is still not quantified by using these procedures. For this reason, we tried to use an approach based on the calculation of the viscosity cumulative integral function of time, denoted as in what follows as "Procedure C". The idea behind this latter procedure is to evaluate the viscosity by using Eq. 2.23 progressively extending the integration interval up to the final SACF time. The result of the procedure C should theoretically be a monotonic

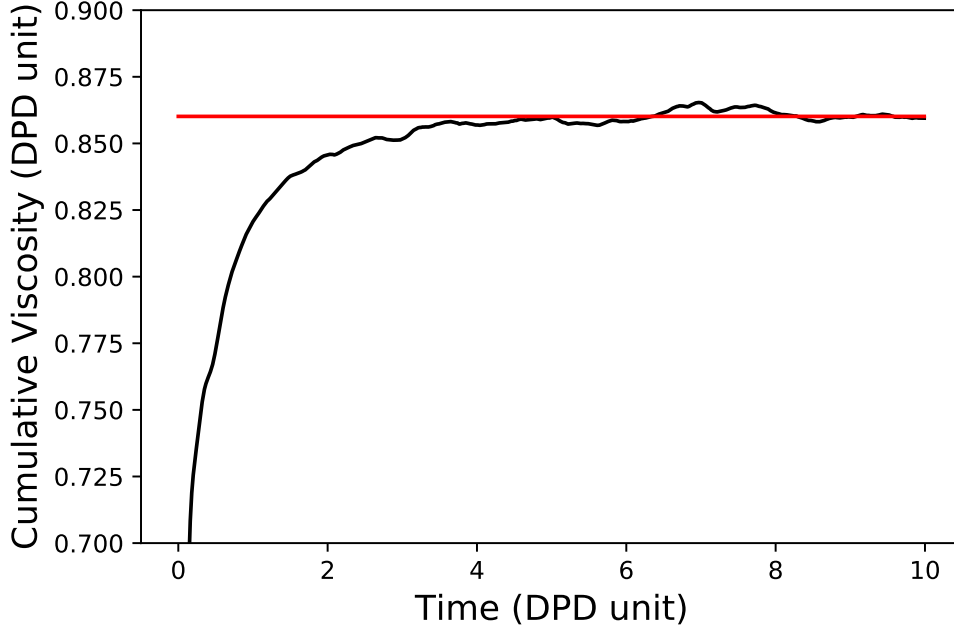


Fig. 3.2 Cumulative integral of viscosity calculated using the Green-Kubo approach. Black line: viscosity cumulative integral. Red line: mean value and coefficient of variation calculated between $t = 5$ and 10.

curve that approaches an asymptotic value that corresponds to the viscosity of the system. However, this is impossible to achieve exactly due to intrinsic stochastic noise within DPD simulations, and an example of such a calculation is depicted in Fig. 3.2. In Fig. 3.2, viscosity values monotonically increase up to approximately 3 DPD units of time, when the viscosity value starts to oscillate around a constant value due to the corresponding oscillations in the SACF. Therefore, we opt to evaluate the system viscosity as the average value of the curve considering only the last 5 DPD time units, avoiding the initial transient period. Moreover, Procedure C is particularly convenient, since it allows us to evaluate a coefficient of variation, providing therefore not only the expected viscosity value but also the associate uncertainty. Tab. 3.1 summarizes the viscosity values calculated using various Green-Kubo procedures together with the viscosity values obtained using the Einstein-Helfand and RNEMD methods, which were used as benchmark. The simulations were run using a standard DPD thermostat, so $\gamma_{\perp} = 0$, and $s = 1$, $r_c^D = 1$ and $N_m = 1$. The results are shown for different values of $\gamma_{\parallel}$ to see the effect of changing one DPD parameter on viscosity. As can be seen in Tab. 3.1, the procedures involving the power-law fit seem to

perform quite poorly when compared to other procedures, especially procedure B, which has the largest deviations. The procedure C gives instead results that are very close to the other two benchmark methods, and this procedure is then used in the rest of the work. Eventually, it is interesting to see that the coefficient of variation of the viscosity calculation for the reported simulations ranges between 0.2 and 0.4%, which is approximately the range found for all the other evaluations carried out in this work. This is due to the fact that the whole numerical methodology to evaluate the viscosity of a fluid has an intrinsic accuracy that derives from the stochastic nature of the DPD model.

3.3.2 Schmidt number scaling

Once a reliable procedure for calculating the viscosity was found, the effect of different DPD parameters on SACF and the Sc number of DPD fluid was investigated by performing different simulations, aiming to reproduce the Sc number of water at 25°C equal to about 370. The parameters varied are the dissipative cutoff radius, r_c^D , the weighting function exponent, s , the degree of CGing, N_m , and the dissipative parameters, $\gamma_{\parallel}$ and $\gamma_{\perp}$. For simplicity and to reduce parameter space, we consider $\gamma_{\perp} = \gamma_{\parallel} = \gamma$. Tab. 3.2 provides a summary of all DPD simulations performed. As expected, the standard choice of DPD parameters reported in the first row of Tab. 3.2, results in a low Sc number, even if the transverse DPD thermostat is used. Moreover, the Sc number does not increase significantly by reducing the GWF exponent s , unless the degree of CGing N_m is increased as well. In Fig. 3.3, we can see the reason of this trend: the SACF goes to zero more rapidly when the degree of CGing is small, regardless of the value of the weighting function exponent, resulting therefore in lower viscosity values according to Eq. 2.23. This means that the conservative force parameter a has a clear influence on the resulting viscosity and self-diffusion coefficient, since N_m is an input to evaluation of a . In fact, by using a higher degree of CGing, the beads in a DPD simulation tend to move less and to exchange more momentum with the surrounding beads, behaving more like a real liquid [74]. Furthermore, as it will be shown in the following, the choice of N_m equal to 10 (i.e., $a = 265$) gives an RDF indicative of the liquid state, as displayed in Fig. 3.7. Conversely, for $N_m > 10$ ($a > 265$) the RDF may present secondary peaks revealing the presence of a crystalline structure [75]. Another interesting aspect to note is an increase of noise in the SACF, evident in Fig. 3.3 when a lower value s is investigated.

Table 3.2 Summary of all DPD simulation performed, including the results in terms of viscosity, self-diffusion coefficient and Schmidt number; all properties are in DPD units. All simulations are carried out with $\rho = 3$. Rows highlighted in bold font are close to the Schmidt number of water at 25°C equal to about 370.

N_m	s	γ	r_c^D	Δt	ν	$\mathcal{D}$	Sc
1	1	4.5	1	0.01	0.430	0.173	2.489
1	0.5	4.5	1	0.01	1.078	0.075	14.373
1	0.25	4.5	1	0.01	2.446	0.038	64.377
1	0.125	4.5	1	0.01	3.687	0.026	141.82
3	1	4.5	1	0.01	0.748	0.099	7.560
3	0.5	4.5	1	0.01	1.612	0.047	34.291
3	0.25	4.5	1	0.01	2.914	0.026	112.06
3	0.125	4.5	1	0.01	4.919	0.017	289.33
10	1	4.5	1	0.01	3.116	0.029	108.53
10	0.733	4.5	1	0.01	3.949	0.024	166.07
10	0.633	4.5	1	0.01	4.245	0.021	203.12
10	0.533	4.5	1	0.01	5.013	0.018	276.35
10	0.5	4.5	1	0.01	5.021	0.017	290.24
10	0.48	4.5	1	0.01	5.055	0.017	300.38
10	0.46	4.5	1	0.01	5.619	0.016	349.02
10	0.434	4.5	1	0.01	5.890	0.016	368.54
10	0.432	4.5	1	0.01	5.950	0.015	384.92
10	0.43	4.5	1	0.01	6.053	0.015	404.09
10	0.42	4.5	1	0.01	6.106	0.015	423.13
10	0.4	4.5	1	0.01	6.253	0.014	444.77
10	0.32	4.5	1	0.01	7.757	0.012	636.88
10	0.3	4.5	1	0.01	7.758	0.011	681.18
10	0.28	4.5	1	0.01	7.834	0.010	725.40
10	0.26	4.5	1	0.01	8.135	0.010	813.46
10	0.25	4.5	1	0.01	8.743	0.010	892.11
10	0.125	4.5	1	0.01	12.13	0.007	1784.1
10	1	9	1	0.01	4.289	0.024	177.00
10	1	18	1	0.01	5.414	0.017	319.96
10	1	20.2	1	0.01	5.880	0.016	367.70
10	1	22	1	0.01	5.879	0.015	397.25
10	1	24	1	0.01	6.361	0.014	453.40
10	1	25	1	0.01	6.387	0.014	458.62
10	1	30	1	0.01	7.862	0.012	650.81
10	1	36	1	0.01	7.999	0.010	767.75
10	1	72	1	0.01	15.33	0.005	2797.2
10	1	4.5	1.2	0.005	4.472	0.021	209.19
10	1	4.5	1.38	0.005	5.870	0.016	366.88
10	1	4.5	1.4	0.005	5.990	0.015	395.63
10	1	4.5	1.6	0.005	14.32	0.007	1962.1

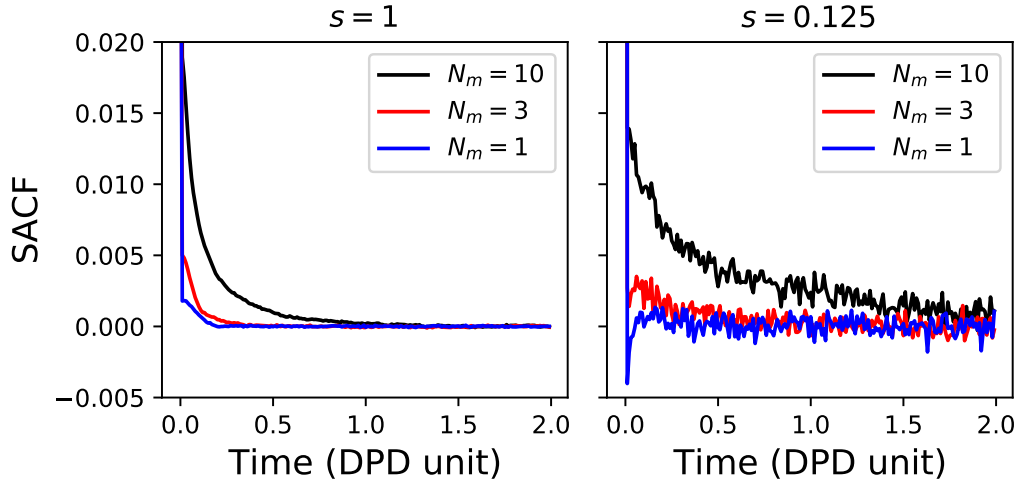


Fig. 3.3 Effect of the degree of CGing, N_m , on the SACF for two different values of the GWF exponent s .

Nevertheless, the resulting viscosities have a coefficient of variation in the same range as in all the other investigated cases, between 0.2 up to 0.8%, confirming the validity of the proposed methodology to assess the rheological properties of a DPD fluid. Fig. 3.4 shows the influence of the other DPD parameters on the SACF. A larger area below the curve corresponds to a higher value of the viscosity according to the Green-Kubo approach. From Fig. 3.4, we can see the following trends. The viscosity increases when the dissipative parameter and dissipative cutoff radius increase, and when the GWF exponent decreases, since these increases/decreases enhance bead momentum dissipation. It is worth mentioning that not all the simulations were run with the same set of all the other DPD parameters, and also, the time step was reduced for the sake of simulation stability when the dissipative cutoff radius was increased. The effect of the variation of individual DPD parameters on the Sc number is presented in Fig. 3.5, where s varied between 1 and 0.125, r_c^D between 1 and 1.8 and γ between 4.5 and 72. Figure 3.5 shows that the Sc number can be increased by up to two orders of magnitude. As mentioned previously, we also observed a strong dependence of Sc number on N_m as depicted in Fig. 3.6. Dependencies of the Sc number on the DPD parameters were fit with a single-variable power-law, since the number of performed simulations does not allow for a multi-variable fitting.

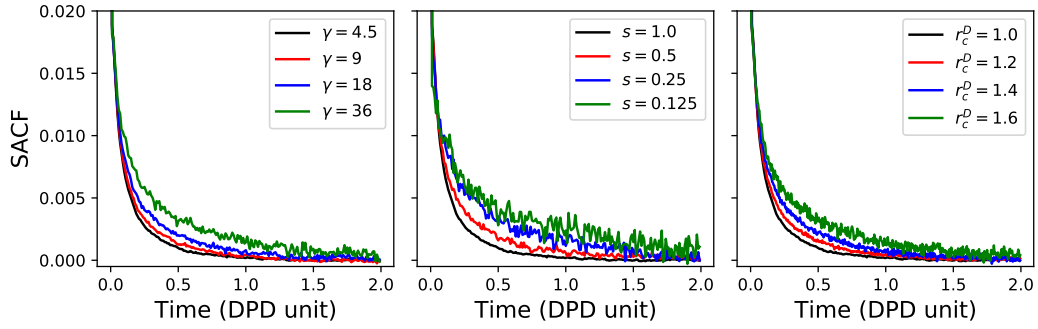


Fig. 3.4 Effects of the (left) dissipative parameter γ , (center) GWF exponent s and (right) dissipative cutoff radius r_c^D on the SACF.

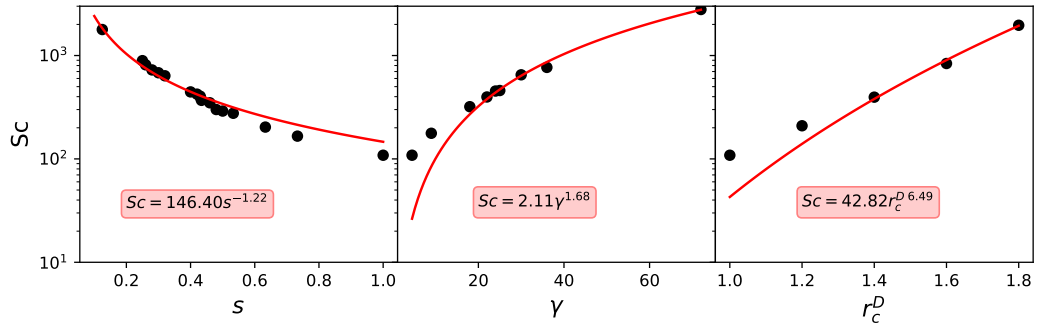


Fig. 3.5 The Schmidt number as a function of (left) dissipative parameter γ , (middle) GWF exponent s and (right) dissipative cutoff radius r_c^D . Dependencies of the Schmidt number on the DPD parameters were fit to the power-law.

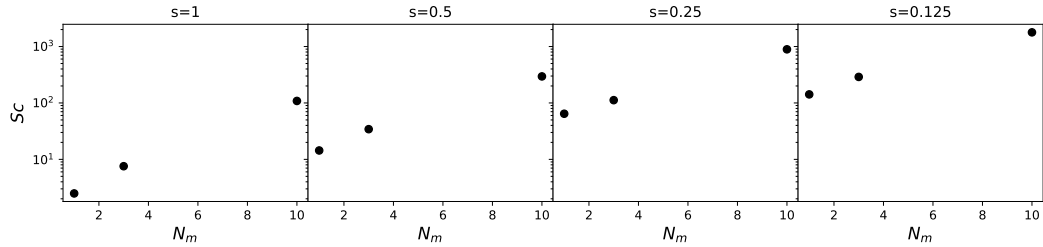


Fig. 3.6 The Schmidt number as a function of the degree of CGing, N_m , for different values of the GWF exponent s .

3.3.3 Tuning water transport properties

In the following, we show in detail the results for the three sets of DPD parameters (highlighted in bold font in Tab. 3.2), that give a Sc number about 370, which corresponds to the Sc number of liquid water at 25°C. The corresponding simulation results are reported in Tab. 3.3 where we compare the viscosity obtained via Green-Kubo approach with that from the Einstein-Helfand and RNEMD methods. We can see that all the viscosities are similar, confirming once again that our procedure based on the Green-Kubo approach can be adopted for evaluating the viscosity. In addition, other aspects are worth to be highlighted. First, the conversion of viscosity and self-diffusion values from DPD units to real units, reported in Tab. 3.3, gives realistic results for liquid water at 25°C. To evaluate the conversion factors, we adopted the approach of Groot and Rabone [76], in which the value of the characteristic length r_c is estimated from the physical volume occupied by ρN_m water molecules, namely:

$$r_c = 3.107(\rho N_m)^{1/3}, \quad (\text{\AA}) \quad (3.8)$$

the characteristic mass of a bead m using the mass of a water molecule $m_w = 2.992 \cdot 10^{-26}$ kg in the following manner:

$$m = N_m m_w, \quad (3.9)$$

and the characteristic time by matching the experimental self-diffusion coefficient $\mathcal{D}^{\text{exp}}$ of the liquid water by using the following expression:

$$\tau = \frac{N_m \mathcal{D}^{\text{DPD}} r_c^2}{\mathcal{D}^{\text{exp}}}. \quad (3.10)$$

The simulation results corresponding to the three sets of DPD parameters are very close to the experimental values of $0.9 \cdot 10^{-6}$ m²/s for viscosity and $2.4 \cdot 10^{-9}$ m²/s for the self-diffusion coefficient of the liquid water. This thus indicates that the transverse DPD thermostat can reproduce the proper momentum and mass transport behavior of liquid water, with at least three different sets of DPD parameters. Another confirmation of this behavior can be seen in Fig. 3.7, which displays the RDFs calculated from the DPD simulations with the three sets of DPD parameters. As it can be seen, the RDFs for all three cases overlap, as expected, since the only

Table 3.3 The properties of DPD fluid corresponding to the three sets of DPD parameters, calibrated against the Schmidt number of liquid water at 25°C equal to about 370.

	Set 1	Set 2	Set 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
v^{DPD} (RNEMD)	5.76	5.26	4.67
v^{DPD} (E-H)	5.74	5.53	5.74
v^{DPD} (G-K)	5.89	5.87	5.88
ν (m ² /s)	$0.895 \cdot 10^{-6}$	$0.892 \cdot 10^{-6}$	$0.893 \cdot 10^{-6}$
$\mathcal{D}^{\text{DPD}}$	0.016	0.016	0.016
$\mathcal{D}$ (m ² /s)	$2.43 \cdot 10^{-9}$	$2.43 \cdot 10^{-9}$	$2.43 \cdot 10^{-9}$
Sc	368.54	366.88	367.70
γ^{eff}	4.36	3.90	4.00

contribution that defines its shape is that of conservative interactions, which are identical in all three cases.

Another important parameter reported in Tab. 3.3 is the so-called effective friction coefficient [77], defined as follows:

$$\gamma^{\text{eff}} = \int_0^{r_c^D} \gamma w(r_{ij}) g(r_{ij}) 4\pi r^2 dr_{ij}, \quad (3.11)$$

where $g(r_{ij})$ is the RDF. The effective friction coefficient is a measure of the overall bead friction and combines the effects of s , r_c^D , γ and implicitly of N_m into one single characteristic quantity. We can see from Tab. 3.3 that values of γ^{eff} for the three sets are rather similar, supporting the following conclusion: sets of parameters (s , r_c^D , γ) having the same γ^{eff} give the same dynamical properties. A last consideration involves the computational time. Set 2 requires more computational resources than the other two cases since Δt for Set 2 is half of Δt for the other cases to keep under control temperature oscillations and to guarantee simulation stability. To allow a consistent comparison of the time statistics for all the three cases, the simulation time was doubled for Set 2. Moreover, a dissipative cutoff radius greater than the conservative cutoff radius in Set 2 decreases the code efficiency, since the use

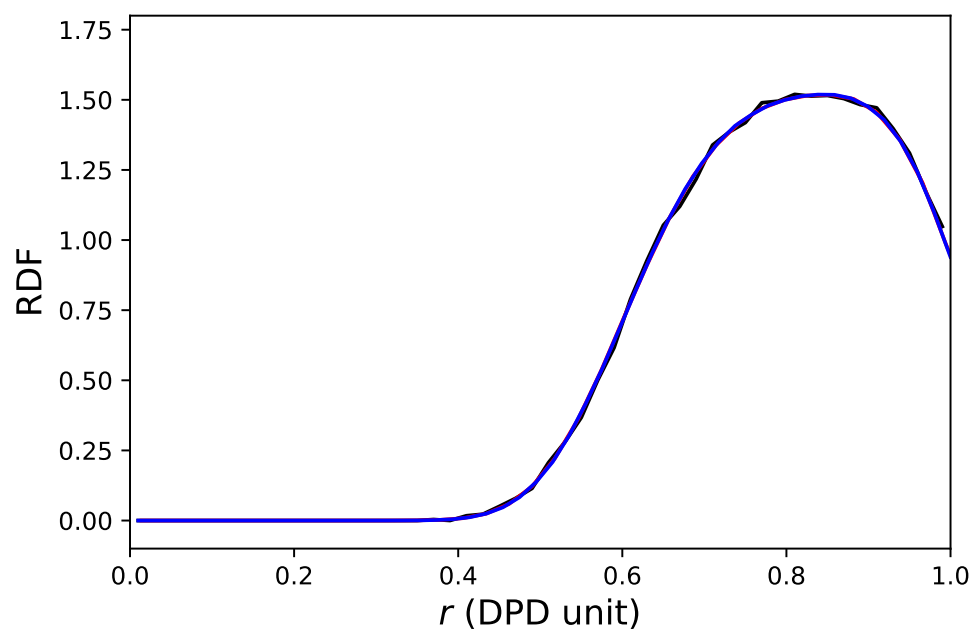


Fig. 3.7 Radial distribution function evaluated for the three sets of DPD parameters, calibrated against the Schmidt number of liquid water at 25°C equal to about 370. Black line: Set 1, red line: Set 2, blue line: Set 3.

of different cutoff radius values for the different forces significantly increases the computational time. Therefore, changing γ or s to increase the Sc number has no or minimal effect on the computational time and simulation stability, unlike tuning r_c^D which can be rather computational demanding.

3.4 Conclusions

The main conclusions of this chapter are summarized in the following.

1. First, we demonstrated that the extended DPD model can be used to simulate fluids with high Sc number, especially when the degree of CGing N_m higher than the standard N_m is adopted. The dependence of the Sc number on the dissipative parameter γ , the dissipative cutoff radius r_c^D and the GWF exponent s was also explored and quantified.
2. Then, we observed that the viscosity coefficient estimated via the Green-Kubo approach is comparable with that obtained by using other methods (i.e., Einstein-Helfand equilibrium and non-equilibrium RNEMD methods) with the advantage that there is no limitation on its usage for equilibrium simulations. This becomes crucial for viscosity prediction in non-Newtonian self-assembled systems. Moreover, the proposed method for treating the noise of the SACF tail is capable of giving not only the mean viscosity value but also the uncertainty of the numerical procedure associated with the viscosity evaluation. In this work, we show that such uncertainty is lower or comparable to the uncertainty of the experimental measurements.
3. Eventually, we observed that three different sets of DPD parameters lead to the same Sc number, RDF and effective friction coefficient. Essentially, we obtained the same fluid with the same rheological behavior with three different sets of DPD parameters. Moreover, we also found that increasing r_c^D to increase the Sc number of the fluid is computational costly when compared to changing γ and s .

3.5 Critical Analysis and Next Steps

To better understand the rationale behind the work presented in Chapters 4 and 5, it is essential to critically analyze the meaning of the aforementioned conclusions in the same order.

1. **Need to explore other time integration schemes.** A correct interpretation of the first conclusion, requires addressing a controversy in the literature concerning the relationship between the repulsive parameter and the degree of CGing. To correctly describe the fluctuations in the system, Groot and Warren related the (like) repulsive parameter, a , to the system's compressibility in their seminal DPD work [13]. Utilizing the Equation-of-State (EoS) for the DPD fluid, Groot and Warren arrived at

$$a = \frac{\kappa^{-1} - 1}{2\alpha\rho} \quad (3.12)$$

where κ^{-1} is the reference dimensionless compressibility, ρ is the bead density, and the EoS parameter $\alpha = 0.101 \pm 0.001$ for $\rho > 2$. Groot and Warren employed the experimental $\kappa^{-1} = 16$ for ambient water which gives $a = 75/\rho$. Later, Groot and Rabone [37] modified Eq. (3.12) by including the level of CGing, Φ , (e.g., the number of water molecules per bead) as

$$a = \frac{\Phi\kappa^{-1} - 1}{2\alpha\rho} \quad (3.13)$$

This limits the use of DPD method since for a large Φ , the system solidifies due to a large a . However, Fuchsli et al. [38] showed that Φ does not need to be taken into account when relating a to κ^{-1} if both a and the cut-off radius r_c are properly scaled by Φ ; specifically as $\Phi^{2/3}a$ and $\Phi^{1/3}r_c$. Then in reduced units, the repulsion parameter becomes invariant under Φ (see Eq. (35) in Ref. [38]), and Eq. (3.12) can be used to relate a to κ^{-1} . In successive works, we aligned our approach with the framework laid out by Fuchsli et al. [38], according to which the interaction parameters a does not scale for any choice of Φ . Consequently, the parameters that influence transport properties are the non-conservative ones. Modifying the Schmidt number to achieve liquid-like dynamics changes the relative magnitude of the applied forces, increasing the

prominence of the dissipative forces. This imbalance between dissipative and random forces, compared to the conservative forces, makes integrating the equations of motion (EOM) particularly challenging. The integration scheme used in this work, which is the most widely adopted in DPD, is not suitable for exploring the full range of non-conservative parameters. Therefore, this highlights the need for improved time integration schemes to simulate systems with high Schmidt numbers. This point is addressed in Chapter 4.

2. **Need to make automatic the G-K based procedure.** The G-K-based procedure proposed in this work demonstrated the reliability of the G-K method compared to benchmark techniques. However, the current approach relies on a manual trial-and-error process for identifying the parameters involved in the calculation, which can lead to wasted computational time. Developing an automated algorithm is essential for performing calculations with controlled accuracy while ensuring computational feasibility. This point is addressed in Chapter 5.
3. **Need to develop a systematic procedure for Schmidt number tuning.** The work presented in the current Chapter demonstrated the influence of three microscopic parameters— s , γ , and r_c^D —on transport properties, with their effects investigated individually. We also showed that different combinations of these parameters can result in the same transport properties, and their combined effect can be represented by a single parameter defined in a mean-field manner. A systematic study of the relationship between transport properties and this γ^{eff} is essential for enhancing the applicability of our findings to simulate fluids with specific Schmidt numbers. This topic is further explored in Chapter 5.

Chapter 4

The role of time integration scheme in the simulation of fluids with increasing Schmidt number.

High Sc number fluids are particularly challenging to simulate due to the imbalanced importance of dissipative and random forces over conservative one, requiring particular attention in the integration of the EoM. In the past few years, a great effort has been devoted to the design of simple, efficient, and accurate numerical methods to integrate the EOM for DPD beads. All methods show spurious effects with increasing time step size, such as unphysical systematic drift of the temperature, uncontrolled spatial correlations between particles, errors in the static and dynamic properties [10]. However, the very small time step size significantly limits the time scales accessible for DPD simulation [78]. Nikunen et al. [79] showed that the performance of the Shardlow's scheme is superior to that of several other schemes for a number of different observables. Recently, Shardlow-like splitting algorithms have been further applied in DPD with various fixed conditions, and its parallel implementation has also been developed [45, 11, 80].

This Chapter presents, for the first time, the application of the Shardlow-splitting algorithm formulated within the velocity-Verlet scheme (VV-SSA) to the extended DPD model, described in Chapter 3 in different friction regimes relying on systematic

The content of this chapter is an adapted version of part of the work published in a peer-reviewed journal. The complete citation is provided below: Lauriello et al., *Computer Physics Communications*, vol. 291, no. 108843, 2023. <https://doi.org/10.1016/j.cpc.2023.108843>.

numerical experiments. The first part of this Chapter details the implementation of the VV-SSA in the open source software LAMMPS [28]. The second part describes the performance assessment of the VV-SSA, in comparison with the most popular modified velocity-Verlet algorithm (VV) with application to the extended DPD model and in different friction regimes relying on systematic numerical experiments.

4.1 Implementation of the VV-SSA integration scheme for the extended DPD model

The standard VV integration scheme has been shown to be relatively accurate in typical MD simulations [81]. Unlike MD, DPD includes not only the conventional conservative forces, but also dissipative and random forces, that in turn depend on the bead velocities and white noise. However, due to its simplicity and good overall performance, the VV algorithm was the starting point for the modified VV scheme [9, 10]. In an attempt to account for the complexity of DPD, the dissipative forces have to be updated for the second time at the end of each integration step by using the up-to-date velocities (momenta), with the first update occurring right after the positions are updated at each step as illustrated in Table 4.1. This considerably improves integration performance, while keeping it computationally efficient, since the additional update of dissipative forces is not particularly time-consuming. For this reason, it is chosen as the reference when a comparison with other integrators is carried out.

Table 4.1 Outline of the modified VV integration scheme (see text for acronym) to update momentum and position of each i -particle ($\mathbf{p}_i^s \rightarrow \mathbf{p}_i^{s+1}; \mathbf{r}_i^s \rightarrow \mathbf{r}_i^{s+1}$).

1. For $i = 1, \dots, N$
a. <i>Momenta First Update:</i> $\mathbf{p}_i^{s+1/2} \leftarrow \mathbf{p}_i^s; \mathbf{f}_i^C(\mathbf{r}^s); \mathbf{f}_i^D(\mathbf{r}^s, \mathbf{p}^s); \mathbf{f}_i^R(\mathbf{r}^s), \text{ where } \mathbf{f}_i^C = \sum_{j \neq i} \mathbf{F}_{ij}^C, \mathbf{f}_i^D = \sum_{j \neq i} \mathbf{F}_{ij}^D \text{ and } \mathbf{f}_i^R = \sum_{j \neq i} \mathbf{F}_{ij}^R$
b. <i>Positions Update:</i> $\mathbf{r}_i^{s+1} \leftarrow \mathbf{r}_i^s; \mathbf{p}_i^{s+1/2}$
2. <i>Force First Calculation:</i> $\{\mathbf{f}_i\}_{i=1}^N$, with $\mathbf{f}_i^C(\mathbf{r}_i^{s+1}); \mathbf{f}_i^D(\mathbf{r}_i^{s+1}, \mathbf{p}_i^{s+1/2}); \mathbf{f}_i^R(\mathbf{r}_i^{s+1})$
3. For $i = 1, \dots, N$
a. <i>Momenta Second Update:</i> $\mathbf{p}_i^{s+1} \leftarrow \mathbf{p}_i^{s+1/2}; \mathbf{f}_i^C(\mathbf{r}^{s+1}); \mathbf{f}_i^D(\mathbf{r}^{s+1}, \mathbf{p}^{s+1/2}); \mathbf{f}_i^R(\mathbf{r}^{s+1})$
4. <i>Dissipative Force Second Calculation:</i> $\{\mathbf{f}_i^D(\mathbf{r}^{s+1}, \mathbf{p}^{s+1})\}_{i=1}^N$

Table 4.2 The modified VV integration scheme (see text for acronym) with detailed integration steps listed. [9, 10].

1. For $i = 1, \dots, N$
a. $\mathbf{p}_i^{s+1/2} = \mathbf{p}_i^s + \frac{1}{2} [\Delta t \mathbf{f}_i^C(\mathbf{r}^s) + \Delta t \mathbf{f}_i^D(\mathbf{r}^s, \mathbf{p}^s) + \sqrt{\Delta t} \mathbf{f}_i^R(\mathbf{r}^s)], \text{ where } \mathbf{f}_i^C = \sum_{j \neq i} \mathbf{F}_{ij}^C, \mathbf{f}_i^D = \sum_{j \neq i} \mathbf{F}_{ij}^D \text{ and } \mathbf{f}_i^R = \sum_{j \neq i} \mathbf{F}_{ij}^R$ b. $\mathbf{r}_i^{s+1} = \mathbf{r}_i^s + \Delta t \frac{\mathbf{p}_i^{s+1/2}}{m_i}$
2. <i>Force Calculation:</i> $\{\mathbf{f}_i\}_{i=1}^N$, with $\mathbf{f}_i^C(\mathbf{r}_i^{s+1}); \mathbf{f}_i^D(\mathbf{r}_i^{s+1}, \mathbf{p}_i^{s+1/2}); \mathbf{f}_i^R(\mathbf{r}_i^{s+1})$
3. For $i = 1, \dots, N$
$\mathbf{p}_i^{s+1} = \mathbf{p}_i^{s+1/2} + \frac{1}{2} [\Delta t \mathbf{f}_i^C(\mathbf{r}^{s+1}) + \Delta t \mathbf{f}_i^D(\mathbf{r}^{s+1}, \mathbf{p}^{s+1/2}) + \sqrt{\Delta t} \mathbf{f}_i^R(\mathbf{r}^{s+1})]$
4. <i>Dissipative Force Updating:</i> $\{\mathbf{f}_i^D(\mathbf{r}^{s+1}, \mathbf{p}_i^{s+1})\}_{i=1}^N$

The basic idea of the Shardlow-splitting algorithm (SSA) is the decomposition of the EoM [Eq. (2.1) and Eq. (2.2)] into differential equations corresponding to the deterministic dynamics and elementary stochastic differential equations (SDEs) corresponding to the fluctuation-dissipation contribution. This approach allows to

adopt larger time step sizes capable of capturing mesoscale events and to recover in this way the practical computational advantage of DPD. The original SSA was formulated for systems with equal-masses particles [45] and successively extended to unequal-mass particles [11]. As reported in Table 4.3 the key idea is to factorize the integration process such that the conservative forces are calculated separately from the dissipative and random terms: the stochastic integration of the dissipative and random forces is performed prior to the deterministic integration of the conservative force. In a typical SSA formulation, both types of differential equations are integrated via the VV algorithm [45]. However, the specific choice of the VV algorithm is not a requirement. Rather, the separated stochastic and deterministic dynamics components may be solved using any traditional MD numerical integration scheme. In this chapter, the SSA was implemented to solve the EoM of the extended DPD thermostat. The stochastic and deterministic differential equations are integrated with the SSA formulated for the VV algorithm (VV-SSA). First, based upon time-splitting, the SSA decomposes the EOM into: differential equations that correspond to the deterministic dynamics; and SDEs that correspond to the stochastic dynamics. Second, based upon operator splitting, the particle stochastic dynamics are updated in a pairwise manner that conserves linear momentum.

Table 4.3 Outline of the VV-SSA integration scheme to update momentum and position of each i -particle ($\mathbf{p}_i^s \rightarrow \mathbf{p}_i^{s+1}; \rightarrow \mathbf{r}_i^s \rightarrow \mathbf{r}_i^{s+1}$) (see text for acronym).

1. *Stochastic Integration: For all $i - j$ pairs of particles*

- a. $\mathbf{p}_i^{s+1/4} \leftarrow \mathbf{p}_i^s; \mathbf{F}_{ij}^D(\mathbf{r}_{ij}^s, \mathbf{v}_{ij}^s); \mathbf{F}_{ij}^R(\mathbf{r}_{ij}^s)$
- b. $\mathbf{p}_j^{s+1/4} \leftarrow \mathbf{p}_j^s; \mathbf{F}_{ij}^D(\mathbf{r}_{ij}^s, \mathbf{v}_{ij}^s); \mathbf{F}_{ij}^R(\mathbf{r}_{ij}^s)$
- c. $\mathbf{v}_{ij}^{s+1/4} \leftarrow \frac{\mathbf{p}_i^{s+1/4}}{m_i} - \frac{\mathbf{p}_j^{s+1/4}}{m_j}$
- d. $\mathbf{p}_i^{s+2/4} \leftarrow \mathbf{p}_i^{s+1/4}; \mathbf{F}_{ij}^D(\mathbf{r}_{ij}^s, \mathbf{v}_{ij}^{s+1/4}); \mathbf{F}_{ij}^R(\mathbf{r}_{ij}^s)$
- e. $\mathbf{p}_j^{s+2/4} \leftarrow \mathbf{p}_j^{s+1/4}; \mathbf{F}_{ij}^D(\mathbf{r}_{ij}^s, \mathbf{v}_{ij}^{s+1/4}); \mathbf{F}_{ij}^R(\mathbf{r}_{ij}^s)$

2. *Deterministic integration #1: For $i = 1, \dots, N$*

- a. $\mathbf{p}_i^{s+3/4} \leftarrow \mathbf{p}_i^{s+2/4} + \frac{\Delta t}{2} \mathbf{f}_i^C(\mathbf{r}^s)$
- b. $\mathbf{r}_i^{s+1} \leftarrow \mathbf{r}_i^s + \Delta t \frac{\mathbf{p}_i^{s+3/4}}{m_i}$

3. *Conservative Force Calculation: $\{\mathbf{f}_i^C(\mathbf{r}^{s+1})\}_{i=1}^N$, where $\mathbf{f}_i^C = \sum_{j \neq i} \mathbf{F}_{ij}^C$*

4. *Deterministic integration #2: For $i = 1, \dots, N$*

$$\mathbf{p}_i^{s+1} \leftarrow \mathbf{p}_i^{s+3/4} + \frac{\Delta t}{2} \mathbf{f}_i^C(\mathbf{r}^{s+1})$$

Table 4.4 The VV-SSA integration scheme for transverse DPD thermostat (see text for acronym) with detailed integration steps listed [11, 12].

1. *Stochastic Integration: For all $i - j$ pairs of particles*

$$\begin{aligned}
 \text{a. } \mathbf{p}_i^{s+1/4} &= \mathbf{p}_i^s - \frac{\Delta t}{2} \gamma_{\parallel} w_{\parallel}^2(r_{ij}^s) (\hat{\mathbf{r}}_{ij}^s \cdot \mathbf{v}_{ij}^s) \hat{\mathbf{r}}_{ij}^s - \frac{\Delta t}{2} \gamma_{\perp} w_{\perp}^2(r_{ij}^s) (\mathbf{I} - \hat{\mathbf{r}}_{ij}^s \hat{\mathbf{r}}_{ij}^{T,s}) \mathbf{v}_{ij}^s + \sigma_{\parallel} w_{\parallel}(r_{ij}^s) \xi_{ij}^s \hat{\mathbf{r}}_{ij}^s \frac{\sqrt{\Delta t}}{2} + \\
 &\quad \sigma_{\perp} w_{\perp}(r_{ij}^s) \xi_{ij}^s (\mathbf{I} - \hat{\mathbf{r}}_{ij}^s \hat{\mathbf{r}}_{ij}^{T,s}) \frac{\sqrt{\Delta t}}{2} \\
 \text{b. } \mathbf{p}_j^{s+1/4} &= \mathbf{p}_j^s + \frac{\Delta t}{2} \gamma_{\parallel} w_{\parallel}^2(r_{ij}^s) (\hat{\mathbf{r}}_{ij}^s \cdot \mathbf{v}_{ij}^s) \hat{\mathbf{r}}_{ij}^s + \frac{\Delta t}{2} \gamma_{\perp} w_{\perp}^2(r_{ij}^s) (\mathbf{I} - \hat{\mathbf{r}}_{ij}^s \hat{\mathbf{r}}_{ij}^{T,s}) \mathbf{v}_{ij}^s - \sigma_{\parallel} w_{\parallel}(r_{ij}^s) \xi_{ij}^s \hat{\mathbf{r}}_{ij}^s \frac{\sqrt{\Delta t}}{2} - \\
 &\quad \sigma_{\perp} w_{\perp}(r_{ij}^s) \xi_{ij}^s (\mathbf{I} - \hat{\mathbf{r}}_{ij}^s \hat{\mathbf{r}}_{ij}^{T,s}) \frac{\sqrt{\Delta t}}{2} \\
 \text{c. } \mathbf{v}_{ij}^{s+1/4} &= \frac{\mathbf{p}_i^{s+1/4}}{m_i} - \frac{\mathbf{p}_j^{s+1/4}}{m_j} \\
 \text{d. } \mathbf{p}_i^{s+2/4} &= \mathbf{p}_i^{s+1/4} - \frac{\Delta t}{2} \frac{\gamma_{\parallel} w_{\parallel}^2(r_{ij}^s)}{1 + \frac{\mu_{ij}}{2} \gamma_{\parallel} w_{\parallel}^2(r_{ij}^s) \Delta t} \left[(\hat{\mathbf{r}}_{ij}^s \cdot \mathbf{v}_{ij}^{s+1/4}) \hat{\mathbf{r}}_{ij}^s + \frac{\mu_{ij}}{2} \sigma_{\parallel} w_{\parallel}(r_{ij}^s) \xi_{ij}^s \hat{\mathbf{r}}_{ij}^s \sqrt{\Delta t} \right] + \sigma_{\parallel} w_{\parallel}(r_{ij}^s) \xi_{ij}^s \hat{\mathbf{r}}_{ij}^s \frac{\sqrt{\Delta t}}{2} - \\
 &\quad \frac{\Delta t}{2} \frac{\gamma_{\perp} w_{\perp}^2(r_{ij}^s)}{1 + \frac{\mu_{ij}}{2} \gamma_{\perp} w_{\perp}^2(r_{ij}^s) \Delta t} \left[(\mathbf{I} - \hat{\mathbf{r}}_{ij}^s \hat{\mathbf{r}}_{ij}^{T,s}) \mathbf{v}_{ij}^{s+1/4} + \frac{\mu_{ij}}{2} \sigma_{\perp} w_{\perp}(r_{ij}^s) \xi_{ij}^s (\mathbf{I} - \hat{\mathbf{r}}_{ij}^s \hat{\mathbf{r}}_{ij}^{T,s}) \sqrt{\Delta t} \right] + \sigma_{\perp} w_{\perp}(r_{ij}^s) \xi_{ij}^s (\mathbf{I} - \hat{\mathbf{r}}_{ij}^s \hat{\mathbf{r}}_{ij}^{T,s}) \frac{\sqrt{\Delta t}}{2} \\
 &\quad \text{where } \mu_{ij} = \frac{1}{m_i} + \frac{1}{m_j} \\
 \text{e. } \mathbf{p}_j^{s+2/4} &= \mathbf{p}_j^{s+1/4} + \frac{\Delta t}{2} \frac{\gamma_{\parallel} w_{\parallel}^2(r_{ij}^s)}{1 + \frac{\mu_{ij}}{2} \gamma_{\parallel} w_{\parallel}^2(r_{ij}^s) \Delta t} (r_{ij}^s) \left[(\hat{\mathbf{r}}_{ij}^s \cdot \mathbf{v}_{ij}^{s+1/4}) \hat{\mathbf{r}}_{ij}^s + \frac{\mu_{ij}}{2} \sigma_{\parallel} w_{\parallel}(r_{ij}^s) \xi_{ij}^s \hat{\mathbf{r}}_{ij}^s \sqrt{\Delta t} \right] - \\
 &\quad \sigma_{\parallel} w_{\parallel}(r_{ij}^s) \xi_{ij}^s \hat{\mathbf{r}}_{ij}^s \frac{\sqrt{\Delta t}}{2} + \frac{\Delta t}{2} \frac{\gamma_{\perp} w_{\perp}^2(r_{ij}^s)}{1 + \frac{\mu_{ij}}{2} \gamma_{\perp} w_{\perp}^2(r_{ij}^s) \Delta t} \left[(\mathbf{I} - \hat{\mathbf{r}}_{ij}^s \hat{\mathbf{r}}_{ij}^{T,s}) \mathbf{v}_{ij}^{s+1/4} + \frac{\mu_{ij}}{2} \sigma_{\perp} w_{\perp}(r_{ij}^s) \xi_{ij}^s (\mathbf{I} - \hat{\mathbf{r}}_{ij}^s \hat{\mathbf{r}}_{ij}^{T,s}) \sqrt{\Delta t} \right] - \\
 &\quad \sigma_{\perp} w_{\perp}(r_{ij}^s) \xi_{ij}^s (\mathbf{I} - \hat{\mathbf{r}}_{ij}^s \hat{\mathbf{r}}_{ij}^{T,s}) \frac{\sqrt{\Delta t}}{2}
 \end{aligned}$$

2. *Deterministic integration #1: For $i = 1, \dots, N$*

$$\begin{aligned}
 \text{a. } \mathbf{p}_i^{s+3/4} &= \mathbf{p}_i^{s+2/4} + \frac{\Delta t}{2} \mathbf{f}_i^C(\mathbf{r}^s) \\
 \text{b. } \mathbf{r}_i^{s+1} &= \mathbf{r}_i^s + \Delta t \frac{\mathbf{p}_i^{s+3/4}}{m_i}
 \end{aligned}$$

3. *Conservative Force Calculation: $\{\mathbf{f}_i^C(\mathbf{r}^{s+1})\}_{i=1}^N$, where $\mathbf{f}_i^C = \sum_{j \neq i} \mathbf{F}_{ij}^C$*

4. *Deterministic integration #2: For $i = 1, \dots, N$*

$$\mathbf{p}_i^{s+1} = \mathbf{p}_i^{s+3/4} + \frac{\Delta t}{2} \mathbf{f}_i^C(\mathbf{r}^{s+1})$$

The VV-SSA was available in LAMMPS only for standard DPD model. A new package was implemented in LAMMPS to make the VV-SSA suitable for the numerical integration of the extended DPD EoM, described in Chapter 3. The VV-SSA algorithm, described in this paragraph, was newly implemented in LAMMPS for the `dpdext/fdt pair_style`. The code is available online on Github¹.

¹<https://github.com/mulmopro/LAMMPS-DPD-EXT>

4.2 Assessing the performance of the VV-SSA in the calculation of transport properties

4.2.1 Simulation details

In this chapter, some numerical experiments were performed to test the time step size stability threshold and the performance of the VV-SSA algorithm, newly implemented in LAMMPS by us for the `dpdext/fdt pair_style`, in comparison with the most popular VV integrator used in many previous papers. Several studies evaluated the integrators, as traditionally done, based on temperature conservation [78, 79, 44]. Stability and accuracy can be achieved using longer time steps with the VV-SSA [80]. As the friction increases in the system the VV algorithm takes more iterations to stabilize the temperature [82] with a smaller time step compared to low friction condition [10, 9]. Although evaluation against dynamic fluid transport properties is essential to have a measure of the dynamical fidelity that numerical methods are able to reproduce, few such studies limited to low friction regime were performed. Furthermore the comparison between VV-SSA and other integration schemes was presented only with application to the standard DPD thermostat [78, 82]. Since to have a deeper understanding of how the physical system evolves it is not enough to just evaluate static quantities, in this analyses, particular attention was devoted to the performance of the investigated time integration schemes in fluid transport properties evaluation with application to the transverse DPD thermostat. In particular, the behavior of the VV-SSA was examined in comparison to the VV for measuring the SACF, which is related to the zero shear viscosity η . Of course, also important static quantities, such as RDF were considered in the analysis. The comparison between integrators was performed investigating also the effect of the dissipative strength γ , the exponent of the GWF s and the dissipative cutoff radius r_c^D .

4.2.2 Results and discussion

The effect of time integration schemes on SACF calculation is analyzed. Figures 4.1, 4.2, 4.3 and 4.4 compare the performance of the two time integration schemes investigated, the VV and the VV-SSA, in terms of SACF calculation. This analysis has been carried out considering two different simple fluids, having two

different values of γ_{eff} , resulting in two different Sc numbers. In this study, the values of N_{rep} and N_{sim} , used to numerically evaluate the SACF, are high enough to ensure significant statistics. For very small values of the effective friction coefficient, Figs. 4.1 and 4.2, the SACFs computed with two different time integration schemes exhibit nearly identical decay profiles. In contrast, significant differences emerge in the high-friction region, as is seen from Figs. 4.3 and 4.4. Therefore, the difference in performance between the two integrators becomes more evident as the γ_{eff} of the system increases, requiring the use of smaller time step sizes. The higher friction results in a higher dissipation, and hence in a faster decorrelation of shear stresses in the system, requiring a smaller time step size in order to have a better temporal resolution of the SACF. Indeed, the smallest relevant time scale in the system is the momentum relaxation time, τ_m which is related to the friction in the system, and can be evaluated as follows [83]:

$$\tau_m = m\gamma^{-1}\rho^{-1}r_c^{-3}. \quad (4.1)$$

The definition of γ_{eff} is [84]:

$$\gamma_{\text{eff}} = \int_0^{r_c^D} \gamma w(r_{ij}) g(r_{ij}) 4\pi r^2 dr_{ij}, \quad (4.2)$$

where $g(r_{ij})$ is the RDF. Comparing the definitions of γ_{eff} and τ_m in Eqs. 4.2 and 4.1, it is evident that τ_m is proportional to γ_{eff}^{-1} . Since γ_{eff} is a measure of the overall friction of the beads in the system, the Sc number is expected to increase for higher values of γ_{eff} . The Sc numbers obtained for the two investigated values of γ_{eff} are in line with expectation. One can conclude that the simulation of high Schmidt number fluids requires the use of smaller time step sizes compared to low Schmidt number fluids. The VV displays integrator-induced artifacts such as increases in γ_{eff} and Sc, since smaller time step sizes are required. The SACF was found to be subject to increasing fluctuations with increasing friction in the case of the VV integrator. The mean of these fluctuations was close to zero at long times. The frequency of SACF data accumulation and the time step size of integration affect the temporal resolution of the SACF. In this study N_{ev} is fixed to 1 and the effect of the time integration scheme on the required time step size is investigated. $N_{\text{rep}} = N_{\text{freq}}$ are chosen high enough to guarantee the convergence of the Cumulative Viscosity Integral (CVI). The main purpose of this part of the thesis is proving that the newly implemented

VV-SSA allows the simulation of high Schmidt number fluids with an higher time step size with respect to the standard VV.

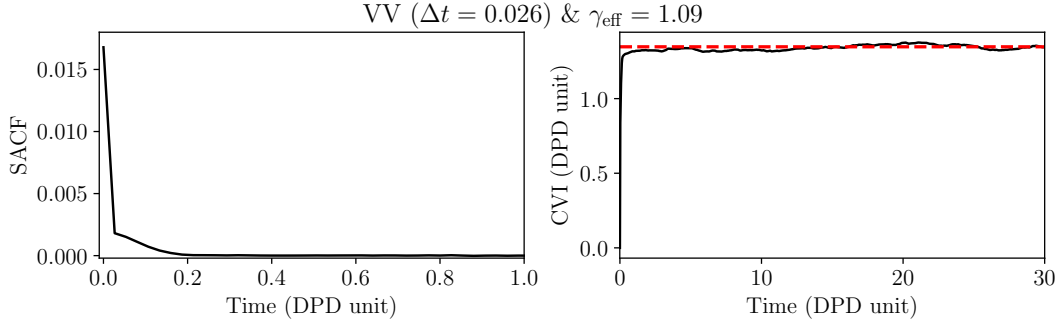


Fig. 4.1 SACF and CVI for VV at $\gamma_{\text{eff}} = 1.09$.

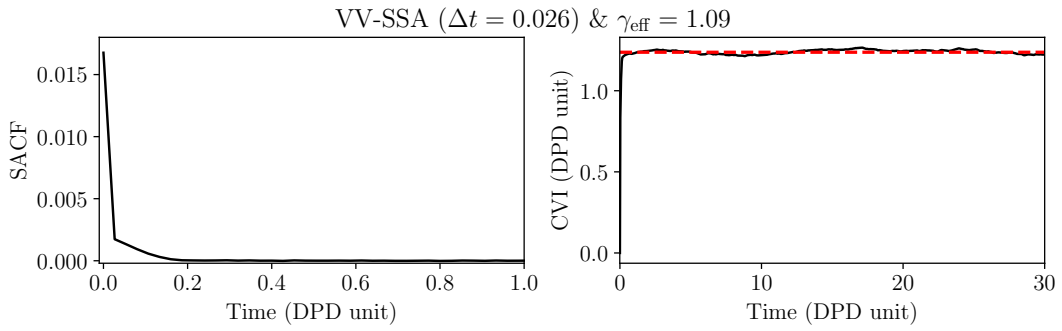
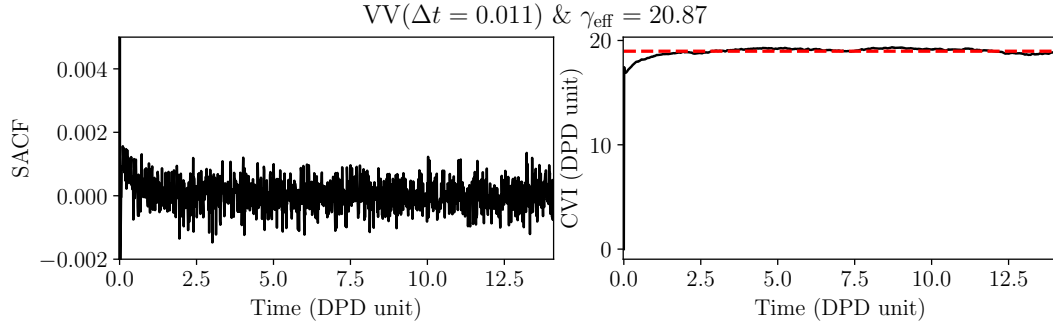
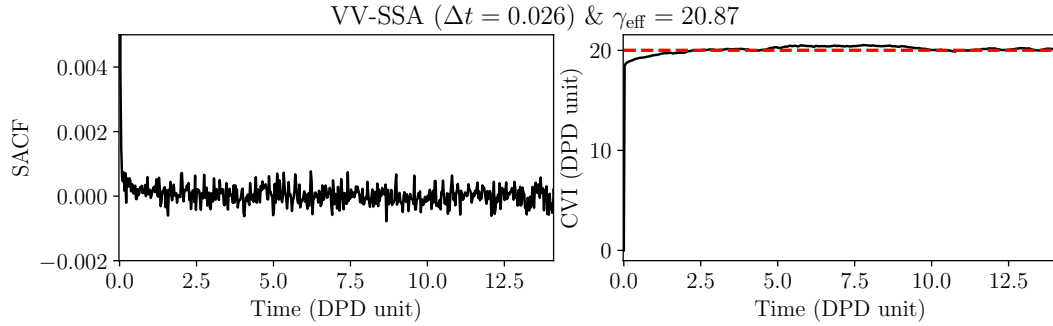


Fig. 4.2 SACF and CVI for VV-SSA at $\gamma_{\text{eff}} = 1.09$.

Figures 4.1 and 4.2 demonstrate that in the case of low γ_{eff} , which corresponds to a low Sc number, the difference in performance of the VV-SSA with respect to the VV is negligible. In this case, there is no reason to prefer the VV-SSA to the VV in terms of efficiency.

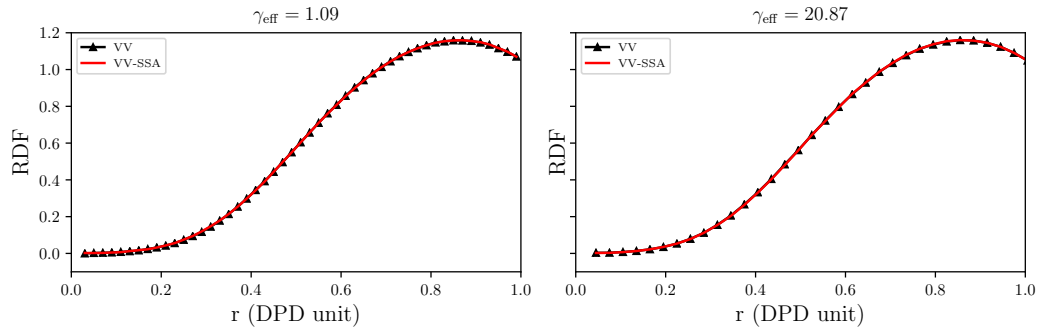
Fig. 4.3 SACF and CVI for VV at $\gamma_{\text{eff}} = 20.87$.Fig. 4.4 SACF and CVI for VV-SSA at $\gamma_{\text{eff}} = 20.87$.

Figures 4.3 and 4.4 show that for high γ_{eff} values, which correspond to high Schmidt numbers, the VV-SSA method outperforms the VV method. The tail oscillations of the SACF are noticeably less noisy. Additionally, the N_{rep} value required to achieve CVI convergence is nearly half of that needed for the VV method. Although the time integration scheme demands double the effort compared to VV, the same viscosity accuracy can be obtained with fewer simulation steps. Overall, the newly implemented VV-SSA scheme proves to be superior to the VV method for simulating high Schmidt number fluids. The most significant parameter to highlight the difference of the two integration schemes are summarized in Tab. 4.5.

The effect of time integration schemes on RDF calculation in different friction regions is also analyzed. Figure 4.5 compares the RDFs obtained in different cases. The two integrators display exactly the same RDF for each effective friction coefficient value investigated, indicating that different systems have the same structure, as expected, despite the employed integrator and Sc . This is expected since the two investigated cases have the same conservative force parameters.

Table 4.5 Comparison between VV and VV-SSA performance.

Integrator	γ_{eff}	Time step size	N_{rep}	Step number	Viscosity	Standard deviation
VV	1.09	0.026	1200	480000	1.33	0.001
VV-SSA	1.09	0.026	1200	480000	1.26	0.001
VV	20.87	0.011	2400	1320000	19.74	0.04
VV-SSA	20.87	0.026	1200	600000	19.16	0.01

Fig. 4.5 Comparison of RDF for VV and VV-SSA integrators at different values of γ_{eff} ; $\gamma_{\text{eff}} = 1.09$ (left), $\gamma_{\text{eff}} = 20.87$ (right).

4.3 Conclusions

In this chapter of the dissertation, we compared the results obtained by two time integration methods for dissipative particle dynamics (DPD), the newly implemented Shardlow-splitting algorithm (VV-SSA) and the commonly used modified velocity-Verlet algorithm (VV), for the solution of the extended DPD model presented in the previous chapter.

We demonstrated the superiority of the VV-SSA in the evaluation of the transport properties of fluids with high Schmidt numbers. More specifically, we found that the VV-SSA reduces the integrator-induced artifacts and performs better in a wider friction region than the VV.

Moreover, larger values of the time step can be adopted in the calculation of transport properties with VV-SSA. This aspect preserves the main advantage of DPD. In fact, in this technique, soft inter-particle forces neither change abruptly nor diverge at short distances, which permits long integration steps as compared to the atomistic simulations. This fact appreciably accelerates the simulations and, together

with the coarse-graining, allows for the study of mesoscale phenomena, making it appealing for modeling the behavior of structured fluids.

Chapter 5

Development of an automated method for transport properties calculation: application to simple fluids.

In some research works [85–87], the reliability of the G-K relations in DPD simulations was doubted, since the presence of stochastic forces creates noise for the resulting stress tensor, generating high statistical errors in the SACF viscosity calculation. To make clearer the rationale behind the method developed in this Chapter, it is worth quickly remarking the main issues related to the G-K method in DPD simulations.

Macroscopic transport coefficients can be calculated from DPD simulations using numerical techniques developed for atomistic simulations. Viscosity can be measured by applying equilibrium and non-equilibrium methods [86]. Non-equilibrium methods are relatively straightforward to implement and do not suffer from statistical uncertainty. At the same time, the simulation setup can be intricate, and it is specific for the property investigated. In fact, non-equilibrium simulations reproduce the experimental setup where external gradients are established giving rise to conjugate fluxes or, conversely, fluxes are imposed which result in conjugate gradients. In contrast, equilibrium methods such as the Green-Kubo (G-K) and Einstein-Helfand (E-H) approaches rely on the measurement of the decay of the spontaneous thermal

The content of this chapter is part of the work published in a peer-reviewed journal. The complete citation is provided below: Lauriello et al., *Computer Physics Communications*, vol. 291, no. 108843, 2023. <https://doi.org/10.1016/j.cpc.2023.108843>.

fluctuations of the corresponding physical property, and they do not require any special simulation setup to create the externally imposed gradients. As a consequence, a single equilibrium simulation with periodic boundary conditions can provide a plethora of transport coefficients for different properties, as commonly found in open-source molecular dynamics codes. Moreover, the equilibrium methods in self-assembled systems allow the estimation of the viscosity in equilibrium without altering their self-assembled equilibrium microstructures. In this Chapter, the G-K approach [50, 88] is used, which allows to calculate the viscosity by numerically integrating the SACF as

$$\eta = \frac{V}{k_B T} \int_0^{+\infty} \langle \Pi_{\alpha\beta}(t_0) \Pi_{\alpha\beta}(t_0+t) \rangle dt, \quad (\alpha \neq \beta), \quad (5.1)$$

where V is the system volume and $\Pi_{\alpha\beta}$ is the off-diagonal component of the stress tensor. Alternative G-K formulas, derived independently in literature [86, 85, 89], exist. It is worth mentioning the alternative G-K formula for η in DPD provided by Ref. [89]. It was obtained by splitting the stress tensor $\mathbf{\Pi}$ into conservative, dissipative, and random force contributions as

$$\begin{aligned} \Pi_{zx} &= \frac{1}{V} \sum_i \left(\frac{p_i^z p_i^x}{m_i} + \sum_{j<i} z_{ij} f_{ij}^{x,C} \right) + \frac{1}{V} \sum_{i,j<i} z_{ij} f_{ij}^{x,D} + \frac{1}{V} \sum_{i,j<i} z_{ij} f_{ij}^{x,R} \\ &= \Pi_{zx}^C + \Pi_{zx}^D + \Pi_{zx}^R \end{aligned} \quad (5.2)$$

and, similarly as done by Ernst and Brito [85], by pre-averaging correlations due to the random contribution. This leads to the alternative G-K formula for η in DPD:

$$\eta = \eta_\infty + \eta^{CC} + \eta^{CD} + \eta^{DC} + \eta^{DD} + \eta^{res}, \quad (5.3)$$

where

$$\begin{aligned} \eta^{AB} &= \frac{V}{k_B T} \int_0^\infty \langle \Pi_{zx}^A(t) \Pi_{zx}^B(0) \rangle dt, \\ \eta^{res} &= \frac{V}{k_B T} \int_0^\infty \langle (\Pi_{zx}^C(t) + \Pi_{zx}^D(t)) \Pi_{zx}^R(0) \rangle dt, \end{aligned}$$

and

$$\eta_{\infty} = \frac{V}{k_B T} \left\langle (\Pi_{zx}^R)^2 \right\rangle \frac{\delta t}{2}.$$

Eq. (5.3) was demonstrated to be the same Ref. [89] of the Ernst-Brito (E-B) G-K formula for η in DPD [85, 86]

$$\eta = \eta_{\infty} + \eta^{CC} - \eta^{CD} + \eta^{DC} - \eta^{DD}, \quad (5.4)$$

which was also obtained via the splitting of Π and evaluating all contributions of correlations due to the random contribution analytically. Implementation of the alternative G-K formulas, Eqs. (5.3) and (5.4), in an automated computational protocol is not straightforward. Therefore, we employed traditional G-K formula, Eq. (5.1), for practical reasons. The automated method for transport properties calculation from DPD equilibrium simulations with controlled precision adopted in this work [90], requires the outputting of the full stress tensor and on-fly evaluation of the SACF. Standard simulation packages such as LAMMPS [28] allows outputting the full stress tensor and on-fly evaluation of its correlation function. This allowed us constructing the automated procedure for DPD viscosity simulations with controlled precision. The alternative G-K formulas, although theoretically superior to the standard G-K formula, require post-processing of outputting values of the stress tensor or its conservative, dissipative, and random force contributions, which in turn may require modifications of the simulation packages and preclude their use within the automated procedure.

The focal points for the viscosity calculation based on G-K method are the achievement of a good statistics in the SACF with the minimum computational cost, and the proper treatment of the noise in the numerical evaluation of the time integral of the SACF. The calculation of such time integral with a simple trapezoidal rule is affected by the SACF tail oscillations and therefore by the selection of the last integration point. To overcome these issues, in our previous work [12] a procedure based on the calculation of the Cumulative Viscosity Integral in function of time (CVI) was proposed and this work aims to automatize and optimize it. Particular attention was devoted to this aspect since the collective correlations, such as the SACF, in literature studies are often determined with poorer statistics than single-particle ones, such as the velocity autocorrelation function, because of the extensive computational

requirements. As mentioned in Chapter 3, the proper evaluation of the SACF, and in turn of the viscosity, using the approximate algorithm implemented in the open source code LAMMPS depends on the choice of the parameters N_{ev} , N_{rep} and N_{freq} . In fact, LAMMPS makes use of three parameters to compute the SACF (N_{ev} , N_{rep} , N_{freq}). Further details on this can be found in the LAMMPS manual [28]. These parameters should be defined by the user, together with the simulation length given by $N_{freq} \times N_{sim}$, where N_{sim} is the number of times the SACF averages are computed, and the time step size Δt . The N_{ev} determines the data accumulation frequencies and for DPD simulations, $N_{ev} = 1$ since the random contribution to η is only non-zero at the first time step by construction, see Eq. (38) in Ref. [91]. Setting $N_{ev} = 1$, the choice of other quantities is crucial to entirely capture the SACF behavior with the proper temporal resolution and a sufficiently good statistics. N_{freq} is always taken equal to N_{rep} . Also the simulation length affects the statistics, and is given by $N_{sim} \times N_{freq}$, where N_{sim} is the number of times the SACF averages are computed. As it will be shown in the following, for each DPD system there is a certain minimum N_{rep} value that guarantees SACF values approach to zero in the decorrelation time interval (the viscosity converges to a mean value), and a certain minimum N_{sim} value sufficiently high to smooth instantaneous tail fluctuations giving an accurate statistics (the standard deviation and the coefficient of variation associated to the mean value decreases). These two parameters N_{rep} and N_{sim} must be sufficiently high to ensure that the mean value of viscosity achieves a plateau value and to reduce the associated standard deviation and coefficient of variation. However, they should not surpass necessary values to prevent excessive computational time and unphysical results. Moreover in this context, also the time step size, Δt , plays a crucial role. It is required a certain level of temporal resolution to obtain a value of viscosity that can be considered independent from the simulation time step size. The "optimal" values of Δt , N_{rep} and N_{sim} depend on the simulated fluid. Fluids with different Schmidt numbers require different parameters. The identification of these parameters before running the simulation is a non-trivial task, and a trial-and-error approach can lead to a waste of computational resources.

This Chapter presents an automated method designed to overcome this problem, since it is based on an algorithm that calculates automatically the minimum values of N_{rep} and N_{sim} in a DPD simulation of a fluid in order to deliver a value of viscosity within an acceptable range of uncertainty. In fact, the algorithm identifies the simulation parameters required to produce the shortest trajectory to achieve

the fixed statistical accuracy in the calculation of the SACF and, consequently, of the viscosity. Additionally, this Chapter details a comprehensive multi-parametric study leveraging the developed method, from which we derived fitting relations for viscosity, self-diffusivity, and the Schmidt number as functions of the effective friction coefficient.

5.1 Description of the method

The automated method makes use of LAMMPS for performing DPD simulations and of a Python script for the management of simulations and post-processing. The workflow of the method is schematized in Fig. 5.1, while its pseudo-code is reported in Algorithm 1. The algorithm is based on three main iterative loops: an external control loop and two internal consecutive control loops. The first is indicated as the Δt -loop and the internal loops as N_{rep} -loop and N_{sim} -loop. The Δt -loop executes different simulations progressively reducing the time step size, in order to find the largest time step for which the viscosity value does not change anymore, namely a time step independent value of viscosity η_{fin} , which is the objective of the aforementioned external loop and more in general of the automated algorithm.

The two inner control loops instead, the N_{rep} -loop and the N_{sim} -loop, give at the end a value of viscosity, called intermediate viscosity, η_{int} , which is the out of loop condition when both subsequent loops are satisfied. The first control loop, the N_{rep} -loop, progressively increases the value of N_{rep} starting from an arbitrary initial guess of N_{rep} , N_{sim} and Δt , and checks if CVI has the expected asymptotic behavior, with the final points of the curve that do not change anymore with time. This is equivalent to capture entirely the expected behavior of the SACF, from the initial condition of maximum stress correlation up to complete decorrelation. If the time interval analyzed by the SACF is not sufficiently large to trace the whole decorrelation dynamics, the tail of the SACF may not reach zero, in turn leading to a CVI curve that still increases in function of time. The size of the time interval to be analyzed is determined by the choice of N_{rep} , which is the value that the N_{rep} -loop tries to identify. The optimal N_{rep} value is the one that includes the full decorrelation behaviour of the SACF in the shortest analyzed time interval. To establish if the CVI asymptote is reached, a condition referred in the following as viscosity plateau, the algorithm uses a special adaptation of the Simple Moving Average method. This

method creates, for each value of N_{rep} , a series of averages, η_{avg} of different time intervals (*Averaging Windows* or *Subsets*) of a fixed extension (*Window Size*) of the full CVI curve and checks the relative error, RE_{CVI} , between them. Specifically, the user must define the distance between subsets, i.e., the *Sliding Window*, that is moved forward in the calculation of the *Current Subset* after the calculation of the average on the previous one, *Reference Subset*. A certain number of subsets, defined by the user through the *Jump Index*, is jumped between the *Reference Subset*=*Current Subset* - *Jump Index* and *Current Subset*, to avoid the possible but undesired identification of a local plateau in the CVI. If the value of the RE_{CVI} is over a chosen threshold, another simulation with an increased value of N_{rep} with increment h_{rep} is performed. This iterative mechanism continues until the RE_{CVI} is below the desired threshold: when this eventually happens the algorithm passes to the next control loop the viscosity plateau, η_{plt} and the reference of its subset, indicated as *Plateau Subset*, together with the associated standard deviation σ_{plt} . In this way, the point up to which integrate the SACF is automatically established and indicated as the viscosity UI point, the upper integration bound of the viscosity integral. There exists the possibility that η_{plt} is not reached in a maximum number of iterations, again decided by the user: in this case, the algorithm firstly increments N_{sim} and then restarts the N_{rep} -loop from the last N_{rep} investigated. We encountered this situation only in a few times during our extensive testing of the algorithm, mainly with fluids with very high viscosity, because such fluids requires from the beginning a higher guess value for N_{sim} . The N_{sim} -loop has a similar structure of the N_{rep} -loop but it serves a different purpose. The act of increasing the N_{sim} value of h_{sim} is performed in order to reduce the noise of the CVI, quantified with the coefficient of variation (COV) of the η_{plt} , namely the ratio between the σ_{plt} and the η_{plt} . The check is done on the absolute error between the COV of two consecutive simulations, and this quantity must be lower than a certain threshold defined by the user, AE_{COV} . Eventually the output of the N_{sim} loop is the previously mentioned η_{int} associated to the current Δt , which is then fed to the external Δt -loop. The Δt -loop keeps iterating until the relative error between two consecutive η_{int} values reaches a threshold specified by the user, $RE_{\eta_{int}}$.

The algorithm returns a value of η_{fin} sufficiently accurate, according to the value of $RE_{\eta_{avg}}$, AE_{COV} and $RE_{\eta_{int}}$ that the user has decided to set. The choice of the N_{rep} , N_{sim} and Δt initial guess is arbitrary done by the user and the results do not depend on their choice. However, the time-to-solution of the algorithm depends on the goodness of the initial guess setup that should be guided from preliminary tests

on the system of interest, as done in this work. The values of h_{rep} and h_{sim} have to be significant for the system simulated. In this work $h_{\text{rep}} = 500$ and $h_{\text{sim}} = 50$ are chosen as increments of the respective loops, since resulting, from a preliminary study on the SACF behaviour in the case of simple fluids, as the minimum values giving a significant improvement of the SACF statistic. The threshold values for the out-of-loop conditions $\text{RE}_{\eta_{\text{avg}}}$, $\text{RE}_{\eta_{\text{int}}}$ and AE_{COV} are chosen as a compromise between the desired accuracy of the viscosity estimate and the computational cost required. Indeed, the checks performed by the automated method on the computed viscosity through the three out-of-loop conditions fix the level of acceptability of the error associated resulting from the whole numerical methodology. The error in the G-K approach is mainly due to inaccuracies in the calculation of the SACF. Such error is quantified with the coefficient of variation (COV) of the viscosity estimate, namely the ratio between the standard deviation and the mean viscosity value. The calculation of the viscosity is reiterated with different combinations of the parameters defining the time interval analyzed by the SACF and the simulation time, until the condition on its coefficient of variation is satisfied. The same procedure is repeated with different time step sizes until the time resolution suitable for the specific system simulated is reached. The optimum threshold values should be the highest below which there is no significant error reduction. In this work we found that $\text{RE}_{\eta_{\text{avg}}} = 0.01$, $\text{RE}_{\eta_{\text{int}}} = 0.05$ and $\text{AE}_{\text{COV}} = 0.01$ work best in our investigation of a simple DPD fluid with Schmidt number from about 1 to 3881.

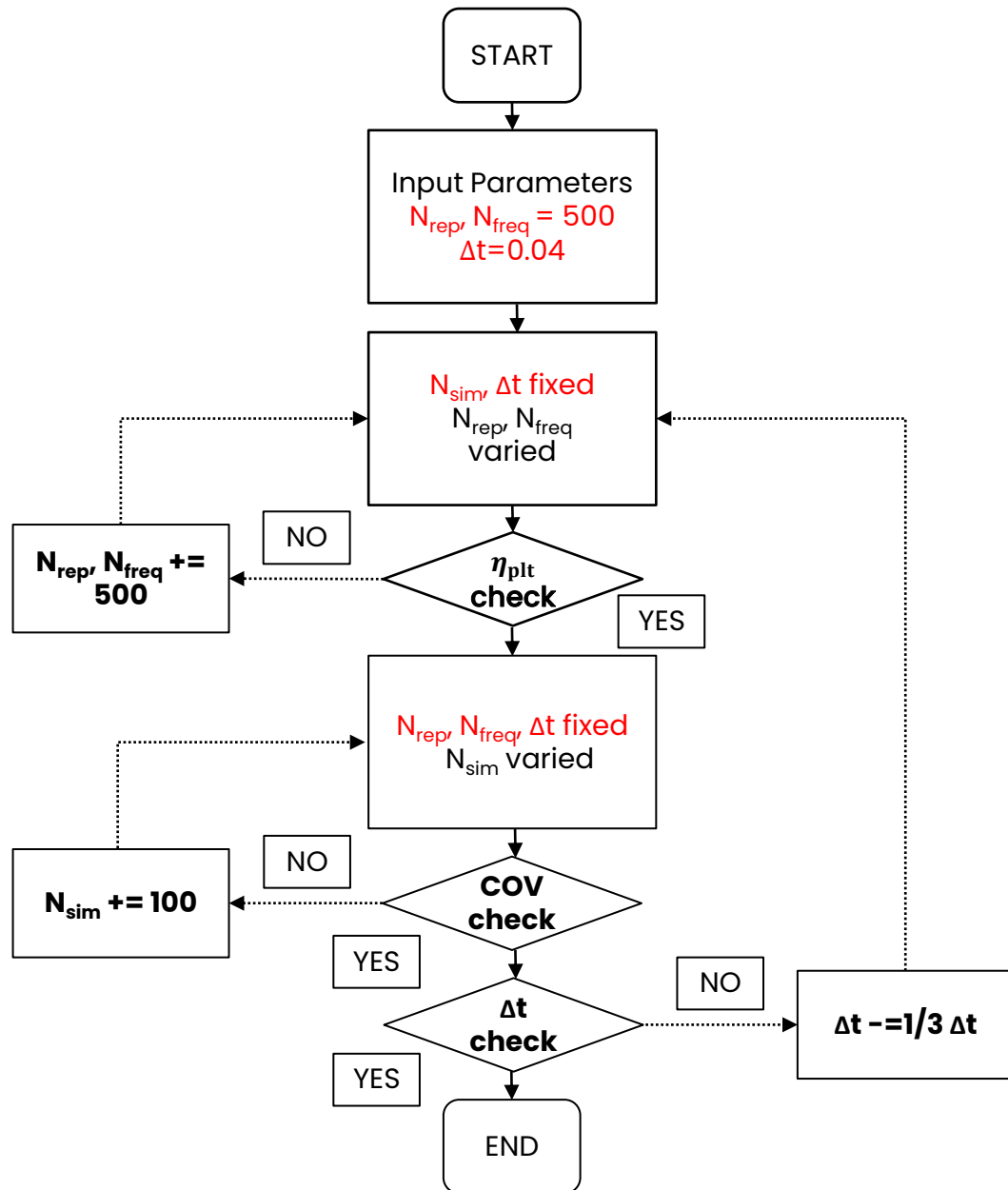


Fig. 5.1 Workflow of the automated method for viscosity calculation from DPD simulation with the Green-Kubo approach.

Algorithm 1 Pseudo-code of the algorithm for automated viscosity calculation from DPD simulation with the Green-Kubo approach.

Setup of the DPD system specific input parameters a, s, γ, r_c^D
Setup of the initial guess for N_{rep} , N_{sim} and Δt to initiate the 1st simulation

```

# Starting the  $\Delta t$ -loop
while True do

  # Starting the  $N_{\text{rep}}$ -loop
  while True do
    • Run simulation with the Current  $N_{\text{rep}}$ 
    • Calculation of the CVI from the SACF (at the Current  $N_{\text{rep}}$ )
    • Adaptation of the Simple Moving Average method to search  $\eta_{\text{plt}}$ 
    • Calculation of different  $\eta_{\text{avg}}$  and  $\sigma_{\text{avg}}$  for different (Averaging Windows)

    for each  $\eta_{\text{avg}}$  ( Current Subset ) do
      if  $\frac{|\eta_{\text{avg}}(\text{Current Subset}) - \eta_{\text{avg}}(\text{Current Subset} - \text{Jump Index})|}{\eta_{\text{avg}}(\text{Current Subset} - \text{Jump Index})} \leq \text{RE}_{\eta_{\text{avg}}}$  then ▷ Out-of- $N_{\text{rep}}$ -loop condition
        break
        ✓  $N_{\text{rep}}$  identified
        ✓ Plateau Subset  $\leftarrow$  Current Subset
        ✓  $\eta_{\text{plt}} \leftarrow \eta_{\text{avg}}$ 
        ✓  $\sigma_{\text{plt}} \leftarrow \sigma_{\text{avg}}$ 
      end if
    end for

    •  $N_{\text{rep}} \leftarrow N_{\text{rep}} + h_{\text{rep}}$ 

  end while
  # End of the  $N_{\text{rep}}$ -loop with Output:  $N_{\text{rep}}$ , Plateau Subset,  $\eta_{\text{plt}}$ ,  $\sigma_{\text{plt}}$ 

  # Starting the  $N_{\text{sim}}$ -loop
  while True do
    • Run simulation with the Current  $N_{\text{sim}}$ 
    • Calculation of the CVI from the SACF (at the Current  $N_{\text{sim}}$ )
    • Calculation of  $\eta_{\text{plt}}$  and  $\sigma_{\text{plt}}$  on the Plateau Subset
    • Calculation of the  $\text{COV}_{\text{plt}} = \frac{\sigma_{\text{plt}}}{\eta_{\text{plt}}}$ 
    if  $|\text{COV}(\text{Current Simulation}) - \text{COV}(\text{Previous Simulation})| < \text{AE}_{\text{COV}}$  then ▷ Out-of- $N_{\text{sim}}$ -loop condition
      break
      ✓  $N_{\text{sim}}$  identified
      ✓  $\eta_{\text{int}} \leftarrow \eta_{\text{plt}}$ 
      ✓  $\text{COV}_{\text{int}} \leftarrow \text{COV}_{\text{plt}}$ 
    end if

    •  $N_{\text{sim}} \leftarrow N_{\text{sim}} + h_{\text{sim}}$ 

  end while
  # End of the  $N_{\text{sim}}$ -loop with Output:  $N_{\text{sim}}$ ,  $\eta_{\text{int}}$ ,  $\text{COV}_{\text{int}}$ 

  if  $\frac{|\eta_{\text{int}}(\text{Current Simulation}) - \eta_{\text{int}}(\text{Previous Simulation})|}{\eta_{\text{int}}(\text{Previous Simulation})} \leq \text{RE}_{\eta_{\text{int}}}$  then ▷ Out-of- $\Delta t$ -loop condition
    break
    ✓  $\Delta t$  identified
    ✓  $\eta_{\text{fin}} \leftarrow \eta_{\text{int}}$ 
  end if

  •  $\Delta t \leftarrow \Delta t - \frac{1}{3} \Delta t$ 

end while
# End of the  $\Delta t$ -loop with Output:  $\Delta t$ ,  $\eta_{\text{fin}}$ 

```

5.2 Computational details

5.2.1 Simulation setup details

All simulations were performed by using the open-source software LAMMPS [28]. The code of the automated method is available online on Github¹. DPD simulations are setup by means of a script file which contains all the input parameters and the instructions for the calculations. The developed algorithm bases the entire procedure of management and submission of simulations by means of automatic generation of these files. In this section simulation features kept constant in all simulations are reported. A system of $N = 10125$ identical particles ($m_i = m = 1$) was simulated in a cubic box of the length equal to 15 DPD length units with periodic boundary in all directions, considering r_c^C as the DPD length unit. In all simulations the potential repulsion parameter $a_{ii} = a$ was set equal to 25 [13]. The system bead density and the system temperature were the same for all the cases and set to $\rho = 3$ and $k_B T = 1$.

5.2.2 Parametric study

Once a reliable automatized procedure for calculating the viscosity was established, a high number of simulations to test extensively the newly implemented code was performed. The effect of different DPD parameters on the Sc number of a simple DPD fluid was investigated. The considered parameters are the dissipative cutoff radius, r_c^D , the weighting function exponent, s and the dissipative parameters, $\gamma_{\parallel}$ and $\gamma_{\perp}$. For simplicity, it was considered $\gamma_{\perp} = \gamma_{\parallel} = \gamma$. The effects of s , γ , and r_c^D can be grouped into one single characteristic quantity measuring the overall bead friction, the so-called "effective friction coefficient", defined as follow:

$$\gamma_{\text{eff}} = 4\pi \int_0^{r_c^D} \gamma w(r_{ij}) g(r_{ij}) r^2 dr_{ij}, \quad (5.5)$$

where $g(r_{ij})$ is the RDF. The simulations cover a large range of Sc from $O(10)$ to $O(10^3)$. A summary of all DPD simulations performed is provided in the Supporting Information. As see below, the SSA-VV performed better in the larger friction region, allowing the exploitation of the DPD parameter space, inaccessible in previous studies [12] due to the limitations emerged using the VV integrator.

¹https://github.com/mulmopro/AMTP_DPD

5.3 Results and discussion

5.3.1 Automated transport properties calculation

Following the conclusion of Chapter 4, all the results showed in the current and in the next paragraphs were produced using the VV-SSA as time integrator. This paragraph presents the intermediate results obtained for a particular DPD system, with the aim of showing how the automated method to calculate the viscosity works, highlighting the choices made in the three loops described in previous paragraph. For each Δt , the sampling interval for the computation of SACF, i.e., $N_{\text{rep}} \times \Delta t$, has to be the minimum needed to capture the entire trend of the SACF, guaranteeing the achievement of the viscosity plateau by means of the N_{rep} -loop. The results obtained during a N_{rep} -loop in terms of the SACF and CVI are reported in Fig. 6.6. In particular, Fig. 6.6 displays both the SACF curves produced for different values of N_{rep} and the correspondent CVI curves. The indicators in the SACFs are the viscosity UI points for different values of N_{rep} . As it can be seen, the last iteration of the N_{rep} -loop is $N_{\text{rep}} = 3000$, for which the condition on RE_{CVI} imposed by the user is satisfied, and the SACF mean values approach to zero. In this way, the N_{rep} -loop automatically identifies as final viscosity UI point the blue one, ensuring the convergence of the CVI to a constant value. The average value of the SACF tail will be zero but it will be affected by instantaneous fluctuations that can be smoothed extending the simulation length given by $N_{\text{sim}} \times N_{\text{freq}}$.

Fig. 6.8 shows the SACF and CVI curves produced progressively increasing the number of iterations N_{sim} , highlighting the effect of the tail oscillation reduction on the computed values of viscosity. As it can be seen, the number of iterations performed in the N_{sim} -loop greatly affects the shape of the SACF and CVI, leading to different values of viscosity. The N_{sim} -loop stops the iterations when both the condition on RE_{CVI} and on AE_{COV} are sequentially satisfied. In this case this condition is reached when $N_{\text{sim}} = 700$, since the curves at $N_{\text{sim}} = 650$ and 700 show little differences. Fig. 5.4 reports that for the last value of N_{sim} both mean value of viscosity and the COV reach a plateau, where the blue lines indicate the values of η_{fin} and COV_{fin} identified by the algorithm. The computed viscosity, $\eta_{\text{fin}} = 6.694 \pm 0.002$, has a very low uncertainty resulting from the whole numerical methodology. In fact, although the higher friction regime, the value of the associated standard deviation, $\sigma_{\text{fin}} = 0.2\%$, is lower or comparable to those obtained in our previous work [12], which exhibits

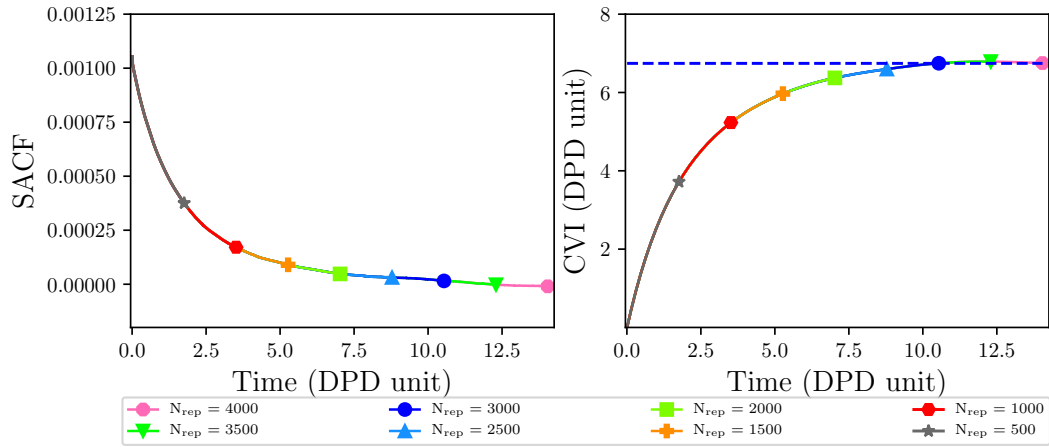


Fig. 5.2 N_{rep} -loop: SACF curves produced for different values of N_{rep} (left) and the correspondent CVI curves (right). The indicators in the SACFs are the viscosity UI points for different values of N_{rep} . The N_{rep} -loop stops at $N_{\text{rep}} = 3000$ and identifies as UI point the blue one.

excellent agreement between the Green-Kubo approach and more accurate standard techniques used as benchmark.

Therefore, for each Δt the SACF is generated with the shortest trajectory giving the fixed statistical accuracy quantified in the COV associated to calculated the viscosity value by means of the N_{rep} -loop and the N_{sim} -loop.

These *optimized* curves still have a time step size dependency. It is required a certain level of temporal resolution different for each system studied, and difficult to predict a priori. Therefore, the method search for the maximum time step size to reach this scale. Fig. 6.5 shows the SACF curves produced for the different time step sizes investigated by the algorithm through the Δt -loop and the correspondent values of η_{int} computed. The Δt -loop stops at $\Delta t = 0.003$, when $\text{RE}_{\eta_{\text{int}}}$ achieves the value fixed by the user, and returns the value of η_{fin} which is the blue one.

The algorithm returns for each DPD system studied the value of the viscosity η with the associated COV, $\mathcal{D}$ and Sc . Furthermore, the algorithm returns the values of Δt , N_{rep} and N_{sim} able to give time step size independent fluid transport properties. Such data are reported in the Supporting Information material of the article for all the systems investigated in this work, with the γ_{eff} ranging from 1 to 220.

To ensure the correctness of the automated algorithm, it is essential to check that the identified time step size gives meaningful results also in terms of static properties. In particular, the system temperature should be equal to the target, with an acceptable

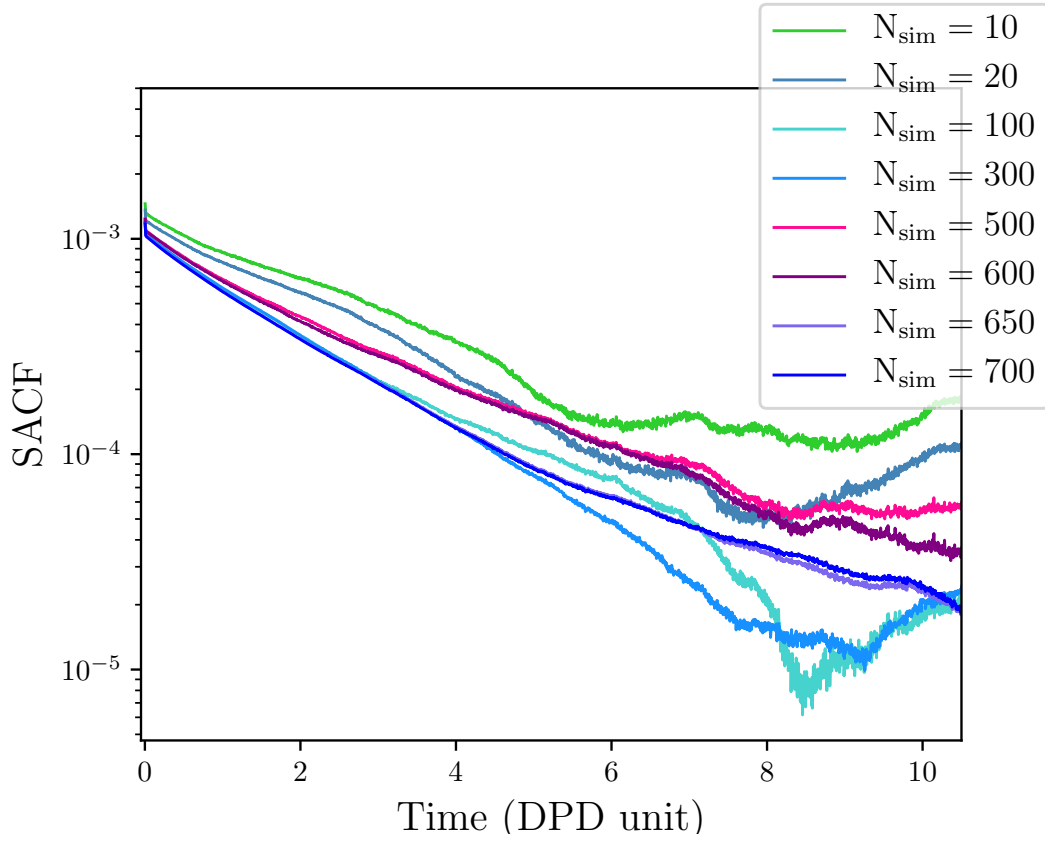


Fig. 5.3 N_{sim} -loop: SACF curves produced for different values of N_{sim} (left) and the correspondent CVI curves (right). The N_{sim} -loop stops at $N_{\text{sim}} = 700$.

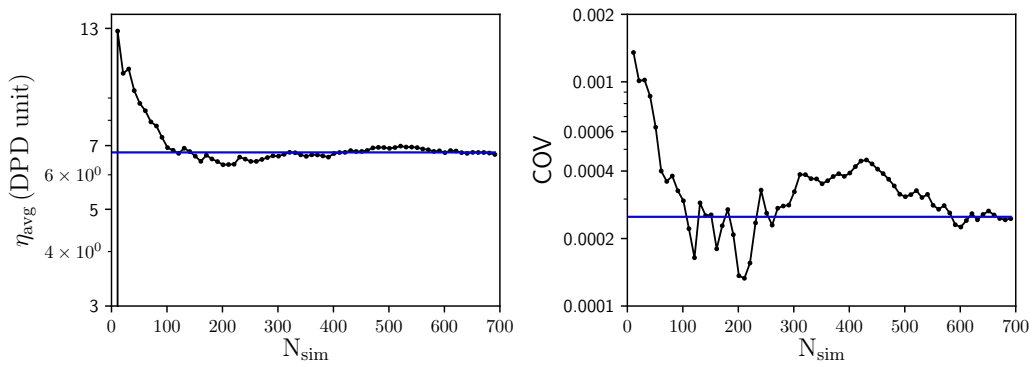


Fig. 5.4 Trend of the mean viscosity value, η_{avg} , and the corresponding coefficient of variation, COV, increasing N_{sim} . Blue lines indicate the values of η_{fin} and COV_{fin} identified by the algorithm.

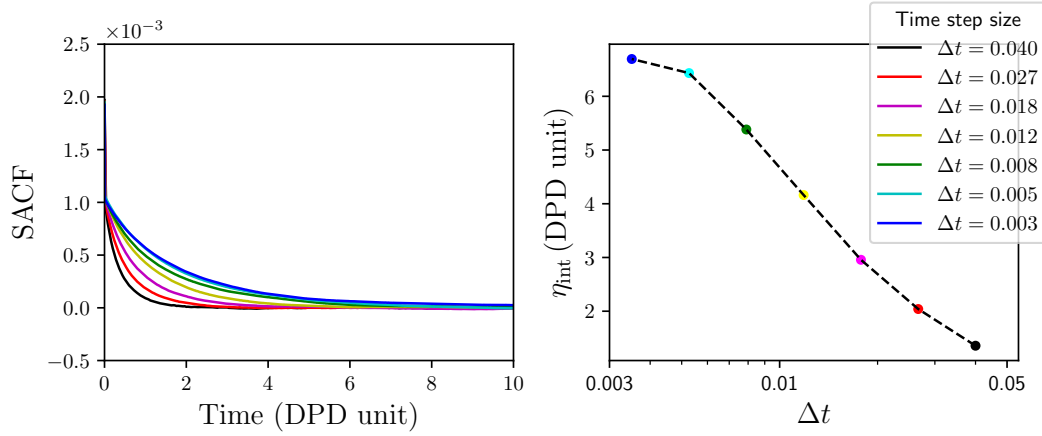


Fig. 5.5 Δt -loop: SACF curves produced for the different time step sizes (left) and the correspondent values of η_{int} computed (right). The Δt -loop stops at $\Delta t = 0.003$ and returns the value of η_{fin} which is the blue one.

deviation of 2% according to the criteria proposed by Groot and Warren [13]. The kinetic temperature control of the VV-SSA scheme for each time step size investigated is shown in Table 6.4. As it can be seen, the temperature fluctuations were lower than about 2% of $k_B T$.

Fig. 5.6 compares the radial distribution function obtained with different time step sizes investigated by the algorithm during the Δt -loop. All the curves are overlapping, which indicates that the impacts of Δt on the structure of the fluid is negligible. This means also that the investigated time steps are already quite low, since the use of high Δt may lead to large deviation of the system temperature together with large fluctuations that may destroy the structure of the fluid. However, to have a quantitative information about the system we defined an average error as

$$\varepsilon = \frac{\sum_{i=1}^{N_r} |g(r_i) - g^*(r_i)|}{N_r}, \quad (5.6)$$

where $g^*(r)$ is the RDF computed with the minimum Δt investigated ($\Delta t = 0.003$), N_r is the number of points in which the RDF is evaluated. The ε displayed in Fig. 5.6 decreases as the time step size decreases. Moreover, the further reduction from $\Delta t = 0.008$ to $\Delta t = 0.005$ produced a marginal variation in the error, showing that the reduction of the time step size plays a greater role in the evaluation of the transport properties rather than the static ones.

Table 5.1 Time step size effect on the system temperature.

Δt	$(k_B T)_{\text{avg}}$	$(k_B T)_{\text{std}} [\%]$
0.040	1.0002	0.82
0.027	1.0000	0.84
0.018	0.9998	0.82
0.012	1.0000	0.84
0.008	1.0001	0.80
0.005	1.0001	0.80
0.003	0.9999	0.80

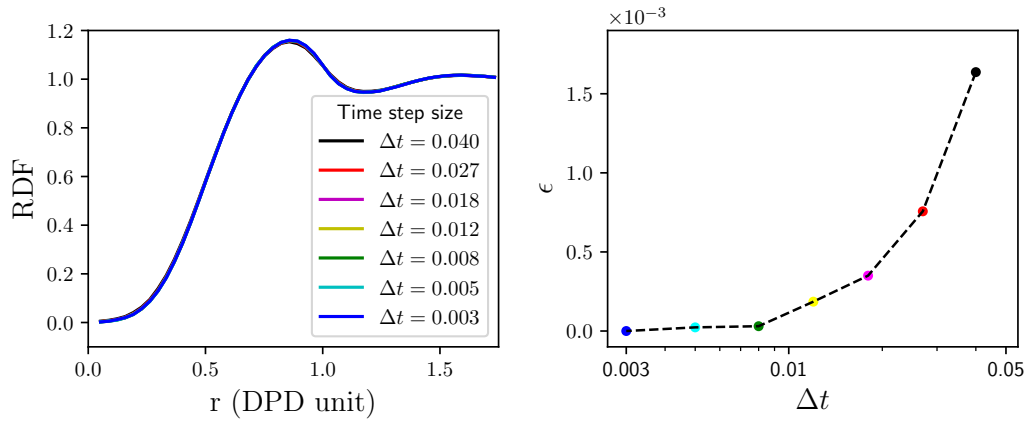


Fig. 5.6 Time step size effect on the RDF (left) and on ϵ (right). The blue RDF curve (left) corresponds is measured with the Δt value identified by the Δ -loop and the blue point (right) is the ϵ associated, which is the lowest one.

5.3.2 Schmidt number scaling with effective friction coefficient

This paragraph shows the results of the parametric study in which the DPD parameter related to bead friction, s , r_c^D and γ , are varied to reproduce the behavior of different simple fluids with different transport properties. As mentioned in Section 5.2.2, for each simulation of this parametric study it is possible to calculate the effective friction coefficient γ_{eff} according to Eq. (6.5), namely a measure of the total friction of the fluid. In Fig. 5.7 the viscosity, the diffusivity and the Schmidt number are plotted against the effective friction coefficient. The fitting curves obtained are reported as red lines in Fig. 5.7. A simplified relation between the effective friction coefficient and the varied DPD parameters is derived by a multi-parametric fitting leading to the following expression:

$$\gamma_{\text{eff}} \simeq \frac{4\pi}{3} \alpha \frac{\gamma}{s^{0.9}} \rho (r_c^D)^3, \quad (5.7)$$

where the constant $\alpha = 0.032$ comes from the fitting procedure. The simplified relationship for the γ_{eff} , Eq. (5.7), is valid for all simple fluids whose potential repulsion parameter a is equal to 25. For different values of a the contribution to γ_{eff} of the variation of the radial distribution function must be taken into account.

While Eq. (5.7) is the proper way to compute the effective friction coefficient at the end of a simulation, Eq. (5.7) can be used before running the simulation in order to estimate the effective friction coefficient for a specific set of parameters. In fact, γ_{eff} results proportional to $4\pi/3\rho(r_c^D)^3$ which represents the number of particles interacting via dissipative terms with each particle in the interaction volume. The r_c^D acts on the number of neighbouring beads increasing the number of dissipative interactions. The values of γ and s module the strength of such interactions. The higher γ and the smaller s yields a stronger particle interactions. However, momentum is transported much faster than mass, since momentum can be transferred to other particles by collisions while diffusion is limited to the bead movement. Thus, the viscosity and the diffusivity are expected to, respectively, increase and decrease with increasing γ_{eff} . In this sense, the results reported in Fig. 5.7 confirm the expectations. In particular, the viscosity η exhibits a linear dependence on the γ_{eff} , i.e.,

$$\eta = 0.0672\gamma_{\text{eff}} + 0.4325. \quad (5.8)$$

Although for low friction fluids the effect of dissipative contribution to the viscosity is negligible, in the zero-friction limit the viscosity has a no zero value because of the

kinetic contribution [92]. While the dissipative contribution to the stress is due to the explicit friction force acting between particles moving on different streamlines, and therefore it is function of γ_{eff} , the kinetic contribution is due to particles diffusing across streamlines and it is not directly related to γ_{eff} . The self-diffusivity was indeed found to be inversely proportional to the γ_{eff} according to the following fitting relationship:

$$\mathcal{D} = 0.2736\gamma_{\text{eff}}^{-1}, \quad (5.9)$$

meaning that the higher the friction, the lower the diffusivity as expected. The Sc number relationship in function of the γ_{eff} is derived as ratio between Eq. (5.8) and Eq. (5.9) resulting in the following:

$$\text{Sc} = 0.0818\gamma_{\text{eff}}^2 + 0.5269\gamma_{\text{eff}}. \quad (5.10)$$

To assess the model goodness-of-fit, the coefficient of determination (R-squared) and the normalized root mean square error (NRMSE) are reported in Table 6.6 for all the fitting relations derived in this study. All the fitting relationships recovered present goodness-of-fit measures of the same order of magnitude.

Table 5.2 Goodness-of-fit measures from the fitting on the effective friction coefficient, the viscosity, the diffusivity the Schmidt number values.

Fitted quantity	R^2	$NRMSE$
γ_{eff}	0.954	0.048
η	0.994	0.017
$\mathcal{D}$	0.876	0.059
Sc	0.978	0.030

The fitting relationships, Eq. (5.8), Eq. (5.9) and Eq. (5.10), were built for values of γ_{eff} ranging from about 1 to 220, covering simple fluids with Sc numbers ranging from approximately 1 to 3880. The Supporting Information reports for each Sc number obtained the corresponding γ_{eff} and the used set of dissipative parameters s , γ and r_c^D to realize that specific simulation. However, it should be noted that, according to Eq. (5.7), there are different infinite combinations of these parameters capable of giving the same γ_{eff} . Another aspect to highlight is the fact that the combined use of Eq. (5.7) and Eq. (5.10) makes possible to simulate other simple fluids with different transport properties, knowing the Sc number of such fluids. In fact, Eq. (5.10) can be exploited to obtain the corresponding value of

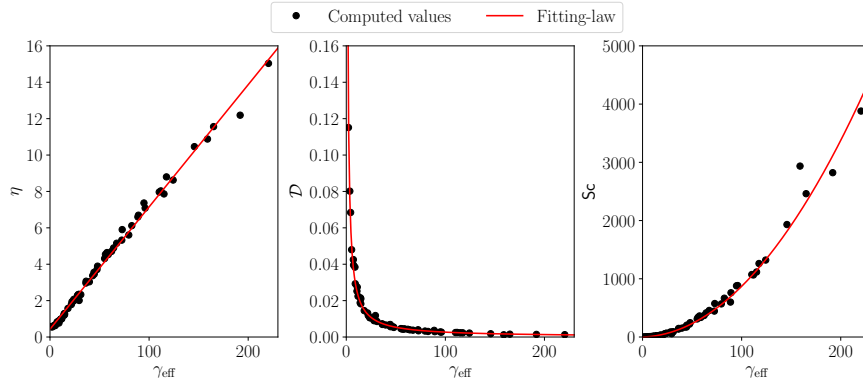


Fig. 5.7 Linear fitting on the viscosity values (left), power-law fitting on the diffusivity values (middle) and the polynomial fitting on the Schmidt number values (right) as a function of the effective friction coefficient parameter. The red lines are to fit the simulation results (black symbols).

γ_{eff} , given the experimental Sc number. Then Eq. (5.7) can be used to obtain a set of fixed parameters giving the desired γ_{eff} ; fixed two parameters, the third one is found. Fixed the γ_{eff} , η and D are fixed according to Eq. (5.8) and Eq. (5.9). For instance, the parameters needed to properly simulate liquid water at 25°C ($Sc \simeq 370$ experimentally) are available in the table in correspondence of γ_{eff} equal to 62.137, giving $Sc = 369.589$. The only limitation is that all the simulations were performed with same conservative soft-potential, meaning that the isothermal compressibility must be the same of water, which is a reasonable assumption for liquids.

5.4 Conclusions

In this Chapter of the thesis, to overcome the numerical issue related to the calculation of the transport quantities from a numerical simulation, we formulated and developed an automated algorithm based on Green-Kubo formula, in which the stress autocorrelation function (SACF) is calculated through an approximated algorithm. Our method automatically finds the parameter of the approximated algorithm for the calculation of the SACF, showing that there is the possibility to get wrong results if these parameters are poorly chosen. Moreover, we found that transport properties can be wrongly evaluated if the time step of the DPD simulation is not accurately selected. The developed automated method is able to find, for a given DPD system, the viscosity value that it is independent on the choice of time step.

Finally from a parametric study in which we varied all the dissipative parameters of the DPD model, we evaluated the effective friction coefficient, viscosity, self-diffusivity and Sc number. From this dataset made by over 60 simulations, we derived a fitting relation for the effective friction coefficient as a function of different DPD parameters, and fitting relations for viscosity, self-diffusivity and Schmidt (Sc) number as a function of the effective friction coefficient. This analysis shows that infinite sets of DPD parameters may give the same effective friction coefficient, in turn leading to the same Sc number and radial distribution function. In other words, it is possible to simulate the same fluid with the same rheological behavior with different sets of DPD parameters. Moreover, we found that viscosity increases linearly with effective friction coefficient, the self-diffusion is inversely proportional, and the ratio between the viscosity and diffusivity recovered fitting curves results in a very good fit for Sc number values computed from simulations. The findings of this work can be used to simulate simple fluids with different Sc number, since the relation between a specific effective friction coefficient and DPD parameters was found. Moreover, although the method proposed in this work was extensively tested and optimized for simple fluids, it is a tool of general validity with the aim of improving the reliability and computational feasibility of the Green-Kubo approach in DPD simulations. Therefore, it is potentially applicable to both simple and complex fluids. However, the application to more topologically complicated systems is not straightforward, since the intrinsic difficulties of the Green-Kubo approach in DPD are accentuated. Indeed, in the case of polymer systems the computational load required for the viscosity prediction is even higher, since the long-time relaxation requires a larger time interval to trace the whole decorrelation dynamics of the SACF. Although there are some challenges to address, the reliability of the method makes it a promising tool to perform a systematic scaling analysis on the polymer systems dynamical properties, such as viscosity and diffusivity, with polymer concentration and length.

Chapter 6

Modeling temperature-dependent transport properties in dissipative particle dynamics: A top-down coarse-graining toward realistic dynamics at the mesoscale

Most of DPD studies on transport properties were carried out without varying temperature, although transport properties usually vary strongly with the system temperature and the effect of temperature on the flow dynamics of non-isothermal mesoscopic systems is not negligible. Since the modification of the non-conservative interaction force field is not straightforward, the temperature effect is just introduced parametrically into non-conservative interactions in order to capture the effect of temperature. However, these studies employed the standard DPD thermostat which, relying on parallel dissipative forces acting in the radial direction only, cannot accurately simulate liquid-like dynamics. Several studies claimed that it fails in the prediction of real fluids transport properties [18, 12]. In fact, Chapter 3 shows that the contribution of shear dissipative forces, in the direction perpendicular to the interatomic axis, is important to ensure a correct description of transport properties. A different thermostat for DPD, a transverse DPD thermostat, is obtained by also describing the perpendicular damping [64]. Previous Chapters 3, 4 and 5 demonstrate

The content of this chapter is part of the work published in a peer-reviewed journal. The complete citation is provided below: Lauriello et al., *The Journal of Chemical Physics*, vol. 161, no. 3, 2024. <https://doi.org/10.1063/5.0207530>.

that a particular extension of the DPD model, based on a transverse thermostat, together with the modification of the weighting function for the dissipative and random (non-conservative) forces [12, 90], allows to finely tune transport properties by calibrating non-conservative interaction parameters against convenient experimental transport properties.

This chapter proposes, for the first time, a temperature-dependent form of the extended DPD model, where both conservative and non-conservative interaction parameters incorporate the system temperature variation obtained by fitting experimental observables.

The extended DPD model of this work is developed using the top-down CGing as its conservative and non-conservative interaction parameters are fitted to experimental observables of a real system. Bottom-up CGings, which rely on atomistic simulations, also exist and a typical example includes multi-scale CGing (force matching) [24, 25]. Here within the Mori-Zwanzig projection operator method [26, 27], the CG equations of motion can be uniquely determined by the underlying microscopic dynamics. They have a form of the generalized Langevin equation and describe the evolution of a subset of the degrees of freedom. Under Markovian approximation, the generalized Langevin equation can be reduced to DPD equations of motion (EoM) with a conservative force, a dissipative force and a random force whose expressions can be obtained directly from atomistic simulations. In this work, instead, a top-down procedure is adopted for the parametrization of non-conservative interactions. This strategy does not require large molecular dynamics trajectories to inform the DPD model reducing the computational effort. The objective is to enable the model to simulate a fluid across different temperatures, describing accurately not only equilibrium properties, but also transport properties at various temperatures. Isothermal simulations across different temperatures are carried out, since the estimation of transport properties at a given temperature is important before evaluating the dynamic behavior induced by thermal gradients. Moreover, transport properties are measured using an equilibrium approach. This is advantageous in self-assembled systems, since it allows the estimation of viscosity in equilibrium without altering their equilibrium self-assembled microstructures. In fact, before evaluating the rheological behavior of self-assembled systems related to the structural changes occurring under fluid motion or temperature change, it is important at first to estimate the viscosity of morphologies in equilibrium without perturbing them.

In particular, we selected liquid water as model system and used experimental data

of transport properties [93, 94] from 5°C to 90°C as target values to develop and validate the extended DPD model. Moreover, we also assessed the performance of the Shardlow-splitting-algorithm-based integration scheme (VV-SSA), which was proved to be superior to the standard velocity-Verlet for the evaluation of transport properties in Chapter 4, to solve the isothermal extended DPD model EoM varying the system temperature. We employed the automated method [90] (AMTP_DPD), presented in previous Chapter 5, to compute transport properties with controlled accuracy from DPD equilibrium simulations carried out across different temperatures. The application of the proposed model provides estimates for liquid water transport properties in excellent agreement with experimental data across the whole liquid temperatures range. In addition, we enabled the open source code LAMMPS[28] to evaluate numerically the different viscosity contributions to investigate their relative weight and their behavior with respect to the system temperature. This type of additional calculation allowed to make comparison with theoretical data available in the literature.

6.1 Temperature dependent-transport properties in DPD simulations

Viscosity and self-diffusivity are temperature-dependent output properties of a DPD simulation. These properties can assume different values depending on input model parameters. Several researchers derived theoretical expressions for dynamical properties in terms of model input parameters. Among them, it is worth mentioning the relationships obtained by Groot and Warren [13], neglecting the conservative interactions. Despite of this simplification, they are in agreement with those derived by more sophisticated approaches [92] and at the same time, they highlight the physical meaning of the different terms. In particular, the effect of temperature on the various terms is clear allowing preliminary considerations on the model capability to capture the temperature-dependence of fluid transport properties. These relationships were obtained by considering the standard DPD thermostat with the SWF. Moving a step forward in the progressive evolution of the DPD model, similar relationships were derived accounting for the substitution of the SWF with a GWF, but still within the standard DPD framework [14], where shear dissipative forces are not included. It is worth highlighting that the GWF differs from the GWF^D employed in our model,

since it uses as dissipative/random cut-off radius the same of the conservative one, $r_c = r_c^C = r_c^D$. For the sake of clarity, Table 6.1 shows the differences among three dissipative/random weighting functions mentioned in this work.

Table 6.1 Main differences among the weighting functions labelled by SWF, GWF and GWF^D.

Label	Exponent	Cutoff radius
SWF	1	r_c
GWF	s	r_c
GWF ^D	s	r_c^D

A re-elaboration of those expressions to highlight the effect of system temperature θ on transport properties (the kinematic viscosity $\nu = \eta/\rho$ and $\mathcal{D}$) is provided in Table 6.2. In both derivations, the first term is directly proportional to the temperature while the second term turns to be independent from the temperature, keeping invariant the model input parameters with temperature. Thus according to theoretical predictions, the kinematic viscosity described by the standard DPD model increases with increasing temperature regardless of the dissipative weighting function employed. Fig. B.1 illustrates the behavior of the transport properties calculated using the SWF and GWF reported in Table 6.2, ranging from $\theta = 0.933$ to $\theta = 1.22$. The kinematic viscosity is almost constant with respect to the temperature showing an unrealistic behavior. In both cases, the typical decaying trend of kinematic viscosity over temperature experimentally observed with many fluids such as water, ethanol, and glycerin cannot be captured. In this work, liquid water transport properties are used as target properties and they are reported in Table 6.3. The experimental data of kinematic viscosity ν and self-diffusivity $\mathcal{D}$ are taken from Refs. [93, 78]. Fig. 6.2 clearly shows that the kinematic viscosity dependence on temperature $T = \theta \times T_R$ is opposite with respect to both the theoretical predictions as already pointed out by other works [14].

The relationship between transport properties and model interaction parameters was investigated independently in different studies also relying on numerical investigations, but not accounting for system temperature variation, usually kept constant at ambient temperature. However, only few studies considered the temperature-dependence of fluid transport properties, but using the DPD conventional formulation, which rely on parallel dissipative and random forces only. Here instead, we

Table 6.2 Re-elaboration of theoretical expressions for the transport properties from literature to highlight their relationship with system temperature θ . The expressions derived in the case of the standard DPD model including the standard dissipative quadratic weighting function [13] (SWF) are compared with the ones obtained in the case of the standard DPD model including a generalized weighting function [14] (GWF^D).

Properties	Standard thermostat (SWF)	Standard thermostat (GWF)
Viscosity, ν	$\alpha_{\text{SWF}} \frac{k_B \theta}{\pi \gamma \rho r_c^3} + \beta_{\text{SWF}} \pi \gamma \rho r_c^5$	$\alpha_{\text{GWF}} \Upsilon(s) \frac{k_B \theta}{\pi \gamma \rho r_c^3} + \frac{\beta_{\text{GWF}}}{\Phi(s)} \pi \gamma \rho r_c^5$
Diffusivity, $\mathcal{D}$	$2\alpha_{\text{SWF}} \frac{k_B \theta}{\pi \gamma \rho r_c^3}$	$2\alpha_{\text{GWF}} \Upsilon(s) \frac{k_B \theta}{\pi \gamma \rho r_c^3}$
Schmidt number, Sc	$\frac{1}{2} + \frac{\beta_{\text{SWF}}}{2\alpha_{\text{SWF}}} \frac{(\pi \gamma \rho r_c^4)^2}{k_B \theta}$	$\frac{1}{2} + \frac{\beta_{\text{GWF}}}{2\alpha_{\text{GWF}} \Phi(s) \Upsilon(s)} \frac{(\pi \gamma \rho r_c^4)^2}{k_B \theta}$

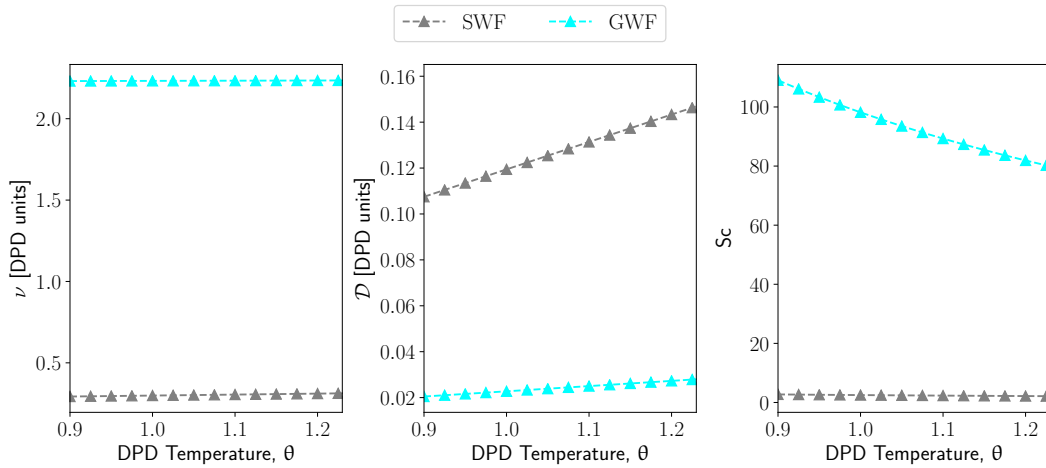


Fig. 6.1 Theoretically predicted behavior of the kinematic viscosity ν (left), the self-diffusivity $\mathcal{D}$ (center) and the Schmidt number Sc (right) of a DPD fluid over the system temperature θ . The SWF curves refer to standard DPD model including standard dissipative quadratic weighting function. The GWF curves refer to standard DPD model including generalized dissipative weighting function.

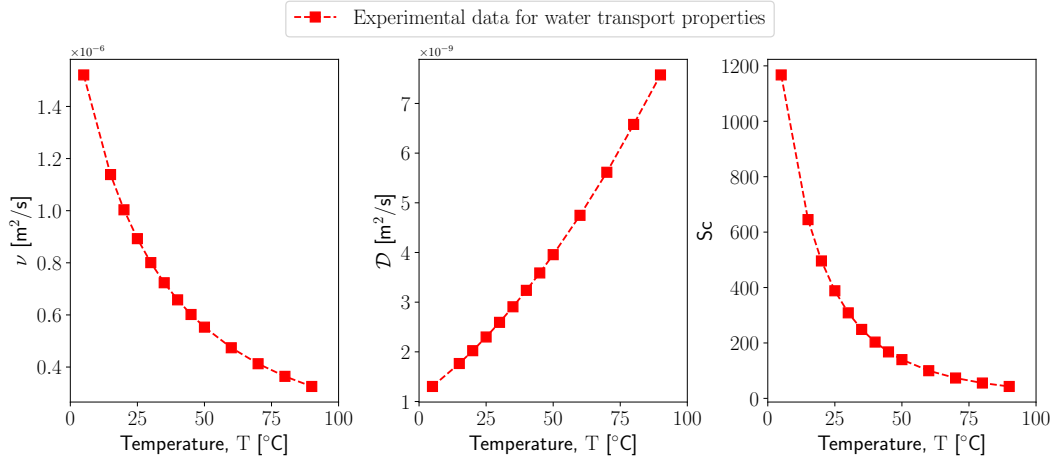


Fig. 6.2 Experimental behavior of the kinematic viscosity ν (left), the self-diffusivity $\mathcal{D}$ (center) and the Schmidt number Sc (right) of liquid water over temperature T .

Table 6.3 Target experimental values in real units and scaled with respect to the reference values. The reference values for the temperature T , the kinematic viscosity ν , and the self-diffusivity $\mathcal{D}$ are highlighted.

T [°C]	T [K]	θ	ν [m²/s]	$\mathcal{D}$ [m²/s]	ν/ν_R	$\mathcal{D}/\mathcal{D}_R$	Sc
5	278.15	0.933	1.520×10^{-6}	1.303×10^{-9}	1.703	0.5668	1167.24
15	288.15	0.966	1.138×10^{-6}	1.765×10^{-9}	1.275	0.7677	645.11
20	293.15	0.983	1.000×10^{-6}	2.023×10^{-9}	1.120	0.8799	459.99
25	298.15	1	8.926×10^{-7}	2.299×10^{-9}	1	1	388.28
30	303.15	1.02	8.007×10^{-7}	2.594×10^{-9}	0.8970	1.128	308.68
35	308.15	1.03	7.234×10^{-7}	2.907×10^{-9}	0.8104	1.264	248.86
40	313.15	1.05	6.578×10^{-7}	3.238×10^{-9}	0.7369	1.408	203.16
45	318.15	1.07	6.016×10^{-7}	3.588×10^{-9}	0.6740	1.561	167.69
50	323.15	1.08	5.531×10^{-7}	3.956×10^{-9}	0.6197	1.721	139.82
60	333.15	1.12	4.740×10^{-7}	4.748×10^{-9}	0.5310	2.065	99.83
70	343.15	1.15	4.127×10^{-7}	5.615×10^{-9}	0.4624	2.442	73.50
80	353.15	1.18	3.643×10^{-7}	6.578×10^{-9}	0.4081	2.861	55.39
90	363.15	1.22	3.254×10^{-7}	7.574×10^{-9}	0.3646	3.294	42.97

employed the extended DPD model detailed in Chapter 3 since in our previous works, we pointed out that the contribution of shear dissipative forces is crucial to ensure correct description of transport properties.

6.2 Interaction parameters

The conservative interaction parameter a is purely repulsive and its value is established via the matching of isothermal compressibility κ . The temperature dependence of the interaction parameter a was first considered by Groot and Warren (GW) [13], and recently reformulated with a generalized approach [22]. The latter allows to include both the simulation temperature θ and the temperature-dependent compressibility κ in the evaluation of the repulsive parameter as follows:

$$a = \frac{\kappa^{-1} - k_B \theta}{2\alpha\rho}, \quad (6.1)$$

where α is a constant. There has been a controversy in the literature whether to take into account the level of CGing (e.g., the number of water molecules per bead) when relating the repulsive parameter to the compressibility. As shown by Fuchsli et al. [38], this issue can be avoided if both the repulsive parameter and the cut-off radius are properly scaled by the level of CGing.

In many previous works, the compressibility was considered independent from temperature. This might be an overly crude approximation as in the case of water where the variation in κ is about 10% between 0°C and 50°C. In our approach, the dimensionless compressibility κ was considered temperature-dependent. The dimensionless bulk modulus κ^{-1} (the inverse of the dimensionless compressibility) depends on temperature according to the following relationship:

$$\kappa^{-1} = \kappa_T^{-1} \frac{(r_c^C)^3}{k_B T_R} = \kappa_T^{-1} \frac{\rho V_0}{k_B T_R}, \quad (6.2)$$

where κ_T is the water compressibility in real units [Pa^{-1}], and V_0 is the molecular volume. κ_T is calculated for different temperatures according to its thermodynamic definition as

$$\kappa_T^{-1} = -\frac{1}{V} \frac{\partial V}{\partial P} \Big|_T. \quad (6.3)$$

The isothermal compressibility is defined as the fractional change in the volume V that results from a change in the pressure P of the system in a process where the temperature T is constant. The thermophysical properties of water, required for the calculation, are taken from the NIST database [93]. For water at room temperature, the simulation temperature and the inverse of dimensionless compressibility are, respectively, $\theta = 1$ and $\kappa^{-1} \simeq 16$, and Eq. (6.1) turns into the GW form

$$a = \frac{15}{2\alpha\rho} = \frac{75}{\rho}. \quad (6.4)$$

where for the standard $\rho = 3$, $a = 25$. The parameter a linearly decreases with temperature as shown in Fig. 6.3, assuming temperature invariance of the compressibility. In Fig. 6.3, the different behavior of a over θ , obtained by accounting for the temperature variation of isothermal compressibility, is also reported. Moreover, Fig. 6.3 provides an estimate of the error on a produced by neglecting the temperature variation of the isothermal compressibility. However, the introduction of temperature-

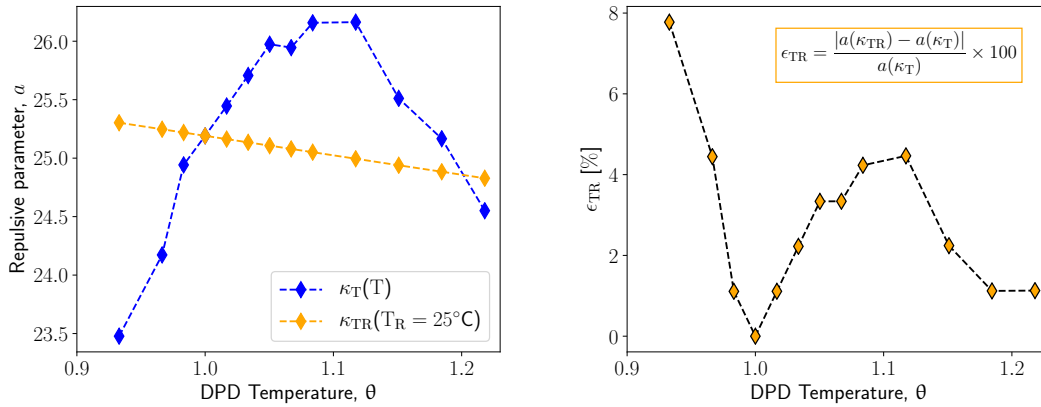


Fig. 6.3 Comparison between the behavior of repulsive parameters a over temperature θ obtained by assuming constant isothermal compressibility κ_{TR} and by considering temperature-dependent isothermal compressibility values κ_T (left). Error estimates on a due to neglecting temperature variation of the isothermal compressibility (right).

dependent repulsive parameters is still not enough to capture temperature-dependent transport properties. In other words, the non-conservative parameters γ , r_c^D , and s capable to reproduce the viscosity η_R and the self-diffusivity $\mathcal{D}_R$ of liquid water at $T_R = 298.15\text{K}$ fail to predict the same properties at different temperatures. This deficiency is clear from Fig. 6.4. Therefore to simulate a DPD fluid with realistic transport properties at different temperatures, at least one of the non-conservative interaction parameters γ , r_c^D , or s needs to be considered as temperature dependent.

Previous chapter [12] and other literature works detailed the effect of the parameters

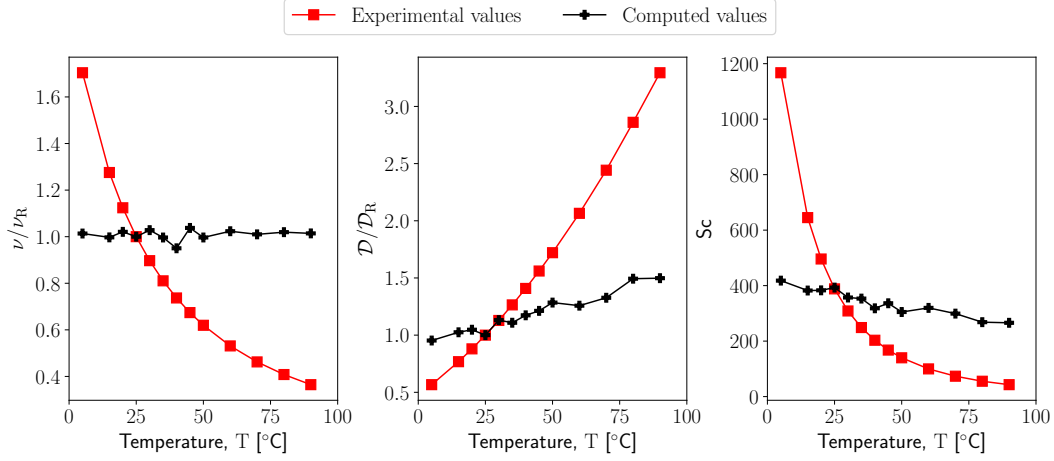


Fig. 6.4 Comparison of transport properties at different temperatures between experimental data and simulation results with temperature-dependent conservative and non-conservative interaction parameters. The kinematic viscosity and self-diffusivity are scaled by their reference values $\nu_R = 8.926 \times 10^{-7} \text{m}^2/\text{s}$ and $\mathcal{D}_R = 2.299 \times 10^{-9} \text{m}^2/\text{s}$ at $T_R = 298.15 \text{K}$. Errors in ν and $\mathcal{D}$ are smaller than points in the viscosity plots

γ , r_c^D , and s on the transport properties. We demonstrated that the effect of γ , r_c^D , and s can be grouped into a single quantity which measures the overall bead friction, the so-called "effective friction coefficient", defined as

$$\gamma_{\text{eff}} = 4\pi \int_0^{r_c^D} \gamma w(r_{ij}) g(r_{ij}) r^2 dr_{ij}, \quad (6.5)$$

where $g(r_{ij})$ is the radial distribution function (RDF). We pointed out that fluids characterized by the same γ_{eff} have the same transport properties. In particular, there is a certain value of γ_{eff} associated to realistic transport properties of liquid water at room temperature.

According to the results shown in previous chapters [12, 90] to simulate liquid water at higher temperatures, the value of γ_{eff} has to be reduced to decrease the viscosity and increase the self-diffusivity consistently with experimental data. This effect can be obtained by increasing s or decreasing γ and/or decreasing r_c^D . In fact, the r_c^D acts on the number of neighbouring beads decreasing the number of the dissipative interactions. Instead, the values of γ and s modulate the strength of the dissipative interactions. The smaller γ and the higher s yield to weaker dissipative interactions, resulting in a lower viscosity and higher self-diffusivity. Similar considerations can

be reached for lower temperatures. To obtain higher viscosity and smaller diffusivity, higher γ and r_c^D values are required as well as smaller s values. The definition of the dissipative coefficient γ as a function of temperature leads to physical inconsistency as already explained, e.g., in Ref. [14]. In fact, since γ and σ are related via the FDT, the reduction of γ means that hotter systems have smaller thermal fluctuations and vice versa for colder systems, which is contrary to the correct physics expectations. The value of γ is fixed for the simulations performed at different temperatures. In this way, the σ depends on θ consistently with physical expectations. Instead, the r_c^D and s parameters can be theoretically considered as temperature dependent.

However, in this work the r_c^D is maintained equal to r_c^C to keep the computational cost of the simulations as low as possible. This choice is feasible since the range of Sc numbers explored by liquid water (from 5°C to 90°C) is rather wide and can be covered by acting on one parameter only. Therefore for each temperature investigated within the liquid temperatures range, the s was calibrated to reproduce the Sc number consistently with experimental data. This methodology led to the derivation of an expression describing the s dependence on the simulation temperature θ .

It is worth mentioning a possible scaling of DPD parameters with a coarse-graining level, Φ . In this work, we consider Fuchslin et al.'s scaling approach which in reduced units, it keeps the DPD parameters and in turn, the system properties invariant under Φ . Specifically: $N' = N/\Phi$, $m'_i = \Phi m_i$, $a' = \Phi^{2/3}a$, $r_c^{C'} = \Phi^{1/3}r_c^C$, $\gamma' = \Phi^{2/3}\gamma$, $\sigma' = \Phi^{5/6}\sigma$, $t' = \Phi^{1/3}t$, and the unit of energy, $(k_B T)' = \Phi(k_B T)$, where primed variables correspond to Φ . Two notes are appropriate for Fuchslin et al.'s scaling approach.

First, Fuchslin et al. pointed out that the validity of their approach does not necessarily hold for the study of dynamical properties, and they demonstrated it on the scaling of diffusion. They showed that the relative particle motions are unaffected by scaling in the reduced unit systems, i.e., trajectories are numerically equal. On the one hand, this leads to the numerical equivalence of the measured diffusion coefficients but on the other hand, the fact that the diffusion coefficient scales like length squared over time causes an apparent problem. It seemingly implies that relative fluctuations of the reduced diffusion coefficient over the reduced length stay constant instead of vanishing with increasing Φ , which is a consequence of Fuchslin et al.'s CGing. It brings a general question to what extent the trajectories of DPD particles can be understood as representing actual transport processes. Several authors [46, 37] resolve the calibration problem by treating the natural time unit as

undetermined and instead obtain the physical time scale from calibrating diffusion coefficients measured in simulation to physical values [37]. Second, Füchslin et al.'s scalings of the non-conservative coefficients, γ and σ , was suggested for the DPD method with the parallel non-conservative forces and the SWF. In this work, we consider the parallel and perpendicular non-conservative forces and the GWF^{D} . This may require examining validity of the scalings of the γ and σ coefficients, and determining a scaling of the GWF^{D} coefficients, s . A possible route towards the scaling of these GWF^{D} coefficients may involve considering the effective friction coefficient, Eq. (17), invariant under Φ in reduced units. However, this is beyond the scope of this work.

6.3 Results and discussion

To ensure the reliability and correctness of the simulation insights, checking the behavior of the system kinetic temperature during simulation runs is essential for each temperature investigated. In particular, the average kinetic temperature, $\theta_{\text{avg}} = \langle \mathcal{T}_{\text{kin}} \rangle$, should be equal to the imposed temperature θ . The standard deviation of θ can be considered acceptable if lower than 2% according to the criteria proposed by Groot and Warren [13]. Table 6.4 shows that θ_{avg} s are around the average value with standard deviations lower than 2%, i.e., the time integration scheme employed properly guarantees the kinetic temperature control during the simulation runs at each temperature investigated. The performance of the VV-SSA to solve the isothermal extended DPD EoM for the evaluation of transport properties was already proven to be superior to other schemes at $\theta = 1$ in our previous chapter [90]. Table 6.4 provides evidence that the VV-SSA performs excellently when θ s vary in the range from 0.933 to 1.22. The kinetic temperature control is important to preserve the meaningfulness of the physical insights provided from each isothermal simulation performed. Otherwise, unphysical drifts of the kinetic temperature would lead to uncontrolled spatial correlations among particles and to significant errors in the simulated static and dynamical properties.

Due to the DPD soft interactions and rather large simulation boxes typically employed in DPD simulations, the finite size error of $\mathcal{D}$ due to periodic boundary conditions, in contrast to molecular dynamics simulations, can be expected to be rather small. Nevertheless, we performed DPD simulations at $\theta = 1$, and varied box

Table 6.4 Targeted system temperature θ , the average kinetic temperature θ_{avg} and associated standard deviation θ_{std} for each temperature T investigated.

T [°C]	θ	θ_{avg}	θ_{std} [%]
5	0.933	0.932	0.7
15	0.966	0.966	0.7
20	0.983	0.983	0.7
25	1	0.999	0.8
30	1.020	1.020	0.8
35	1.030	1.029	0.8
40	1.050	1.049	0.8
45	1.070	1.070	0.8
50	1.080	1.088	0.8
60	1.120	1.119	0.9
70	1.150	1.149	0.9
80	1.180	0.180	0.9
90	1.220	1.220	0.9

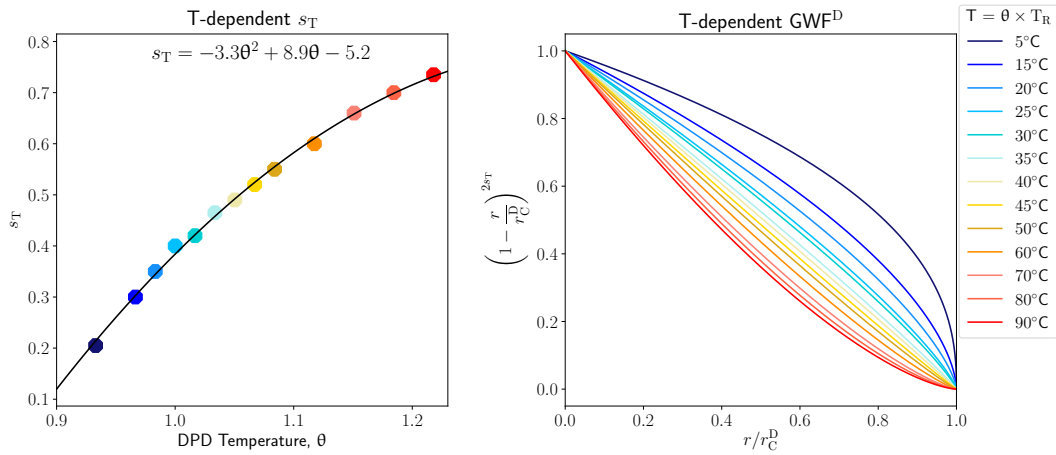


Fig. 6.5 The s power coefficient as a function of temperature θ which provides the correct Sc number of liquid water at different real temperatures $T = T_R \times \theta$. The fitting of those data gives s as a function of θ (left). Dissipative weighting functions at different temperatures $T = T_R \times \theta$ (right).

lengths of the cubic simulation box, L , from 12 to 40. We then plotted evaluated values of the self-diffusion coefficient, $\mathcal{D}^{\text{PBC}}$, as a function of $1/L$ as shown in Fig. 2 of the Supporting Information. Analogously to MD simulations [95], we represented $\mathcal{D}^{\text{PBC}}$ as

$$\mathcal{D}^{\text{PBC}} = \mathcal{D}^0 - \frac{k_{\text{B}}T\xi}{6\pi\eta L} \quad (6.6)$$

where $\mathcal{D}^0$ is the “true” self-diffusion coefficient, and $\xi = 1.855157$. Note that $\xi = 1.855157$ is smaller than $\xi = 2.837297$ relevant for MD simulations [95], and in turn, $\mathcal{D}^{\text{PBC}}$ vs $1/L$ is less steep in DPD simulations than that in MD simulations. We then use Eq. (6.6) to evaluate values of $\mathcal{D}^0$ at different temperatures. We found that differences between values of $\mathcal{D}^{\text{PBC}}$ and $\mathcal{D}^0$ are up to 2%, which has no or negligible effect on the adjustment of the non-conservative DPD parameters, see the Supporting Information for details. The values of $\mathcal{D}^{\text{PBC}}$ were employed below.

For each temperature investigated, the s power coefficient was calibrated to capture the correct Sc number of liquid water. The Sc number measured by DPD is compared with the experimental Sc for each temperature investigated in the range from $T = 5^\circ\text{C}$ to $T = 90^\circ\text{C}$, which corresponds to the DPD temperature range from $\theta = 0.933$ to $\theta = 1.22$. Table 6.5 reports the experimental Sc, Sc_{exp} , and the DPD Sc, Sc_{DPD} , for each temperature T together with the relative error. The level of acceptability of the error was fixed to be lower than 5% a priori. The s values capable to satisfy this matching condition for each temperature are also reported in Table 6.5, and are plotted in Fig. 6.5.

The s power coefficient modulates the strength of non-conservative interactions within the volume defined by r_c^D . To get transport properties behavior consistently with experiments, the s parameter needs to incorporate the system temperature variation. Fig. 6.5 shows how different values of s lead to different behavior of the GWF^D at different temperatures. Room temperature water transport properties are properly reproduced with $s = 0.40$. It is clear that higher/smaller temperatures require smaller/higher values of s to accurately describe momentum and mass transport. Since smaller s yields stronger non-conservative interactions, resulting in turn into a higher Sc and vice versa, the obtained s trend over temperature was expected. The fitting of those data gives s as a function of θ according to the following expression:

$$s_{\text{T}} = -3.3\theta^2 + 8.9\theta - 5.2, \quad (6.7)$$

Eq. (6.7) allows to reproduce liquid water transport properties consistently with experimental data. Fig. 6.6 exhibits the comparison of the temperature-dependent

Table 6.5 Values of the non-conservative parameter, s , providing an excellent match between the experimental Schmidt number, Sc_{exp} , and the DPD Schmidt number, Sc_{DPD} , for each temperature, T , as evident from the relative error ϵ_{Sc} .

$T [^{\circ}\text{C}]$	θ	Sc_{exp}	Sc_{DPD}	$\epsilon_{Sc}[\%]$	s
5	0.933	1167.24	1125.19	3.6	0.205
15	0.966	645.11	654.04	1.3	0.300
20	0.983	459.99	504.19	1.6	0.350
25	1.00	388.28	392.91	1.1	0.400
30	1.02	308.68	317.33	2.7	0.420
35	1.03	248.86	249.92	0.6	0.465
40	1.05	203.16	211.98	4.3	0.490
45	1.07	167.69	173.35	3.2	0.520
50	1.08	139.82	142.95	2.1	0.550
60	1.12	99.83	101.35	1.4	0.600
70	1.15	73.50	73.74	0.3	0.660
80	1.18	55.39	55.52	0.2	0.700
90	1.22	42.97	53.44	1.0	0.735

kinematic viscosity, self-diffusivity and Sc number between the DPD simulations and experiments. The extended DPD model is able to properly capture temperature variations of both the viscosity and self-diffusivity over temperature in the range from $T = 5^{\circ}\text{C}$ to $T = 90^{\circ}\text{C}$. This outcome remarks the importance of properly modeling non-conservative interactions to correctly describe real fluids transport properties. The use of the extended DPD model in which the s parameter is reconsidered as a temperature-dependent parameter s_T is crucial in the achievement of the realistic decaying trend of the kinematic viscosity over temperature. In fact, the left subplot of Fig. 6.7 clearly highlights that the main contribution to the kinematic viscosity ν is given by $\nu - \nu^{CC}$, where $\nu^{CC} = \eta^{CC}/\rho$, cf. Eq. (5.3); similarly for $\nu - \nu^{CC}$. The latter ignores the contribution to the viscosity that arises from $\Pi_{\alpha\beta}^C$, which gathers the contribution to the stress tensor given by the kinetic and conservative forces, see Eq. (5.2). The $\nu - \nu^{CC}$ represents the contribution to the viscosity that arises primarily from non-conservative interactions. It is quantitatively most of the viscosity and it is more sensitive than ν^{CC} to temperature variation. This outcome is the proof

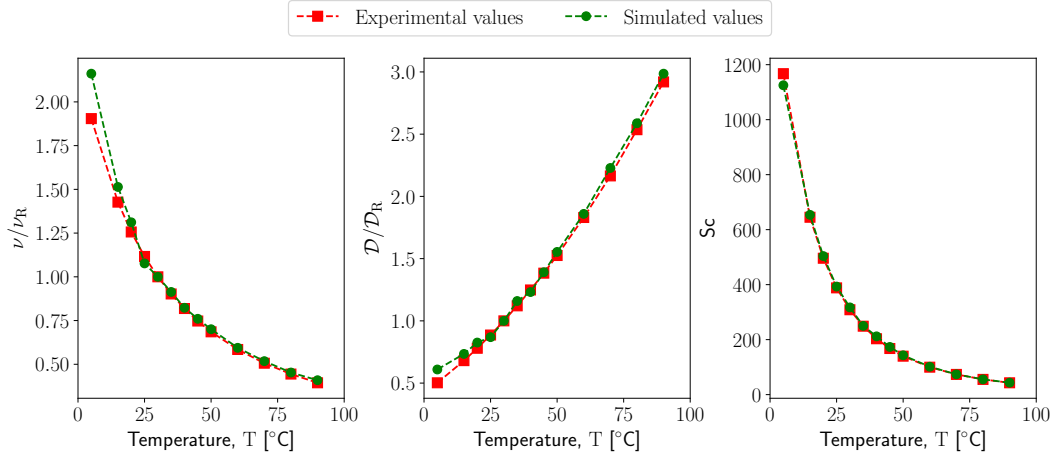


Fig. 6.6 Comparison of the transport properties at different temperatures between experimental data and the results of DPD simulations with the temperature-dependent conservative interaction parameter a and the temperature-dependent non-conservative interaction parameter s . The kinematic viscosity and diffusivity are scaled by their reference values at ν_R and $\mathcal{D}_R$ at $T_R = 25^\circ\text{C}$.

that the capability to properly capture the viscosity dependence on temperature basically depends on the definition of the temperature-dependent non-conservative parameter s_T .

Moreover, the evaluation of $\nu - \nu^{CC}$ allows comparison with theoretical predictions since they are derived by neglecting the conservative forces. The comparison is still approximate since the theoretical expressions also do not take into account the cross terms ν^{CD} and ν^{DC} of Eq. (5.3), but it is significant for the investigation carried out in this work. The right plot of Fig. 6.7 compares our viscosity values $\nu - \nu^{CC}$ at different temperatures and the ones predicted by two different theoretical expressions available in literature and previously discussed. The ν^{SWF} theoretical expression is obtained considering the standard DPD thermostat with the SWF; the s parameter is fixed to one for all fluids. Instead, ν^{GWF} theoretical expression is obtained assuming a standard DPD thermostat with the GWF^D . In this case, s can assume different values depending on type of fluids simulated.

In the right subplot of Fig. 6.7, ν^{GWF} is computed assuming $s = 0.4$. However in both expressions, s as an input parameter is independent from the system temperature. Although both theoretical predictions can give only a rough estimate of the viscosity since they are derived via a simplified approach, the behavior of the curves makes clear that the effect of temperature on viscosity cannot be described by the DPD

models assumed in their derivation. In fact, ν^{SWF} are much smaller than the realistic ones due to the absence of shear dissipation, whose inclusion has been proved to be essential to describe liquid-like dynamics. Moreover, they are almost constant with respect to the system temperature. The ν^{GWF} values are higher than the ν^{SWF} ones as expected, since the s parameter is smaller and it yields stronger non-conservative interactions. However, the ν^{GWF} viscosities due to the introduction of a smaller s are still almost constant and independent of the system temperature. Instead, our $\nu - \nu^{\text{CC}}$ s are high enough to provide realistic Sc numbers for liquid water and they also vary with temperature consistently with the expected experimental trend.

First, this result indicates that the extended DPD model employed and based on the transverse DPD thermostat with the GWF^D , provides the quantitative match between simulated and experimental transport properties. Second, it demonstrates that the methodology based on modeling the s parameter as a function of the system temperatures is successful to capture temperature-dependent transport properties. Figure 6.8 reports the values of the effective friction coefficient γ_{eff} measured at

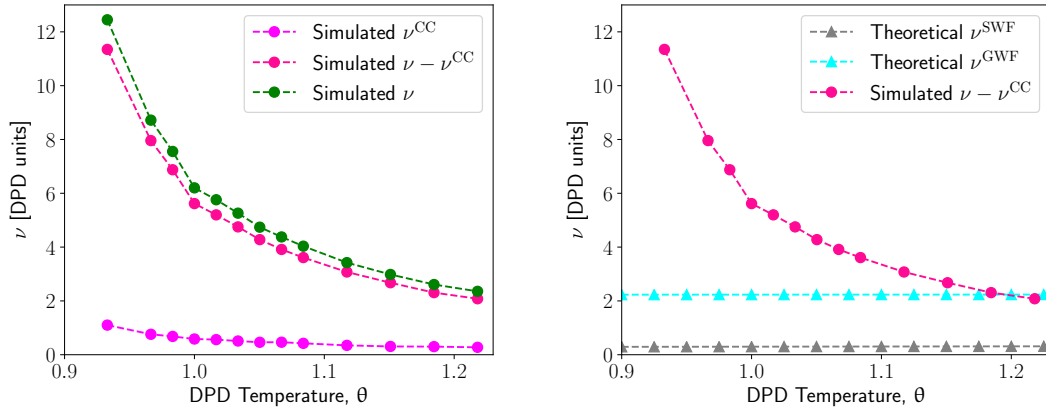


Fig. 6.7 Kinematic viscosity ν obtained from DPD simulations at different temperatures θ splitted into ν^{CC} and $\nu - \nu^{\text{CC}}$ contributions (left). Comparison between $\nu - \nu^{\text{CC}}$ simulated and the theoretical predictions of Fig. B.1 (right).

different temperatures. As expected, it decreases as the temperature increases. The simulation γ_{eff} s are fitted to

$$\gamma_{\text{eff}} = 102.9\theta^{-3} - 269.1\theta^{-1.5} + 236.7\theta^{-1} - 69.5. \quad (6.8)$$

The main contribution to γ_{eff} 's variation with temperatures comes from non-conservative interactions since the contribution from the RDF integration is almost constant. The RDFs do not vary significantly with the system temperature as shown in the left

subplot of Fig. 6.8 where the RDFs at different temperatures almost overlapped. To assess the model goodness-of-fit, the coefficient of determination (R-squared)

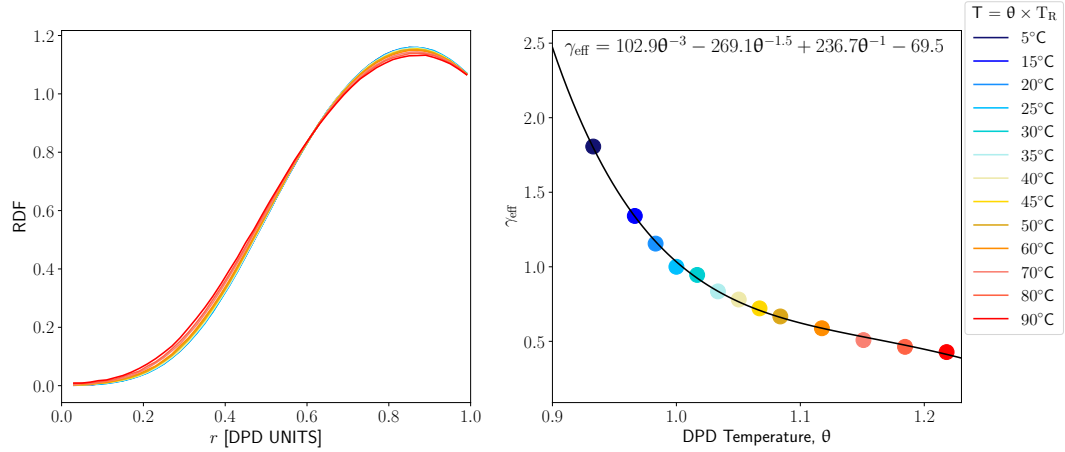


Fig. 6.8 Radial distribution function for each temperature investigated $T = T_R \times \theta$ (left). Effective friction coefficient γ_{eff} obtained at different temperatures (right).

and the normalized root-mean-squared error (NRMSE) are reported in Table 6.6 for all the fitting relations derived in this study. All the fitting relationships recovered present goodness-of-fit measures with the same order of magnitude.

Table 6.6 Goodness-of-fit measures from the fitting of the s values and the effective friction coefficient values γ_{eff} .

Fitted quantity	R^2	$NRMSE$
s_T	0.9980	0.0128
γ_{eff}	0.9983	0.0112

6.4 Conclusions

This Chapter investigated the use of the extended dissipative particle dynamics (DPD) model, numerically solved with the Shardlow-splitting-algorithm-based integration scheme, as a computational tool to predict temperature-dependent transport properties at the mesoscale following a top-down coarse-graining. The transport properties calculation from equilibrium simulations was performed using an in-house developed automated and reliable method, available online.

A temperature-dependent form of the extended DPD model, where both conservative and non-conservative interaction parameters incorporate the system temperature variation, was proposed. In particular, liquid water was used as model system in this work. Experimental data of transport properties [93, 94] from 5°C to 90°C were used as target values to develop and validate the model. The extended DPD model allows the simulation of water dynamics across different temperatures within the whole liquid range consistently with experimental data of viscosity and self-diffusivity. Although the model was developed and validated against transport properties of liquid water, it can also be used as a first rough estimation for the non-conservative parameters of other liquids.

The temperature-dependent form of the extended DPD model of this work can be further extended to multicomponent systems or systems with chain-like molecules. For the conservative interaction parameters, which are based on Groot-Warren's DPD parameterization [13], the temperature-dependent compressibility can be used to determine like repulsive parameters, a_{ii} , at various temperatures. Unlike repulsive parameters, a_{ij} , can be determined from a linear relationship between a_{ij} and the Flory-Huggins interaction parameter, which is inherently temperature-dependent. Alternatively for systems with chemically distinctly different beads or local density variations, Kacar et al.'s DPD parameterization can be used, which explicitly involves a temperature dependence [96, 97]. Parameters of non-conservative forces can be adjusted analogously as in this work to the viscosity and mutual self-diffusion coefficients. However, adjustment will involve different *itype* – *jtype* pairs, e.g., 1-1, 1-2, and 2-2 type-pairs for a binary system, and may become rather complex and demanding. Such the adjustment may benefit from machine learning methods.

The outcomes of this part of the thesis would also be useful for future developments of non-isothermal DPD models. In fact, the energy conserving DPD (DPDe) is slowly growing as a powerful tool in capturing temperature-induced mesoscopic phenomena, but it requires the definition of temperature-dependent interactions. The incorporation of a temperature-dependent conservative force-field was presented in different studies and it was shown to be capable of predicting the behavior of equilibrium properties induced by a thermal gradient. Instead on the other hand, the incorporation of temperature-dependent dissipative and random forces, such as those proposed in this work, is less straightforward. It requires further theoretical development to properly modify the DPDe framework. However, this extension is

interesting and crucial to exploit DPDe as a computational tool to simulate at the mesoscale also the transport properties behavior induced by a thermal gradient.

Chapter 7

Understanding the complex interplay between morphologies and material properties in Pluronic systems: insights from dissipative particle dynamics.

Pluronics are tri-block copolymers composed of PEO and PPO chains, as described in Section 2.1 of Chapter 2, they can form various microstructures depending on composition, temperature, and shear stress. This peculiar morphological behavior leads to complex rheology, which presents challenges for industrial processing. Predicting these rheological properties is crucial to designing and scaling up processes and equipment, as viscosity directly impacts flow behavior and stability in manufacturing.

Computational tools, especially DPD, offer an efficient alternative to labor-intensive experimental methods, allowing a rapid and cost-effective exploration of structural changes and their effects.

The study described in this chapter leverages some of the findings of previous chapters to apply DPD in conducting virtual experiments on Pluronic systems in water, with the results validated against real-world data. This chapter, in particular, focuses on two applications: first, simulating binary mixtures of Pluronic L64 and water to investigate how concentration variations affect microstructure formation and viscosity using an equilibrium approach; and second, exploring the behavior of

The content of this chapter, modified and integrated, is part of two publications under preparation.

Pluronic F68 in water at 45 wt% concentration at different temperatures. The latter study highlights the formation of an ordered domain, body-centered cubic (BCC), of micellar structures at elevated temperatures, revealing a reversible liquid-to-solid transition and a corresponding increase in viscosity.

7.1 Predicting concentration-dependent zero-shear viscosity in Pluronic L64/water mixtures

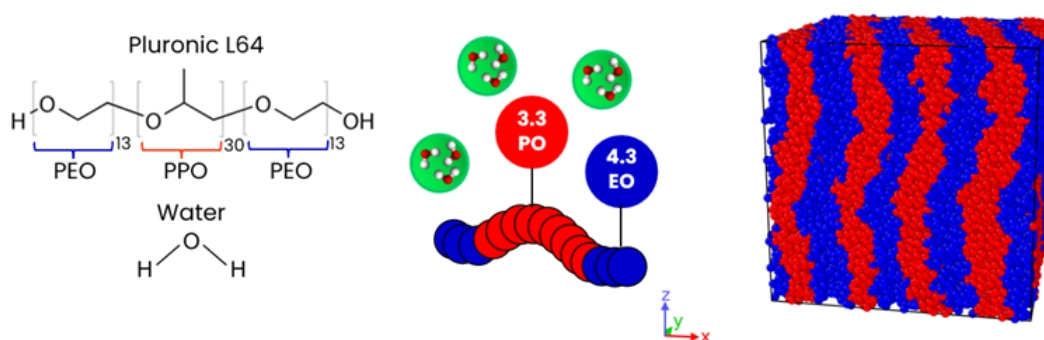


Fig. 7.1 Pluronic L64 and water molecules (left); coarse-grained model used in DPD (center); lamellar microstructure from DPD equilibrium simulation (right).

The aim is to explore how variations in concentration influence the evolution of microstructures and rheological properties in Pluronic L64-water mixtures. In fact, by varying the concentration, a spectrum of morphologies emerges that encompasses micellar, hexagonal, lamellar, and reverse micellar structures, as better described in Chapter 2. This diversity in microstructures results in complex rheology. An equilibrium approach for viscosity estimation is employed. This is an important step before evaluating the rheological behavior of structured fluids occurring under fluid motion or temperature change. In fact, it is important at first to assess the zero-shear viscosity of morphologically different formulations without perturbing them. Since the standard DPD model cannot describe real fluid transport properties as largely discussed in previous chapters, the “extended” DPD model presented in Chapter 3 is employed. This research introduces a criterion aimed at estimating non-conservative interaction parameters. The objective is to accurately capture the experimental trend of viscosity as a function of concentration.

7.1.1 Model Description, Methodology and Computational details

DPD Model for Pluronic L64

A Pluronic L64 molecule has the structural formula depicted in Fig. 7.2. The

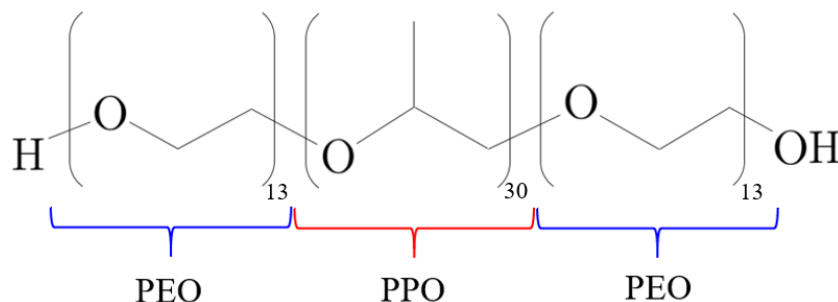


Fig. 7.2 Structural formula of the Pluronic L64 molecule (image adapted from Ref. [7]).

mapping process of the polymer chain requires a CGing procedure, where physical atoms are grouped in DPD-particle representations, so called beads. The mapping scheme adopted for the CG model allows to have beads that are intended to represent the same volumes within the chain [98]. According to this scheme, an EO bead, indicated with A consists of 4.3 repeating units of ethylene oxide, while a PO bead, denoted with B, contains 3.3 repeating units of propylene oxide. The Pluronic L64 macromolecule is simplified into a chain of 15 beads, as shown in Fig. 7.3. It can be schematically represented as $A_3B_9A_3$, where A and B correspond to the EO and PO beads, respectively. Water molecules are represented by a third type of bead

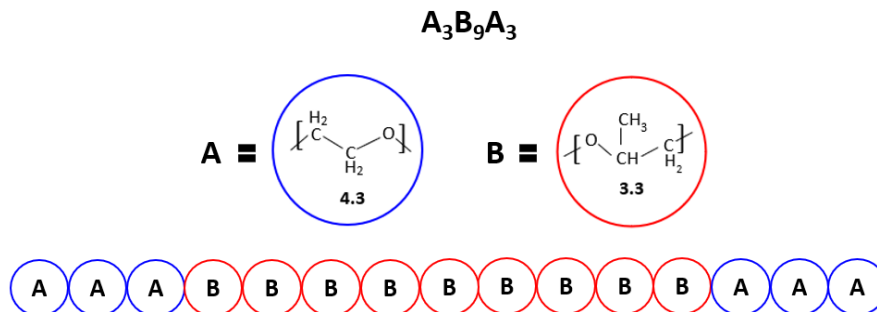


Fig. 7.3 DPD model resulting from the coarse-graining procedure of a Pluronic L64 molecule.

(C), which corresponds to ten water molecules. The re-normalization factors are chosen such that all beads have the same mass and volume [99]. The repulsion

Table 7.1 Coarse-grained beads, molecular volumes of H₂O, EO, and PO calculated according to the approach by Durchschlag and Zipper [15], and the corresponding bead volumes expressed in Å³.

CG bead	Molecular volume (Å ³)	Volume bead (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

coefficients between different types of beads in the DPD model were taken from the literature [16]. They are listed on Tab 7.2 and a schematic representation of the conservative interactions between different types of beads is presented in Fig. 7.4

The chemical bonds between the monomers forming the polymer molecule are

Table 7.2 Interaction parameters [16], reported in DPD units, for the three species: PEO, PPO, and water.

	A (PEO)	B (PPO)	C (H ₂ O)
A (PEO)	25.0	48.9	25.9
B (PPO)	48.9	25.0	38.4
C (H ₂ O)	25.9	38.4	25.0

represented using the bead-and-spring model, in which adjacent beads in the polymer chain interact via a harmonic potential, described by the equation:

$$E_{\text{harm}} = K_b(r_{ij} - r_0)^2. \quad (7.1)$$

The choice of the nominal equilibrium distance r_0 and the harmonic constant K_b for each pair of beads must ensure that the topology of the polymer chains is preserved, which is crucial for accurately modeling rheological properties. Various values for these harmonic potential parameters have been tested through different simulations. Excessively high K_b values caused excessive rigidity in the polymer chain. An optimal K_b value was identified, which balances repulsive interactions and bonding

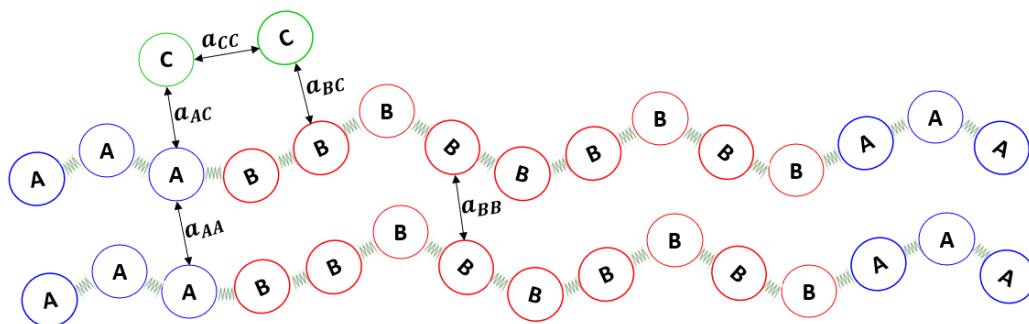


Fig. 7.4 Conservative interactions between the different types of beads in the system: A (PEO), B (PPO), and C (H_2O).

strength to maintain flexible and realistic polymer chain configurations. We fixed a unitary nominal equilibrium distance in DPD units and tested different values for the harmonic constant K_b .

Methodology and computational details

Binary mixtures of Pluronic L64 are investigated using DPD, with polymer concentrations ranging from 5 to 85 wt% (5, 15, 25, 30, 35, 45, 55, 65, 75, and 85 wt%). Since the standard DPD model cannot describe real fluid transport properties, the “extended” DPD model, described in Chapter 3, is employed.

Morphological characterization is performed using a clustering algorithm, as described in Chapter 2, which allows the identification of distinct aggregates and offers insight into their size and shape variations. This approach facilitates the detection of different microstructures across concentrations.

Viscosities are computed using the G-K based equilibrium approach, discussed in the previous chapters.

After defining the model and method for computing viscosity, the next step, central to this work, is identifying the model parameters required to capture our physical target, the zero-shear viscosity. The previous chapters, particularly Chapter 6, focused on parameter identification for liquid water at different temperatures. Here, we extend this process to Pluronic interactions at a fixed temperature. With the solvent accurately described in terms of its thermodynamic and transport properties, as established in Chapter 6, we explored two approaches to modeling Pluronic interactions. The first applied uniform dissipative parameters to all interactions, leading to underestimated viscosities relative to the experimental data. To address this, we

introduced a criterion for estimating non-conservative interaction parameters, making them interaction-specific and concentration-dependent. This approach aims to better capture the experimental viscosity trend as a function of concentration. To this purpose, the dissipative coefficient γ_{12} , known as *cross γ -parameter*, is calculated as follows:

$$\gamma_{12} = x_2\gamma_{11} + x_1\gamma_{22} , \quad (7.2)$$

that is as a linear combination of γ_{11} , the dissipation coefficient related to the interaction of two water beads, and γ_{22} , the dissipation coefficient related to the interaction of two Pluronic beads; x_1 is the mass fraction of the aqueous component in the mixture, and $x_2 = 1 - x_1$ is the mass fraction of the copolymer in the mixture. The index 1 corresponds to the water beads, while the index 2 represents the Pluronic beads (both PEO and PPO). This formula is developed in analogy to Darken's model for the calculation of the mutual diffusion coefficients of two components in a mixture based on their respective self-diffusion coefficients [100].

7.1.2 Results and discussion

Phase diagram

The study successfully replicates the morphological transitions induced by the changes in polymer concentration, closely aligned with the experimental phase diagram, as shown in Fig. 7.8. At low polymer concentrations (5-45 wt%), in agreement with the behavior of the experimental phase, a micellar morphology was found. At these concentrations, the block copolymer macromolecules self-assemble into spherical micelles. The model is able to capture the concentration-induced transition from micellar phase to gel-like phase around 45 wt% of polymer concentration, as evidenced by the behavior of shape descriptors in Fig. 7.5. The absolute values of λ_x , λ_y , λ_z increase by an order of magnitude with respect to the micellar phase and their mutual ratios are almost equal as expected for the gel phase. In addition, the transition from the gel phase to the lamellar phase is properly captured, as emerges from the significant increase in λ_z/λ_x .

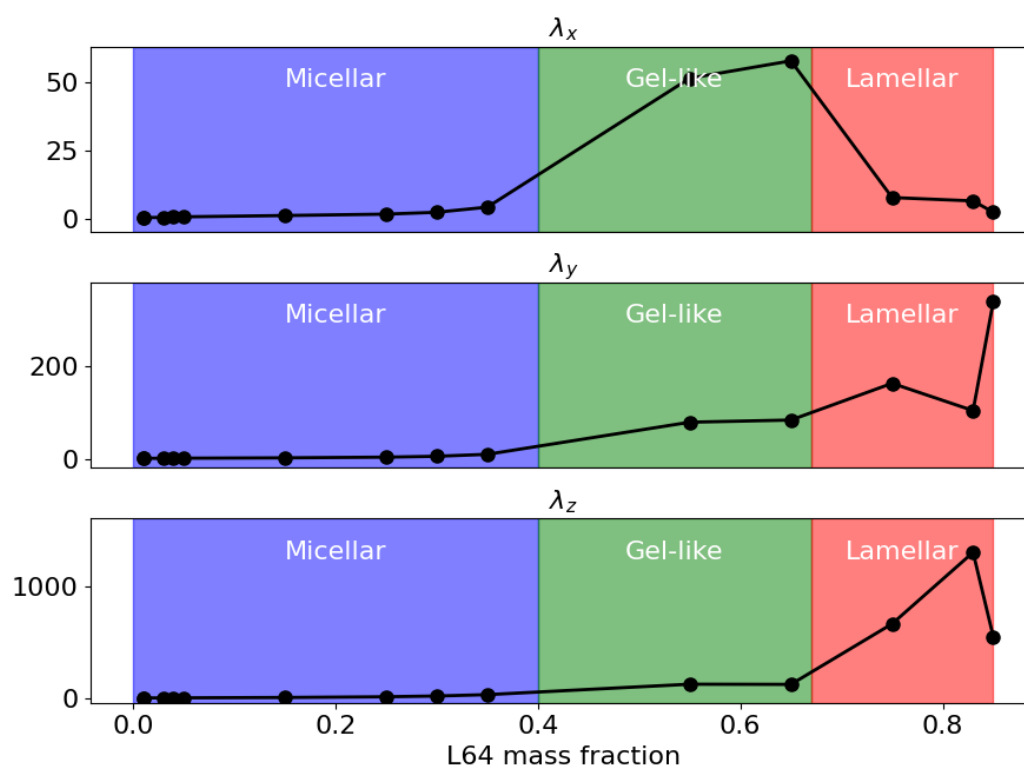


Fig. 7.5 Concentration-induced phase transition analyzed through shape descriptors. This figure illustrates the relationship between eigenvalues and varying concentrations, highlighting the detection of phase transitions induced by concentration changes.

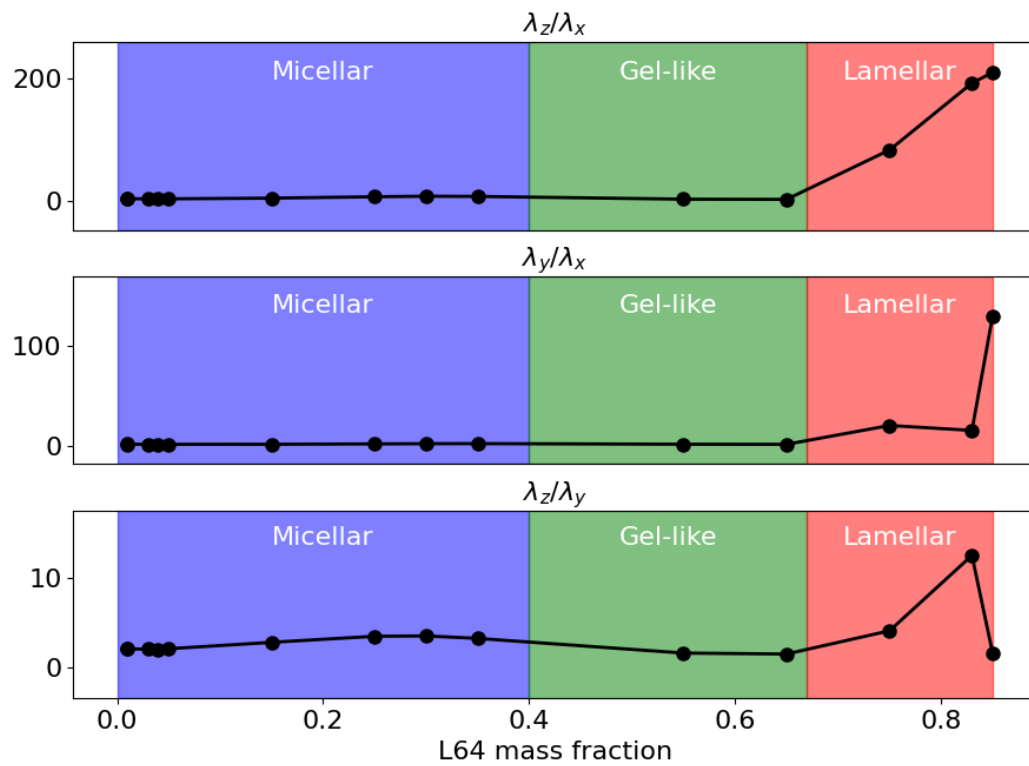


Fig. 7.6 Concentration-induced phase transition analyzed through shape descriptors. This figure presents the relationship between the ratios of eigenvalues and varying concentrations, emphasizing the detection of phase transitions induced by changes in concentration.

Zero-shear viscosity predictions

The approach used to identify non-conservative parameters, as described in Section 7.1.1, establishes a strong correlation with the experimental zero-shear viscosity data. Figure 7.8 shows that the viscosity predictions derived from the simulations are in strong agreement with the experimental data, particularly at lower polymer concentrations. This confirms the model's effectiveness in capturing the system's rheological behavior within the Newtonian region. However, further refinements in both the model and the methodology for computing rheological properties are needed to achieve a comprehensive characterization in the non-Newtonian range.

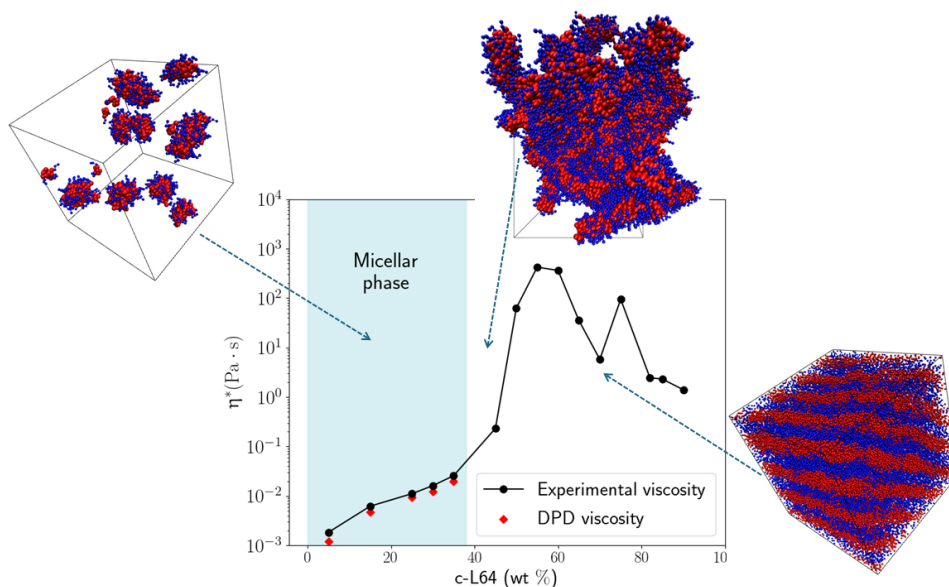


Fig. 7.7 Complex viscosity, evaluated at 1 rad s^{-1} , plotted against Pluronic L64 weight concentration. Predicted microstructures using DPD at various concentrations are illustrated alongside. The figure also includes a comparison with DPD zero-shear viscosity at different concentrations within the micellar range.

7.2 Predicting temperature-dependent liquid-to-solid transition in Pluronic F68/water mixtures

This part of the research work involves simulating the behavior of Pluronic F68 in water, particularly focusing on its intriguing behavior at a concentration of 45% by weight (wt) across various temperatures. Notably, at elevated temperatures, the sys-

tem exhibits a distinctive characteristic: it assembles into an ordered supramolecular structure known as the BCC phase, resembling a solid-like structure. This observation holds significant implications, especially in the realm of drug delivery systems. Indeed, since many drug delivery systems rely on ordered and thermodynamically stable nanostructures, such as BCC phase, Pluronic F68 shows good promise as a candidate for therapeutic applications. The ability of adjusting and controlling the morphology and rheological properties of drug delivery systems in which a pharmaceutical molecule is loaded stands as hot topic in therapeutic applications where the release of a drug cargo in human body needs to be controlled and smart. By leveraging computational tools such as DPD in conjunction with real experiments, a wide range of scenarios can be explored accelerating the design process. The study conducts DPD simulations of systems composed by 45 wt% of Pluronic F68 and water. To ensure the reliability and validity of the simulation results, they are compared and validated against experimental data obtained from an experimental work involving the use of scattering techniques and rheological measurements to characterize the nanostructure and rheological properties of the system under similar conditions.

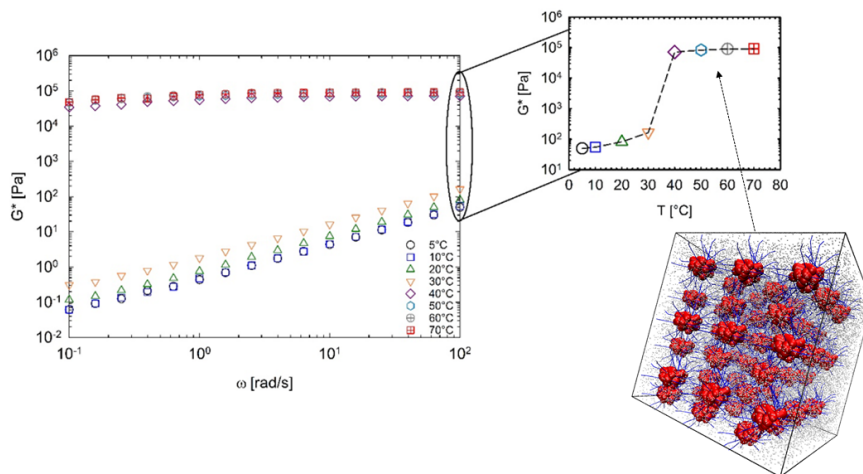


Fig. 7.8 Pluronic F68 water solution plotted at various temperatures. A zoomed-in view at 100 rad/s is provided, highlighting the transition from a liquid-like to a solid-like phase with increasing temperature. Additionally, the figure includes a snapshot of the simulated body-centered cubic (BCC) lattice.

7.2.1 Model Description, Methodology and Computational details

DPD Model for Pluronic F68

The mapping scheme used to model Pluronic F68 is the same as that adopted for L64: the number of monomers clustered into one bead is 4.3 for the ethylene oxide (EO) repeating units and 3.3 for the propylene oxide (PO) repeating units, resulting in 45 beads for a Pluronic F68 chain with the structure $(A_{18}B_9A_{18})$. Conservative parameters are mapped onto Flory-Huggins parameters as explained in the Section 2.2.2 of the Chapter 2. They were then adjusted to match the experimental radii of both micelle core and corona at $T=25^\circ\text{C}$, which is the reference temperature. Figures 7.9 and 7.10 compare experimental and computed data. These parameters are temperature-dependent, as previously reported in the literature. As the temperature increases, a_{AW} and a_{BW} increase. The chemical bonds between the monomers that

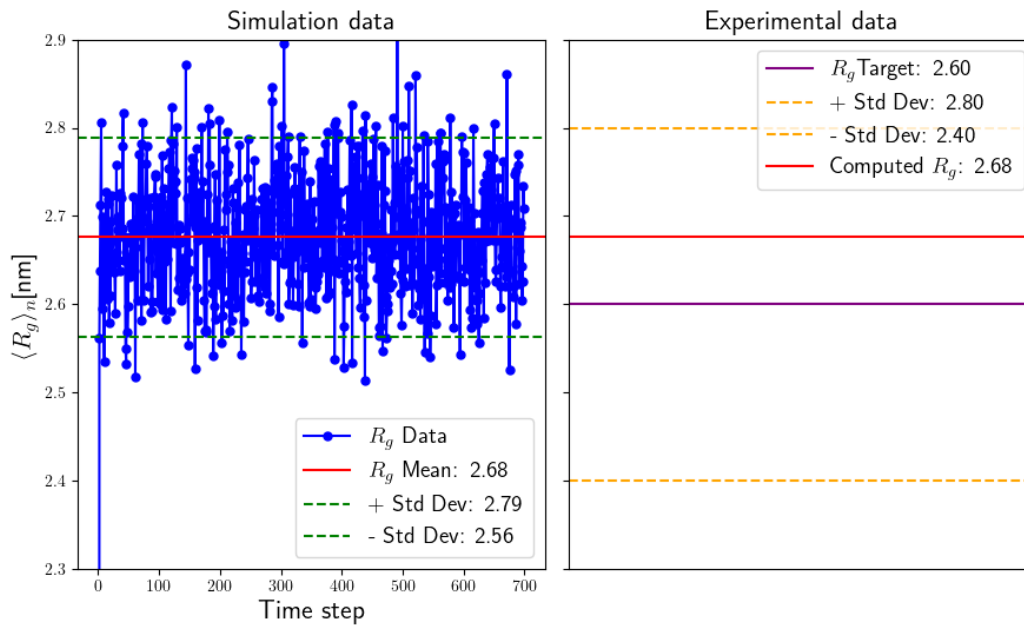


Fig. 7.9 Comparison of the computed core radius and the experimental core radius of Pluronic F68 at 25°C .

form the polymer molecule are represented using the bead-and-spring model as well as for the L64 system in the previous paragraph. Additionally, for temperatures at which gel phase formation occurs, it is necessary to introduce the Segmental

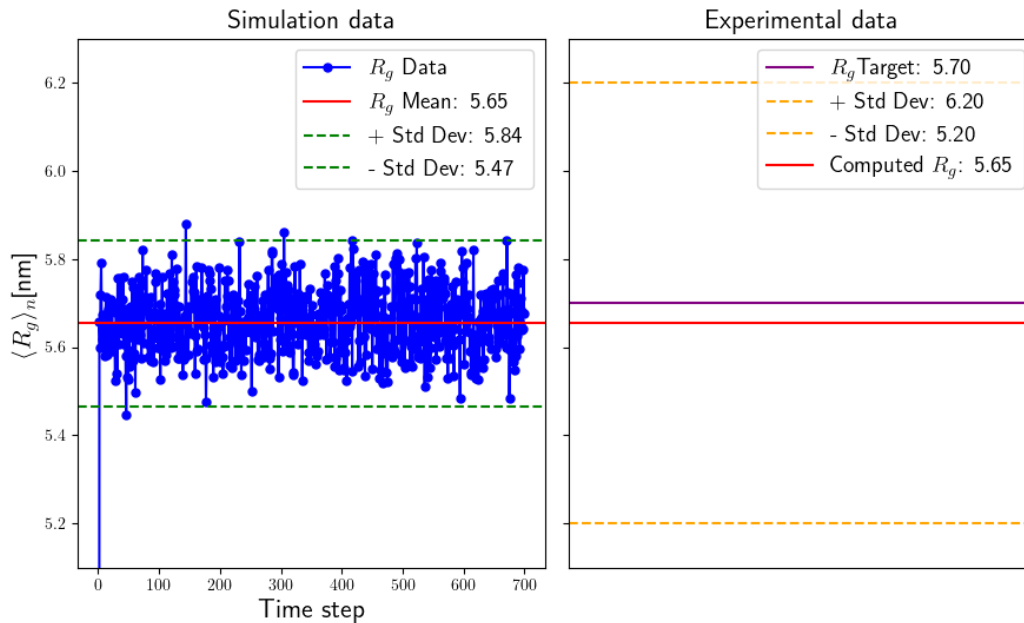


Fig. 7.10 Comparison of the computed micellar radius and the experimental micellar radius of Pluronic F68 at 25°C.

Repulsive Potential (SRP), which prevents the nonphysical crossing of the chain beads bonds, otherwise allowed by the soft repulsive interactions of the DPD model. The rheological properties depend on the physical entanglements or chain crossing characteristic of the gel. In the context of the DPD simulation, it is necessary to isolate and consider only the contribution of the latter for the calculation of the viscosity, in order not to have nonphysical increases of the latter [101–105]. In fact, while a long-standing advantage of the standard DPD model of polymer chains has been that microscale systems can reach an equilibrated structure within a reasonable simulation time, the dynamic path that such a simulation takes does not necessarily correspond to the true path. One reason is that the DPD polymer chains are able to artificially cross through each other. DPD model can reproduce equilibrium thermodynamic and structural properties of polymer systems, but not rheological behavior such as stress-strain curves or dynamic response to external stimuli. The SRP model presented here helps to overcome this deficiency by reproducing the equilibrium properties of the standard DPD model, while also providing a means to reproduce the non-equilibrium behavior. The repulsive interaction force between non-adjacent

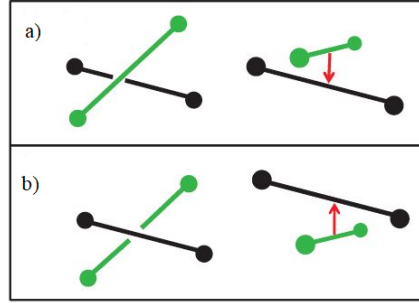


Fig. 7.11 Example of topological violation for the nonphysical crossing of bonds (green line and black line) between the beads of the polymer chain represented by the filled circles, known as bond crossing (image processing extracted from Ref. [8]). The minimum distance vector (in red) changes its orientation by 180° between the time instants t and $t + \Delta t$. Configuration (a) is relative to the time instant t . Configuration (b) is relative to the time instant $t + \Delta t$.

bonds has a form similar to the conservative force through which they interact:

$$\mathbf{F}_{kl}^{SRP} = b \left(1 - \frac{d_{kl}}{d_c} \right) \hat{\mathbf{d}}_{kl}, \quad (7.3)$$

where $\mathbf{F}_{kl}^{SRP}$ is the force acting between the bonds k and l separated by distance d_{kl} ; b e d_c are respectively the force constant and the distance of *cutoff bond-bond*. The distance between the two bonds can be calculated as the minimum distance, *minimum bond-bond distance* or as the distance between the midpoints of the bonds, *midpoint-to-midpoint distance*. In this latter case, the repulsive potential between the bonds is designated as mSRP (modified SRP) since it results from an optimization of the previous model [8]. In this study, mSRP is used to mitigate the undesirable effects of SRP, such as the increased repulsive interaction between beads, which introduces an excluded volume around the bond. This alteration compromises the accurate reproduction of the system's topology and negatively impacts the formation of microstructures and rheological properties.

Methodology and computational details

DPD equilibrium simulations were conducted to investigate the self-assembly behavior of F68/water mixtures at 45 wt% across different temperatures. Five temperatures were considered in this work: 5°C , 25°C , 28°C , 40°C , and 70°C . A system of $N =$ identical particles with $m_i = m = 1$ was simulated in a cubic box of length equal to

$40r_c^C$ with periodic boundary conditions in all directions. The number density was set to $\rho = 3$.

The temperature T varies within the temperatures range, i.e., from 278.15 K (5°C) to 343.15 K (70°C). To carry out isothermal DPD simulations across different temperatures, it was useful to introduce a temperature scale [14], as already done for liquid water in Chapter 6. The temperature of reference $T_R = 298.15$ K was used to scale the temperatures T . Therefore, the simulation temperature $\theta = T/T_R$ ranges from 0.933 to 1.15. It is equal to unity only in the particular case of $T = 298.15$ K. DPD fluid interaction parameters are setup coherently with the simulation temperature θ . The conservative repulsive parameter, a_{WW} , is defined accordingly to what explained in Chapter 6. a_{AA} and a_{BB} are assumed to be equal to a_{WW} . The interaction parameters between various types of polymer beads are modeled as temperature-dependent to reflect changes in how the hydrophobic block interacts with other species. According to Alexandris et al., the hydrophobic block of Pluronic induces micellization due to the weakening hydrogen bonding between water and the PPO block as the temperature rises. As a result, the χ interaction parameters for PEO-water and PPO-water increase with temperature, as already discussed in other works [106].

Equilibrium trajectories generated at various temperatures are analyzed using the post-processing algorithm detailed in Chapter 2 to calculate the micelle aggregation number (A_s) and the radius of gyration (R_g) of micelles and PPO cores. As discussed in Chapter 2, the analysis begins by identifying Pluronic aggregates, where two molecules are considered part of the same aggregate if they share at least a minimum number of contact pairs. A contact pair is defined as two beads from different molecules that are within a specified distance. The weight distribution function of aggregation numbers, F_w , is given by:

$$F_w(A_s) = \frac{m_{A_s} N_{A_s}}{\sum_i m_{A,i} N_{A,i}}, \quad (7.4)$$

where $m_{A,i}$ represents the weight of an aggregate with an association number $A_s = i$ and $N_{A,i}$ denotes the number of aggregates with that association number. The aggregation number, A_s , represents the number of chains in an aggregate. Its weight-average value, computed as

$$\langle A_s \rangle_w = \frac{\sum_i m_{A,i}^2}{\sum_i m_{A,i}}, \quad (7.5)$$

is used for comparison with experimental data and is referred to simply as A_s in this work. Finally, R_g is derived from the gyration tensor, with its weight-average value compared to the experimental results and denoted as R_g for simplicity. Further details of these calculations are provided in Chapter 2.

In non-equilibrium simulations, the friction coefficient γ is also treated as a temperature-dependent parameter and calibrated to reproduce the experimental trend of the loss modulus with respect to temperature. After the equilibrium structures were obtained and characterized, the non-equilibrium oscillatory shear technique was applied by oscillating the tilt factor sinusoidally with the specified amplitude and period. A triclinic (non-orthogonal) cubic box was adopted in the simulations, where periodic boundary conditions and the NVE ensemble were applied. All simulations are performed using LAMMPS, and specifically the “extended” model explained in previous chapters. The storage modulus and the loss modulus are measured by applying the small amplitude oscillatory shear technique, described in Chapter 2.

7.2.2 Results and discussion

By comparing the simulation results with the experimental data, this study aims to elucidate the relationship between the morphological behavior and the macroscopic rheological properties of the Pluronic F68 water system at different temperatures.

7.2.3 Morphology

Experimental rheological data reveal a liquid-to-solid transition around 35°C, where the system adopts the characteristics of a soft solid. It behaves as a non-Newtonian shear-thinning material, exhibiting well-defined viscoelasticity and relaxation time. To further investigate the mechanisms underlying this behavior across temperatures, we also performed morphological measurements. Two factors that may influence this transition are the presence of ordered domains and entanglements, as illustrated in Fig. 7.12. Entanglements may result from the formation of larger micelles or an increase in the number of micelles. Experimental data indicate that the micelle radius remains nearly constant as the temperature increases. Simulations offer rapid insight into details such as the number of aggregates and their average number of aggregations between temperatures. Post-processing calculations enable distinguishing sub-micellar aggregates from micelles by applying a cut-off aggregation number,

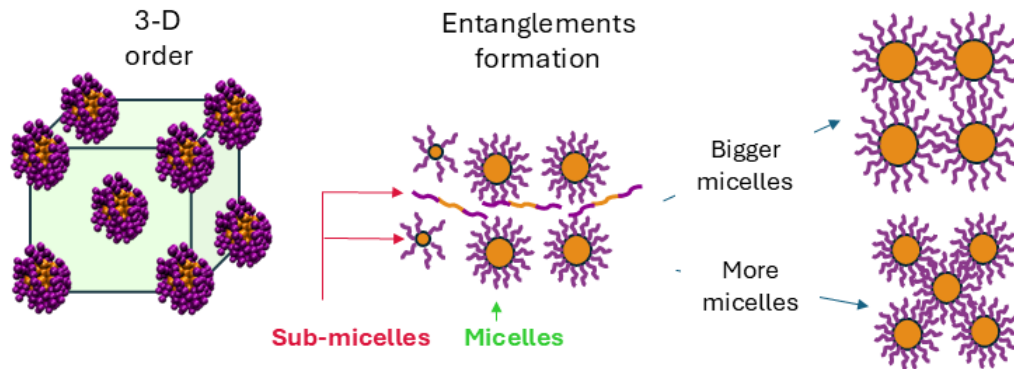


Fig. 7.12 Illustration of the possible mechanisms behind elasticity: ordered domains and entanglements.

facilitating the study of their evolution across varying temperatures. Figures 7.14 and 7.13 illustrate the total aggregates (sub-micelles and micelles) and the average aggregation number over the examined temperature range. Figures 7.13 and 7.14

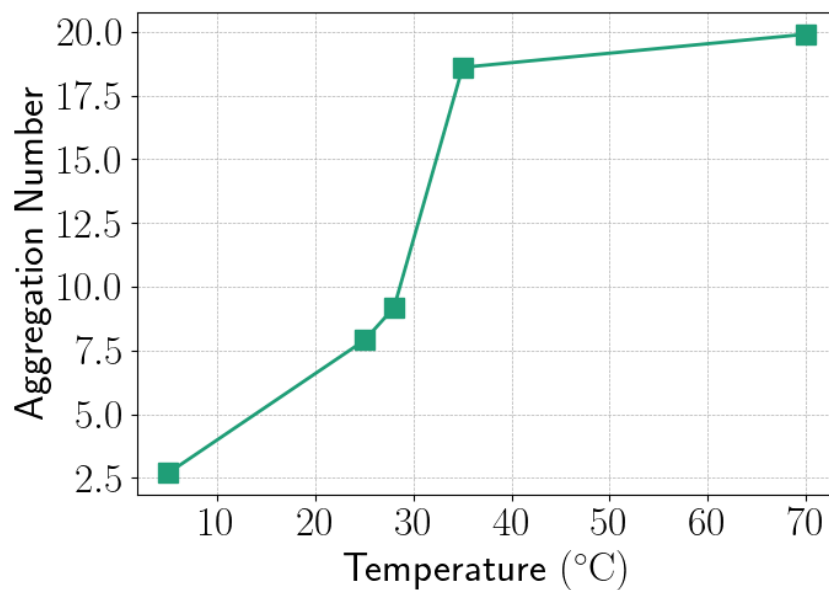


Fig. 7.13 Aggregation number as a function of temperature.

show that from 5 to 25°C, sub-micelles decrease as micelles form. Between 25 and 28°C, sub-micelles reduce as existing micelles grow. This is demonstrated by an increase in the average number of aggregation and, at the same time, an almost stable value of the micelle number. With further temperature increase, sub-micelles

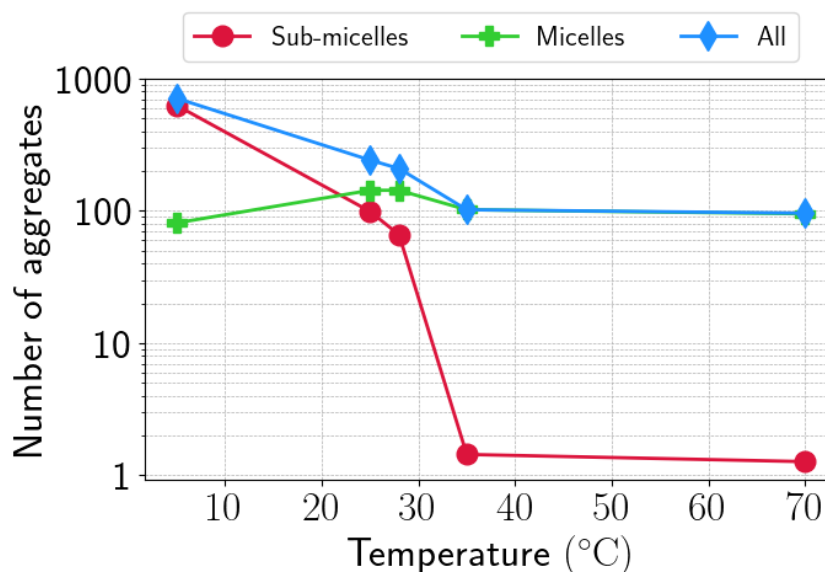


Fig. 7.14 Number of aggregates (sub-micellar and micellar) as a function of temperature.

essentially disappear. This drop in their number occurs around 35°C, confirming the experimental evidence.

The weight-averaged distribution functions of the aggregation number help to provide a better interpretation of these plots. At 5°C (Fig. 7.15), the absence of a peak in the distribution indicates the lack of aggregates large enough to qualify as micelles. At 25°C (Fig. 7.16), the emergence of a peak signals the onset of micelle formation. At this stage, the system contains both sub-micelles and micelles of varying sizes, freely fluctuating. This promotes the formation of a disordered domain, leading to a viscous macroscopic behavior. The disordered domain is clearly visible in the snapshot shown in Fig. 7.19a. The corresponding rheological properties will be discussed in the next paragraph. At 28°C (Fig. 7.17), the distribution function remains nearly unchanged, reflecting the continued growth of the micelles. At 40 °C (Fig. 7.18), the distribution becomes sharply compact, indicating micelles of uniform size, packed tightly together. Their reduced mobility leads to the formation of an ordered domain, where the proximity of micelles promotes the entanglement or weak crossing of the PEO chain, resulting in a macroscopic viscoelastic response. The ordered domain is clearly visible in the snapshot shown in Fig. 7.19b. The corresponding rheological properties will be discussed in the next paragraph.

The micellar radius and PPO core obtained from simulations are compared to

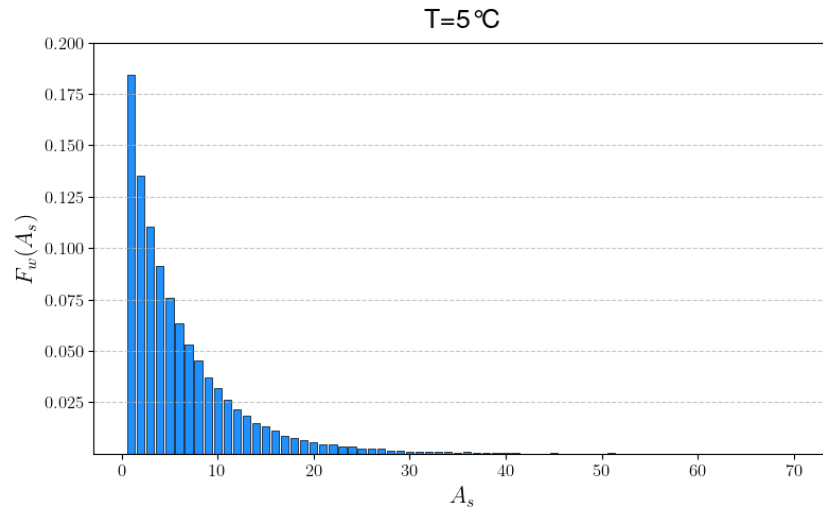


Fig. 7.15 Weight-averaged distribution function at T=5°C

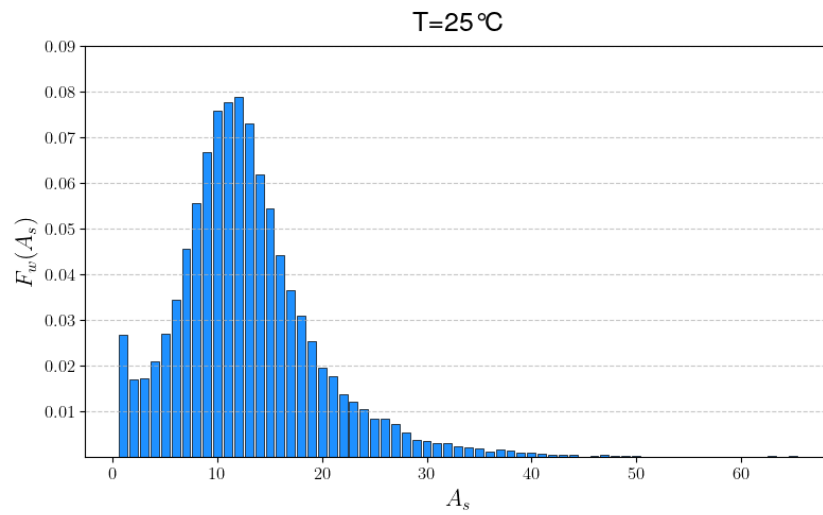


Fig. 7.16 Weight-averaged distribution function at T=25°C

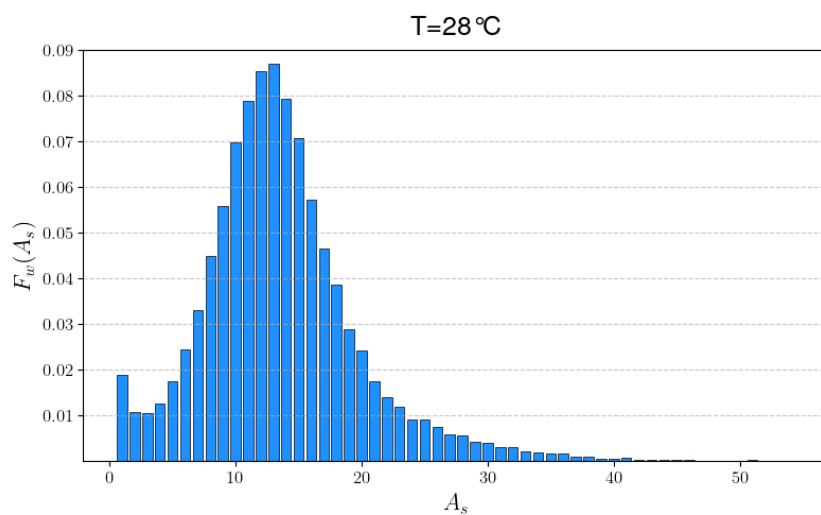


Fig. 7.17 Weight-averaged distribution function at $T=28^\circ\text{C}$

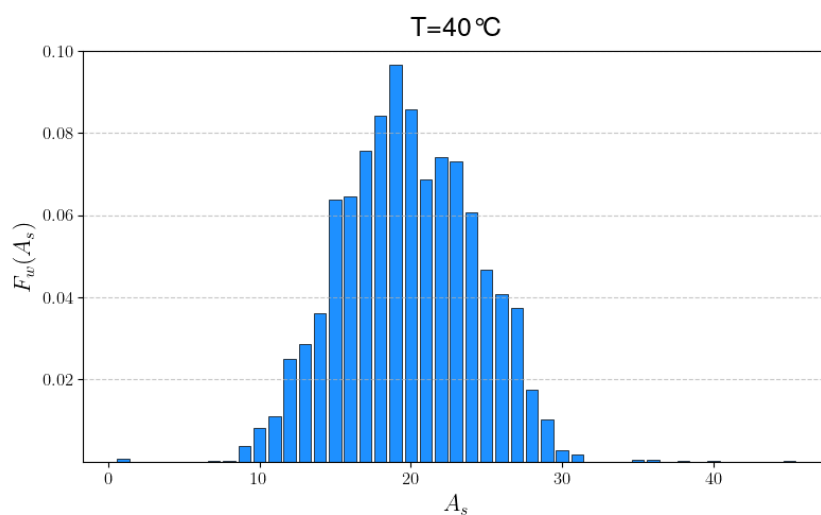


Fig. 7.18 Weight-averaged distribution function at $T=40^\circ\text{C}$

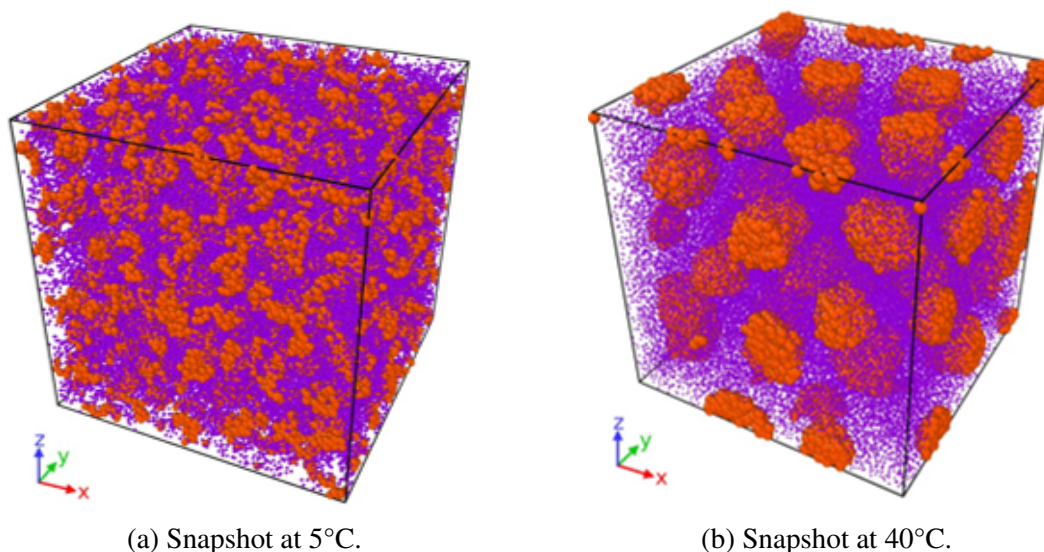


Fig. 7.19 Snapshots at different temperatures.

SAXS data. The experimental results in Fig. 7.20 reveal that the radius of the micelle remains nearly constant, while the radius of the core shows a slight increase as the temperature rises. As a result, the PEO shell radius slightly decreases with increasing temperature. These values are compared with the values calculated in Figs. 7.21, 7.22 and 7.23. In light of the uncertainty linked to both experimental and simulated parameters, the match is quite good at low temperature. As the temperature increases, a misalignment occurs. Simulations predict that the micelle radius increase and the PPO core is almost constant. Therefore, it can be deduced that there is an expansion of the PEO shell. This result is significant for understanding the rheological behavior of the system. The contact between chains could explain the microstructural basis underlying the well-defined viscoelasticity and relaxation time. Further investigations are needed to evaluate the experimental models used to extract these values and to deepen the interpretation. RDFs at different temperatures (Figs. 7.24, 7.25, 7.26 and 7.27) confirm the shift in microstructural order with temperature.

7.2.4 Rheology

This section presents the results for the linear viscoelasticity of F68/water mixtures at temperatures ranging from 5°C to 70°C, at 45 wt%. Although preliminary, these results are promising and can provide a functional foundation for future research. The trends for the storage modulus (G') and the loss modulus (G'') are promising,

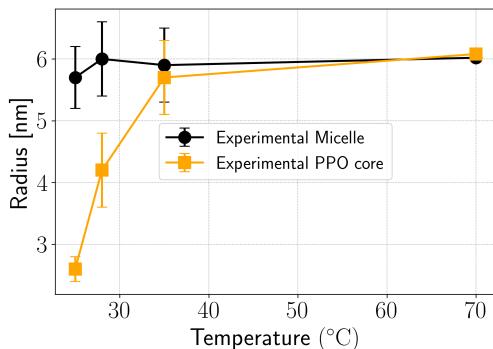


Fig. 7.20 Experimental measurements of micellar and PPO core radii across a range of temperatures.

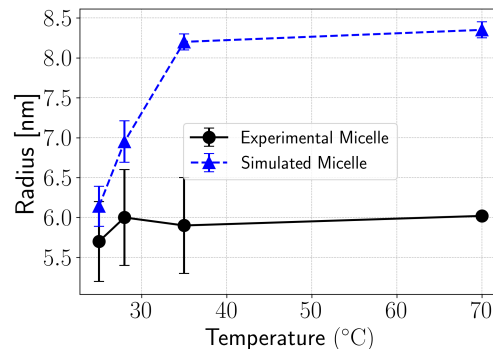


Fig. 7.21 Comparison of experimental and computed micellar radius measurements across a range of temperatures.

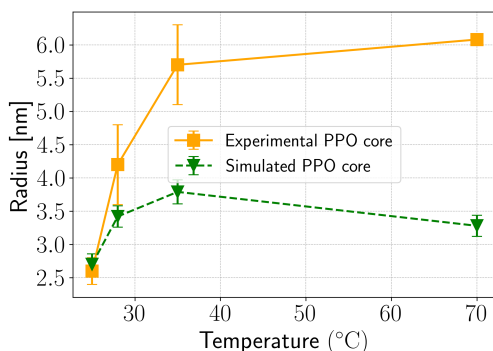


Fig. 7.22 Comparison of experimental and computed PPO core radius measurements across a range of temperatures.

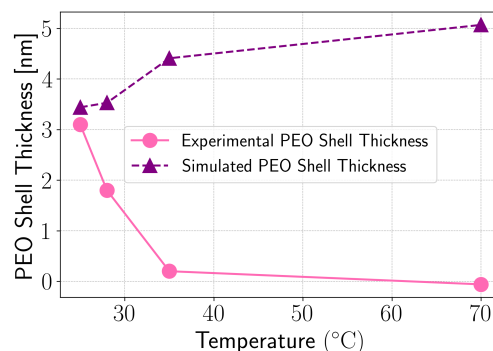


Fig. 7.23 Comparison of experimental and computed PEO shell radius measurements across a range of temperatures.

with a step change around 40°C that aligns with the experimental observations. However, the results are reported in DPD units. More work is needed to establish a reliable conversion framework into real units.

7.3 Conclusions

In this chapter, we demonstrated, through two distinct applications, that DPD is a valuable tool to investigate the interplay between morphology and rheology in Pluronic systems and to have a deeper understanding of the phenomena involved. In both cases, the results showed a strong correlation with the experimental data.

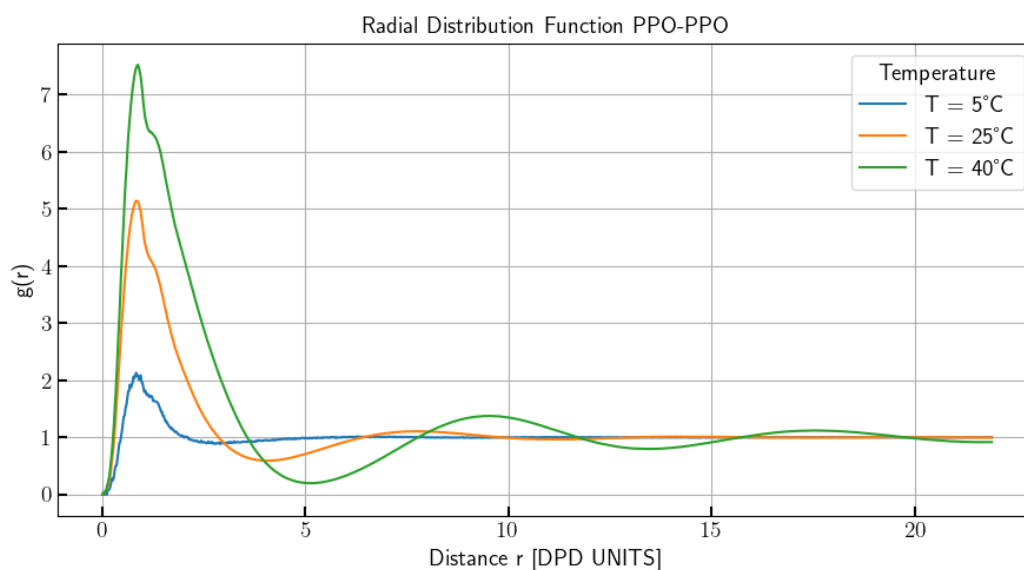


Fig. 7.24 Radial distribution functions (RDFs) of PPO-PPO interactions at varying temperatures.

In the first case study, we showed that DPD can successfully predict the concurrent changes in morphology and zero-shear viscosity values induced by variations in concentration in a binary mixture of Pluronic L64 and water.

In the second case study, we demonstrate that DPD can effectively capture the liquid-to-solid transition through both morphological characterization and rheological measurements induced by a temperature change in a binary mixture of Pluronic F68 and water.

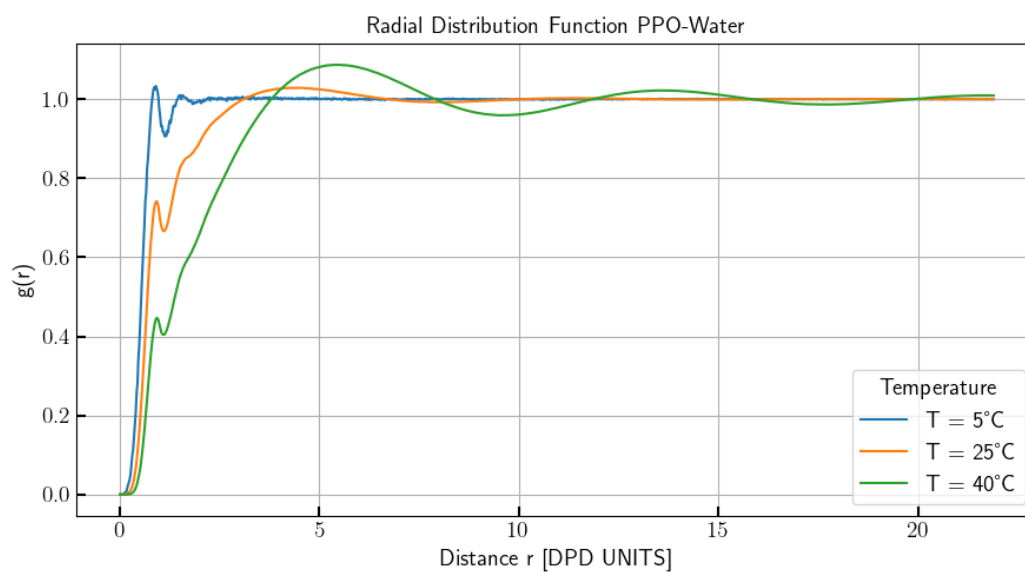


Fig. 7.25 Radial distribution functions (RDFs) of PPO-Water interactions at varying temperatures.

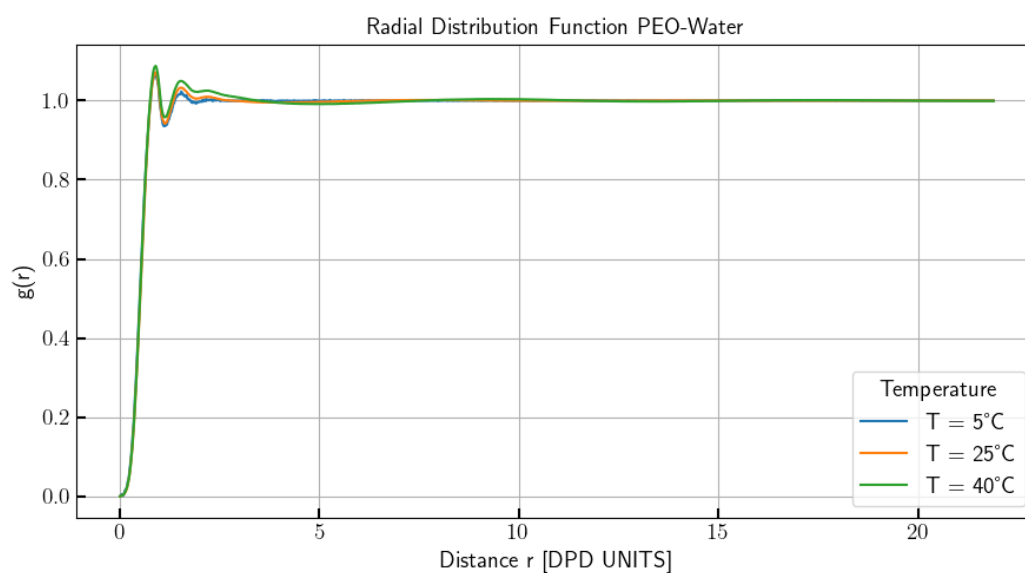


Fig. 7.26 Radial distribution functions (RDFs) of PEO-Water interactions at varying temperatures.

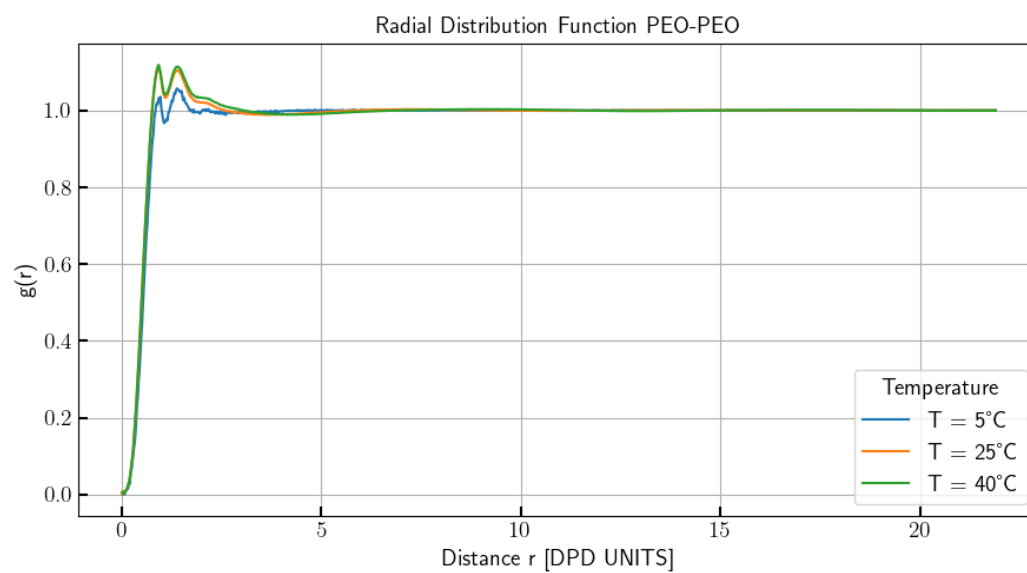


Fig. 7.27 Radial distribution functions (RDFs) of PEO-PEO interactions at varying temperatures.

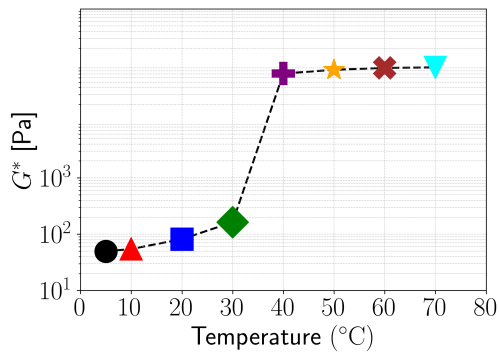


Fig. 7.28 Experimental measurements of the complex modulus across different temperatures.

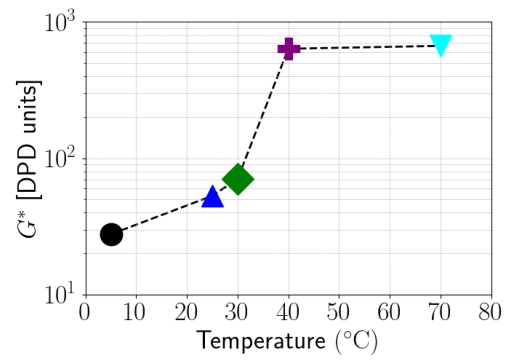


Fig. 7.29 Computed values of the complex modulus in DPD units across different temperatures.

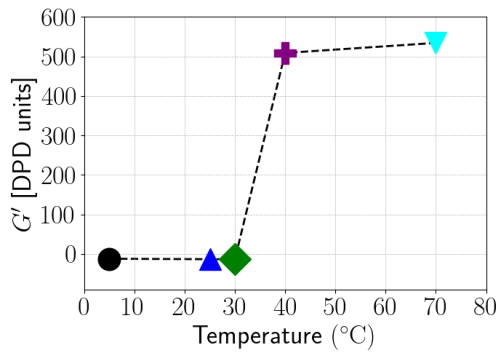


Fig. 7.30 Computed values of the elastic modulus in DPD units across different temperatures.

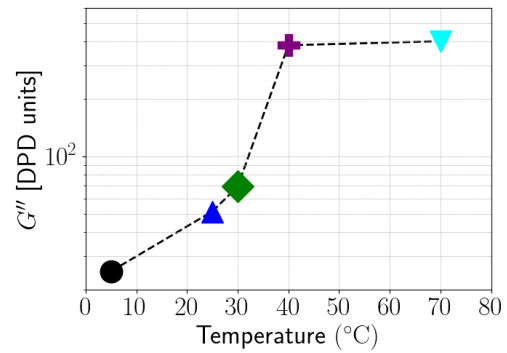


Fig. 7.31 Computed values of the loss modulus in DPD units across different temperatures.

Chapter 8

Machine-learning enhanced parameterization methodology for Pluronics Water Mixtures in DPD simulations.

As detailed in Sec. 2.1 of Chapter 2, Pluronics, or poloxamers, are synthetic amphiphilic copolymers with a triblock structure: a hydrophobic PPO core flanked by two hydrophilic PEO chains. They are available in numerous types. Their versatility, biocompatibility, low toxicity, temperature sensitivity, and self-assembly capabilities make them ideal for a wide range of applications. These include drug delivery, tissue engineering, bioprinting, and nanomedicine, along with uses in dispersion stabilization, emulsification, lubrication, formulation, and surface modification to improve biocompatibility in medical applications. They are also commonly applied in the cosmetics industry. The wide range of commercially available Pluronics is due to the flexibility in adjusting the lengths of the PPO and PEO chains. This allows for the customization of their properties and overall molecular weight by modifying the lengths of the hydrophobic and hydrophilic segments. Their molecular weight ranges from 2000 to 20000 g / mol, with a PEO content of 10 to 80 wt. %. These materials are named using an alphanumeric code that combines a letter and two or three digits. The letter represents the physical state at room temperature: L for liquids, P for pastes, and F for flakes. The digits denote the molecular weight and composition. Specifically, the molecular weight of the hydrophobic PO block is calculated by

The content of this chapter, modified and integrated, is part of a publication under preparation.

multiplying the first one or two digits by 300, while the hydrophilic EO content (in weight percentage) is determined by multiplying the final digit by 10. Pluronics are often referred to as $\text{EO}_x\text{PO}_y\text{EO}_x$, where x and y represent the number of EO and PO units, respectively. The polymerization degree of the Pluronic hydrophilic and hydrophobic segments determines its hydrophilic-lipophilic properties. The relative lengths of the EO and PO chains dictate these properties. This ratio also defines the hydrophilic-lipophilic balance (HLB), which quantifies the Pluronic degree of hydrophilicity or lipophilicity. The HLB of Pluronic molecules can be calculated using the following equation:

$$\text{HLB} = -36.0 \frac{N_{\text{PO}}}{N_{\text{PO}} + N_{\text{EO}}} + 33.2, \quad (8.1)$$

where N_{PO} and N_{EO} refer to the PO and EO block repeating units, respectively.

Depending on factors such as HLB, concentration, temperature, and shear stresses, these materials can exist in aqueous solutions either as individual macromolecules, unimers, or self-assemble into micelles once they surpass the CMC and the CMT. They can evolve in various microstructures, which in turn govern the ultimate material properties they exhibit. The CMC represents the concentration threshold at which self-assembly in micelles occurs, which impacts various functionalities such as drug delivery. Micelles can exhibit a variety of shapes and sizes, typically described by the radius of gyration R_g and the number of aggregation A_s .

Pluronic micelles, widely researched for medical applications, are among the most studied systems for drug delivery. These micelles can encapsulate hydrophobic drugs within the PPO core, allowing them to be transported throughout the body. Recently, Pluronic micelles have also been used in cancer therapy. For these reasons, the micellar phase has been investigated in depth using numerous experimental techniques. For example, the CMC has been studied by surface tension measurements. Micelle characterization is performed by static light scattering or small-angle X-ray and/or neutron scattering. Key properties such as CMC, R_g , and A_s , which significantly influence functionalities like drug delivery, are determined by factors such as concentrations, temperatures, and the types of Pluronics. Although predicting and controlling these properties is vital for optimizing performance, the experimental campaigns required to explore different conditions, such as composition, concentration, and temperature, are both time-intensive and costly.

Computational tools offer valuable support that enables the efficient and cost-effective exploration of various scenarios. Atomistic simulations, however, face

limitations due to the large time and length scales involved, making them unsuitable for studying self-assembly phenomena directly. Coarse-grain methods, which represent groups of atoms or molecules as single entities called beads, bridge the gap between micro- and macro-scale models. These methods operate at scales inaccessible to molecular-level approaches. Among these, DPD is one of the most widely used techniques for such systems. It offers greater computational efficiency and can simulate time and length scales that accurately reproduce the morphologies of diverse systems. However, its primary drawback is the challenge of capturing chemical specificity and linking the model to atomic-level structures.

These challenges are reflected in the parameterization problem: identifying and transferring the parameters of a CG model, such as DPD, is not straightforward but crucial for accurately representing physical properties. The model parameters of a CG model are not easily identifiable, transferable to accurately capture physical properties at the specific thermodynamic state for which it was parametrized. Both issues are problematic whether or not the CG model is developed by reference to experimental data, top-down CGing, or by reference to an underlying atomistic model, bottom-up coarse-graining. Using a top-down parameterization approach, the interaction parameters are usually related to thermodynamic properties or fit to experimental observables. Although using a bottom-up approach, these mesoparticle interactions can also be parameterized by deriving CG conservative interactions from structure matching, force matching, or relative entropy approaches. The top-down approach is often preferred since it does not require large molecular dynamics trajectories to inform the DPD model, reducing the computational effort. Although manual fitting against experimental data might be manageable for simple fluids, it becomes complex and demanding for multi-component systems or those with chain-like molecules, such as Pluronics Water mixtures investigated in this work.

Pluronic systems in DPD are typically mapped onto FH theory as well as other polymer systems. In fact, Groot and Warren introduced an important link between DPD and the FH parameter theory for polymer solutions. In polymer science, established theories connect the χ -parameter to the solubility and the mixing energies of polymeric components. These parameters can be obtained through either atomistic simulations or experimental data. Since FH parameters are experimentally derived, their calculation does not require additional simulations. However, the FH-based mapping procedure has intrinsic limitations, which hinder the accurate quantitative prediction of critical properties, such as the CMC, the micelle radius of gyration

(R_g), and the aggregation number (N_a), especially under varying conditions such as concentration or temperature. In principle, the parameters of a CG model are not easily transferable. In particular, they can depend on composition and temperature. This dependency cannot be known a priori and need to be investigated for every CG model. Therefore, the reason of the disagreement between experimental results and simulations is, in part, in the use of fixed parameters across different concentrations, temperatures, and types of Pluronics. Better agreement could be achieved with a more flexible model, allowing for the proper variation of χ_{AW} , χ_{BW} , and χ_{AB} based on concentration, temperature, and Pluronic type. Preliminary work is required to tune and validate the parameters for each system to closely match the experimental results. In such cases, the adjustment process must account for various *itype* – *jtype* interactions (e.g., 1-1, 1-2, 2-2 for a binary system), requiring numerous trial-and-error iterations to find suitable parameters. Indeed, by employing a top-down approach, these can be acquired through multiple iterations of DPD simulations and a systematic comparison with experimental data. This extensive search through the parameter space is not only labor-intensive and costly but also demands substantial expertise from researchers. Machine learning techniques could potentially streamline this parameter optimization process by reducing the effort and enhancing accuracy. However, adjustments are needed to satisfactorily reproduce the physical properties of the system under study. In a top-down parameterization approach, the parameter tuning task is typically performed through a manual trial-and-error process. This involves using a penalty function that compares the calculated values of selected properties with their corresponding target values from experiments:

$$\mathcal{P}(\boldsymbol{\theta}) = \sqrt{\sum_i w_i \left(1 - \frac{A_i(\boldsymbol{\theta})}{A_i^T}\right)^2} \quad (8.2)$$

Each evaluation of $\mathcal{P}(\boldsymbol{\theta})$ requires running a full DPD simulation, followed by a post-processing step to extract the relevant properties. In addition, for each time variable like temperature or the type of Pluronic change, a new simulation is required, adding complexity to the process. DPD simulations are known to be computationally expensive, making them time-consuming and impractical for high-dimensional optimization tasks. In addition, they do not provide gradient information, which is essential for efficiently solving optimization problems.

At the heart of this workflow lies the development of an accurate data-driven surrogate model to replace traditional physics-based DPD simulations. The key parameters

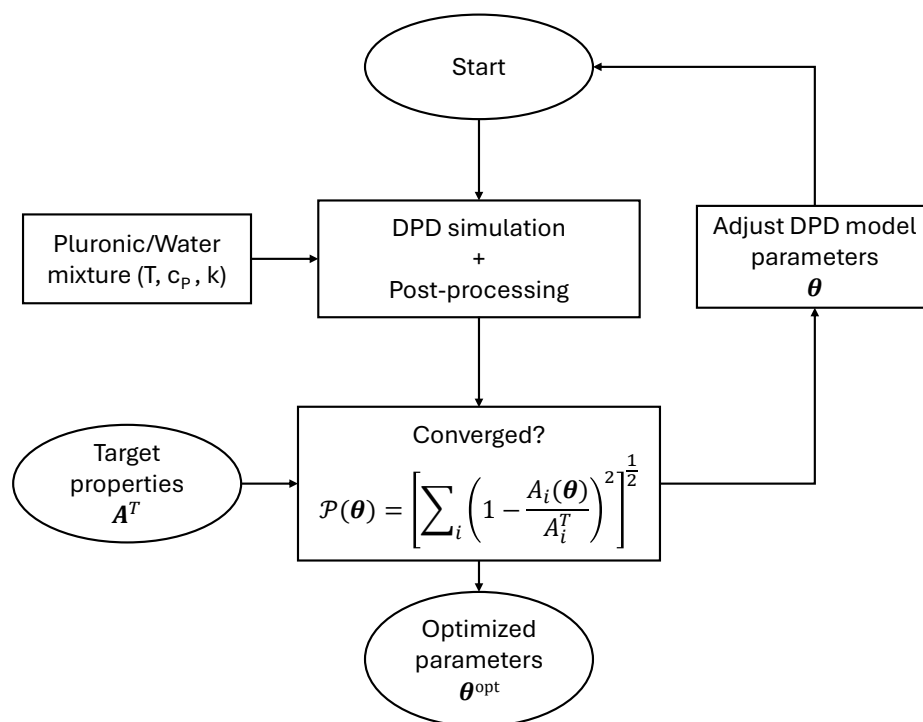


Fig. 8.1 Typical trial-and-error parameters identification workflow.

of the DPD model are identified as input features, and a series of simulations are performed with these features varied within physically meaningful ranges. For each simulation, physical properties are calculated, creating a dataset in which each sample represents input features paired with their corresponding outcomes. The GPR model is trained on a subset of the dataset, and its performance is evaluated on the remaining portion, referred to as the validation dataset. The GPR accurately models the relationships between input features and outputs, and once trained, it can make predictions at a much faster rate. This increased speed of the GPR-based surrogate model is then utilized to perform a grid-search optimization of the parameters.

DPD simulations are conducted using the open-source software LAMMPS. Physical observables are extracted through a post-processing workflow involving C++ code and a Python routine. The GPR model is developed, trained, and tested using the Python library *Scikit learn*, while grid search optimization is performed with a dedicated Python script.

8.1 Data-driven approach: methods and computational details

8.1.1 DPD simulations: setup and post-processing

The first step involves running DPD simulations to generate the dataset required to train the GPR algorithm. The data obtained from these simulations serve as the "ground truth" for the data-driven model and are divided into two subsets: a training set and a validation set. The first key step is the simulation setup, which involves identifying the most relevant DPD input parameters to accurately reproduce the target physical properties: radius of gyration of the micelle (R_g), aggregation number (A_s), and CMC. The second key step is the extraction of these physical properties from simulation trajectories through a post-processing routine. In the following paragraphs these two parts are described. These two parts of the workflow are elaborated upon the subsequent paragraphs.

Setup

It is essential to evaluate the contribution of all potentially relevant parameters to the modeling of these target properties. It is worth highlighting that the selected prop-

erties are equilibrium properties of the system. Consequently, they are determined solely by conservative interactions. For this reason, in this study the standard DPD model, described in Chapter 2, is employed.

In the DPD model, a Pluronic macromolecule is represented as a chain composed of A and B bead types. An EO *bead*, denoted as A, represents 4.3 repeating units of ethylene oxide, while a PO *bead*, labeled as B, corresponds to 3.3 repeating units of propylene oxide. For instance, the Pluronic L64 macromolecule is simplified into a chain of 15 *beads*. This chain is schematically represented as $A_3B_9A_3$, where A and B correspond to the EO and PO *beads*, respectively. Table 8.1 summarizes the DPD models for all the Pluronic systems analyzed in this work. The water molecules are represented by a third type of beads (W), which corresponds to ten water molecules. The re-normalization factors are chosen such that all beads have the same mass and volume. All the conservative model parameters involved are reported in the

Table 8.1 Coarse-Grained Representation of the analyzed Pluronic systems

Pluronic Name	Composition	DPD model composition
L64	$EO_{13}PO_{30}EO_{13}$	$A_3B_9A_3$
P65	$EO_{20}PO_{30}EO_{20}$	$A_5B_9A_5$
P104	$EO_{27}PO_{63}EO_{27}$	$A_4B_{18}A_4$
F68	$EO_{82}PO_{31}EO_{82}$	$A_{18}B_9A_{18}$

Tab. 8.2. The repulsive parameters are denoted as a_{ij} and the cutoff distances are represented as $r_{C,ij}$. The chemical bonds between the monomers forming the polymer molecule are represented using the bead-and-spring model, in which adjacent beads in the polymer chain interact via a harmonic potential as explained in previous chapters. Thus, for consecutive beads in the chain, two additional parameters are involved: the nominal equilibrium distance $R_{0,ij}$ and the harmonic constant $K_{b,ij}$. Accounting for all parameters creates a high-dimensional parameter space, making the problem computationally demanding. To address this, we initially restrict the number of parameters by exploiting physical insights. Future work may explore additional parameters to refine the model. First, since ij -interactions are identical to ji -interactions, redundant parameters below (or above) the diagonal in Table 8.2 are excluded. Second, since all beads represent the same volume, all cutoff distances are considered equal and set to 1. Third, the repulsive parameter a_{WW} is set to match the isothermal compressibility of water, as is commonly done in DPD simulations, and

Table 8.2 Conservative interaction parameters for the three species: PEO, PPO, and water.

	A (PEO)	B (PPO)	W (H ₂ O)
A (PEO)	$a_{AA}; r_{C,AA}; K_{b,AA}; R_{0,AA}$	$a_{AB}; r_{C,AB}; K_{b,AB}; R_{0,AB}$	$a_{AW}; r_{C,AW}$
B (PPO)	$a_{BA}; r_{C,BA}; K_{b,BA}; R_{0,BA}$	$a_{BB}; r_{C,BB}; K_{b,BB}; R_{0,BB}$	$a_{BW}; r_{C,BW}$
W (H ₂ O)	$a_{WA}; r_{C,WA}$	$a_{WB}; r_{C,WB}$	$a_{WW}; r_{C,WW}$

all repulsive interactions similar to each other are assumed equal to a_{WW} . As a result, the free parameters are: a_{AW} , a_{BW} , a_{AB} , $R_{0,AA}$, $R_{0,BB}$, $R_{0,AB}$, $K_{b,AA}$, $K_{b,BB}$ and $K_{b,AB}$. At the coarse-graining level, all bonded interaction parameters are assumed equal (R_0 and K_b for all bonds). These parameters have a marginal impact on CMC and A_s , but they can influence the morphology and compactness of the micelles, and then the R_g . Preliminary values of $K_b = 50$ and $R_0 = 0.7$ are set using the bond-bond distribution function. Additionally, a_{AB} is assumed equal to a_{BW} , reducing the free parameters to a_{AW} and a_{BW} . These simplifications are reasonable, as the self-assembly process is primarily driven by solubility differences between the hydrophobic and hydrophilic blocks and water.

Thus, a_{BW} and a_{AB} are the key parameters of the DPD model to capture the chosen physical targets.

Post-processing routine

The post-processing is handled through an automated routine, which orchestrates the calculations necessary to derive from the equilibrium trajectories, produced by the DPD simulations, the system observables: the CMC, R_g , and A_s . The post-processing of saved trajectories starts with the identification of Pluronic aggregates (micelles): two molecules are grouped into the same aggregate if they share a minimum number of contact pairs. A contact pair is defined as a pair of beads from different molecules that are within a specific distance of each other.

This step is followed by the calculation of the aggregation number, A_s . The aggregation or association number, denoted as A_s , represents the number of chains within a single aggregate. The weight-average aggregation number is computed for

comparison with experimental data and is given by

$$\langle A_s \rangle_w = \frac{\sum_i m_{A,i}^2}{\sum_i m_{A,i}}, \quad (8.3)$$

where $m_{A,i}$ is the weight of the aggregate i . For simplicity, in this work, $A_s \equiv \langle A_s \rangle_w$. A more detailed explanation of this calculation is provided in Chapter 2.

The CMC is computed. The CMC is the concentration above which Pluronics macromolecules self-assemble into micelles. It is computed as the concentration of sub-micellar aggregates, using a cut-off aggregation number [107] to distinguish them from micellar aggregates. The sub-micellar portions of the number distribution function of association numbers is used to determine the CMC. Finally, R_g is calculated from the gyration tensor, as described in chapter 2. Analogously to the aggregation number, the weight-averaged radius of gyration is computed for comparison against experimental data. For simplicity, in this work, $R_g \equiv \langle R_g \rangle_w$.

8.1.2 Dataset creation

Four different Pluronics, as reported in Tab. 8.3, are analyzed in this work.

For each of them, a dataset consisting of 40 data points is created. The dataset is

Table 8.3 Physiochemical properties of the considered Pluronics [17]

Pluronic Name	Composition	2x (EO)	y (PO)	Avg. Mol. Wt. (g/mol)	PO Mol. Wt. (g/mol)	EO Content (wt. %)	HLB Value
L64	EO ₁₃ PO ₃₀ EO ₁₃	26	30	2900	1750	40	15
P65	EO ₂₀ PO ₃₀ EO ₂₀	40	30	3400	1750	50	17
P104	EO ₂₇ PO ₆₃ EO ₂₇	54	63	5900	3250	40	13
F68	EO ₈₂ PO ₃₁ EO ₈₂	164	31	8350	1750	80	28

constructed using data from DPD simulations, which serve as the "ground truth" for training the surrogate model. Key input features, a_{BW} and a_{AB} , are selected based on their relevance to the system behavior, as discussed in Sec. 8.1.1. A series of simulations is then performed, varying these parameters within physically meaningful ranges. a_{BW} and a_{AB} are varied across different simulations, while all other parameters are kept constant across all simulations, accordingly to Sec. 8.1.1.

For each simulation, physical properties are calculated, resulting in a dataset where each sample consists of input features paired with their corresponding outcomes.

8.1.3 GPR as surrogate model

Once the dataset for each Pluronic system (e.g., L64 at 30°C) is generated from the simulations, it is used to construct separate surrogate models for each output property (e.g., R_g , A_s , CMC). Each model captures the relationship between the respective output property and the input parameters (a_{AW} , a_{BW}). The same process is applied to all four Pluronic systems (L64, P65, P104, F68). To establish these input-output relationships, we employ Gaussian process regression (GPR), a Bayesian machine learning technique that leverages non-parametric modeling through Gaussian processes. It excels in handling small datasets, provides uncertainty estimates for predictions, and offers flexibility through its kernel-based approach, making it well-suited for our regression tasks. The estimated uncertainty, quantified by 95% confidence intervals, can inform the required training data size to achieve a desired level of confidence. For each Pluronic system, we construct three distinct Gaussian process models of the form $y = GP(\mathbf{x})$, where the input vector $\mathbf{x} = [a_{AW}, a_{BW}]$ represents the water-interaction parameters. Each model predicts one of our target properties: R_g , A_s , or CMC. Using R_g as our an example, we will demonstrate the modeling process that is subsequently applied to A_s and CMC. Consider a training set with N_{train} samples. We can represent the input and output as:

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_{N_{\text{train}}} \end{bmatrix} \in \mathbb{R}^{N_{\text{train}} \times 2} \quad \text{and} \quad \mathbf{Y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_{N_{\text{train}}} \end{bmatrix} \in \mathbb{R}^{N_{\text{train}}}, \quad (8.4)$$

respectively. Here, $\mathbf{x}_i = [a_{AW,i}, a_{BW,i}]$ represents the input parameters for the i -th sample from simulation, and y_i is its corresponding output ($R_{g,i}$ in this example).

The Gaussian process assumes that the output values $\mathbf{Y}(\mathbf{X})$ at any given set of input points $\mathbf{X}$ are drawn from a multivariate normal distribution. The Gaussian process is fully characterized by a mean function $\mathbf{U}(\mathbf{X})$ and a covariance function,

which describes the relationships between input features:

$$\mathbf{Y}(\mathbf{X}) \sim \text{GP}(\mathbf{U}(\mathbf{X}), \mathbf{\Sigma}(\mathbf{X}, \mathbf{X})) , \quad (8.5)$$

where $\mathbf{U}(\mathbf{X}) \in \mathbb{R}^{N_{\text{train}}}$ is the mean function, and $\mathbf{\Sigma}(\mathbf{X}, \mathbf{X}) \in \mathbb{R}^{N_{\text{train}} \times N_{\text{train}}}$ is the covariance matrix. The covariance function is defined using the Radial Basis Function (RBF) kernel, which measures the similarity between two input points $\mathbf{x}_i$ and $\mathbf{x}_j$. The RBF kernel is given by:

$$\Sigma(\mathbf{x}_i, \mathbf{x}_j) = \theta_a^2 \exp\left(-\frac{1}{2} \frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{\theta_l^2}\right) , \quad (8.6)$$

where the θ_a is the amplitude parameter, and θ_l is the length scale, which control the smoothness of the mean function.

The GPR model training involves optimizing the kernel hyperparameters by maximizing the log-likelihood function:

$$\log p(\mathbf{Y}|\mathbf{X}; \mathbf{\Theta}) = -\frac{1}{2} \left[\mathbf{Y}^\top \tilde{\mathbf{\Sigma}}^{-1} \mathbf{Y} + \log |\tilde{\mathbf{\Sigma}}| + N_{\text{train}} \log(2\pi) \right] , \quad (8.7)$$

where $\tilde{\mathbf{\Sigma}} = \mathbf{\Sigma}(\mathbf{X}, \mathbf{X}) + \sigma^2 \mathbf{I}$ and $|\tilde{\mathbf{\Sigma}}|$ is the determinant of matrix $\tilde{\mathbf{\Sigma}}$. Here, $\mathbf{\Theta} = [\sigma, \theta_a, \theta_l]$, which include the noise level σ and kernel parameters θ_a and θ_l , are optimized using the L-BFGS Quasi-Newton optimization algorithm [108].

Once the GPR model is trained, for any new input $\mathbf{x}'$, the posterior distribution of the output can be obtained as:

$$y(\mathbf{x}') | \mathbf{Y}(\mathbf{X}) \sim \text{GP}(\hat{\mathbf{u}}, \hat{\mathbf{\Sigma}}) , \quad (8.8)$$

where

$$\begin{aligned} \hat{\mathbf{u}} &= \mathbf{\Sigma}(\mathbf{x}', \mathbf{X}) [\mathbf{\Sigma}(\mathbf{X}, \mathbf{X}) + \sigma^2 \mathbf{I}]^{-1} [\mathbf{Y} - \mathbf{U}(\mathbf{X})] + \mathbf{u}(\mathbf{x}'), \\ \hat{\mathbf{\Sigma}} &= \mathbf{\Sigma}(\mathbf{x}', \mathbf{x}') - \mathbf{\Sigma}(\mathbf{x}', \mathbf{X}) [\mathbf{\Sigma}(\mathbf{X}, \mathbf{X}) + \sigma^2 \mathbf{I}]^{-1} \mathbf{\Sigma}(\mathbf{x}', \mathbf{X})^\top . \end{aligned} \quad (8.9)$$

Here, $\hat{\mathbf{u}}$ denotes the GPR model's predicted mean value of the output, while the covariance matrix $\hat{\mathbf{\Sigma}}$ indicates the prediction confidence and reveals the associated uncertainty of this prediction. The 95% confidence interval is determined from the mean prediction and the standard deviation as: $[\text{mean_prediction} - 1.96 \times \text{std_prediction}, \text{mean_prediction} + 1.96 \times \text{std_prediction}]$, where the standard deviation is calculated as the square root of each diagonal element of the covariance matrix.

8.1.4 Grid-search for parameters optimisation

The GPR-based surrogate model is employed to perform an adaptive grid-search optimization of the parameters. This optimization process addresses a multi-target objective, defined as:

$$\begin{aligned}\boldsymbol{\theta}^{\text{opt}} &= \arg \min_{\boldsymbol{\theta}} \mathcal{P}(\boldsymbol{\theta}) \\ \mathcal{P}(\boldsymbol{\theta}) &= \sqrt{\sum_i \left(1 - \frac{A_i(\boldsymbol{\theta})}{A_i^T}\right)^2}\end{aligned}\quad (8.10)$$

where $\boldsymbol{\theta} = [a_{AW}, a_{BW}]$ represents the input parameters, and $\mathbf{A} = [\text{CMC}, R_g, A_s]$ denotes the output properties of interest with $\mathbf{A}^T$ representing their target experimental values [109–111]. The target experimental values for the output properties are taken

Table 8.4 Experimental values for the Critical Micellar Concentration (CMC), radii of spherical aggregates (R_g), and aggregation numbers (A_s) of the considered Pluronics at 30°C and low concentrations.

Pluronic Name	CMC [wt. %]	R_g [nm]	A_s
L64	2.5	2.28	10
P65	3.2	2.27	3.8
P104	0.002	5.36	81
F68	10	2.30	3.5

from the literature and summarized in Tab. 8.4. The grid-search systematically evaluates combinations of parameter values across a predefined search space to identify the configuration that minimizes the loss function $\mathcal{P}(\boldsymbol{\theta})$. Each evaluation involves querying the GPR surrogate models to estimate the output properties for a given set of input parameters. The optimization employs an adaptive grid-search strategy, which iteratively refines the search space based on the results of previous evaluations. Unlike a traditional grid-search that evaluates all combinations of parameter values in a predefined grid, the adaptive approach dynamically adjusts the resolution and boundaries of the grid to focus on regions of interest where the loss function $\mathcal{P}(\boldsymbol{\theta})$ is minimized. Initially, a coarse grid is defined over the parameter space, and the surrogate models are used to evaluate $\mathcal{P}(\boldsymbol{\theta})$ at each grid point. Based on these results, regions with low penalty values are identified, and the grid resolution in these regions is refined iteratively. Each evaluation of $\mathcal{P}(\boldsymbol{\theta})$ involves querying the GPR surrogate models to estimate the output properties for a given set of input parameters.

The process continues until the optimal parameter configuration $\theta^{\text{opt}} = [a_{AW}^{\text{opt}}, a_{BW}^{\text{opt}}]$ that minimizes the loss function is identified.

8.2 Results and discussion

8.2.1 Performance of the GPR

In our analysis, we employed GPR as an effective surrogate model for DPD simulations. GPR demonstrated its capability to provide accurate predictions, successfully capturing the relationships between DPD model input parameters (a_{AW} , a_{BW}) and the resulting properties of the system (CMC, R_g and A_s). Three different GPR surrogate models are employed, each tailored to predict one of the specific properties of the system based on the two input parameters.

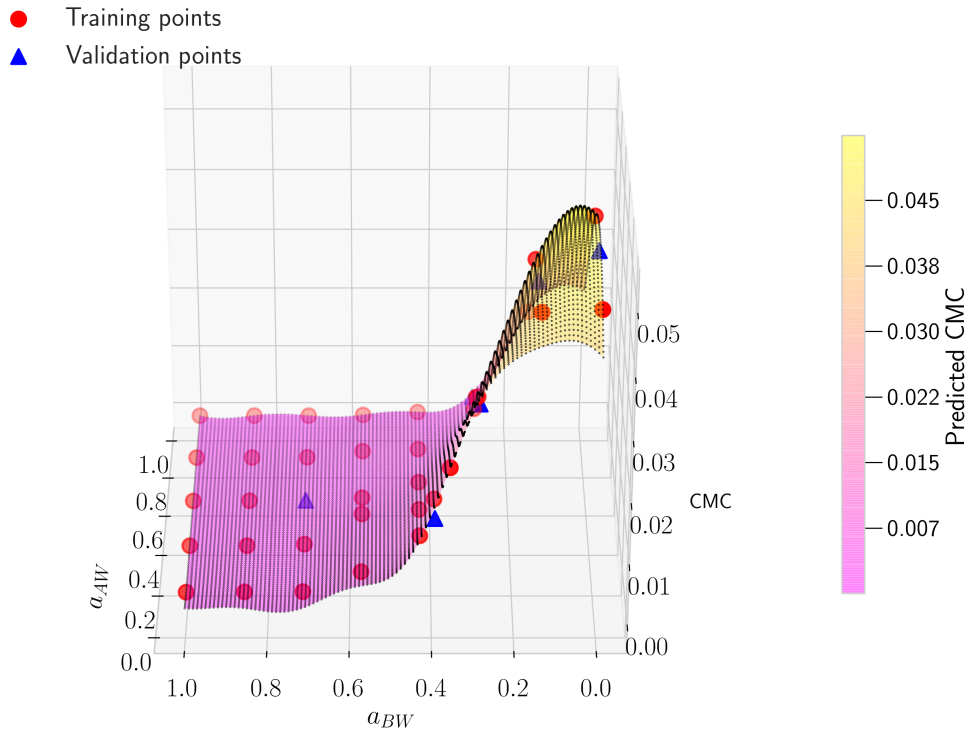


Fig. 8.2 GPR surrogate model predictions for CMC based on DPD model input parameters a_{AW} , a_{BW} .

Fig. 8.2 demonstrates the performance of the GPR model for predicting the CMC across a two-dimensional input space defined by the variables a_{AW} and a_{BW} . The predicted CMC values are displayed using a color gradient, with yellow shades indicating higher values and pink shades indicating lower values. The red circles represent the training data points used to fit the GPR model, while the continuous surface displays the model predictions over the entire input space. Additionally, blue triangles mark the validation data points, allowing for an assessment of the model's predictive accuracy.

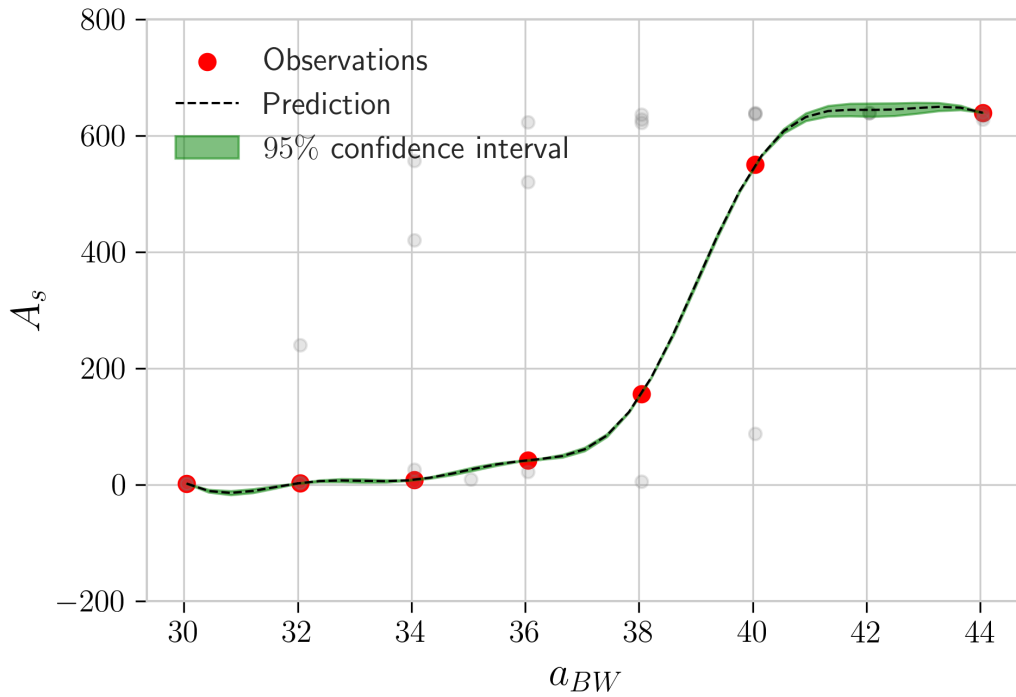


Fig. 8.3 GPR surrogate model predictions of CMC based on a_{BW} at fixed $a_{AW} = 27.346$.

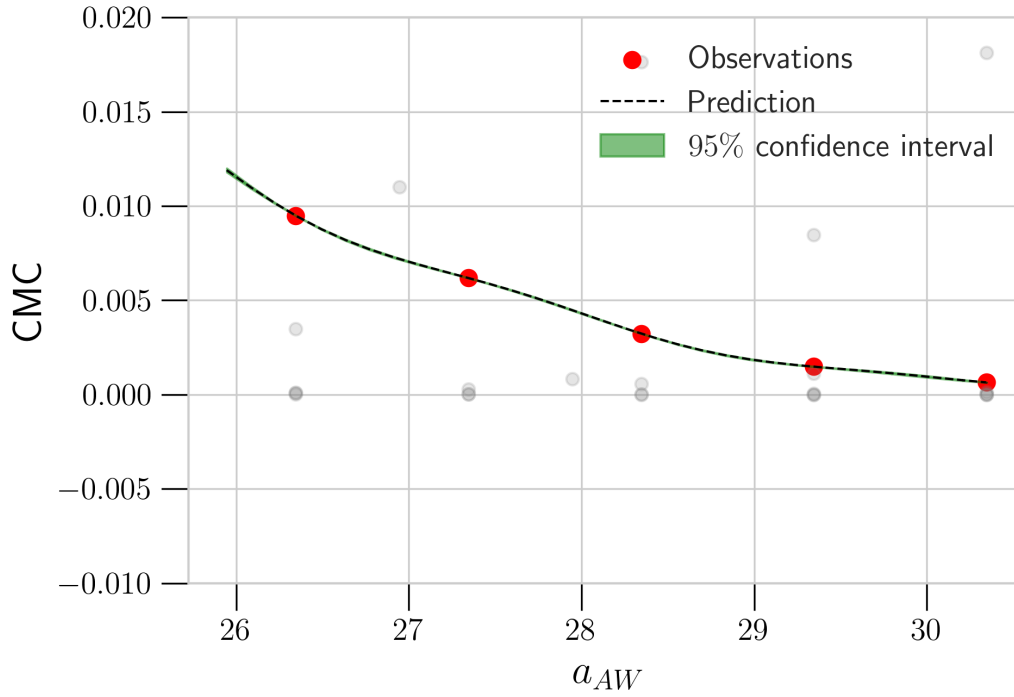


Fig. 8.4 GPR surrogate model predictions of CMC based on a_{AW} at fixed $a_{BW} = 36.046$.

Figs. 8.3 and 8.4 are presented to provide a detailed view of specific sections of the two-dimensional input space. These plots show slices along the a_{AW} and a_{BW} directions, respectively, to visualize the confidence intervals of the GPR model. The plots display the predicted CMC values with green regions representing the confidence intervals, which indicate a very good agreement with the training data and highlight the reliability of the GPR model predictions. By examining these one-dimensional sections, the model accuracy and robustness are further emphasized, demonstrating its capability to effectively capture variations and uncertainties within the input domain.

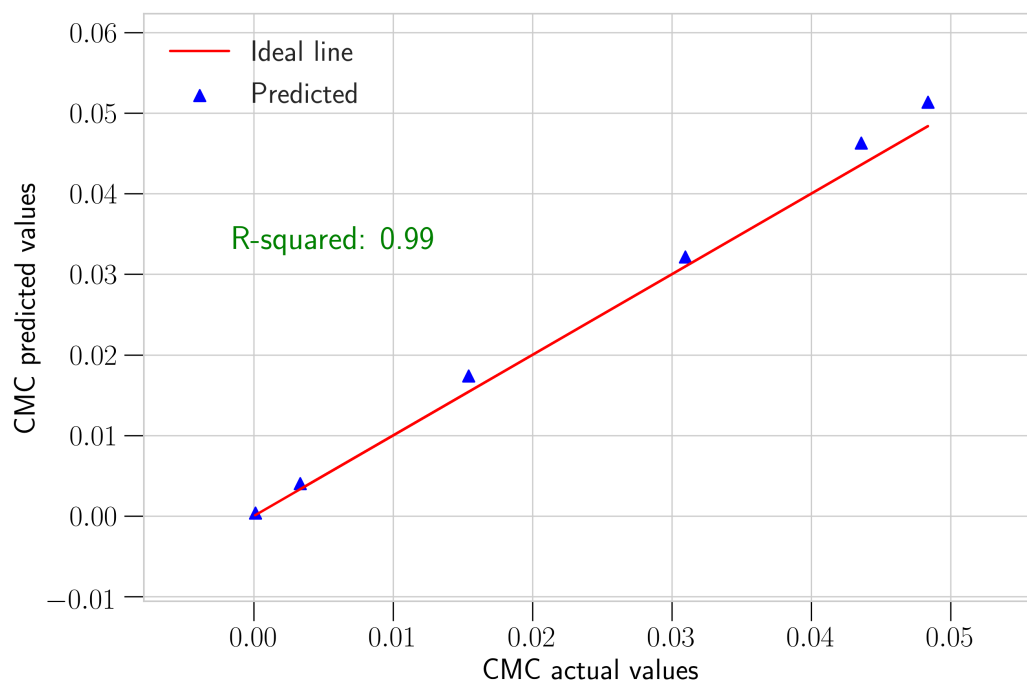


Fig. 8.5 Parity diagram of the CMC on the test set for the GPR.

Fig. 8.5 provides a visual comparison between the predicted CMC values from the GPR surrogate model and the reference values obtained from the DPD simulations. In this plot, the predicted values are plotted against the corresponding reference values. The strong correlation between the predicted and actual values, affirms the reliability of the model.

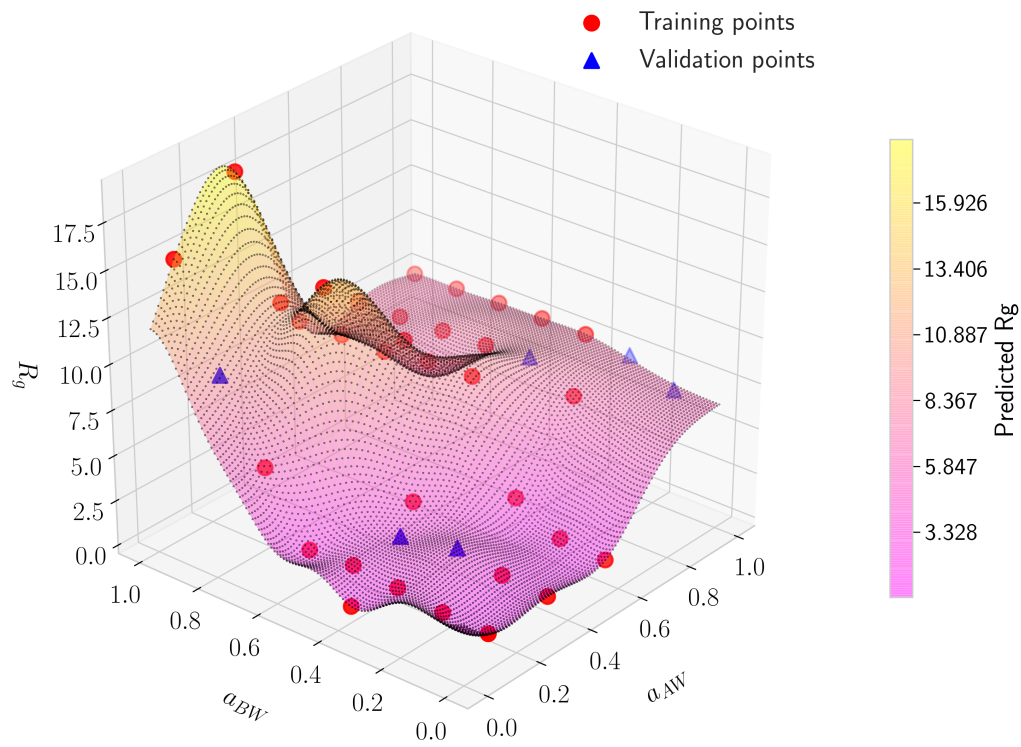


Fig. 8.6 GPR surrogate model predictions for R_g based on DPD model input parameters a_{AW} , a_{BW} .

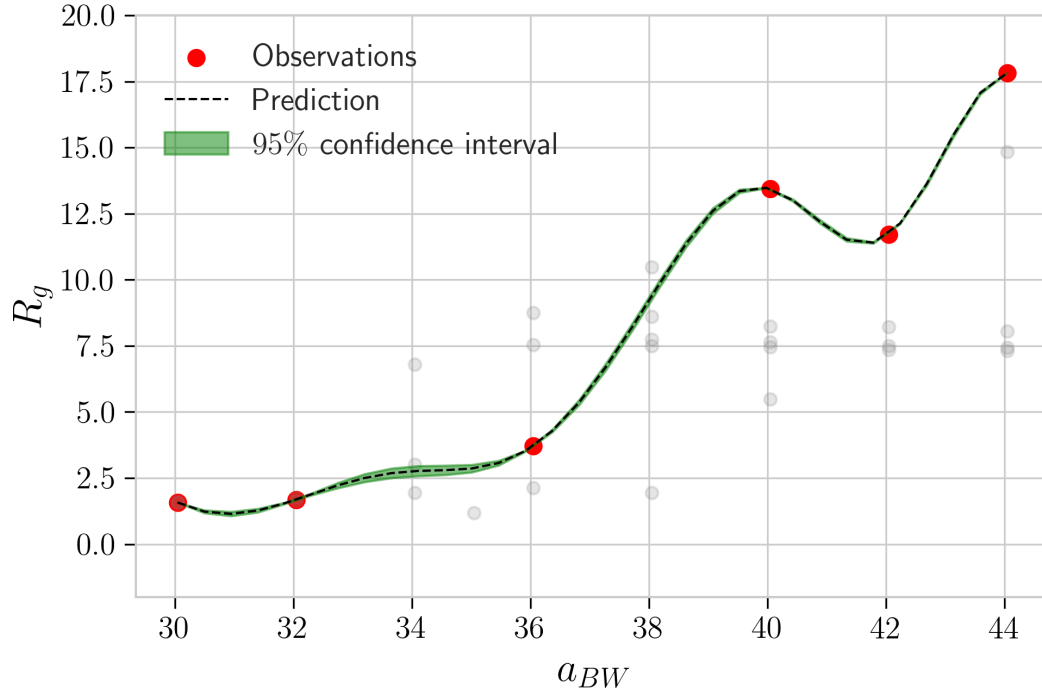


Fig. 8.7 GPR surrogate model predictions of R_g based on a_{BW} at fixed $a_{AW} = 27.346$.

Fig. 8.6 illustrates the effectiveness of the GPR model in predicting the radius of gyration (R_g) within a two-dimensional input space characterized by the variables a_{AW} and a_{BW} . To enhance our understanding of specific segments of this input space, Fig. 8.7 and Fig. 8.8, display one-dimensional sections along the a_{AW} and a_{BW} axes, respectively. These plots reveal the confidence intervals associated with the GPR model, highlighting its accuracy and dependability in estimating R_g values.

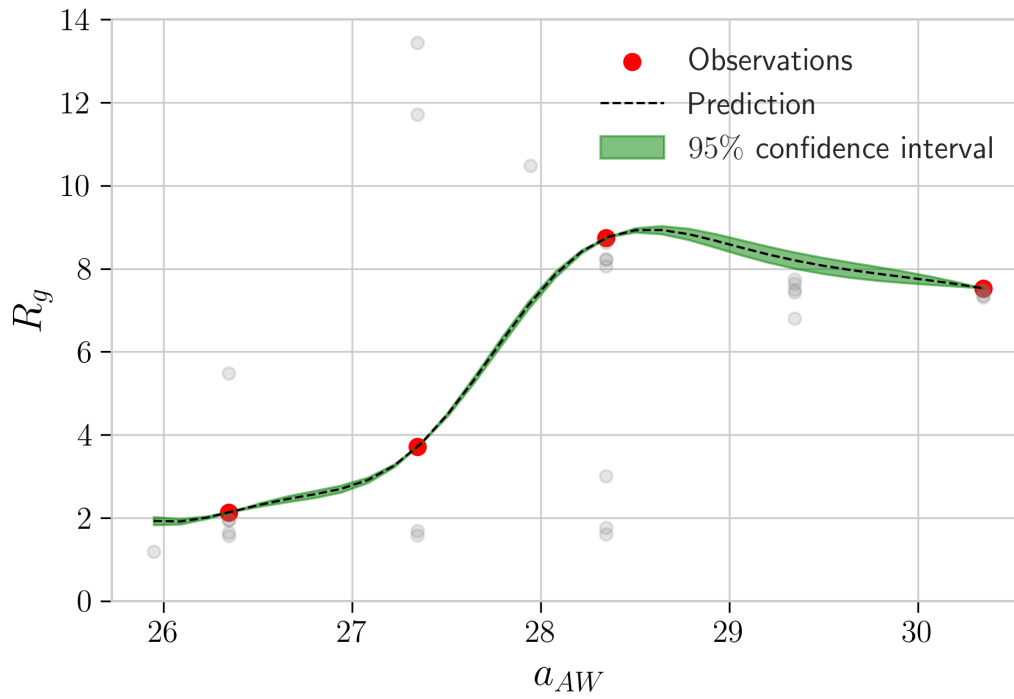


Fig. 8.8 GPR surrogate model predictions of R_g based on a_{AW} at fixed $a_{BW} = 36.046$.

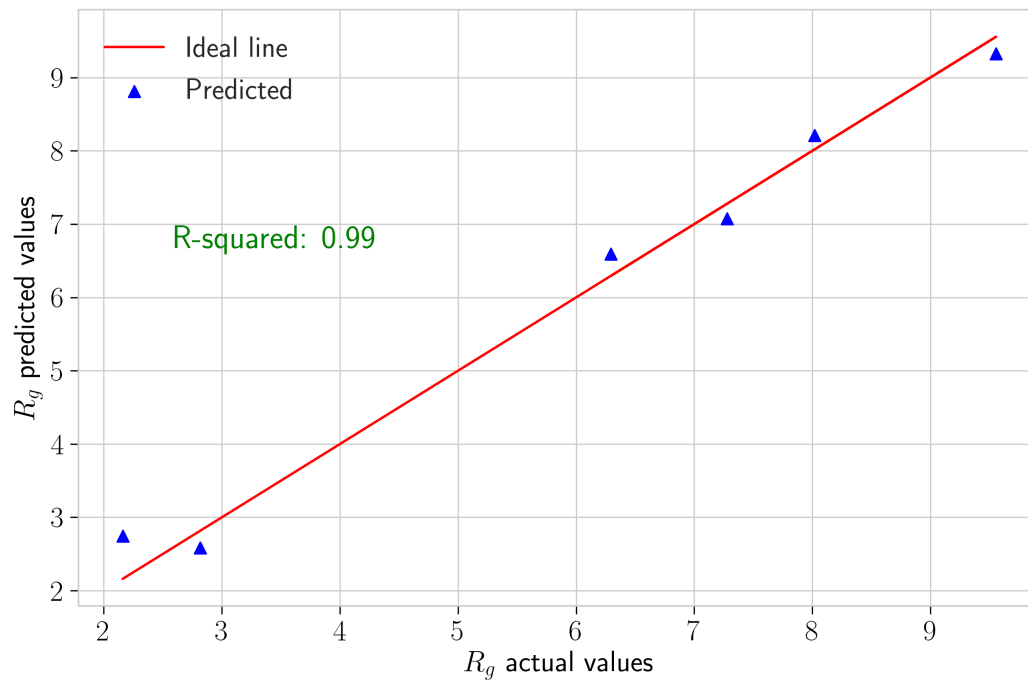


Fig. 8.9 Parity diagram of the R_g on the test set for the GPR.

Fig. 8.9 shows the parity diagram that compares R_g actual and predicted value on a test set unseen from the GPR. The predicted values are plotted against the corresponding reference values, with a 45-degree line indicating perfect agreement. Points falling along the line effectively illustrates the predictive accuracy of the GPR model in estimating R_g values based on the input parameters.

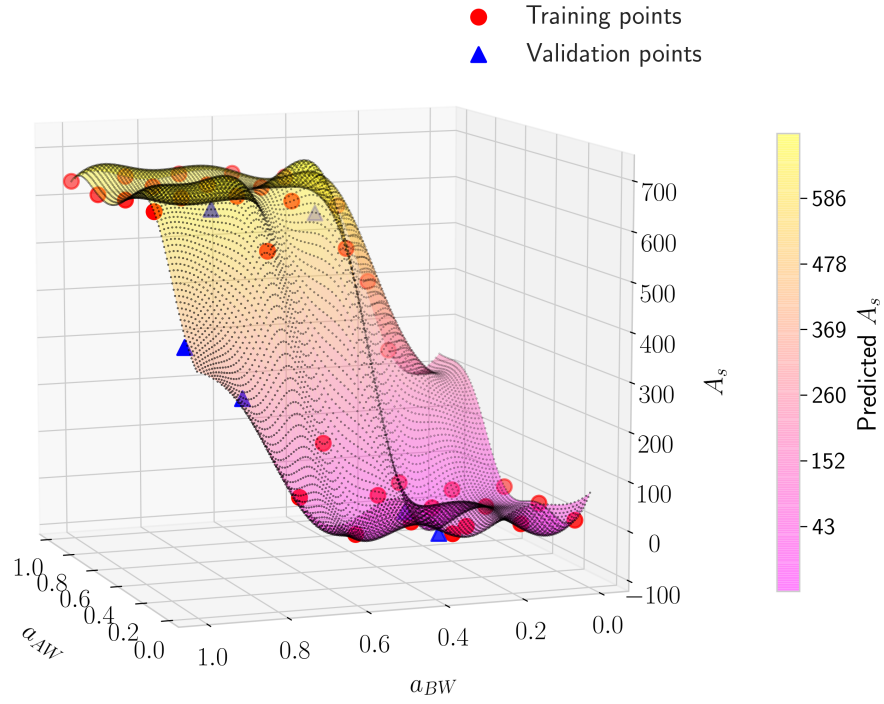


Fig. 8.10 GPR surrogate model predictions for A_s based on DPD model input parameters a_{AW} , a_{BW} .

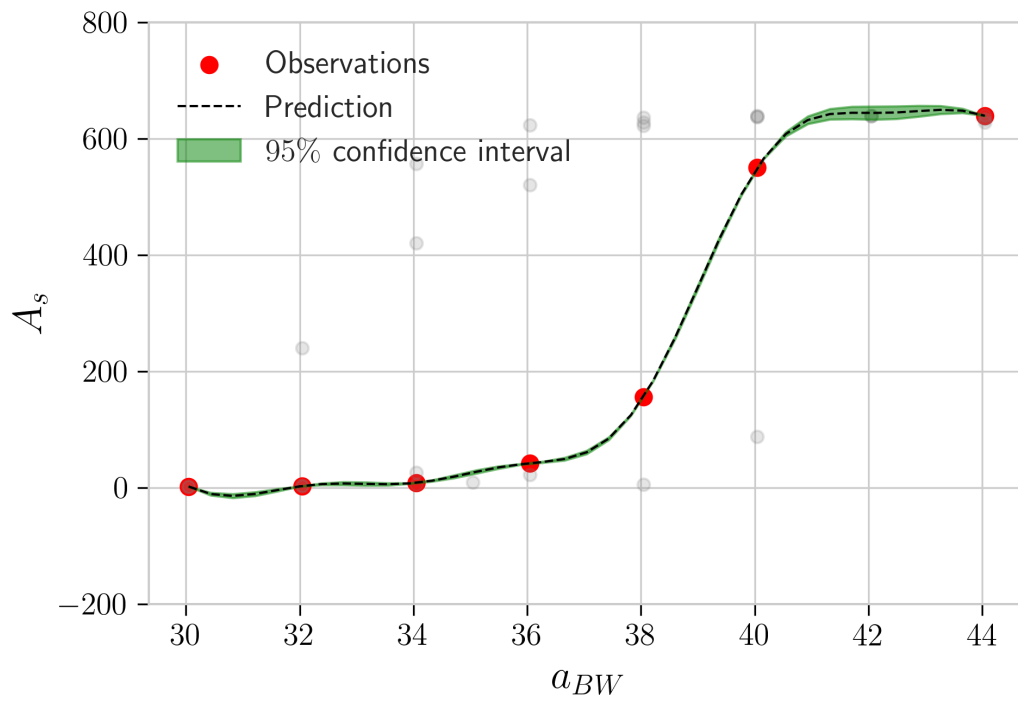


Fig. 8.11 GPR surrogate model predictions of A_s based on a_{BW} at fixed $a_{AW} = 27.346$.

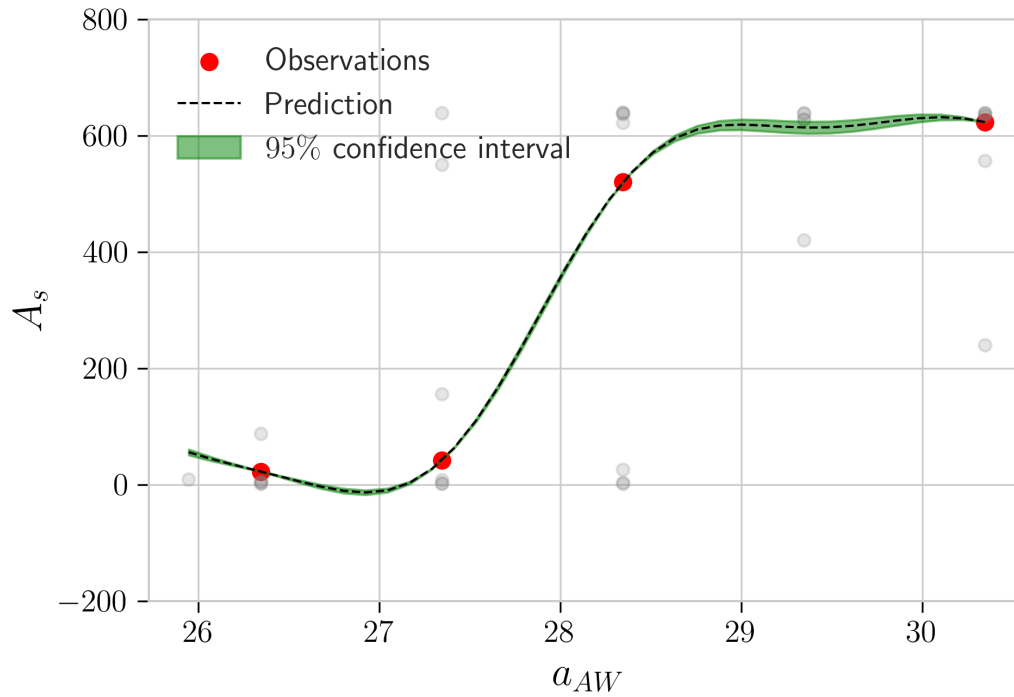


Fig. 8.12 GPR surrogate model predictions of A_s based on a_{AW} at fixed $a_{BW} = 36.046$.

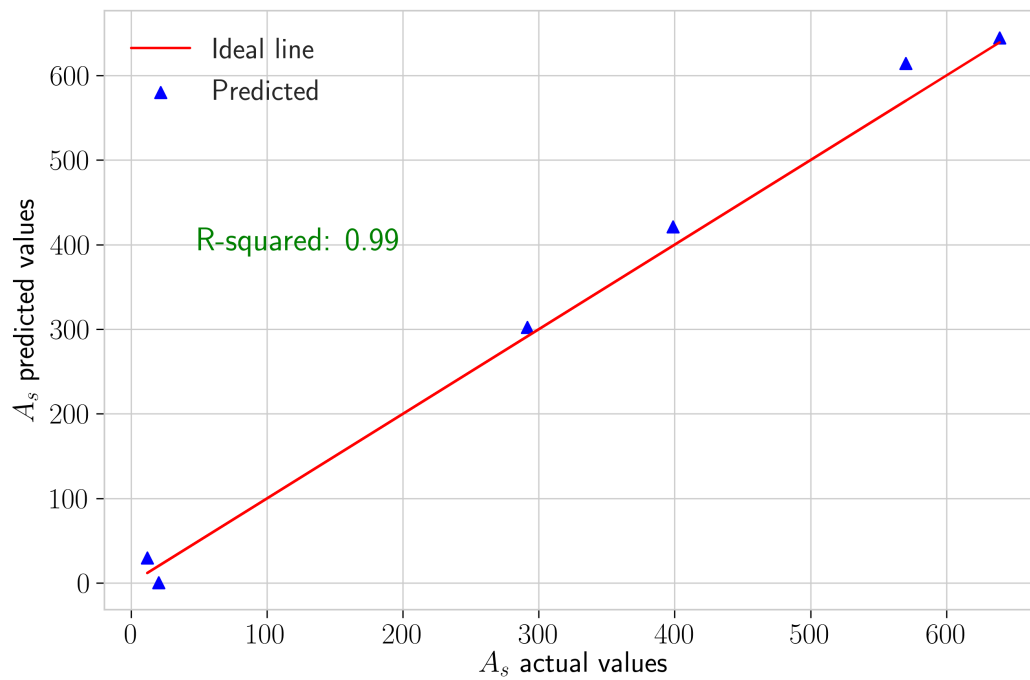


Fig. 8.13 Parity diagram of the A_s on the test set for the GPR.

Figures 8.10, 8.11, 8.12, and 8.13 present similar analyses for the GPR surrogate model used to predict A_s based on the input parameters of the DPD model a_{AW} and a_{BW} . The model demonstrates high accuracy and reliability in estimating the values of A_s .

8.2.2 Optimization results

By leveraging the availability of GPR surrogate models for CMC, R_g and A_s based on the input parameters of the DPD model, the optimization step is performed using an adaptive grid search method. The average GPR execution time is 0.001s. The total execution time to identify the minimum of the penalty function is contingent on the grid resolution. In this analysis, we used an adaptive resolution approach that progressively increases within the area of interest to improve efficiency, as shown in Figs. 8.14 and 8.15. The optimal parameter values identified for all Pluronic systems are summarized in Tab. 8.5.

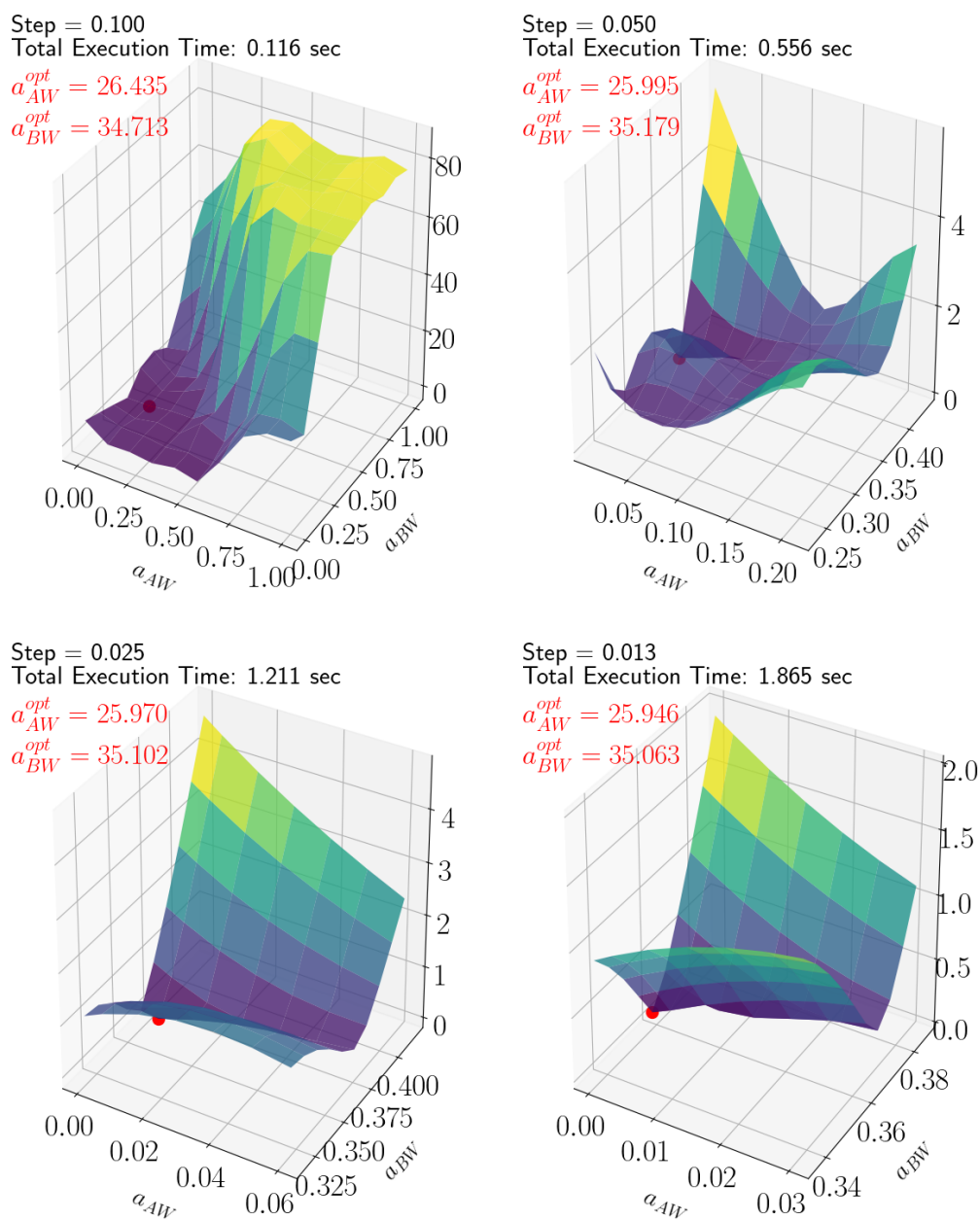


Fig. 8.14 Adaptive grid search for optimizing L64 parameters, showing results at different step sizes.

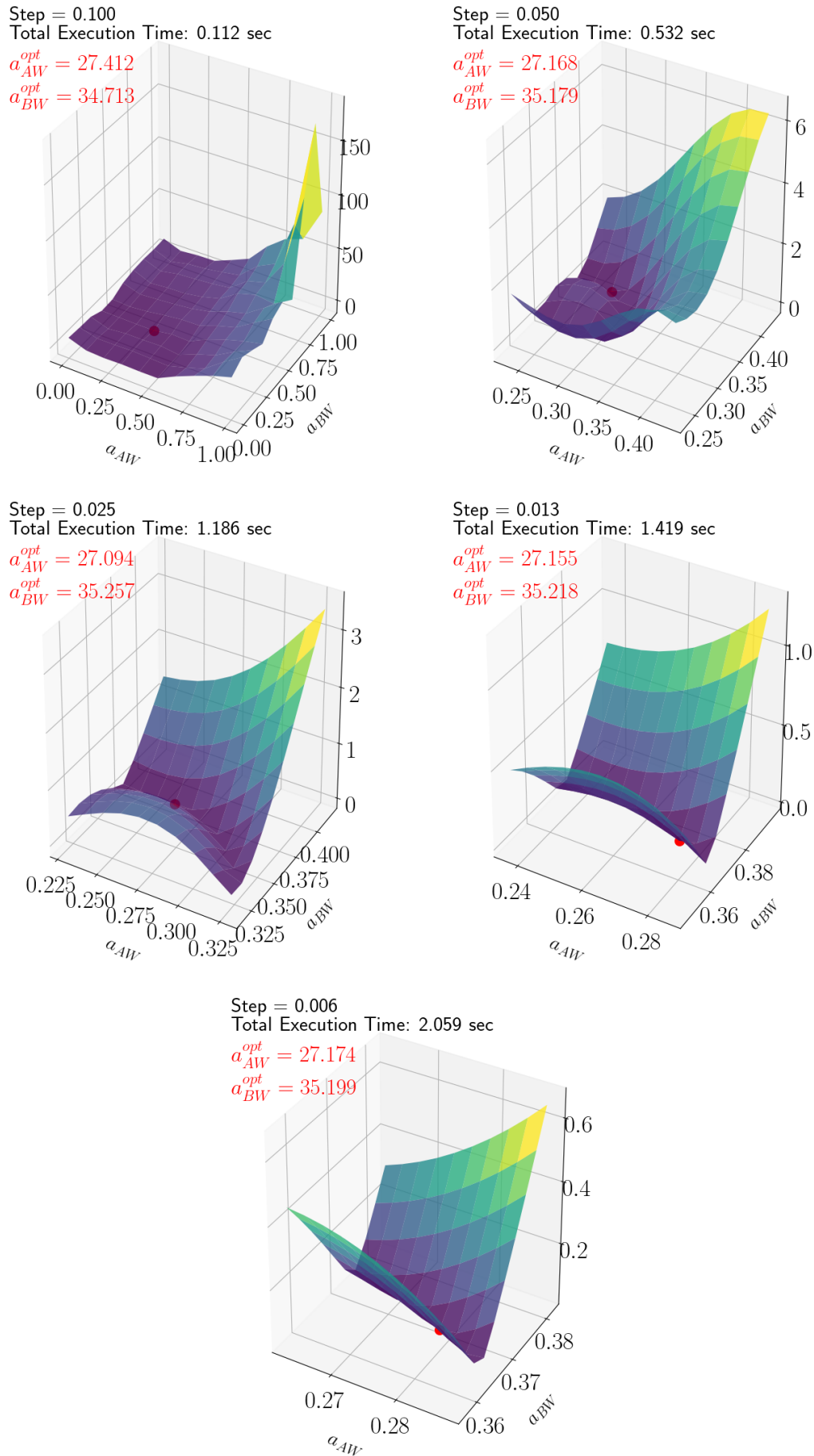


Fig. 8.15 Adaptive grid search for optimizing F68 parameters, showing results at different step sizes.

Table 8.5 Optimized DPD model parameters for the considered Pluronics.

Pluronic Name	Composition	DPD model composition	a_{AW}	a_{BW}
L64	EO ₁₃ PO ₃₀ EO ₁₃	A ₃ B ₉ A ₃	25.946	35.063
P65	EO ₂₀ PO ₃₀ EO ₂₀	A ₅ B ₉ A ₅	26.341	35.120
P104	EO ₂₇ PO ₆₃ EO ₂₇	A ₄ B ₁₈ A ₄	26.472	40.894
F68	EO ₈₂ PO ₃₁ EO ₈₂	A ₁₈ B ₉ A ₁₈	27.152	35.228

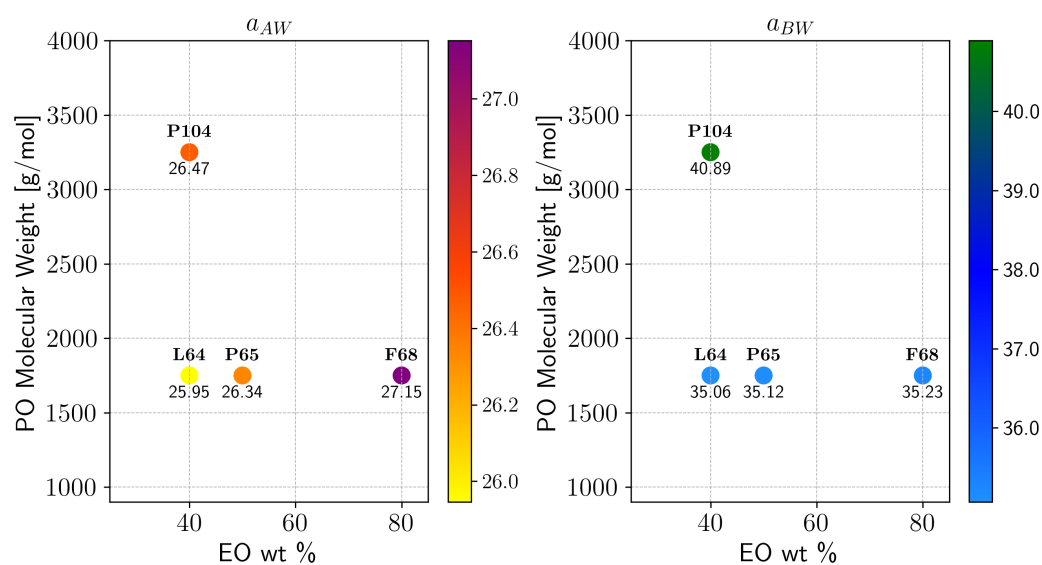


Fig. 8.16 Optimized DPD model parameters for the considered Pluronics systems, defined by PO molecular weight and EO content.

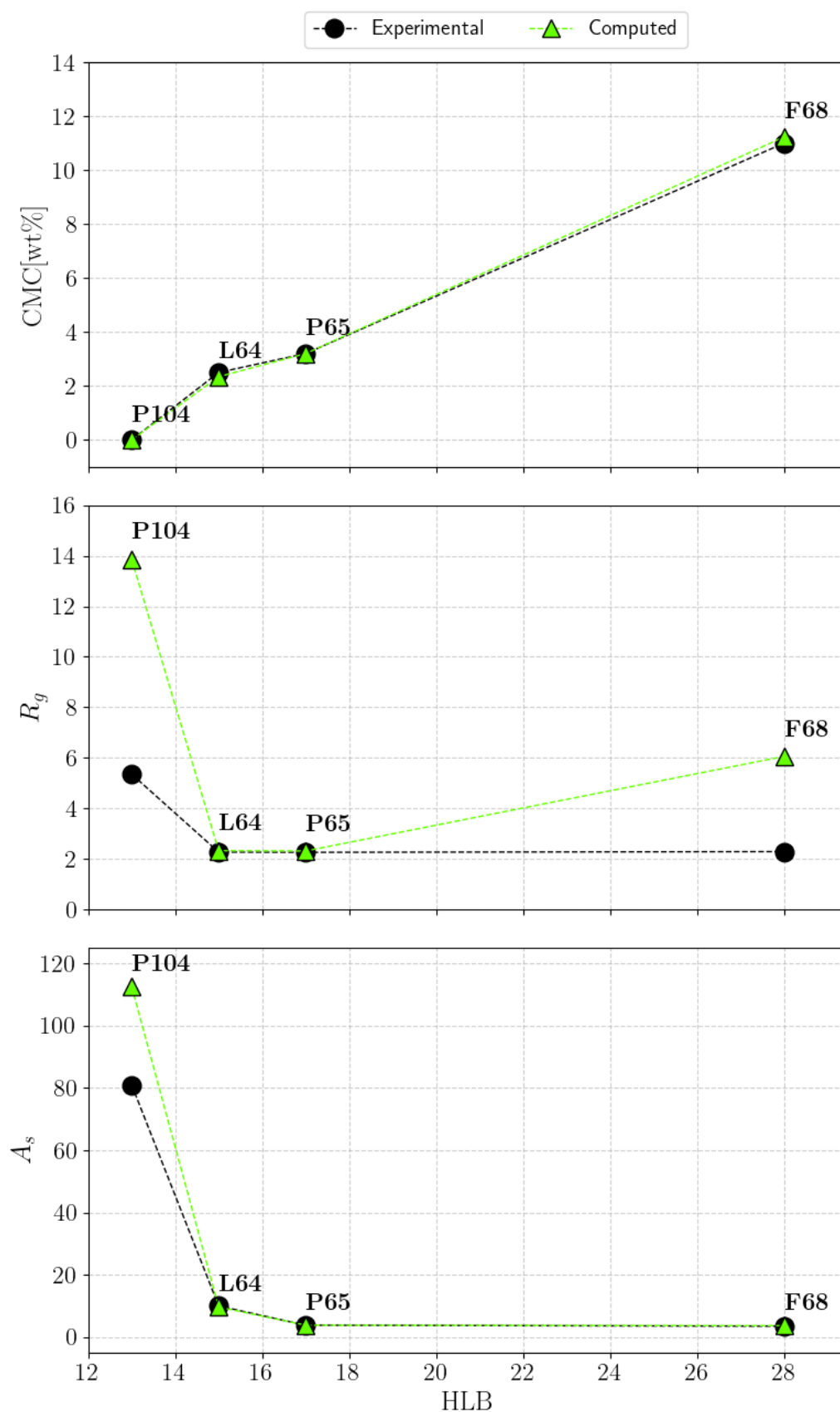


Fig. 8.17 Comparison of experimental targets and computed results obtained through optimized parameters, for the Critical Micellar Concentration (CMC), radii of spherical aggregates (R_g), and aggregation numbers (A_s) of the considered Pluronics at 30°C and low concentrations.

8.3 Conclusions and future steps

This study demonstrates that the top-down parameterization of Pluronic systems significantly benefits from the application of machine learning techniques. The presented workflow outlines a successfully a data-driven strategy for accurately determining model parameters tailored to different Pluronic systems.

In particular, GPR-based surrogate models exhibit a strong capability to replicate the results of Dissipative Particle Dynamics (DPD) simulations. In particular, we developed a reliable model that effectively acts as a surrogate for Dissipative Particle Dynamics (DPD) simulations.

This model ability to deliver faster responses facilitates more efficient parameter optimization and enables the calibration of various Pluronic systems against experimental "ground truth". The ability of this model to deliver quick responses enhances the efficiency of parameter optimization and supports the calibration of various Pluronic systems with experimental data. Although this method is straightforward and exhaustive, it can be computationally intensive, especially for DPD simulations, where each evaluation necessitates a full simulation and subsequent post-processing for property extraction. The GPR-based surrogate model offers a solution to these challenges by facilitating a more efficient grid-search process. By modeling the relationship between input parameters and output properties, GPR accelerates the optimization process and allows for a more informed exploration of the parameter space. This reduces the need for extensive simulations, making parameter optimization more practical and efficient.

This work lays the groundwork for a potential transfer learning framework, aiming to streamline the parameterization process across various Pluronic systems and conditions, such as differing temperatures. The goal is to generalize the parameterization process by incorporating additional DPD model input parameters and extending it to various Pluronic systems and conditions, including differing temperatures. The next section outlines the envisioned framework.

8.3.1 Transfer learning

To adequately capture the variations in R_g (or A_s or CMC) across all θ for a given Pluronic type at a specific temperature, data collection requires N_p points. Gener-

alizing the parameterization process by incorporating additional DPD model input parameters and extending it to various Pluronic systems and conditions, including differing temperatures and concentrations, necessitates a higher number of data points. This requires performing additional simulations.

In this context, we define $\mathbf{x} = [v, \boldsymbol{\theta}] \in \mathbb{R}^{(d+1)}$ and $y = R_g(\mathbf{x}) \in \mathbb{R}$ (or $y = A_s(\mathbf{x}) \in \mathbb{R}$ or $\text{CMC}(\mathbf{x}) \in \mathbb{R}$) as the input and output of GPR, respectively. The vector $\boldsymbol{\theta} \in \mathcal{P}$, with $\mathcal{P} \in \mathbb{R}^d$ and $d = 5$, represents the input model parameters:

$$\boldsymbol{\theta} = [a_{\text{AW}}, a_{\text{BW}}, a_{\text{AB}}, R_0, K]. \quad (8.11)$$

Instead, v refers to a variable of interest for the system that can be the temperature (T) or the polymer concentration (c_p) or the polymer length (L). In any case $v \in \Upsilon = [v_{\min}, v_{\max}]$. Therefore, $(\boldsymbol{\theta}, v) \in \mathcal{P} \times \Upsilon$ with $\mathcal{P} \in \mathbb{R}^d$ and $\Upsilon = [v_{\min}, v_{\max}]$, representing the domain of model parameters, respectively, where $d = 5$ is the dimension of the parameter space, and Υ the domain of the variable of interest. The objective is to forecast the value of $R_g(\boldsymbol{\theta}; v^*)$ (or $A_s(\boldsymbol{\theta}; v^*)$ or $\text{CMC}(\boldsymbol{\theta}; v^*)$) at any given point along the variable of interest $v^* \in \Upsilon$, utilizing the dataset of $R_g(\boldsymbol{\theta}; v)$ across N_p training instances of parameters. The Gaussian Process Regression (GPR) model's training dataset comprises $N_{\text{train}} = N_p \times N_v$ data points, where N_p represents the number of parameter instances and N_v denotes the number of instances along the variable v .

The dependence of the output variables (e.g., R_g , A_s , CMC) represented by the vector $\mathbf{Y} = [y_1, y_2, \dots, y_{N_{\text{train}}}]^T \in \mathbb{R}^{N_{\text{train}}}$, obtained from DPD simulations, on the input variables (model parameters and the variable of interest) $\mathbf{X} = [x_1, x_2, \dots, x_{N_{\text{train}}}]^T \in \mathbb{R}^{N_{\text{train}} \times (d+1)}$, is modeled as a Gaussian process: $\mathbf{Y}(\mathbf{X}) \sim GP(\mathbf{U}(\mathbf{X}), \boldsymbol{\Sigma} \mathbf{X} \mathbf{X})$. In this context, $\mathbf{U}(\mathbf{X}) \in \mathbb{R}^{N_{\text{train}}}$ represents the mean function, and $\boldsymbol{\Sigma} \mathbf{X} \mathbf{X} \in \mathbb{R}^{N_{\text{train}} \times N_{\text{train}}}$ denotes the covariance matrix.

It's clear that $N_{\text{train}} = N_p \times N_v$. To adequately capture the variations in R_g (or A_s 's or CMC 's) across all v of interest, data collection should span numerous N_p . For each v , We need the snapshot data at some fixed $\boldsymbol{\theta}$ with a number of N_p . Thus, at each v , the snapshot data can be considered as a vector with dimension of N_p . A strategy to mitigate training expenses involves minimizing the necessary N_p . We use this to do variable decomposition and build model to transfer on v . In this sense, an interesting help comes from the Proper Orthogonal Decomposition (POD) technique, which is an eigenvalue method used to reduce a high-dimensional data set into fewer dimensions while retaining important information. POD, also known

as Principal Component Analysis (PCA) in some contexts, identifies the dominant modes of variation within the dataset and represents the data using a smaller number of basis functions or modes. In the following R_g (or A_s or CMC) are designated as f . By projecting the original data onto the POD modes, the dimensionality can be effectively reduced while preserving the essential features of the f variations across different v values. This allows to capture the most significant variations in f with fewer degrees of freedom.

POD is a powerful model reduction technique. We need snapshot data of $f(\boldsymbol{\theta}; v_i)$ at v_i with $i = 1, 2, \dots, N_v$. A reduced order model (ROM) is constructed for f by decomposing the snapshot data:

$$f(\boldsymbol{\theta}; v_i) = \bar{f}(\boldsymbol{\theta}) + \hat{f}(\boldsymbol{\theta}; v_i), \quad (8.12)$$

where $\bar{f}(\boldsymbol{\theta}) = \frac{1}{N_v} \sum_{i=1}^{N_v} f(\boldsymbol{\theta}; v_i)$ denotes the mean of all snapshots. The task is to find a set of POD basis functions and POD coefficients such that the fluctuating part $\hat{f}(\boldsymbol{\theta}; v_i)$ can be reproduced as:

$$\hat{f}(\boldsymbol{\theta}; v_i) = \sum_{k=1}^{N_v} \alpha_k(v) \phi_k(\boldsymbol{\theta}). \quad (8.13)$$

For this purpose, the correlation matrix $\mathbf{A} \in \mathbb{R}^{N_v \times N_v}$ of the fluctuating parts:

$$A_{ij} = \int_{\mathcal{D}} \hat{f}(\boldsymbol{\theta}; v_i) \hat{f}(\boldsymbol{\theta}; v_j) d\boldsymbol{\theta}, \quad (8.14)$$

where i and j refer to the i -th and j -th snapshots, respectively. Then the POD bases $\phi_k(\boldsymbol{\theta})$ can be obtained via eigendecomposition of $\mathbf{A}$:

$$\phi_k(\boldsymbol{\theta}) = \frac{1}{\sqrt{\lambda_k}} \sum_{i=1}^{N_v} w_i^k \hat{f}(\boldsymbol{\theta}; v_i), \quad (8.15)$$

where $\lambda_1, \lambda_2, \dots, \lambda_M$ are eigenvalues in descending order and $\mathbf{w}^k$ is the eigenvector corresponding to λ_k . The POD basis functions satisfy the condition of orthogonality:

$$(\phi_k, \phi_l) = \begin{cases} 1, & \text{if } k = l, \\ 0, & \text{if } k \neq l. \end{cases}$$

where $(\phi_k, \phi_l) = \int_{\mathcal{D}} \phi_k(\boldsymbol{\theta}) \phi_l(\boldsymbol{\theta}) d\boldsymbol{\theta}$ defines the inner-product of two functions. In model order reduction, only the first $R \ll N_v$ largest eigenvectors (or POD modes) are preserved. As a result, a snapshot $\hat{f}(\boldsymbol{\theta}; v_i)$ can be approximated as:

$$\hat{f}(\boldsymbol{\theta}; v_i) \approx \hat{f}^{POD}(\boldsymbol{\theta}; v_i) = \sum_{k=1}^R \sqrt{\lambda_k} w_k^i \phi_k(\boldsymbol{\theta}), \quad (8.16)$$

where $\alpha_k(v_i) = \sqrt{\lambda_k} w_k^i$. The number of modes, R , should be empirically defined in such a way that most of the "energy" of the system is captured. Given the Eq. (8.16) we can establish the ROM for f as:

$$f(\boldsymbol{\theta}; v^*) \approx f^{ROM}(\boldsymbol{\theta}; v^*) = \bar{f}(\boldsymbol{\theta}) + \sum_{k=1}^R \alpha_k(v^*) \phi_k(\boldsymbol{\theta}), \quad (8.17)$$

by which the function can be effectively predicted at a given variable of interest instance v^* by the reduced POD bases $\phi_k(\boldsymbol{\theta})$ and parameter modes $\alpha_k(v^*)$. Consequently, we only need to train and predict $\alpha_k(v^*)$ instead of $f(\boldsymbol{\theta}; v^*)$, leading to significantly reduced training and prediction expenses. We can still utilize GPR to model each parameter mode $\alpha_k(v)$. Now, the required training data for GPR is $N_{\text{train}} = N_v$, significantly smaller than $N_{\text{train}} = N_p \times N_v$. Once the GPR model of each parameter mode is constructed, we can predict $\alpha_k(v^*)$ and in turn f by Eq. (8.17) at any given parameter instance v^* . In addition, the uncertainty for the predicted $\alpha_k(v^*)$ can be quantified by the standard deviation $\hat{\sigma}_k(v^*)$ in the GPR model.

Chapter 9

Conclusions

9.1 Overview of the key contributions of this research

As discussed in the earlier chapters, the primary objective of this dissertation is to demonstrate that DPD is a valuable computational tool in chemical engineering. Although the original version of the method has limitations in tackling real-world engineering problems, this work shows that many of these can be effectively addressed. The selected research field is the modeling of a distinctive class of structured fluids, Pluronic systems, which serve as test cases for soft matter systems commonly encountered in chemical engineering. Despite that, a significant portion of this work involves liquid water since it represents a crucial physical example that necessitates a high Schmidt number and, additionally, serves as a foundational step for simulating more complex fluids, such as aqueous Pluronic solutions.

A significant portion of this research is dedicated to enhancing DPD methodologies to accurately predict the transport properties of high Schmidt number fluids, a topic that has frequently been overlooked in existing DPD studies. The main contributions linked to this aspect are summarized as follows.

1. **Extended DPD model for high Schmidt number fluids.** This research introduces an extended DPD model that significantly improves the tunability of transport properties in DPD fluids. This model facilitates the simulation of fluids at specific Schmidt numbers without altering their physical state, effectively addressing a challenge frequently encountered in previous studies.

2. **New Time Integration Scheme for high Schmidt number fluids.** A more performant time integration scheme has been implemented in this extended DPD version, which markedly enhances performance in simulating high Schmidt number fluids. This improvement reduces the occurrence of simulation artifacts and eliminates the need for excessively low time steps, a common requirement in existing studies.
3. **Automated Method for Transport Properties Computation.** Another contribution of this research is the development of an automated method for accurately computing transport properties from equilibrium simulations. This computational protocol allows for calculations with controlled accuracy while minimizing computational effort. Striking the right balance between accuracy and computational time is crucial for effectively applying equilibrium techniques, as emphasized in numerous existing DPD studies.

The code related to the first two points is available online on Github¹. Also the code of the automated method is available online on Github².

Another important portion of this research work is dedicated to the parametrization of real-systems, with particular attention given to temperature and multi-component systems.

- **Methodology to incorporate temperature into the parametrization.** Within this theme, the first aspect explored is the incorporation temperature effects into the DPD model, often neglected, with the aim to accurately capture both temperature-dependent equilibrium and transport properties. In this context, liquid water is chosen as the physical system for parametrization, as it represents a simple system suitable for investigation with a high Schmidt number and temperature-dependent transport properties.
- **Methodology to parametrize real multi-component systems.** The second aspect explored regards the development of a machine-learning assisted parametrization methodology for multi-component systems following a top-down procedure, with the aim to accurately capture equilibrium properties. This part regards the parametrization of Pluronic systems.

¹<https://github.com/mulmopro/LAMMPS-DPD-EXT>

²https://github.com/mulmopro/AMTP_DPD

Finally, another minor part of the work proposes several studies in which DPD is employed as a complementary tool to real experiments, aiming to gain a deeper understanding of the intricate interplay between morphology and rheology in Pluronic systems. In particular, two applications are investigated.

- **Predicting concentration-dependent zero-shear viscosity in Pluronic L64/water mixtures**
- **Predicting temperature-dependent liquid-to-solid transition in Pluronic F68/water mixtures**

9.2 Conclusions summary

In conclusion, the main results obtained throughout this work are summarized in a chapter-by-chapter format, emphasizing their significance and impact on research as well as their practical applications in chemical engineering.

In Chapter 3 we demonstrated that an extended version of the DPD model effectively transforms gas-like dynamics into liquid-like dynamics. We introduce a robust viscosity evaluation procedure, demonstrating the reliability of equilibrium approaches when compared to benchmark non-equilibrium methods. More specifically, we observed that the viscosity coefficient estimated via the Green–Kubo approach is comparable with that obtained by using other methods (i.e., Einstein–Helfand equilibrium and non-equilibrium RNEMD methods) with the advantage that there is no limitation on its usage for equilibrium simulations. This becomes crucial for viscosity prediction in self-assembled non-Newtonian systems. Moreover, the proposed method of treating the noise of the SACF tail is capable of giving not only the mean viscosity value, but also the uncertainty of the numerical procedure associated with the viscosity evaluation. In this work, we show that such uncertainty is lower or comparable to the uncertainty of the experimental measurements. Eventually, we showed that the transport properties of the fluid can be finely tuned by adjusting specific model parameters, enabling the simulation of high Schmidt number fluids. Importantly, the model successfully reproduces the momentum and mass transport behavior characteristic of liquid water. The findings of this Chapter

laid the groundwork for research work successively done and presented in 4, 5, and 6.

In Chapter 4 we compared the results obtained by two time integration methods for dissipative particle dynamics (DPD), the new implemented Shardlow-splitting algorithm (VV-SSA) and the commonly used modified velocity-Verlet algorithm (VV), for the solution of a DPD model with transverse thermostat. We demonstrated the superiority of the VV-SSA in the evaluation of transport properties of fluids with high Schmidt numbers. More specifically, we found that the VV-SSA eliminates the integrator-induced artifacts and performs better in a wider friction region than the VV. Moreover, larger values of the time step can be adopted in transport properties calculation with VV-SSA. This aspect preserves the main advantageous of DPD. In fact in this technique, soft interparticle forces neither change abruptly nor diverge at short distances, which permits long integration steps as compared to the atomistic simulations. This fact appreciably accelerates the simulations and, together with the coarse-graining allows for the study of mesoscale phenomena making it appealing for modeling the behaviour of structured fluids.

In Chapter 5 we focused on the numerical issue related to the calculation of the transport quantities from a numerical simulation. We formulated and developed an automated algorithm, based on Green-Kubo formula, to compute transport properties with controlled accuracy adopting a well-defined computational protocol. Our method automatically finds the simulation parameters employed by LAMMPS to compute the Stress Auto-Correlation Function with the "best" trade-off between statistical accuracy achieved and computational effort. This tool, firstly, improves the reliability of the results. In fact, we showed that there is the possibility to get wrong results if these parameters are poorly chosen. Secondly, it addresses the issue of computational efficiency. The optimal choice of these parameters is system-dependent and, typically, for each system investigated, they are identified through a trial-and-error procedure, which consumes a significant amount of computational time. Moreover, we found that transport properties can be wrongly evaluated if the time step of the DPD simulation is not accurately selected. The developed automated method is able to find, for a given DPD system, the viscosity value that is independent on the choice of time step. Finally from a parametric study in which we varied all the dissipative parameters of the DPD model, we evaluated the effective friction coefficient, viscosity, self-diffusivity and Sc number. From this

dataset made by over 60 simulations, we derived a fitting relation for the effective friction coefficient as a function of different DPD parameters, and fitting relations for viscosity, self-diffusivity and Schmidt (Sc) number as a function of the effective friction coefficient. This analysis shows that infinite sets of DPD parameters may give the same effective friction coefficient, in turn leading to the same Sc number and radial distribution function. In other words, it is possible to simulate the same fluid with the same rheological behavior with different sets of DPD parameters. Moreover, we found that viscosity increases linearly with effective friction coefficient, the self-diffusion is inversely proportional, and the ratio between the viscosity and diffusivity recovered fitting curves results in a very good fit for Sc number values computed from simulations. The findings of this work can be used to simulate simple fluids with different Sc number, since the relation between a specific effective friction coefficient and DPD parameters was found. Moreover, although the method proposed in this work was extensively tested and optimized for simple fluids, it is a tool of general validity with the aim of improving the reliability and computational feasibility of the Green-Kubo approach in DPD simulations. Therefore, it is potentially applicable to both simple and complex fluids. However, the application to more topologically complicated systems is not straightforward, since the intrinsic difficulties of the Green-Kubo approach in DPD are accentuated. Indeed, in the case of polymer systems the computational load required for the viscosity prediction is even higher, since the long-time relaxation requires a larger time interval to trace the whole decorrelation dynamics of the SACF. Although there are some challenges to address, the reliability of the method makes it a promising tool to perform a systematic scaling analysis on the polymer systems dynamical properties, such as viscosity and diffusivity, with polymer concentration and length.

In Chapter 6 we investigated the use of the extended dissipative particle dynamics (DPD) model, numerically solved with the Shardlow-splitting-algorithm-based integration scheme, as a computational tool to predict temperature-dependent transport properties at the mesoscale following a top-down coarse-graining. The transport properties calculation from equilibrium simulations was performed using the automated method presented in Chapter 5. A temperature-dependent form of the extended DPD model, where both conservative and non-conservative interaction parameters incorporate the system temperature variation, was proposed. In particular, liquid water was used as model system in this work. Experimental data of

transport properties [93, 94] from 5°C to 90°C were used as target values to develop and validate the model. The extended DPD model allows the simulation of water dynamics across different temperatures within the whole liquid range consistently with experimental data of viscosity and self-diffusivity. Although the model was developed and validated against transport properties of liquid water, it can also be used as a first rough estimation for the non-conservative parameters of other liquids. The temperature-dependent form of the extended DPD model of this work can be further extended to multicomponent systems or systems with chain-like molecules. For the conservative interaction parameters, which are based on Groot-Warren's DPD parameterization [13], the temperature-dependent compressibility can be used to determine like repulsive parameters, a_{ii} , at various temperatures. Unlike repulsive parameters, a_{ij} , can be determined from a linear relationship between a_{ij} and the Flory-Huggins interaction parameter, which is inherently temperature-dependent. Alternatively for systems with chemically distinctly different beads or local density variations, Kacar et al.'s DPD parameterization can be used, which explicitly involves a temperature dependence [96, 97]. Parameters of non-conservative forces can be adjusted analogously as in this work to the viscosity and mutual self-diffusion coefficients. However, adjustment will involve different *itype* – *jtype* pairs, e.g., 1-1, 1-2, and 2-2 type-pairs for a binary system, and may become rather complex and demanding. Such the adjustment may benefit from machine learning methods. The outcomes of this work would also be useful for future developments of non-isothermal DPD models. In fact, the energy conserving DPD (DPDe) is slowly growing as a powerful tool in capturing temperature-induced mesoscopic phenomena, but it requires the definition of temperature-dependent interactions. The incorporation of a temperature-dependent conservative force-field was presented in different studies and it was shown to be capable of predicting the behavior of equilibrium properties induced by a thermal gradient. Instead on the other hand, the incorporation of temperature-dependent dissipative and random forces, such as those proposed in this work, is less straightforward. It requires further theoretical development to properly modify the DPDe framework. However, this extension is interesting and crucial to exploit DPDe as a computational tool to simulate at the mesoscale also the transport properties behavior induced by a thermal gradient.

In Chapter 7, we demonstrated, through two distinct applications, that DPD is a valuable tool to investigate the interplay between morphology and rheology in Pluronic systems and to have a deeper understanding of the phenomena involved.

In the first case study, we showed that DPD can successfully predict the concurrent changes in morphology and zero-shear viscosity values induced by variations in concentration in a binary mixture of Pluronic L64 and water. In the second case study, we demonstrated that DPD can effectively capture the liquid-to-solid transition through both morphological characterization and rheological measurements induced by temperature change in a binary mixture of Pluronic F68 and water. In both cases, the results exhibited a strong correlation with experimental data.

In Chapter 8, we demonstrated that the top-down parametrization of Pluronic systems significantly benefits from the application of machine learning techniques. We described a successful parametrization workflow that employs a data-driven approach to determine the model parameters for Pluronic systems. Specifically, a reliable data-driven model was developed to accurately surrogate DPD simulations. This model ability to provide faster responses enables more efficient parameter optimization and facilitates the calibration of various Pluronic systems against experimental data. Additionally, we proposed a transfer learning based framework to enhance the efficiency of the parametrization process across different types of Pluronic systems.

References

- [1] M. Karttunen, A. Lukkarinen, and I. Vattulainen, *Novel Methods in Soft Matter Simulations* (Springer Berlin, Heidelberg, 2013).
- [2] N. A. Di Spirito, N. Grizzuti, M. Casalegno, F. Castiglione, and R. Pasquino, *International Journal of Pharmaceutics* **644**, 37647976 (2023).
- [3] M. Khimani, H. Patel, V. Patel, P. Parekh, and R. L. Vekariya, *Polymer Bulletin* **77**, 5783 (2020).
- [4] K. Ulrich, P. Galvosas, J. Kärger, and F. Grinberg, *Materials* **5**, 966 (2012).
- [5] P. Alexandridis and T. A. Hatton, *Colloids and Surfaces A* **96**, 1 (1995).
- [6] F. Müller-Plathe, *Physical Review E* **59**, 4894 (1999).
- [7] A. Pitto-Barry and N. P. E. Barry, *Polymer Chemistry* **5**, 3281 (2014).
- [8] T. W. Sirk, Y. R. Sliozberg, J. K. Brennan, M. Lisal, and J. W. Andzelm, *Journal of Chemical Physics* **136**, 134903 (2012).
- [9] G. Besold, I. Vattulainen, M. Karttunen, and J. M. Polson, *Physical Review E* **62**, R7611 (2000).
- [10] I. Vattulainen, M. Karttunen, G. Besold, and J. M. Polson, *The Journal of Chemical Physics* **116**, 3967 (2002).
- [11] M. Lísal, J. K. Brennan, and J. Bonet Avalos, *AIP Publishing* **135**, 1 (2011).
- [12] N. Lauriello, J. Kondracki, A. Buffo, G. Boccardo, M. Bouaifi, M. Lísal, and D. Marchisio, *Physics of Fluids* **33**, 073106 (2021).
- [13] R. D. Groot and P. B. Warren, *The Journal of Chemical Physics* **107**, 4423 (1997).
- [14] Z. Li, Y.-H. Tang, H. Lei, B. Caswell, and G. E. Karniadakis, *Journal of Computational Physics* **265**, 113 (2014).
- [15] P. Durchschlag and H. Zipper, *Progress in Colloid and Polymer Science* **94**, 20 (1994).

-
- [16] A. Prhashanna, S. A. Khan, and S. B. Chen, *The Journal of Physical Chemistry B* **119**, 572 (2015).
- [17] N. A. D. Spirito, N. Grizzuti, and R. Pasquino, *Physics of Fluids* **36**, 111302 (2024).
- [18] S. Izvekov and B. M. Rice, *Physical Chemistry Chemical Physics* **17**, 10795 (2015).
- [19] J. B. Avalos, M. Lísal, J. P. Larentzos, A. D. Mackie, and J. K. Brennan, *Physical Chemistry Chemical Physics* **21**, 24891 (2019).
- [20] J. B. Avalos, M. Lísal, J. P. Larentzos, A. D. Mackie, J. K. Brennan, and J. K. Brennan, *Physical Review E* **103**, 062128 (2021).
- [21] M. Lísal, J. P. Larentzos, J. B. Avalos, A. D. Mackie, and J. K. Brennan, *Journal of Chemical Theory and Computation* **18**, 2503 (2022).
- [22] P. Vanya, J. Sharman, and J. A. Elliott, *The Journal of Chemical Physics* **150**, 064101 (2019).
- [23] E. Mayoral and A. G. Goicochea, *The Journal of Chemical Physics* **138**, 094703 (2013).
- [24] S. Trément, B. Schnell, L. Petitjean, M. Couty, and B. Rousseau, *The Journal of Chemical Physics* **140**, 134113 (2014).
- [25] S. Izvekov and B. M. Rice, *The Journal of Chemical Physics* **140**, 104104 (2014).
- [26] H. Mori, *Progress of Theoretical Physics* **33**, 423–455 (1965).
- [27] R. Zwanzig, *The Journal of Chemical Physics* **33**, 1338 (1965).
- [28] A. P. Thompson, H. M. Aktulga, R. Berger, D. S. Bolintineanu, W. M. Brown, P. S. Crozier, P. J. in 't Veld, A. Kohlmeyer, S. G. Moore, T. D. Nguyen, R. Shan, M. J. Stevens, J. Tranchida, C. Trott, and S. J. Plimpton, *Computer Physics Communication* **271**, 108171 (2022).
- [29] P. Alexandrilis and B. Lindman, *Amphiphilic Block Copolymers - Self-Assembly and Applications* (Elsevier, 2000).
- [30] P. Alexandrilis, *Current Opinion in Colloid and Interface Science* **1**, 490 (1996).
- [31] G. Wanka, H. Hoffmann, and W. Ulbricht, *Macromolecules* **27**, 4145 (1994).
- [32] P. Alexandrilis, D. Zhou, and A. Khan, *Langmuir* **12**, 2690 (1996).
- [33] R. Larson, *The Structure and Rheology of Complex Fluids* (Oxford University Press, 1999).

- [34] R. D. Groot and P. B. Warren, *The Journal of Chemical Physics* **107**, 4423 (1997).
- [35] P. Español and P. Warren, *Europhysics Letters (EPL)* **30**, 191 (1995).
- [36] M. P. Allen and F. Schmid, *Molecular Simulation* **33**, 21 (2007).
- [37] R. Groot and K. Rabone, *Biophysical Journal* **81**, 725 (2001).
- [38] R. M. Füchslin, H. Fellermann, A. Eriksson, and H.-J. Ziock, *The Journal of Chemical Physics* **130**, 214102 (2009).
- [39] P. Flory, *Principles of Chemistry Cap.12* (Cornell University Press, Itacha, NY, 1953).
- [40] R. D. Groot and K. L. Rabone, *Biophysical Journal* **81**, 725 (2001).
- [41] C. Wijmans, B. Smit, and R. Groot, *J. Chem. Phys.* **114**, 7644 (2001).
- [42] P. Español, *Physical Review E* **57**, 2930 (1998).
- [43] K. Novik and P. Convey, *Journal of Chemical Physics* **109**, 7667 (1998).
- [44] I. Pagonabarraga, M. H. J. Hagen, and D. Frenkel, *Europhysics Letters* **42**, 377 (1998).
- [45] T. Shardlow, *Journal on Scientific Computing* **24**, 1267 (2003).
- [46] A. F. Jakobsen, G. Besold, and O. G. Mouritsen, *The Journal of Chemical Physics* **124**, 094104 (2006).
- [47] J.-P. Hansen and I. R. McDonald, *Theory of Simple Liquids* (Academic Press, Amsterdam, Netherlands, 2013).
- [48] A. W. Lees and S. F. Edwards, *Journal of Physics C: Solid State Physics* **5**, 1921 (1972).
- [49] J. A. Backer, C. P. Lowe, H. C. J. Hoefsloot, and P. D. Iedema, *The Journal of Chemical Physics* **122**, 154503 (2005).
- [50] M. S. Green, *The Journal of Chemical Physics* **22**, 398 (1954).
- [51] R. Kubo, *Journal of the Physical Society of Japan* **12**, 570 (1957).
- [52] E. Helfand, *Physical Review* **119**, 1 (1960).
- [53] W. Hong, J. Lin, X. Tian, and L. Wang, *Macromolecules* **53**, 10955 (2020).
- [54] K. Šindelka and M. Lísal, *Molecular Physics* **119**, e1857863 (2021).
- [55] G. R. Newkome, C. N. Mooreld, and V. F. *Dendrimers and Dendrons: Concepts, Syntheses, Applications* (Wiley-VCH, Weinheim, 2004).

- [56] L. Cheng and D. Cao, [Langmuir](#) **25**, 2749 (2009).
- [57] D. J. Bray, A. Del Regno, and R. L. Anderson, [Molecular Simulation](#) **46**, 308 (2020).
- [58] J. Partington, R. Hudson, and K. Bagnall, *Nature* **169**, 583 (1952).
- [59] R. Groot, *Langmuir* **16**, 7493 (2000).
- [60] P. Español and P. Warren, [Europhysics Letters \(EPL\)](#) **30**, 191 (1995).
- [61] X. Fan, N. Phan-Thien, S. Chen, X. Wu, and T. Y. Ng, [Physics of Fluids](#) **18**, 063102 (2006).
- [62] W. Pan, I. V. Pivkin, and G. E. Karniadakis, [EPL \(Europhysics Letters\)](#) **84**, 10012 (2008).
- [63] S. Yaghoubi, E. Shirani, A. R. Pishevar, and Y. Afshar, [EPL \(Europhysics Letters\)](#) **110**, 24002 (2015).
- [64] C. Junghans, M. Praprotnik, and K. Kremer, [Soft Matter](#) **4**, 156 (2008).
- [65] R. C. Krafnick and A. E. García, [The Journal of Chemical Physics](#) **143**, 243106 (2015).
- [66] S. D. Stoyanov and R. D. Groot, [The Journal of Chemical Physics](#) **122**, 114112 (2005).
- [67] P. J. Hoogerbrugge and J. M. V. A. Koelman, [Europhysics Letters \(EPL\)](#) **19**, 155 (1992).
- [68] S. Chen, N. Phan-Thien, B. C. Khoo, and X. J. Fan, [Physics of Fluids](#) **18**, 1 (2006).
- [69] P. D. Palma, P. Valentini, and M. Napolitano, [Physics of Fluids](#) **18**, 1 (2006).
- [70] Z. Li and G. Drazer, [Physics of Fluids](#) **20**, 1 (2008).
- [71] N. Mai-Duy, N. Phan-Thien, T. Nguyen, and T. Tran-Cong, [Physics of Fluids](#) **32**, 1 (2020).
- [72] H. Reddy and J. Abraham, [Physics of Fluids](#) **21**, 1 (2009).
- [73] G. Jung and F. Schmid, [The Journal of Chemical Physics](#) **144**, 204104 (2016).
- [74] W. Dzwinel and D. A. Yuen, [International Journal of Modern Physics C](#) **11**, 1 (2000).
- [75] I. V. Pivkin and G. E. Karniadakis, [The Journal of Chemical Physics](#) **124**, 184101 (2006).
- [76] R. Groot and K. Rabone, [Biophysical Journal](#) **81**, 725 (2001).

- [77] F. Lahmar and B. Rousseau, *Polymer* **48**, 3584 (2007).
- [78] B. Leimkuhler and X. Shang, *Journal of Computational Physics* **280**, 72 (2015).
- [79] P. Nikunen, M. Karttunen, and I. Vattulainen, *Computer Physics Communications* **153**, 407 (2022).
- [80] J. P. Larentzos, J. K. Brennan, J. D. Moore, M. Lísal, and W. D. Mattson, *Computer Physics Communications* **185**, 1987 (2014).
- [81] L. Verlet, *Physical Review* **159**, 98 (1967).
- [82] A. Chaudhri and J. R. Lukes, *Phys. Rev. E* **81**, 026707 (2010).
- [83] E. a. J. F. Peters, *Europhysics Letters* **66**, 311 (2004).
- [84] F. Lahmar and B. Rousseau, *Polymer* **48**, 3584 (2007).
- [85] M. H. Ernst and R. Brito, *Europhysics Letters* **73**, 183 (2006).
- [86] G. Jung and F. Schmid, *The Journal of Chemical Physics* **144**, 204104 (2016).
- [87] M. Panoukidou, C. R. Wand, and P. Carbone, *Soft Matter* **17**, 8343 (2021).
- [88] R. Kubo, *Journal of the Physical Society of Japan* **12**, 570 (1957).
- [89] D. C. Malaspina, M. Lísal, J. P. Larentzos, M. A. D. Brennan, J. K., and J. Bonet Avalos, *Physical Chemistry Chemical Physics* **25**, 12025 (2023).
- [90] N. Lauriello, G. Boccardo, D. Marchisio, M. Lísal, and A. Buffo, *Computer Physics Communications* **291**, 108843 (2023).
- [91] D. C. Malaspina, M. Lísal, J. P. Larentzos, J. K. Brennan, A. D. Mackie, and J. Bonet Avalos, *Physical Chemistry Chemical Physics* **26**, 1328 (2024).
- [92] C. A. Marsh, G. Backx, and M. H. Ernst, *Physical Review E* **56**, 1676 (1997).
- [93] E. W. Lemmon, M. O. McLinden, and D. G. Friend, in *NIST Chemistry Webbook, NIST Standard reference database*, Vol. 69, edited by P. J. Linstrom and W. G. Mallard (National Institute of Standards and Technology, Gaithersburg MD, 20899, USA, 1998).
- [94] M. Holz, S. R. Heil, and A. Sacco, *Physical Chemistry Chemical Physics* **2**, 4740 (2023).
- [95] I.-C. Yeh and G. Hummer, *The Journal of Physical Chemistry B* **108**, 15873 (2004).
- [96] G. Kacar, E. A. J. F. Peters, and G. de With, *Europhys. Lett.* **102**, 40009 (2013).

- [97] G. Kacar, E. A. J. F. Peters, and G. de With, *Soft Matter* **9**, 5785 (2013).
- [98] B. A. C. van Vlimmeren, N. M. Maurits, A. V. Zvelindovsky, G. J. A. Sevink, and J. G. E. M. Fraaije, *ACS Publications* (1999).
- [99] R. L. Anderson, D. J. Bray, A. S. Ferrante, M. G. Noro, I. Stott, and P. B. Warren, *J. Chem. Phys.* **147**, 094503 (2017).
- [100] R. Sharma and K. Tankeshwar, *The journal of chemical physics* **108**, 2601 (1997).
- [101] Y. Araki and N. Arai, *Chemical Engineering Communications* **208**, 1 (2019).
- [102] T. L. Chantawansri, T. W. Sirk, and Y. R. Silozberg, *The Journal of Chemical Physics* **138**, 024908 (2013).
- [103] G. Dorenbos, *Polymer* **173**, 43 (2019).
- [104] R. Ketkaew and Y. Tantirungrotechai, *Macromolecular Theory and Simulations* **1700093**, 1 (2018).
- [105] G. S. Jiao, Y. Li, H.-J. Quian, J. Liu, and Z.-Y. Lu, *Molecular Simulations* **44**, 137 (2017).
- [106] A. De Nicola, T. Kawakatsu, and G. Milano, *Macromolecular Chemistry and Physics* **214**, 1940 (2013).
- [107] M. A. Johnston, W. C. Swope, K. E. Jordan, P. B. Warren, M. G. Noro, D. J. Bray, and R. L. Anderson, *The Journal of Physical Chemistry B* **120**, 6337 (2016).
- [108] D. C. Liu and J. Nocedal, *Mathematical Programming* **45**, 503 (1989).
- [109] G. Wanka, H. Hoffmann, and W. Ulbricht, *Macromolecules* **27**, 4145 (1994).
- [110] P. Alexandridis, D. Zhou, and A. Khan, *Langmuir* **12**, 2690 (1996).
- [111] A. V. Kabanov, I. R. Nazarova, I. V. Astafieva, E. V. Batrakova, V. Y. Alakhov, A. A. Yaroslavov, and V. A. Kabanov, *Macromolecules* **28**, 2303 (1995).

Appendix A

Appendix A

This appendix contains supporting information material for Chapter 5 which deals with the development of an automated reliable method to compute transport properties from DPD equilibrium simulations.

A.1 Schmidt number scaling with effective friction coefficient

In the following, Table A.1 reports the parameters for all the simulation performed in the work.

Table A.1 Summary of all DPD simulation performed, including the results in terms of viscosity, self-diffusion coefficient and Schmidt number; all properties are in DPD units. All simulations are carried out with $a = 25$ and $\rho = 3$.

γ_{eff}	s	r_c^D	γ	η	$\mathcal{D}$	Sc	Δt	N _{rep}	N _{sim}
1.092	1	1	4.5	0.574	0.173	1.103	0.0119	1000	900
2.184	1	1	9	0.550	0.115	1.593	0.0119	1000	900
4.372	1	1	18	0.645	0.068	3.142	0.0178	1000	800
8.758	1	1	36	0.885	0.038	7.675	0.0267	500	700
5.386	1	1.5	4.5	0.631	0.048	4.387	0.0267	1000	700
10.772	1	1.5	9	0.981	0.025	12.968	0.0178	500	800
21.539	1	1.5	18	1.785	0.0119	45.011	0.0119	1000	900

43.051	1	1.5	36	3.370	0.007	172.164	0.0053	1500	1100
9.063	1	1.75	4.5	0.769	0.029	8.759	0.0267	1000	700
18.125	1	1.75	9	1.572	0.015	36.127	0.0119	1000	900
36.246	1	1.75	18	2.966	0.007	136.045	0.0053	1500	1000
72.448	1	1.75	36	5.324	0.004	445.207	0.0178	2000	800
11.862	1	1.9	4.5	0.998	0.022	14.834	0.0267	500	700
23.722	1	1.9	9	1.976	0.011	62.194	0.0079	1000	1000
47.441	1	1.9	18	3.710	0.006	221.847	0.0053	2000	800
94.819	1	1.9	36	7.365	0.003	879.229	0.0023	4000	600
13.996	1	2	4.5	1.187	0.019	21.024	0.0178	1000	800
27.993	1	2	9	2.291	0.009	85.015	0.0023	2000	900
55.982	1	2	18	4.543	0.004	337.940	0.0016	3500	700
111.896	1	2	36	8.030	0.003	1065.770	0.0035	3500	700
3.576	0.5	1	4.5	0.600	0.080	2.496	0.0267	500	700
7.141	0.5	1	9	0.826	0.043	6.472	0.0267	1000	600
14.263	0.5	1	18	1.270	0.022	19.649	0.0119	1000	900
28.506	0.5	1	36	2.334	0.011	72.434	0.0053	1000	1100
14.699	0.5	1.5	4.5	1.229	0.018	22.473	0.0178	1000	800
29.400	0.5	1.5	9	2.346	0.009	86.368	0.0079	1000	1000
58.795	0.5	1.5	18	4.533	0.004	336.425	0.0053	4500	800
117.523	0.5	1.5	36	8.800	0.002	1262.328	0.0035	2500	800
24.002	0.5	1.75	4.5	2.069	0.011	64.757	0.0016	5000	600
48.009	0.5	1.75	9	3.894	0.005	246.170	0.0016	7500	600
96.008	0.5	1.75	18	7.085	0.003	883.700	0.0119	1000	900
191.899	0.5	1.75	36	12.190	0.001	2822.270	0.0053	1500	700
31.065	0.5	1.9	4.5	2.332	0.009	91.042	0.0007	9500	600
62.137	0.5	1.9	9	4.708	0.004	369.589	0.0119	1000	900
124.269	0.5	1.9	18	8.625	0.002	1321.755	0.0053	1500	700
36.433	0.5	2	4.5	3.066	0.007	147.764	0.0007	9500	600

72.864	0.5	2	9	5.903	0.003	575.094	0.0053	1500	1200
145.731	0.5	2	18	10.462	0.002	1931.647	0.0016	4500	600
11.091	0.125	1	4.5	0.988	0.027	11.996	0.0010	8000	600
22.193	0.125	1	9	1.940	0.012	54.895	0.0267	1000	700
44.392	0.125	1	18	3.554	0.007	171.912	0.0079	1000	1000
88.714	0.125	1	36	6.597	0.004	599.961	0.0053	2000	1100
39.729	0.125	1.5	4.5	3.035	0.007	152.936	0.0035	3000	1200
79.475	0.125	1.5	9	5.607	0.003	568.600	0.0079	1000	1000
158.920	0.125	1.5	18	10.880	0.001	2935.451	0.0007	7000	600
64.166	0.125	1.75	4.5	4.895	0.004	414.639	0.0023	5000	700
82.541	0.125	1.9	4.5	6.119	0.003	666.431	0.0035	2500	1200
165.080	0.125	1.9	9	11.563	0.002	2461.411	0.0035	2500	1300
7.352	0.25	1	4.5	0.797	0.040	6.725	0.0010	9000	600
14.706	0.25	1	9	1.295	0.021	20.493	0.0178	1000	800
29.409	0.25	1	18	2.009	0.012	56.710	0.0178	1000	800
27.560	0.25	1.5	4.5	2.258	0.010	77.412	0.0119	1000	900
55.124	0.25	1.5	9	4.313	0.005	310.391	0.0016	3500	800
110.234	0.25	1.5	18	7.967	0.002	1073.213	0.0016	5000	600
220.479	0.25	1.5	36	15.036	0.001	3880.659	0.0010	10500	600
44.623	0.25	1.75	4.5	3.459	0.006	203.423	0.0053	1500	600
89.254	0.25	1.75	9	6.694	0.003	760.439	0.0035	3000	700
57.516	0.25	1.9	4.5	4.640	0.004	361.797	0.0035	2500	1200
115.042	0.25	1.9	9	7.863	0.002	1116.612	0.0035	3000	600
67.287	0.25	2	4.5	5.154	0.004	456.452	0.0040	2000	900

Appendix B

Appendix B

This appendix contains supporting information material for Chapter 6 which deals with modeling temperature-dependent transport properties in dissipative particle dynamics.

B.1 Error Estimation in Viscosity Calculations

Employing the automated method, the error in the viscosity is dictated by the threshold values set for out-of-loop conditions, selected as a balance between desired accuracy and computational expense. Specifically, the automated method conducts three checks on the computed viscosity through the three out-of-loop conditions as illustrated in Fig. 5.1. More in depth, the error strongly depends on the inaccuracies in the calculation of the SACF.

First, if the time interval analyzed by the SACF is not sufficiently large to trace the whole decorrelation dynamics, the tail of the SACF may not reach zero. This can result in the CVI curve continuing to increase with time, leading to an inaccurate estimation of viscosity. The viscosity estimate is acceptable when its variation with an increasing SACF sampling time interval is negligible. In other words, when the tail of the CVI oscillates around a plateau corresponding to the mean value of viscosity, η_{avg} .

Second, also the noise of the CVI may produce an error in the viscosity calculation. Such error is quantified with the COV of the viscosity estimate - the ratio between the standard deviation η_{std} and the mean viscosity η_{avg} value. We showed

that the noise of CVI strongly depends on the simulation length varied in the N_{sim} -loop, the second loop of the automated algorithm. The corresponding check is done on the absolute error between the COV of two consecutive simulations, and this quantity must be lower than a certain threshold.

Eventually, timestep size dependency of the results, namely the Δt -loop, contributes to the final error of the automated procedure. A certain level of temporal resolution, different for each system studied, has to be reached to obtain the independence of the viscosity value. The relative error between two viscosity values computed with decreasing the time step size is set to be lower than a certain threshold value. Once the proper Δt is found with the Δt -loop, it is possible to define the error of the viscosity calculation procedure, which is related to the out-of-loop conditions in the first two loops at that specific Δt : η_{avg} , η_{std} and the COV. The error estimates are now included in the Table B.1.

Table B.1 Error estimates in the viscosity calculations across different temperatures considered in the work. η_{avg} is the mean value of viscosity, η_{std} and COV are the associated standard deviation and coefficient of variation, respectively.

T [$^{\circ}\text{C}$]	η_{avg}	η_{std}	COV[%]
5	37.344	0.162	0.40
15	26.155	0.086	0.30
20	22.663	0.038	0.20
25	18.604	0.065	0.40
30	17.279	0.1	0.60
35	15.783	0.04	0.30
40	14.22	0.025	0.20
45	13.132	0.032	0.20
50	12.103	0.027	0.20
60	10.267	0.043	0.40
70	8.949	0.008	0.10
90	7.066	0.022	0.30

B.2 Assessing the reliability of self-diffusivity calculations

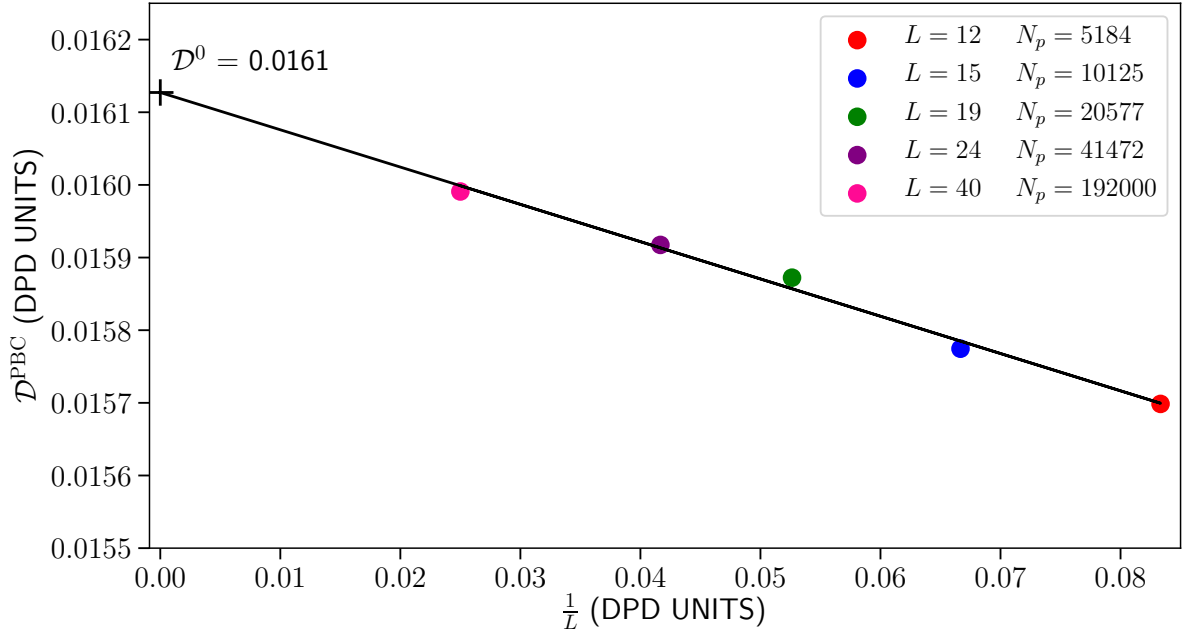


Fig. B.1 Self-diffusion coefficient of water $\mathcal{D}^{\text{PBC}}$ at ambient temperature as a function of the inverse box length $1/L$ with different number of water beads $N = 5,184 - 192,000$. The black line is a linear fit of the simulated self-diffusion coefficients, $\mathcal{D}^{\text{PBC}}$, to $1/L$. The linear fit provides "true" self-diffusion coefficient $\mathcal{D}^0 = 0.0161$.

Table B.2 Impact of the finite-size correction on self-diffusivity calculations.

$k_B\theta$	η	$\mathcal{D}^{\text{PBC}}$	$\mathcal{D}^0$	$ \mathcal{D}^{\text{PBC}} - \mathcal{D}^0 $	$\frac{ \mathcal{D}^{\text{PBC}} - \mathcal{D}^0 }{\mathcal{D}^0}$
0.933	37.344	0.0111	0.0112	0.0002	1.46%
0.966	26.155	0.0133	0.0136	0.0002	1.79%
0.983	22.663	0.0150	0.0153	0.0003	1.86%
1	18.604	0.0158	0.0161	0.0004	2.19%
1.02	17.279	0.0182	0.0185	0.0004	2.08%
1.03	15.783	0.0211	0.0215	0.0004	2.00%
1.05	14.22	0.0224	0.0228	0.0005	2.12%
1.07	13.132	0.0252	0.0258	0.0005	2.07%
1.08	12.103	0.0282	0.0288	0.0006	2.04%
1.12	10.267	0.0338	0.0345	0.0007	2.07%
1.15	8.949	0.0405	0.0413	0.0008	2.04%
1.18	7.826	0.0470	0.0480	0.0010	2.07%
1.22	7.066	0.0542	0.0553	0.0011	2.04%

Table B.3 Impact of the corrected self-diffusivity values on the final Sc number.

$k_B\theta$	$\text{Sc}(\mathcal{D}^0)$	$\epsilon_{\text{Sc}(\mathcal{D}^0)}[\%]$	$\text{Sc}(\mathcal{D}^{\text{PBC}})$	$\epsilon_{\text{Sc}(\mathcal{D}^{\text{PBC}})}[\%]$
0.933	1111.429	4.80	1125.19	3.6
0.966	641.053	0.60	654.04	1.3
0.983	493.747	0.50	504.19	1.6
1	385.176	0.80	392.91	1.1
1.02	311.333	0.90	317.33	2.7
1.03	244.697	1.70	249.92	0.6
1.05	207.894	2.30	211.98	4.3
1.07	169.664	1.20	173.35	3.2
1.08	140.081	0.20	142.95	2.1
1.12	99.198	0.60	101.35	1.4
1.15	72.227	1.70	73.74	0.3
1.18	54.347	1.90	55.52	0.2
1.22	42.591	0.90	53.44	1