Thesis for The Degree of Licentiate of Engineering

Exploring Brownian Phenomena using Hydrodynamic Memory Kernels

Anand Joseph Michael

Department of Mechanics and Maritime Sciences
Chalmers University of Technology
Gothenburg, Sweden, 2025

Exploring Brownian Phenomena using Hydrodynamic Memory Kernels

Anand Joseph Michael

Department of Mechanics and Maritime Sciences
Division of Fluid Dynamics
Chalmers University of Technology
SE-412 96 Göteborg,
Sweden
Phone: +46(0)31 772 1000

Printed by Chalmers Digitaltryck,
Gothenburg, Sweden 2025.

To my family and friends

Exploring Brownian Phenomena using Hydrodynamic Memory Kernels

Anand Joseph Michael

Department of Mechanics and Maritime Sciences
Chalmers University of Technology

Abstract

Accurate modelling of Brownian phenomena at low particle-fluid density ratios can improve the design of many microfluidic devices. Two novel simulation methods are presented that rely on the generalised Langevin equation (GLE) and its associated memory kernel to accurately model Brownian motion in complex flow scenarios. The methods are capable of capturing the drag, added mass and history effects that are relevant at low particle-fluid density ratio (ρ_p/ρ_f close to 1). The first method is a pure Direct Numerical Simulation (DNS) method that relies on numerically solving the Navier-Stokes equations to measure the total hydrodynamic force on the particle. The memory kernel required for the GLE based model is obtained via optimisation procedures that rely on the hydrodynamic force and the velocity history of the particle. The random colored Brownian force required for the GLE is then generated using the memory kernel. Finally, the particle is moved under the influence of the hydrodynamic force and the Brownian force. Unhindered and wall-adjacent Brownian motion are accurately simulated using the method.

The second method is a multiscale Lagrangian particle tracking (LPT) method, which is demonstrated to accurately capture hydrodynamic forces and Brownian motion by relying on solely the memory kernel associated with a particle's position. The method is used to study the migration of a Brownian particle towards a wall under the influence of a constant attractive force. A memory kernel library mapping the hydrodynamic forces (including nonlocal history and added mass effects) is generated using short DNS that utilise the optimisation routine developed as part of the previous pure DNS approach. The effects of the distance to the wall and the strength of the pulling force on the impact region of the particle on the wall are analysed and presented. The multiscale method is seen to be computationally more efficient than the pure DNS method while being capable of capturing detailed flow physics. The capabilities of the memory kernel to capture the effects associated with different fluid, particle and domain properties open up the possibility to extend these methods to more complex flow cases relevant to various microfluidic applications.

Keywords

Colored Brownian motion, Generalised Langevin Equation, Memory Kernel, Multiphase Direct Numerical Simulations, Lagrangian Particle Tracking

List of Publications and Conferences

Appended publications

This thesis is based on the following publications:

[**Paper A**] **A. J. Michael**, A. Mark, S. Sasic, H. Ström, *Generalized Langevin dynamics in multiphase direct numerical simulations using hydrodynamically optimized memory kernels*
Physics of Fluids 37 (2025) 033317.
DOI: https://doi.org/10.1063/5.0254930

[**Paper B**] **A. J. Michael**, A. Mark, S. Sasic, H. Ström, *A nonlocal multiscale model for Brownian particles: application to hindered deposition in microfluidic systems*
Submitted to a scientific journal.

Conferences

Parts of this conference have been presented at the following conferences:

[**Conference I**] Swedish Mechanics Day, *2024*
Gothenburg, Sweden.

[**Conference II**] International Conference on Numerical Methods in Multiphase Flows, *2024*
Reykjavik, Iceland.

[**Conference III**] International Conference on Multiphase Flows, *2025*
Toulouse, France.

Acknowledgment

I want to begin by thanking my supervisor, Henrik Ström, for all the guidance he has provided me so far in this project. It has been great to know you have my back when I feel stuck with a problem. This work would not have been possible without my co-supervisor, Andreas Mark, whose insight has been especially helpful when it comes to `IBUFlow`. I want to extend my gratitude to my examiner, Srdjan Sasic, for all the advice and support he has provided through the journey so far. I want to thank the Swedish Research Council (Vetenskapsrådet) for funding this project via grant agreement no 2021-05175. The computations required for this work were enabled by resources provided by the National Academic Infrastructure for Supercomputing in Sweden (NAISS), partially funded by the Swedish Research Council (Vetenskapsrådet) through grant agreement no. 2022-06725.

I would like to thank my colleagues at the division for the great work environment you have provided. A special shoutout to Akhilesh for all the chats within and outside the department. I would also like to thank all my friends, old and new, who have helped make Gothenburg a new home for me. A special mention goes to my cousin Vineeth, who has heard the best and worst of me over many late-night phone calls. Last but not least, I would like to thank my parents and my sister for believing in me and supporting me all this time that I have been away from home.

Anand Joseph Michael

Contents

Part I

Summary

Chapter 1

Introduction

A particle immersed in a fluid experiences frequent, incessant collisions from the kinetic fluctuations of fluid molecules in its surroundings. These collisions cause the particle to move in a random fluctuating manner referred to as Brownian motion, and the particle undergoing such a fluctuating motion is often referred to as a Brownian particle[1] . Brownian motion causes particles to diffuse through the fluid in directions that are not always aligned with the convective currents of the fluid, leading them to affect the behaviour of many microfluidic systems profoundly.

Many biological and physicochemical phenomena at the microscale rely on directing the Brownian behaviour of particles immersed in fluids. Artificial Brownian motors are an example of such a device that makes use of external input signals or spatial asymmetries to direct the motion of Brownian particles immersed in the fluid flowing in the device [2] . Concentration gradients in the systems are another way to direct motion[3]–[5] . Alternatively, electric, magnetic or gravitational forces can also be utilised to pull the particle in a particular direction[6]–[10] . All such devices require an optimal balance between different design parameters to allow for specific Brownian particle motion at the microscale. While experimental techniques are being improved upon to analyse microfluidic phenomena, the minute time and length scale associated with Brownian phenomena still pose challenges in acquiring accurate and detailed results[11] . Simulations, then, become the alternative choice for analysing these microfluidic phenomena.

Over the years since its discovery, many simulation techniques have been developed to model Brownian motion. Since the jittery motion associated with the Brownian motion has a molecular origin, the most direct approach to modelling Brownian motion is molecular dynamics[12]–[14] . In molecular dynamics approaches, Newtonian or Hamiltonian mechanics is used to recreate the fluctuating motion associated with molecules or clusters of molecules[13] . Various force models are used to create interparticle interactions, often modelled as Lennard-Jones potential forces. While these simulations are accurate, the small time scale required to account for molecular interactions along with the computational cost associated with multiparticle simulations often limit the size

and duration of phenomena that can be modelled using molecular dynamics[14] .

An alternative approach is to model Brownian motion from a continuum perspective using a Langevin description[15] . Here, the molecular collisions associated with the Brownian phenomena can be considered to give rise to two continuum scale forces. A random Brownian impulse force that generates the fluctuating motion associated with a Brownian particle and a hydrodynamic friction force that resists the relative motion between the particle and the surrounding fluid[15], [16] . Models based on the Classical Langevin Equation (CLE) rely on such a description of Brownian motion to model Brownian phenomena. The hydrodynamic force in this case is usually the Stokes drag associated with the Brownian particle, often modelled as a quantity proportional to the current velocity of the particle[17]–[19] . The common molecular origin for the hydrodynamic force and the random Brownian force allows for the strength of the hydrodynamic force to be related to the covariance of the random Brownian force using a relationship known as the fluctuation-dissipation relationship. Based on this relationship, the CLE models the Brownian force as a random Markovian white noise process. Stokesian or Brownian dynamic simulations rely on such a CLE based model to carry out Brownian motion simulations in larger domains with multiple particles incorporated. However, these simulations are often restricted to high particle-fluid density ratios. At low particle-fluid density ratios, in addition to the instantaneous Stokes drag, the added mass and history effects also start to play a major role in the hydrodynamic resistance experienced by the Brownian particle[20]–[22] . The Markovian white noise model of the Brownian force is no longer valid, and hence, the CLE approaches are not accurate at smaller particle-fluid density ratios (close to neutrally buoyant), which are of relevance in many microfluidic devices[2] .

An alternative Langevin description is the generalised Langevin equation (GLE), where the hydrodynamic force acting on the particle includes the relevant added mass and history effects. The net hydrodynamic force in this case is often modelled as a convolution integral between a memory kernel and the velocity history of the particle. The fluctuation-dissipation relationship in this case shows that the covariance between the Brownian forces is proportional to the memory kernel and therefore acquires a non-Markovian colored nature[1], [20], [21] . The memory kernel thus becomes a continuum representation of the molecular phenomena that generates the hydrodynamic force and the random Brownian forces required in GLE based simulations. The memory kernel required for such simulations can be determined from molecular dynamics simulations, where the Mori-Zwansig projection formalism is used to derive the memory kernels from molecular correlations[23], [24] . Volterra equation inversion techniques and deep learning methods are used to derive the memory kernels from such molecular simulations[25], [26] . These simulations, which rely on the memory kernel to create continuum approximations from molecular dynamics, are often referred to as coarse-graining techniques[24], [27] . Since the GLE is already a continuum-scale equation, it is alternatively possible to derive the memory kernel from the hydrodynamic force directly using continuum-scale

Direct Numerical Simulations (DNS) of the particle motion. The continuum scale DNS approach allows us to simulate larger domains that are harder to incorporate in molecular simulations while accounting for all the hydrodynamic interactions the particle undergoes, especially when low particle-fluid densities are involved. It also opens up the possibility to utilise well-developed multiphase simulation techniques to simulate multiparticle and multifluid interactions that might be required to simulate complex microfluidic flow fields, possibly also incorporating particle reaction with the surrounding.

In this work, a pure DNS method is presented that determines the memory kernel directly from the hydrodynamic force experienced by the particle[28] . The hydrodynamic force includes the effects of the fluid properties (viscosity and density relative to the Brownian particle), the particle geometry and the neighbouring domain (including other particles and walls). Hence, the memory kernel derived in such an approach already contains information regarding changes to the drag, added mass and history effect, providing for an accurate modelling of the Brownian phenomena required to study microfluidic applications at low particle-fluid density ratios. The method is shown to be able to accurately capture Brownian particle dynamics in unhindered and hindered flow cases. In order to extend the GLE based approach to more complex microfluidic simulations, a lower-cost multiscale model that derives hydrodynamic memory kernels using short DNS simulations and then runs LPT simulations based on them is also presented. In many microfluidic applications, the Brownian particle moves towards a wall under the influence of various guiding forces[3], [9], [10] . The multiscale method is used here to analyse such systems by simulating the migration of a Brownian particle towards a wall. The effects that the domain size and strength of the attractive force have on the impact region of the particle are discussed to show how the simulations can be used to optimise such systems. In both these methods, the shape of the memory kernel developed as part of the simulations can be used as a "fingerprint" capturing all the various hydrodynamic effects felt by the Brownian particle in a particular flow scenario.

Chapter 2

Aim

The statistical characteristics of Brownian motion, if understood better, could be utilised to achieve desired results in microfluidic applications. Three main factors could influence the statistical nature of the Brownian motion of a particle. The nature of the fluid surrounding the particle (i.e. its viscosity and its density in relation to the particle) determines the strength and non-local nature of the hydrodynamic hindrance that the Brownian particle experiences. Another major factor is the geometry of the domain surrounding the particle (including the presence of other particles), which can hamper the Brownian oscillations in certain directions. Finally, the shape of the particle itself can influence the Brownian motion in different directions. Through this work, simulation techniques, based on a Langevin description of Brownian motion, are developed to accurately model Brownian motion and the effects on it due to variations in these factors. The major objectives of the work can then be described as:

G1: Develop accurate simulation techniques for Brownian motion capable of capturing the complete hydrodynamic effects on the particle that are valid at low particle-fluid density ratios

G2: Utilise the developed simulation techniques to study the effects of fluid properties, domain geometry and particle shape on the optimal functioning of microfluidic devices

The rest of the thesis aims to show the methods developed so far and their capabilities. Chapter 3 of the thesis contains a discussion of the theory involved in the Langevin description of Brownian motion, which is then used in all the numerical techniques presented in chapter 4 . After this, the key results obtained using the simulation methods are presented in chapter 5 . Finally, the key conclusions obtained as part of this work are presented in chapter 6, which also includes an outlook on the future directions this work can take.

Chapter 3

Modelling Brownian motion

The molecular description of Brownian motion can be utilised to create a continuum description of the particle trajectory. If the fluid in which the Brownian particle is immersed is considered as a continuum, then the effects of the molecular collisions can be modelled using two major forces. On the one hand, some of the molecular collisions give rise to random impulses that cause the meandering path that is so characteristic of Brownian motion. On the other hand, some of the collisions also resist the relative motion of the particle with respect to the fluid[16] . The Langevin description of Brownian motion relies on this understanding of the effects of molecular collisions to represent these phenomena using two competing forces that affect the particle motion: a random Brownian force to represent the impulsive pushes and a hydrodynamic friction force to represent resistance to relative motion[15] . The Langevin model thus uses Newtonian mechanics to model the trajectory of a Brownian particle as the result of a hydrodynamic friction force and a random Brownian force. The classical Langevin equation (CLE), valid at high particle-fluid density ratios is presented first here. This is followed by the generalised Langevin equation (GLE), which also includes added mass and history effects into the hydrodynamic force, allowing for a more accurate representation of Brownian motion at low particle-fluid density ratios.

3.1 The Classical Langevin Description

In the classical Langevin equation (CLE), the hydrodynamic force is modelled as a drag force ($\mathbf{F_D}$) that is proportional to the velocity ($\mathbf{V}$) of the particle [15], [20], [29]:

$$\begin{aligned} m_p \frac{d\mathbf{V}(t)}{dt} &= \mathbf{F_D}(t) + \mathbf{F_B}(t), \\ &= -\gamma \mathbf{V}(t) + \mathbf{F}(t). \end{aligned} \tag{3.1}$$

Here, m_p refers to the mass of the particle while γ is the drag coefficient allowing for the representation of the hydrodynamic drag force as a function of

the velocity at time t. For a spherical particle, the drag coefficient is given as $\gamma = 6\pi\mu r_p$, where μ is the viscosity of the fluid and r_p is the particle radius[30] . The force $\mathbf{F_B}$ is the random Brownian force that changes the direction of the particle's motion.

Due to the common molecular origin of the two forces, it is possible to derive a relation between the fluctuating random forces and the dissipative hydrodynamic force, commonly referred to as the fluctuation-dissipation relation. For the classical Langevin equation, the corresponding fluctuation-dissipation relation is [1], [20], [29]:

$$\langle F_{Bi}(t-\tau)F_{Bj}(t)\rangle = 2k_B T\gamma\delta_{ij}\delta(\tau). \tag{3.2}$$

In this equation, k_B is the Boltzmann constant, T is the absolute temperature of the surrounding fluid, and τ is the lag time between the two Brownian forces. The fluctuation-dissipation relationship thus states that the Brownian forces at two time instances separated by a lag time τ are uncorrelated, as represented by the Dirac delta function $\delta(\tau)$ in equation 3.2 . There is also no relationship between the random forces in two orthogonal directions, as shown by the Kronecker delta δ_{ij}, where the subscripts i and j represent any orthogonal directions in the coordinate system. Due to this relationship, the random Brownian force in the CLE can be modelled as a Markovian (memoryless) white noise process with a variance proportional to the drag coefficient of the Brownian particle, as shown in equation 3.2 [19], [31], [32] .

Due to the random nature of Brownian motion, it is common to study the statistical behaviour of the Brownian particle rather than individual instances of the particle trajectory. Two such common statistical properties are the Velocity Auto-Correlation Function (VACF or C_V) and the Mean Square Displacement (MSD).

The VACF is a measure of how quickly the velocity of the particle decorrelates over time. From the CLE, the VACF can be shown to be [20]:

$$\frac{C_V(\tau)}{V_{RMS}^2} = \frac{\langle V(t)V(t+\tau)\rangle}{\langle V(t)^2\rangle} = \exp\left(-\frac{\tau}{\tau_p}\right). \tag{3.3}$$

The particle response time, $\tau_p = m_p/\gamma$, is used to characterise the time taken by a particle to respond to external stimuli[1], [20] . For a spherical particle, the particle response time then becomes $\tau_p = \frac{2}{9}r_p^2\rho_p/\mu$. At thermal equilibrium, the equipartition theorem can be used to derive the root mean square velocity (RMS) as $V_{RMS} = \langle V(t)^2\rangle = dk_BT/m_p$, where d is the number of dimensions considered in the model[1] . The particle diffusivity in this case can be obtained in this case as $D = \int_0^\infty C_V d\tau = k_bT/\gamma$.

The MSD is a measure of how far the particle has diffused or moved in a certain period of time. From the CLE, the MSD can be shown to be equal to [20]:

$$\langle [X(t+\tau) - X(t)]^2\rangle = 2dD\left\{\tau - \tau_p\left[1 - \exp\left(\frac{-\tau}{\tau_p}\right)\right]\right\}. \tag{3.4}$$

As the lag time (τ) used to measure particle diffusion tends to infinity, the MSD tends to the classical relationship derived by Einstein [33], i.e. $\langle[X(t+\tau)-X(t)]^2\rangle = 2dD\tau$. The CLE thus allows for short-term ballistic motion while also capturing the long-term diffusive behaviour characteristic of Brownian particles.

3.2 The Generalised Langevin Description

While the CLE provides a good continuum description of Brownian motion, there are some issues associated with it. The various phenomena that are involved in the continuum description of an immersed particle's motion and their associated characteristic time scales give an indication of the deficiencies of the classical Langevin description. Although the CLE considers the instantaneous Stokes drag proportional to the particle's current velocity, it does not consider the inertia of the fluid in the vicinity of the particle, nor the viscous memory effects associated with the boundary layer development [1], [22], [28] . The viscous effects, in particular, propagate over the particle on a time scale of $\tau_f = r_p^2/\nu$ (where $\nu = \mu/\rho_f$ is the kinematic viscosity of the fluid). When the particle density is much larger than the fluid density, the viscous history effects become inconsequential to the particle dynamics since they operate at time scales much smaller than the particle response time (i.e. $\tau_p/\tau_f = \frac{2\rho_p}{9\rho_f} >> 1$). At such high particle-fluid density ratios, the CLE, devoid of any history effect considerations, is an accurate continuum representation of Brownian phenomena. On the other hand, when the particle density is quite close to the fluid density, the fluid inertia as well as the history effects become more prominent. The characteristic fluid time scale is then quite close in magnitude to the particle response time (i.e. $\tau_p/\tau_f = \frac{2\rho_p}{9\rho_f} \approx 1$), and an accurate modelling of Brownian phenomena then requires the capture of the unsteady drag effects brought about by the fluid inertia and history effects. Molecular simulations have shown that the VACF for a Brownian particle shows a decay proportional to $t^{-3/2}$ instead of the exponential decay predicted by the CLE[12] . Experimental works have also shown that the Brownian force experienced by a particle has a coloured nature to it in contrast to the white noise process predicted by the CLE[34], [35], indicating the requirement for a modified modelling approach that can capture these differences.

The generalised Langevin equation (GLE) provides an alternative description of Brownian motion that captures the unsteady drag effect as shown in equation 3.5[21], [36] . The total hydrodynamic force is represented in this description using a convolution integral between a memory kernel ($\mathbf{K}(t,\tau)$) and the velocity history of the particle. In addition to the Stokes drag, the hydrodynamic friction force here incorporates the added mass and history forces to account for the fluid inertia and viscous memory effects, respectively [1] .

$$\begin{aligned} m_p \frac{d\mathbf{V}(t)}{dt} &= \mathbf{F_H}(t) + \mathbf{F_B}(t) \\ &= -\int_0^t \mathbf{K}(t,\tau) \cdot \mathbf{V}(t-\tau) d\tau + \mathbf{F_B}(t). \end{aligned} \tag{3.5}$$

The fluctuation-dissipation relationship that depicts the evolution of the covariance of the random force then changes to [21], [29], [37]:

$$\langle F_{B,i}(t-\tau) F_{B,j}(t) \rangle = k_B T K_{ij}(t,\tau). \tag{3.6}$$

The random Brownian forces at two time instances separated by a lag time of τ are no longer uncorrelated. Instead, they show covariance that is proportional to the memory kernel value associated with that time lag. The Brownian force is thus a non-Markovian coloured process that has memory effects embedded into its covariance as predicted from experiments[34], [35] .

The hydrodynamic friction force in a single direction contains memory effects from all three orthogonal directions. The memory kernel value at a particle time lag is a 3×3 tensor which allows for the particle motion in the three directions to influence the hydrodynamic force in a particular direction via the convolution integral with the velocity histories in the three directions (valid at $Re << 1$)[24], [38]:

$$F_{H,i}(t) = -\int_0^t K_{ij}(t,\tau) V_j(t-\tau) d\tau. \tag{3.7}$$

The memory kernel in this modelling approach serves two purposes, the first of which is to relate the history of the particle to the hydrodynamic force in each direction. The hydrodynamic force acting on a particle is influenced by the properties of the fluid surrounding the particle (i.e. its viscosity and density relative to the particle), the geometry of the particle and the geometry of the domain in which the fluid flows. Since the kernel models the hydrodynamic force, its values in turn represent the influences of these factors on the particle trajectory. Thus, the kernel can be thought of as a "fingerprint" of the particle and its surroundings obtained via the hydrodynamic force[21], [28] .

Since, the simulations considered in this work utilise the same particle-fluid combinations within a single simulation, the influence due to the fluid properties with respect to the particle as well as the particle shape can be considered to be fixed for a single simulation. The possible cause for the change in the memory kernel values is then the domain features, which influence the hydrodynamic force depending on the position of the particle relative to the domain during the course of a simulation run. Thus, memory kernel values depend on the particle's position in the domain at time t. At the time instance t, the total hydrodynamic force on the particle is influenced to different degrees by the velocity history of the particle at different lag times τ in the past. Thus, the memory kernel is represented as the function of two factors, the current time instance t and the lag time τ, leading to the representation $K(t,\tau)$.

The second purpose of the kernel is to model the evolution of the covariance of the random Brownian force via the fluctuation-dissipation relationship. The memory kernel links the hydrodynamic force to the random force. Knowledge of its shape allows for the generation of random Brownian forces that follow the required covariance pattern expected from equation 3.6 .

Similar to the CLE, it is possible to derive the VACF associated with a particle in this scenario. Mainardi et al. derived the VACF for an unhindered spherical particle[20] . Here, unhindered implies that there are no domain constraints in the neighbourhood of the particle that could influence its hydrodynamic behaviour. In this case, the added mass effect causes some amount of fluid to be dragged along with the particle which leads to the effective mass of the paritcle becoming $m_e = m_p(1 + \rho_f/2\rho_p)$, which in turn modifies the particle response time to $\tau_e = \tau_p(1 + \rho_f/2\rho_p)$. The ratio between the fluid response time and the modified particle response time is expressed as $\beta = \tau_f/\tau_e = 9\rho_f/(2\rho_p + \rho_f)$. The RMS velocity is modified to include the added mass effect as $V_{RMS} = dk_BT/m_e$. The analytical value of the VACF then becomes:

$$\frac{C_V(\tau)}{V_{RMS}^2} = \frac{\langle V(t)V(t+\tau)\rangle}{\langle V(t)^2\rangle} = \left\{ \frac{a_+E_{1/2}(a_+\sqrt{\tau}) - a_-E_{1/2}(a_-\sqrt{\tau})}{a_+ - a_-} \right\}. \quad (3.8)$$

Here, $E_{1/2}(a\sqrt{t}) = e^{a^2t}\text{erfc}(-a\sqrt{t})$ is the Mittag-Leffler function of order $1/2$ with erfc representing the complementary error function. On the other hand, $a_\pm = (-\sqrt{\beta} \pm (\beta - 4)^{1/2})/(2\sqrt{\tau_e})$.

Thus, the VACF no longer decays exponentially as predicted by the CLE. In the long term, the VACF decay instead shows a slope proportional to $t^{-3/2}$ as observed from molecular dynamics [12] simulations, showing that the GLE is a more accurate representation of the Brownian motion.

The MSD calculated by Mainardi et al. for the spherical unhindered Brownian particle is:

$$\langle [X(t+\tau) - X(t)]^2\rangle = 2dD\left\{ \tau - 2\sqrt{\frac{\beta\tau_e\tau}{\pi}} + \frac{a_+^3[1 - E_{1/2}(a_-\sqrt{\tau})] - a_-^3[1 - E_{1/2}(a_+\sqrt{\tau})]}{(a_+ - a_-)(a_+a_-)^2} \right\}. \quad (3.9)$$

The MSD in this case also reduces to the Einstein pure diffusion expression of $\langle [X(t+\tau) - X(t)]^2\rangle = 2dD\tau$ at large lag times. Thus, the GLE gives a more accurate VACF and MSD trend compared to the CLE due to its comprehensive modelling of hydrodynamic memory effects and the associated coloured nature of the Brownian force.

3.3 Hydrodynamic Force on a Particle

The Langevin description, both the classical version and the generalised version requires the hydrodynamic force acting on the particle. The hydrodynamic force on the particle can be determined in different ways depending upon on the particle shape and the surrounding domain.

In the case of an unhindered spherical particle, there are mathematical models available for the representation of the hydrodynamic force on the particle [1], [20], [39]:

$$\begin{aligned}\mathbf{F_H}(t) = \quad & -\gamma \mathbf{V}(t) - \frac{2}{3}\pi r_p^3 \rho_f \frac{d\mathbf{V}(t)}{dt} \\ & -\gamma r_p \int_0^t \frac{d\mathbf{V}(\tau)}{d\tau} \frac{d\tau}{\left[\pi\mu(t-\tau)/\rho_f\right]^{1/2}}.\end{aligned} \tag{3.10}$$

The first term on the right represents the steady Stokes drag that is proportional to the instantaneous particle velocity. The second term represents the effects of the fluid inertia as an added mass force proportional to the acceleration of the particle, while the third term is a history force that accounts for the viscous memory effects.

For a particle that is non-spherical or experiences hindrance from the presence of domain boundaries in its vicinity, the derivation of such a mathematical model is not always straightforward. In such cases, the surface integral of the pressure (p) and viscous stresses (τ_{ij}) exerted on the particle by the surrounding fluid is a more general way of determining the hydrodynamic force on the particle [16]:

$$\begin{aligned}\mathbf{F_H}(t) &= \int_S (-p\delta_{ij} + \tau_{ij}) n_j dS, \\ &= \int_S \left(-p\delta_{ij} + \mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)\right) n_j dS.\end{aligned} \tag{3.11}$$

where u represents the velocity of the fluid. To determine the pressure and stress fields around the particle, it is necessary to solve the Navier-Stokes equations:

$$\frac{\partial u_j}{\partial x_j} = 0, \tag{3.12}$$

$$\rho_f \left(\frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j}\right) = -\frac{\partial p}{\partial x_j} + \mu \frac{\partial^2 u_i}{\partial x_j^2} + f_i, \tag{3.13}$$

where f_i represents any external source term. The pressure and stress fields developed using the Navier-Stokes equations inherently contain information about the fluid properties, particle shape and the presence of hindrance in the flow field. Simultaneously, the surface integral representation of the hydrodynamic force incorporates the drag, added mass and history effects into

the hydrodynamic force without the need for specific mathematical models to represent these subcomponents of the hydrodynamic force. Thus, the hydrodynamic force model shown in equation 3.11, is a generic model that can capture changes to the drag, added mass and history effects brought about by variation in fluid properties, particle shape or domain geometry.

The hydrodynamic force as previously shown in equation 3.7 can also be modelled as the convolution integral between a memory kernel and the velocity history associated with the particle trajectory. This representation of the hydrodynamic force also accounts for the drag, added mass and history effects. Any changes to the strength of these effects from variations in fluid property, particle shape and domain changes via changes in the memory kernel values. Thus, the surface integral and the convolution integral represent the same hydrodynamic force[21]:

$$\mathbf{F_H} = \int_S (-p\delta_{ij} + \tau_{ij}) n_j dS = -\int_0^t \mathbf{K}(t,\tau) \cdot \mathbf{V}(t-\tau) d\tau \tag{3.14}$$

The interchangeable forms of the hydrodynamic force will be utilised along with the GLE to simulate the motion of a Brownian particle in the upcoming sections.

Chapter 4

Numerical Methodology

Two novel methods simulating Brownian motion are presented in this section, both of which are based on the GLE presented in the previous sections. The first method is a Direct Numerical Simulation (DNS) method that utilises the Navier-Stokes equations to solve the flow field around a particle and then simulating the particle motion based on the GLE [28] . The second method is a multiscale approach that relies on short DNS to develop memory kernels that are then utilised in a Lagrangian Particle Tracking (LPT) approach based on the GLE to evolve the particle trajectory [40] . The second approach allows for the study of Brownian phenomena in more complex flow scenarios, incorporating the nonlocal flow effects, such as history effect, at a lower cost compared to the pure DNS approach. The first section in this chapter will focus on the DNS approach, while the second section will focus on the multiscale approach.

4.1 Direction Numerical Simulations of Brownian Motion

The DNS method is based on the GLE and its associated fluctuation-dissipation relationship (equations 3.5 and 3.6 respectively). These equations require the knowledge of the hydrodynamic force acting on the particle, the associated memory kernel and random coloured Brownian forces that follow the covariance relationship depicted in the fluctuation-dissipation equation. The main steps associated with the simulations then are:

1. Obtain the hydrodynamic force on the Brownian particle

2. Optimise associated memory kernel values utilising previously obtained hydrodynamic force and velocity history of the particle

3. Generate coloured random Brownian forces based on the optimised memory kernel

4. Evolve particle trajectory using GLE by employing the obtained hydrodynamic force and generated Brownian force

The steps outlined here will be explained in detail in the upcoming sections.

4.1.1 Hydrodynamic Force

The hydrodynamic force ($\mathbf{F_H}$) in the DNS appraoch has been obtained by numerically solving the Navier-Stokes equation for the fluid flow around the particle. In this work, the mirroring Immersed Boundary Method has been applied along with an appropriate Fluid Structure Interaction model (IB-FSI) to obtain the required velocity and pressure fields needed for the hydrodynamic force calculations[41], [42] . The Immersed Boundary (IB) method uses the implicit Dirichlet IB condition of $u_i = u_i^{IB}$. The mirroring IB method mirrors the velocity over the IB to cells on the interior of the particle such that linear interpolation between the mirrored velocity and the external point gives the velocity at the boundary, thereby enforcing the IB condition. The coefficients obtained while discretising the Navier-Stokes equations are closed using the implemented IB condition. The fictitious velocity generated in the interior of the boundary is replaced with the IB velocity in all flux calculations to ensure fulfilment of the continuity equation.

While the IB-FSI method has been implemented using the `IBOFlow` software here, alternate multiphase simulation techniques can also be utilised to obtain flow fields. Once the required flow quantities are obtained, the surface integral of the pressure and viscous stresses can be utilised to obtain the total hydrodynamic force on the particle as shown in equation 3.11 .

4.1.2 Optimising Memory Kernel

The memory kernel ($\mathbf{K}(t,\tau)$) connects the hydrodynamic force to the random Brownian force via the fluctuation-dissipation relationship. Hence, knowing the kernel values gives a way of generating the temporally correlated random Brownian force. The memory kernel value at a certain lag time is a 3×3 tensor that represents the influence of the history of particle motion in all directions on the force in a particular direction as shown in equation 3.7 . The tensor representation of the memory kernel at time t and lag time τ is shown in equation 4.1 . When the kernel is used in the fluctuation-dissipation relationship shown in equation 3.6, it depicts the auto-covariance (given by the diagonal terms in the tensor, i.e. K_{ii}) and the cross-covariences (given by the off-diagonal terms in the tensor, i.e. K_{ij} where $i \neq j$) of the random Brownian forces[21] .

$$\mathbf{K}(t,\tau) = \begin{bmatrix} K_{11}(t,\tau) & K_{12}(t,\tau) & K_{13}(t,\tau) \\ K_{21}(t,\tau) & K_{22}(t,\tau) & K_{23}(t,\tau) \\ K_{31}(t,\tau) & K_{32}(t,\tau) & K_{33}(t,\tau) \end{bmatrix} \tag{4.1}$$

The memory kernels have been proven to be symmetric, reducing the number of independent terms to be determined in the kernel[21] . Also, the simulations

performed as part of this work utilise spherical particles, which allow for certain simplifications to the memory kernel tensor. From the symmetry of the particle and flat walls used in the simulations, it can be assumed that the non-diagonal terms of the memory kernel are negligibly small[43], [44] . This allows for the approximation that $K_{ij} = K_{ii}\delta_{ij}$. In the case of an unhindered particle, the memory kernel has been shown to reduce to a single term represented by $K(t,\tau)$ as all the diagonal terms take on the same value [21], [29] . For the hindered case with flat walls in the vicinity of a spherical particle, the kernel can be split into components that are normal ($\perp$) to the surface and parallel ($\parallel$) to the surface.

The memory kernel is obtained by an optimisation routine that makes use of the hydrodynamic force acting on a particle and the associated velocity history of the particle. The approximations mentioned earlier keep the optimisation routine used to obtain the memory kernel relatively simple and reduces numerical noise that can creep into the optimised kernel values. The process presented here, however, can be extended to account for cross-covariance between the Brownian forces in different directions by extending the optimisation routine to obtain the off-diagonal terms of the memory kernel. The simplification of the memory kernel transforms equation 3.7, into the simplified form:

$$F_{H,i} = -\int_0^t K_i(t,\tau)V_i(t-\tau)d\tau. \tag{4.2}$$

Similarly, the fluctuation-dissipation relationship also reduces to:

$$\langle F_{B,i}(t-\tau)F_{B,j}(t)\rangle = k_B T K_{ii}(t,\tau)\delta_{ij}. \tag{4.3}$$

In order to determine the values of the memory kernel at different lag times, the convolution integral form of the hydrodynamic force shown in equation 4.2 is used. The cost function is created by taking all the terms in equation 4.2 to one side and then dividing the whole quantity so obtained by the total hydrodynamic force as shown:

$$C(K_i(t,\tau)) = \frac{|F_{H,i}(t) + \int_0^t K_i(t,\tau)V_i(t-\tau)d\tau|}{|F_{H,i}(t)|}. \tag{4.4}$$

Here, the hydrodynamic force ($F_{H,i}$) on the particle in a particular direction is the force obtained via the surface integral of the pressure and stress fields obtained from the multiphase simulation of the particle as mentioned in section 4.1.1 . The second term in the numerator, i.e. the convolution integral between the memory kernel ($K_i(t,\tau)$) and the velocity ($V_i(t)$) history of the particle in the same direction, also represents the hydrodynamic force as shown in equation 4.2 . Thus, the numerator ideally sums up to zero. The goal of the optimisation routine is to modify the terms of the memory kernel until the difference between the two forms of the hydrodynamic force in the numerator becomes negligibly small. Since the simulation utilises discrete time steps to evolve the particle trajectory, the memory kernel and the velocity history are

also discretised to convert the convolution integral into a summation. The final discretised form of the cost function then becomes:

$$C(K_i(t,\tau)) = \frac{|F_{H,i}(t) + \sum_{n=0}^{N} K_i(t, n\Delta t) V_i(t - n\Delta t)\Delta t|}{|F_{H,i}(t)|}. \tag{4.5}$$

Here, N is the total number of values in the discredited memory kernel while Δt is the size of the time step used in the simulation.

In this work, the Broyden–Fletcher–Goldfarb–Shanno optimisation routine (`L-BFGS-B`) [45]–[47] is implemented by utilising the `scipy` library from `python` [48], to optimise the memory kernel values. At every time step, the whole discretised memory kernel is passed on to the optimisation function along with the associated velocity history corresponding to each term in the memory kernel. The optimisation routine iteratively modifies all the values of the discretised memory kernel until the cost function is minimised below a limit (set to 10^{-6} in this work). At the first time step of the simulation, the memory kernel starts with a single value associated with lag time zero, and then at each new time step, a new value is added to the kernel, thereby increasing the total number of terms in the discretised memory kernel. Every new term added to the kernel is initialised with the value zero, while the remaining terms retain the value obtained for that lag time after optimisation at the previous time step. Thus, although the optimisation routine modifies the kernel values at all lag times during iterations of the optimisation routine, all the terms except for the new term added at the current step begin from a good initial guess value obtained from the previous time step. This is based on the assumption that the change in particle position from one time step to the next does not significantly change the hydrodynamic neighbourhood of the particle, and hence the associated memory kernel, dependent in this case only on the domain features, would not undergo very drastic changes. Therefore, it can be assumed that the values obtained at the previous time step for the kernel would be quite close to the expected values in this time step, allowing for quicker convergence to the final optimised value of the kernel. The whole optimisation process is repeated using the forces and velocity histories in each direction to obtain the optimised memory kernels associated with each direction.

Since the covariance of the Brownian forces is proportional to the memory kernel, the covariance matrix created for the Brownian forces from the kernel values must be positive semi-definite (PSD)[24] . The PSD condition can be enforced by utilising the Trust Region method that allows for the addition of constraints to the optimisation routine. The `trust-constr` routine in the `scipy` library allows for this optimisation route. However, the enforcement of the constraint can considerably increase the optimisation time. So in the simulations presented, the `L-BFGS-B` routine is utilised first without constraints, and then in case the covariance matrix produced with the kernel so obtained is not PSD, then the constrained routine is applied for further optimisation. It can also be noted here that the size of the memory kernel is usually restricted to contain values up to a lag time of $10\tau_P$, so that the computational cost on the simulation

is minimised. Beyond this point, the kernel values are seen to have negligible effects on the hydrodynamic force (since $|\mathbf{K}(t, \tau > 10\tau_p)|/\max|\mathbf{K}(t,\tau)| < 10^{-5}$). The memory kernel obtained via the optimisation routine detailed here is then utilised to generate the Brownian force necessary to complete the GLE.

4.1.3 Generating Colored Brownian Force

The generation of the random colored Brownian force ($\mathbf{F_B}$) relies on the simplified fluctuation-dissipation relationship shown in equation 4.3 . This relationship can be expanded to show the covariance matrix ($N \times N$)of the Brownian forces over different lag times using the discretised memory kernel (containing N terms):

$$\begin{aligned} Cov(F_{B,i})_{N,N} &= [\langle F_{B,i}(t - n\Delta t)F_{B,i}(t)\rangle]_{N,N}, \\ &= k_B T[K(t, n\Delta t)]_{N,N}, \\ &= k_B T \begin{bmatrix} K(t,0) & K(t,\Delta t) & \dots & K(t,N\Delta t) \\ K(t,\Delta t) & K(t,0) & \ddots & \vdots \\ \vdots & \ddots & \ddots & K(t,\Delta t) \\ K(t,N\Delta t) & \dots & K(t,\Delta t) & K(t,0) \end{bmatrix}. \end{aligned} \tag{4.6}$$

The variance of the Brownian forces is represented by the diagonal terms of the covariance matrix, while the off-diagonal terms represent the covariance between Brownian forces separated by a lag time $n\Delta t$. Since the covariance matrix satisfies the PSD condition, the Cholesky decomposition can be used to factorise the matrix into root matrices represented by a lower triangular matrix ($L_{N,N}$) and its transpose($L^T_{N,N}$):

$$Cov(F_{Bi}) = L_{N,N}L^T_{N,N}. \tag{4.7}$$

The lower triangular matrices are then multiplied by N normally distributed random numbers ($\mathcal{N}$) with zero mean and unit variance to obtain N random colored numbers that obey the covariance scheme laid out in equation 4.6 as shown:

$$[F]_{N,1} = [L]_{N,N}[\mathcal{N}(0,1)]_{N,1}. \tag{4.8}$$

The random Brownian force required for the GLE-based simulation is then chosen as the last number in the colored random number series:

$$\mathbf{F}_{B,i}(t) = [F]_{N,1}[N]. \tag{4.9}$$

The normal random number series used to generate the colored force is cycled through by deleting the first number and adding a new random number at the end of the series at each time step. This ensures that the colored random number generated from it at each time step has a unique value while following the fluctuation-dissipation relationship.

4.1.4 Evolution of Particle Trajectory

Once the forces required for the GLE, namely the hydrodynamic force ($\mathbf{F_H}$) and the random Brownian force ($\mathbf{F_B}$) is determined, the particle trajectory can be evolved using the time stepping scheme presented here. The net acceleration acting on the particle can be obtained by adding the hydrodynamic force and the random Brownian force, and then dividing by the particle mass (m_p) as shown:

$$\mathbf{a}(t) = \frac{\mathbf{F}_H(t) + \mathbf{F}_B(t)}{m_p}. \tag{4.10}$$

The acceleration so obtained is then used in the IB-FSI framework to update the particle velocity ($\mathbf{V}$) and position ($\mathbf{X}$) at the next time step, using an implicit Newmark time stepping scheme [49]:

$$\mathbf{V}(t + \Delta t) = \mathbf{V}(t) + \frac{\Delta t}{2} \left[\mathbf{a}(t) + \mathbf{a}(t + \Delta t)\right], \tag{4.11}$$

$$\mathbf{X}(t + \Delta t) = \mathbf{X}(t) + \mathbf{V}(t)\Delta t + \frac{\Delta t^2}{2}\mathbf{a}(t). \tag{4.12}$$

Here, the term $\mathbf{a}(t + \Delta t)$ refers to the implicit acceleration of the particle that is iteratively obtained. The implicit acceleration associated with time instance $t + \Delta t$ is initialised with a guess value and used to move the particle to its future position using equations 4.11 and 4.12 . The hydrodynamic force at this new position is calculated and used in determining the implicit acceleration $\mathbf{a}(t + \Delta t)$ at this position according to equation 4.10, which is compared with the previous guess value. The new value is taken as the guess value for the next step of the iteration until both the guess and new values converge to approximately the same values (i.e. their threshold drops below a certain value).

Alternatively, explicit time stepping methods can also be used to evolve the particle velocity and position:

$$\mathbf{V}(t + \Delta t) = \mathbf{V}(t) + \Delta t\mathbf{a}(t), \tag{4.13}$$

$$\mathbf{X}(t + \Delta t) = \mathbf{X}(t) + \mathbf{V}(t)\Delta t + \frac{\Delta t^2}{2}\mathbf{a}(t). \tag{4.14}$$

The IB-FSI framework used to carry out the DNS method employs the implicit Newmark scheme, while some of the LPT simulations described in this work utilise an explicit time stepping scheme. The simulations using the explicit scheme will be specified in the section describing different cases.

4.1.5 Simulation Cases

The simulation cases presented in this section utilise the DNS method based on the IB-FSI framework to determine the fluid forces and the subsequent motion of the particle based on the GLE. The simulations presented here describe particle motion at a Reynolds number ($Re = 2r_p V_{RMS}/\nu$) of approximately

$4.45 \cdot 10^{-7}$, which is relevant for many practical applications[44] . The particle-fluid density ratio (ρ_p/ρ_f) is set to 1 to demonstrate the capabilities of the DNS method in accurately modelling Brownian motion in different flow scenarios involving added mass and history effects. The two different cases simulated are unhindered Brownian motion and hindered Brownian motion near a wall, which will be discussed further in the upcoming sections. The simulation cases discussed here are run using a time step size of $\tau_p/50$ for a duration of at least $1000\tau_p$ so that sufficient data is available to generate the MSD and VACF graphs.

4.1.5.1 Unhindered Case

In the unhindered case, the particle is assumed to be surrounded by fluid that extends infinitely into the surrounding, thereby removing the influence of any domain boundaries, such as walls, on the particle hydrodynamics. Since the MSD and VACF for this case are well documented in literature [1], [20], this case is used as a verification case for the DNS method. For practical purposes, the simulation domain cannot be infinite, and hence in the IB-FSI framework used to implement the DNS method, the domain is reduced to a cube of side $30r_p$ as shown in figure 4.1 . Since the boundaries of the domain are considerably far from the particle, they have a negligible influence on the hydrodynamics of the particle. The particle-fluid density ratio is set to 1 to simulate a neutrally buoyant particle, where the influence of the added mass and history effects are quite strong.

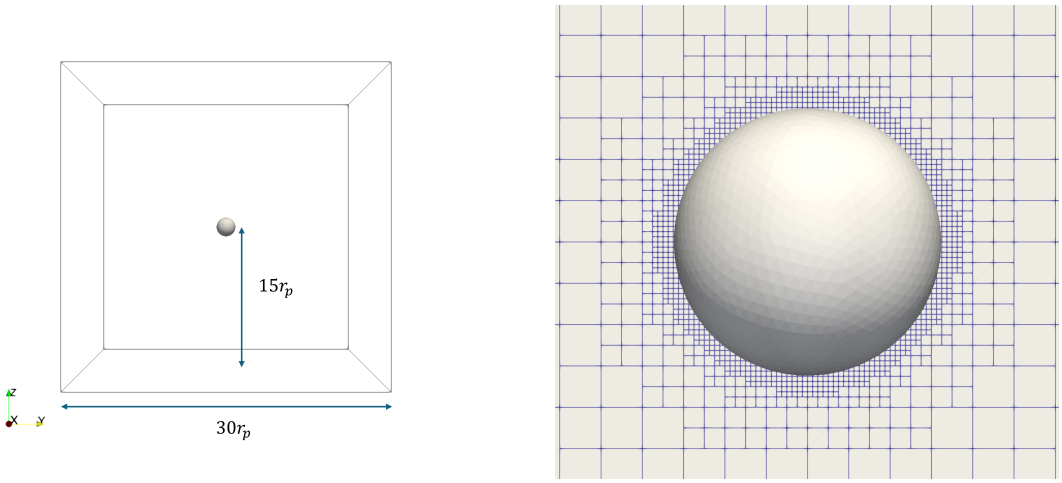
In the unhindered case, mathematical models as shown in equation 3.10 are directly available for modelling the hydrodynamic force on the particle[39] . This provides an alternate way for determining the hydrodynamic force ($\mathbf{F_H}$) used to determine the memory kernel. The time-discretised form of the hydrodynamic force in this case is:

$$\begin{aligned}\mathbf{F_H}(t) = \quad & -\gamma\mathbf{V}(t) - \frac{1}{2}m_p\frac{\rho_f}{\rho_p}\mathbf{a}(t) \\ & -\gamma\Delta t\sqrt{\frac{r_p^2\rho_f}{\pi\mu}}\sum_{\tau=0}^{\tau=t-\Delta t}\frac{\mathbf{a}(\tau)}{(t-\tau)^{1/2}},\end{aligned} \tag{4.15}$$

where $\mathbf{a}(t)$ refers to the acceleration of the particle obtained from simulation information at times t and $t - \Delta t$.

The LPT method is used in the unhindered case to verify the viability of the GLE approach using a different way of generating the hydrodynamic force from the surface integral approach used in the IB-FSI framework. Since, mathematical models are used in contrast to numerically solving the Navier-Stokes equations, the LPT method is considerably faster than the DNS approach.

The LPT method is also used to model the particle dynamics based on the CLE. In this case, the random Brownian force used in the simulations is generated as a white noise process that follows the variance given by the



(a) Setup (b) Grid Discretisation

Figure 4.1: The unhindered case setup is depicted in (a). The particle is placed at the centre of the domain with a distance of $15r_p$ between the particle centre and the walls, which makes any effect from the walls negligibly small. In (b), the grid discretisation used in the IPS-FSI framework simulation is shown. The smallest cells have a size of $5r_p/128$.

fluctuation-dissipation relationship in equation 3.2 . The Brownian force then takes the form:

$$F_{B,i} = \sqrt{\frac{2k_B T\gamma}{\Delta t}}\mathcal{N}(0,1). \tag{4.16}$$

In both versions of the LPT simulation, the explicit time-stepping scheme is used.

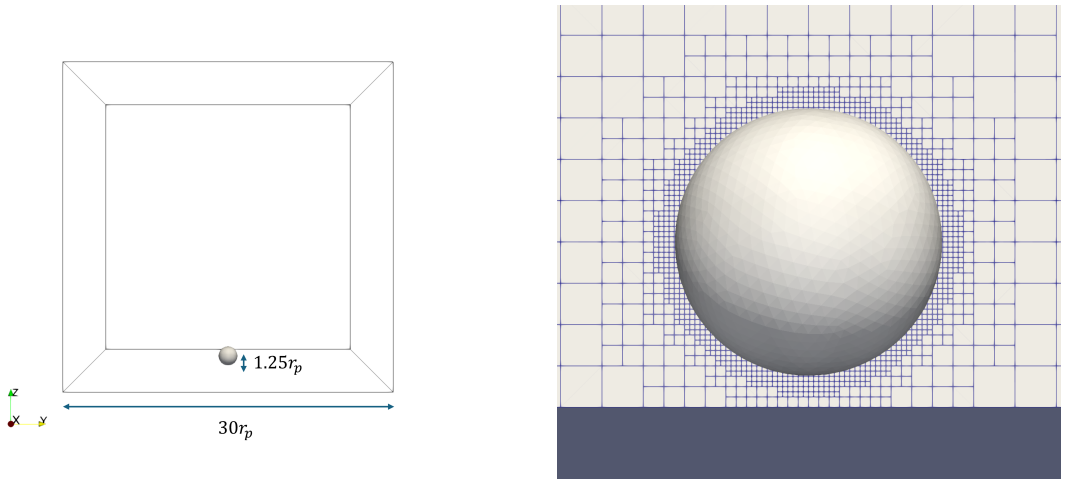
Thus the unhindered case is simulated in three ways:

- IB-FSI with memory-kernel-based GLE
- LPT with memory-kernel-based GLE
- LPT with CLE (no memory effects)

The statistical behaviour of the particle from these three approaches is compared against each other as well as theoretical results to verify the GLE-based method and contrast it against the CLE-based method.

4.1.5.2 Wall Adjacent Case

The effect of hindrance on the Brownian motion of a particle is studied by placing the particle close to a wall at a distance of $1.25r_p$. The same domain as used in the unhindered scenario is replicated here, with the only shift being that of moving the particle close to the wall in the z-direction as shown in figure 4.2 . Since there are no generic mathematical models for accurate modelling of



(a) Setup (b) Grid Discretisation

Figure 4.2: The wall-adjacent case setup is shown in (a). The particle is placed with its centre at a distance of $1.25r_p$ from the wall in the z-direction so that the changes in the particle behaviour due to the presence of the wall can be observed. In (b), the grid discretisation used in the IPS-FSI framework simulation is shown. The smallest cells have a size of $5r_p/128$.

hydrodynamic forces, only the DNS method, using the IB-FSI framework, is utilised in this case. The particle-fluid ratio is set to 1 in this case as well.

The statistical behaviour of the particle is again studied here by calculating the MSD and VACF.

4.2 Nonlocal Multiscale Simulation of Brownian Motion

While the DNS approach allows for the accurate modelling of Brownian phenomena, the computational cost associated with it can be quite high. As the domain size and the number of particles involved increase, the cost of simulation also proportionally increases. This is especially problematic when extending the method to study applications involving multiparticle Brownian phenomena that act over extended time durations and large domains. An alternative multiscale approach based on the GLE that still accounts for nonlocal hydrodynamic effects from the history of the particle motion is explained in this section.

At a fundamental level, the multiscale approach relies on the understanding that the memory kernel can be utilised to model the total hydrodynamic force while also being capable of generating the coloured Brownian force. In addition, the memory kernel provides knowledge of the total hydrodynamic force on the particle without explicit analytical models for the drag, added mass and history effects that contribute towards the total hydrodynamic force. This is especially useful in scenarios involving complex geometries where explicit mathematical models for the hydrodynamic force are not always readily available (unlike

the unhindered spherical particle scenario, where mathematical models are available). Thus, knowledge of the memory kernel associated with the particle at different points in the domain provide a mapping of the hydrodynamic force that acts on the particle at those positions.

Many microfluidic applications also have Brownian particle motion that is influenced by magnetic [8], [9], gravitational [10], or electric [6], [7] fields, requiring the modelling of a conservative body force acting on the particle apart from the hydrodynamic and Brownian force. Therefore, the GLE is modified in this case to account for conservative forces that might guide the particle's motion:

$$\begin{aligned} m_p \frac{d\mathbf{V}(t)}{dt} &= \mathbf{F_H}(t) + \mathbf{F_B}(t) + \mathbf{F_C}(t) \\ &= -\int_0^t \mathbf{K}(t,\tau) \cdot \mathbf{V}(t-\tau) d\tau + \mathbf{F_B}(t) + \mathbf{F_C}(t), \end{aligned} \tag{4.17}$$

where $\mathbf{F_C}$ is a conservative force acting on the particle. The nonlocal multiscale model based on the understanding of the GLE and the associated memory kernel has the following major steps:

1. Generate memory kernel library mapping the domain using short DNS
2. Determine memory kernel associated with current particle position from kernel library
3. Generate hydrodynamic force and coloured Brownian force from the memory kernel
4. Utilise hydrodynamic force, Brownian force and conservative force to evolve particle trajectory in a memory kernel based LPT framework

The steps listed here are discussed in further detail in the upcoming subsections.

4.2.1 Mapping the Domain using Memory Kernel Library

The memory kernel is utilised in the multiscale approach to model the hydrodynamic force acting on the particle while also generating the colored random Brownian force required for the GLE. As previously shown in equation 3.7, the convolution integral of the memory kernel and the velocity history of the particle provides the total hydrodynamic force on the particle. The hydrodynamic force varies based on the position of the particle in the domain, and hence, to accurately model the force, the memory kernel needs to be obtained at different positions in the domain based on the simulation requirement. Similar to the DNS method previously discussed, the simulations run using the nonlocal multiscale method also use spherical particles. The symmetry of the memory kernel tensor as well as the symmetry of the particle shape, as discussed in section 4.1.2, allow for the memory kernel to be reduced to just its diagonal

terms (i.e. $K_{ij} = K_{ii}\delta_{ij}$) which account for normal ($\perp$) and parallel ($\parallel$) hydrodynamic effects from nearby domain boundaries[21], [44] . The relationship between the hydrodynamic force and the memory kernel then simplifies to equation 4.2, which will be utilised in the generation of the memory kernel.

In order to generate the memory kernel at a particular position ($\mathbf{K}(x, \tau)$), a short DNS is performed in the IB-FSI framework. In the simulation, the particle is pushed, starting from the position of interest, using a small constant force for a short duration. The force is chosen so that the particle neighbourhood does not change considerably during the duration of the motion. This allows for the assumption that the memory kernel associated with the particle does not change drastically during the course of the simulation. The hydrodynamic force acting on the particle in response to its motion is captured by performing the surface integral of the pressure and viscous stresses acting on the particle's surface. The velocity of the particle is also recorded simultaneously. The hydrodynamic force and the velocity history are then used to obtain the memory kernel values assumed to be associated with the initial position of the particle by using the optimisation routine described in section 4.1.2 . The optimisation routine uses the `L-BFGS-B` method and iteratively optimises the memory kernel against the cost function shown in equation 4.5 . Refer to section 4.1.2 for a more detailed explanation. The process involving short DNS runs while optimising for the kernel values is repeated at all the positions in the domain where the memory kernel is required, thereby creating a library of memory kernels that map the hydrodynamic effects that various positions in the domain have on the particle.

4.2.2 Determination of Memory Kernel Associated with Particle Position

The memory kernel associated with the particle at every single point in the domain is not readily available in the kernel library since only select positions have been used to generate the library. Linear interpolation is used during the LPT simulations to determine the memory kernel at the current particle position by utilising the predetermined kernels from the library associated with the nearest positions. The kernel ($\mathbf{K}(t, \tau)$) at the current particle position ($\mathbf{x}(t)$) is then given by:

$$\mathbf{K}(t, \tau) = \mathbf{K}(\mathbf{x_1}, \tau) + \frac{(\mathbf{x}(t) - \mathbf{x_1})}{\mathbf{x_2} - \mathbf{x_1}}(\mathbf{K}(\mathbf{x_2}, \tau) - \mathbf{K}(\mathbf{x_1}, \tau)), \tag{4.18}$$

where $\mathbf{x_1} < \mathbf{x}(t) < \mathbf{x_2}$. The kernels $\mathbf{K}(\mathbf{x_1}, \tau)$ and $\mathbf{K}(\mathbf{x_2}, \tau)$ are predetermined kernels available in the library for the positions $\mathbf{x_1}$ and $\mathbf{x_2}$ respectively.

4.2.3 Generation of Hydrodynamic Force and Coloured Brownian Force

Once the memory kernel at the current particle position in the domain is determined, it is used to generate the hydrodynamic force as shown in equation 4.2 . The numerically discretised version of the same equation that is used in the simulation is given as:

$$F_{H,i}(t) = -\sum_{n=0}^{N} K_i(t, n\Delta t) V_i(t - n\Delta t)\Delta t. \tag{4.19}$$

Thus, the discretised version of the memory kernel is multiplied by the corresponding values from the stored velocity history of the particle motion and summed up to model the hydrodynamic force. This is a numerical approximation of the convolution integral form of the hydrodynamic force presented in equation 4.2 .

The coloured Brownian force is generated from the memory kernel at the current particle position using the fluctuation-dissipation relationship shown in equation 4.3 . The covariance matrix for the Brownian forces can be then generated from this and utilised to generate a colored random sequence of Brownian forces with the appropriate covaraince as shown in section 4.1.3 . Apart from the hydrodynamic and Brownian forces, a conservative force is also applied in the simulations presented in this work leading to the following equation of motion for the particle:

$$\begin{aligned} m_p \frac{dV_i(t)}{dt} &= F_{H,i}(t) + F_{B,i}(t) + F_{C,i}(t) \\ &= -\sum_{n=0}^{N} K_i(t, n\Delta t) V_i(t - n\Delta t)\Delta t + F_{B,i}(K(t)) + F_{C,i}(t). \end{aligned} \tag{4.20}$$

4.2.4 Evolution of Particle Trajectory

The acceleration of the particle can be determined from the net force acting on the particle as shown:

$$\mathbf{a}(t) = \frac{\mathbf{F}_H(t) + \mathbf{F}_B(t) + \mathbf{F}_C(t)}{m_p}. \tag{4.21}$$

The acceleration of the particle is then used to evolve the particle velocity and position over time using an LPT framework utilising the implicit Newmark time stepping scheme (detailed in section 4.1.4).

To summarise, the simulations begin with the mapping of the memory kernels associated with particle at different positions in the domain using short DNS simulations. The kernel library so developed is used in a memory kernel based LPT method, which estimates the kernel associated with the current particle position using linear interpolation. The hydrodynamic force and the random colored Brownian force associated with the particle motion at the current position are then generated using the estimated kernel. Finally, an implicit time-stepping scheme is used to propagate the particle motion in time utilising the GLE, incorporating any additional external forces apart from hydrodynamic and Brownian forces, which are already included.

4.2.5 Simulation Cases

A variety of simulations are performed for a neutrally buoyant ($\rho_p/\rho_f = 1$) spherical particle using the nonlocal multiscale model. The multiscale method is first verified against the DNS method by simulating two unhindered scenarios. In the first case, an unhindered spherical particle is pushed with the small force ($\mathbf{F_C}$) and the resulting hydrodynamic response ($\mathbf{F_H}$) in the absence of Brownian force ($\mathbf{F_B}$) is measured in the LPT simulation and compared with the same simulation in the DNS case to ensure that the multiscale method is capable of generating an accurate hydrodynamic response by using the memory kernel. In the second case, an unhindered spherical particle is simulated undergoing Brownian motion in the absence of any external force ($\mathbf{F_C}$). The VACF generated from the multiscale method is then compared against the same obtained using the DNS method to ensure that the multiscale method can accurately model Brownian phenomena.

Once the multiscale approach is verified, it is utilised to simulate the motion of a Brownian particle moving towards a wall under the influence of an attractive conservative force. The domains used in these simulations contain two walls in the positive and negative z-direction. The domain is considered to extend infinitely in the xy-plane. The distance between the walls is set to $10r_p$ and $30r_p$ to understand how variation in wall distance can influence the particle behaviour. The domains used are depicted in figure 4.3 . The force used to pull the particle towards the wall is set to $m_p g$ and $5m_p g$ to study the effects variation in the external forcing on the particle dynamics. In all cases, the particles are introduced in the centre of the domain, between the two walls and then allowed to move until they come in contact with the wall when the simulation is concluded. Finally, simulations are also run where the particle covers the same distance as it would from the centre of the domain to the wall, but with the wall considered to be far away from the particle, making its effect on the hydrodynamics of the particle negligible. This set of simulations is done so as to understand the differences between hindered and unhindered particle motion. The random numbers needed to generate the Brownian forces used in the simulations are obtained using a seeded random number generator from the `random` module in `python`[50] . In order to obtain statistical average behaviours for the particle, each combination of domain, force and hindrance is simulated using ten different seed values for the random number generator.

In order to perform the wall bounded LPT simulations, the memory kernels associated with various particle positions from the centre of the domain to the wall need to be obtained from DNS simulations. In the neutrally buoyant scenario, the added mass effect is a significant contributor to the hydrodynamic force, with theoretical models predicting the following contributions in a wall-adjacent scenario[44]:

$$\begin{aligned} m^*_\perp &= m_p + \tfrac{m_f}{2}\left[1 + \tfrac{3}{8}\left(\tfrac{r_p}{h}\right)^3\right], \\ m^*_\parallel &= m_p + \tfrac{m_f}{2}\left[1 + \tfrac{3}{16}\left(\tfrac{r_p}{h}\right)^3\right]. \end{aligned} \tag{4.22}$$

(a) $10r_p$ (b) $30r_p$

Figure 4.3: The two domains used in the multiscale simulations of Brownian particle migration: (a) domain with walls separated by a distance of $10r_p$ in the z direction, (b) domain with walls separated by $30r_p$.

Domain	**Particle position ($\mathbf{h/r_p}$)**											
30 $\mathbf{r_p}$	15.00	11.91	9.45	7.50	5.95	4.72	3.75	2.98	2.36	1.88	1.49	1.18
15 $\mathbf{r_p}$	5.00	4.37	3.82	3.33	2.91	2.54	2.22	1.94	1.70	1.48	1.29	1.13

Table 4.1: The particle positions from the wall used for sampling the memory kernel in the IB-FSI simulations for the two different domain sizes.

Here, $m_f = m_p(\rho_f/\rho_p)$, is the mass of fluid that occupies the same volume as the particle. The added mass effects in the both, the normal ($\perp$) and parallel ($\parallel$) directions to the wall scale with the distance (h) from the wall as multiples of $(r_p/h)^3$. Hence, the positions at which to calculate the memory kernels have been picked based on these equations as the positions at which the quantity $(r_p/h)^3$ doubles, starting from the centre of the domain. The chosen positions are given in table 4.1 . In the work presented here, each DNS required for the cases was run for $20\tau_p$, with a time step size of $\tau_p/50$ to generate a memory kernel with a max lag time of $10\tau_p$. The LPT simulations also use the same time step size, allowing for the memory kernels generated from the DNS to be directly used without any changes in temporal discretisation.

Chapter 5

Highlighted Results

The DNS method and the nonlocal multiscale model have been used to simulate various cases. In this chapter, the different results obtained from each of these cases will be presented and discuss. The results obtained using the DNS method is discussed first in section 5.1 followed by discussions of the results from the multiscale approach in section 5.2 .

5.1 Explorations using the DNS Method

In this section, the results from the unhindered and wall-adjacent cases of Brownian motion are presented. While only the neutrally buoyant particle is presented here, the results for simulations performed at higher particle-fluid density ratios can be found in paper A[28] . The MSD and the VACF are presented initially for the Brownian motion of the unhindered particle, followed by the wall-adjacent particle. The MSD in each case is calculated from the particle position (X) data as:

$$\langle X^2(t)\rangle = \langle \Delta X^2(\Delta t)\rangle = \frac{1}{M}\sum_{t=0}^{t=T-\Delta t}\left[X(t+\Delta t) - X(t)\right]^2 . \tag{5.1}$$

Here M is the number of data points used to arrive at the average. The VACF on the other hand is calculated from the particle velocity (V) data as:

$$C_V(\Delta t) = \langle V(0)\cdot V(\Delta t)\rangle = \frac{1}{M}\sum_{t=0}^{t=T-\Delta t}\left[V(t)\cdot V(t+\Delta t)\right] . \tag{5.2}$$

The MSD and VACF are compared against theoretical results and trends. While the unhindered case is used to verify the simulation technique, the wall-adjacent case demonstrates its applicability in more complex flow scenarios. Finally, the optimised memory kernel obtained in each case is also discussed.

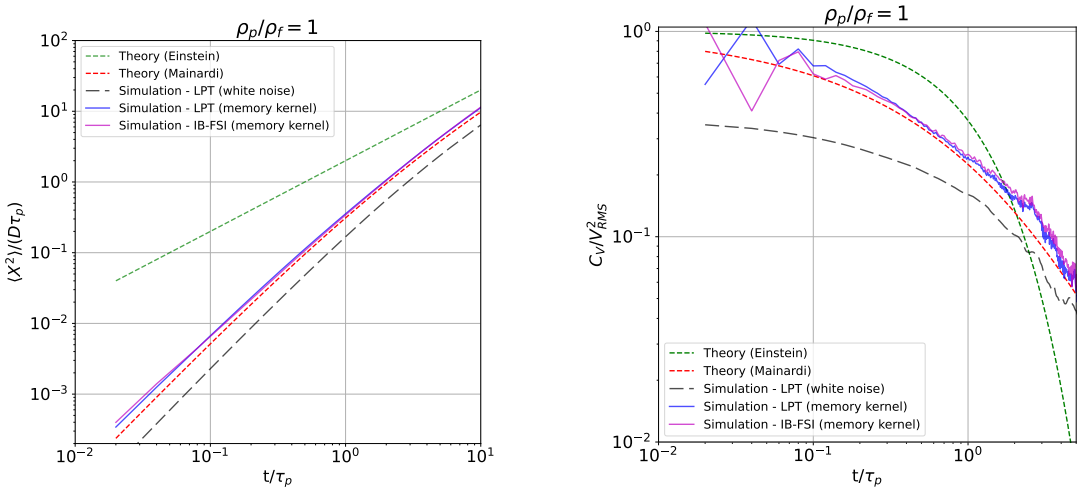


Figure 5.1: The MSD and VACF for an unhindered particle are shown on the left and right, respectively. The simulations contain results obtained from the DNS simulations run using the IB-FSI framework, as well as LPT simulations run using the white noise and memory kernel-based generation of the random Brownian force. These results have been compared against theoretical predictions.

5.1.1 Unhindered Brownian Motion

In the case of unhindered Brownian motion, the results obtained from the DNS using IB-FSI framework are plotted along with the LPT simulation results in figure 5.1 . Since the unhindered case is symmetric along all three directions, the data plotted has been averaged over the three orthogonal directions used to run the simulations. The fluctuations in the long-term regime of the VACF graphs are due to the low number of data points available to obtain a stable average value.

The figure on the left shows the results of the MSD from the simulations compared against the theoretical MSD values obtained from theory [20] as calculated by Mainardi et al., as shown in equation 3.9 . A line corresponding to the Einstein model of diffusion, where the MSD is given as being equal to $2dDt$ is also plotted in the figure. All the MSD data have been normalised with the value $D\tau_P$ to make it dimensionless[33] . The GLE based simulations (both the DNS and LPT version) show that the MSD of the particle closely follows the expected theoretical values calculated by Mainardi et al. The ballistic motion that the Einstein model is unable to represent has been captured in the short term ballistic regime seen at the beginning of the graphs. In the long term, the GLE simulations tend towards the pure diffusion model predicted by Einstein. This is visible from the gradual change in slope of the MSD line at higher lag times. The added mass effect and history effects slow down the approach of the MSD towards the long term pure diffusion line predicted by Einstein. In contrast, the CLE based simulation results under-predict the MSD compared to the theoretical values. The white noise generation of the Brownian

force based on the Stokes drag is, hence, insufficient to generate enough motion in the Brownian particle to create an accurate MSD.

On the right side of figure 5.1, the VACF graph from the three types of simulations are shown. Here, again, the simulation results are compared against the theoretical predictions given in equation 3.8 as calculated by Mainardi et al[20] . The exponential decay for the VACF derived from the CLE is also shown. All the data has been normalised by the RMS velocity calculated, including the added mass effect. The VACF from the GLE simulations are very close to the values obtained by Mainardi et al. in their theoretical calculations for an unhindered particle. The CLE based LPT simulation results, however, underpredict the VACF due to the absence of the coloured Brownian force that can account for the added mass and history effects. The GLE based simulations create an oscillation in the VACF graph in the short lag time region, which can be attributed to the dependence of the added mass effect on the acceleration of the particle, which in the simulations here depends on data at time steps of the simulation. These oscillations can be avoided by taking shorter time step sizes for the simulation, which would shift the oscillations to shorter time durations.

Thus, the GLE based DNS and LPT methods are capable of accurately capturing unhindered diffusion of a spherical particle. The results from the wall-adjacent case simulated using the DNS method are presented next to show its applicability to more general use cases.

5.1.2 Wall-Adjacent Brownian Motion

The MSD and VACF are presented for a neutrally buoyant spherical particle placed at a distance of $1.5r_p$ from a wall. The MSD and VACF have been calculated as shown in equation 5.1 and 5.2, respectively. The results from the simulation have been normalised as discussed in section 5.1.1 and displayed in figure 5.2 .

In the wall-adjacent case, the diffusion coefficient in the long time limit for the particle reduces from its unhindered value (D) to the following[43], [51]–[53]:

$$D_{\parallel} = D\left(1 - \frac{9r_p}{16h} + \frac{1r_p^3}{8h^3} - \frac{45r_p^4}{256h^4} - \frac{1r_p^5}{16h^5}\right), \tag{5.3}$$

and

$$D_{\perp} = D\frac{6h^2 - 10r_p h + 4r_p^2}{6h^2 - 3r_p h - r_p^2}, \tag{5.4}$$

This reduction in the diffusivity of the particle is attributed to the increased hydrodynamic hindrance that the particle experiences close to a wall[1], [54] . The MSD from the wall-adjacent case, shown on the left in figure 5.2, depicts this reduction in diffusivity in the normal and parallel directions to the wall, with the reduction being more pronounced in the normal direction due to increased hindrance in this direction. The MSD is within 20% of its expected long-time value in the normal direction while it is within 7% of the same in

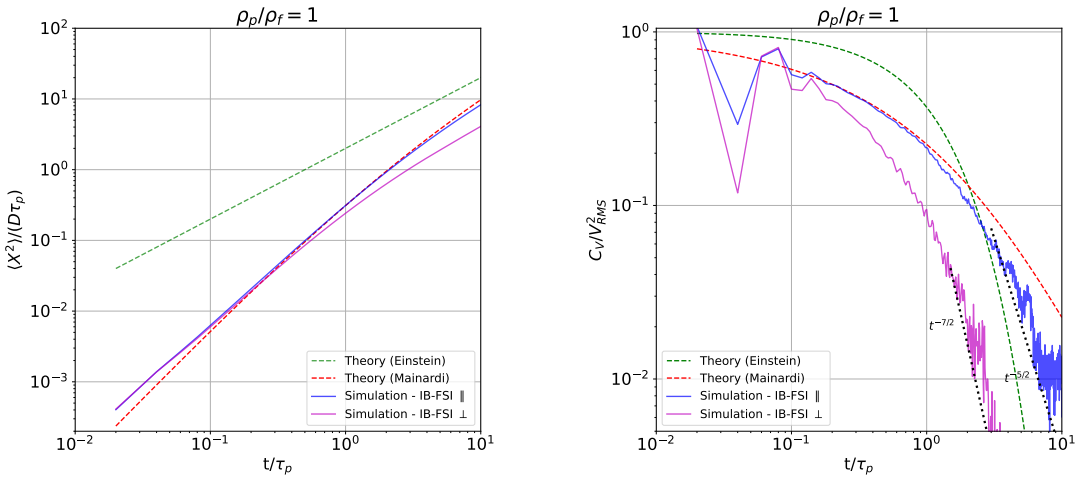


Figure 5.2: The MSD and VACF for a wall-adjacent particle are shown on the left and right, respectively. The simulations contain results obtained from the DNS simulations run using the IB-FSI framework, as well as LPT simulations run using the white noise and memory kernel-based generation of the random Brownian force. These results have been compared against theoretical predictions.

the parallel direction. The wall-adjacent MSD predictions, thus, capture the expected reduced diffusion from hindrance effects with good agreement with theoretical predictions.

Similarly, the wall-adjacent VACF can be expressed as[55]:

$$\frac{C_{V,\parallel}(\tau)}{\langle V^2(t) \rangle} = \frac{3B}{2} \frac{\tau_w}{\tau_f} \left(\frac{\tau}{\tau_f} \right)^{-5/2}, \tag{5.5}$$

in the directions parallel to the wall, and

$$\frac{C_{V,\perp}(\tau)}{\langle V^2(t) \rangle} = \frac{3B}{2} \left[A \left(\frac{\tau}{\tau_f} \right)^{-5/2} + \frac{\tau_w^2}{4\tau_f^2} \left(\frac{\tau}{\tau_f} \right)^{-7/2} \right], \tag{5.6}$$

Here, $B = (\rho_p/\rho_f + 1/2)/9\sqrt{\pi}$ and $A = (2\rho_p/\rho_f - 5)/9$ are numerical constants that depend only on the particle-to-fluid density ratio. The characteristic time $\tau_w = h^2/\nu$, is the time for hydrodynamic effects of the wall to reach the particle. In the parallel direction, the VACF scales as $\tau^{-5/2}$ in contrast to the $\tau^{-3/2}$ scaling seen in the unhindered case. Meanwhile, in the normal direction, the VACF scales as $\tau^{-7/2}$ in the initial stages (when $\tau_w \lesssim t \ll \tau_w^2/\tau_f$), which then changes to a $\tau^{-5/2}$ in the long term. Experiments conducted by Jeney et al. [56], were able to capture these scaling trends for a particle at a distance of $h = 2.2r_p$ at a density ratio of $\rho_p/\rho_f = 1.96$. The characteristic time scales associated with the experiments where $\tau_f = 2.25\tau_p$ and $\tau_w = 23\tau_p$. In the simulations shown, the characteristic time scales were $\tau_f = 4.5\tau_p$ and $\tau_w \approx 7\tau_p$, which is quite close to the experimental values. Here, $\tau_w > \tau_f > \tau_p$, whereas $\tau_w \gg \tau_f, \tau_p$ required for the theoretical predictions is not fulfilled, so an exact

match to theoretical trends cannot be expected. The VACF graphs obtained from the wall-adjacent simulations, shown on the right of figure 5.2, do depict the expected scaling of $\tau^{-5/2}$ in the parallel direction. In the normal direction, the scaling of $\tau^{-7/2}$, expected in the initial part of the VACF is observed. The VACF decorrelates much faster in both the parallel and normal directions since the wall limits the development of flow structures around the particle. The decorrelation is more pronounced in the normal direction than the parallel direction due to the increased hydrodynamic resistance in the normal direction. The simulation is able to accurately capture the quicker VACF decorrelation trends predicted by theory and experiments[55], [56] . Thus, MSD and VACF graphs show that the DNS method is capable of accurately capturing the trends of wall-adjacent diffusion, showing that the method can be utilised in complex flow scenarios.

5.1.3 Notes on Memory Kernels

The memory kernel carries information about the drag, added mass and history effects that act on a particle. A theoretical estimate of the memory kernel associated with a spherical particle in the unhindered scenario is[29]:

$$K_{ij}(t,\tau) = \begin{cases} \gamma\left(2\delta(\tau) - \sqrt{\tau_f}\frac{\tau^{-3/2}}{2\sqrt{\pi}}\Theta(\tau)\right), & \text{if } i = j, \\ 0, & \text{if } i \neq j. \end{cases} \tag{5.7}$$

Here, the $2\gamma\delta(\tau)$ represent the instantaneous drag effect at zero lag time. The memory kernel graphs have been normalised using the numerical representation of this function, i.e. $2\gamma/\Delta\tau$. The memory kernel significantly changes only when its geometric neighbourhood is altered significantly, which does not happen during the diffusion experienced by the Brownian particle in either of the cases. Hence, the normalised memory kernel (K^*) is also averaged over all the time steps in the simulation to obtain an average kernel for each case (i.e. $K^*(t,\tau) = K^*(\tau)$).

The memory kernels derived from the unhindered and the wall-adjacent cases are shown in figure 5.3 . The initial part of the kernel is represented in a linear graph while the tail is compared against the theoretical expression in a logarithmic inset. The kernel is the factor by which the velocity history is multiplied with before being summed up to create the hydrodynamic force. Hence, each value of the kernel represents the strength of the contribution towards the force from the velocity of the particle at that lag time in the past. The memory kernel in both cases contain a peak at zero lag time that corresponds to the strength of the combined drag and added mass effect. This is followed by a dip in the kernel representing the negative contribution from the added mass effect since it depends on the acceleration of the particle, which contains a negative contribution from the velocity at the previous time step. The tail of the kernel represents the history force acting on the particle.

The unhindered case, shown on the left in figure 5.3, shows the memory kernels obtained from the LPT simulations and the DNS run in an IB-FSI framework. The differences in the in the optimised kernels between the two

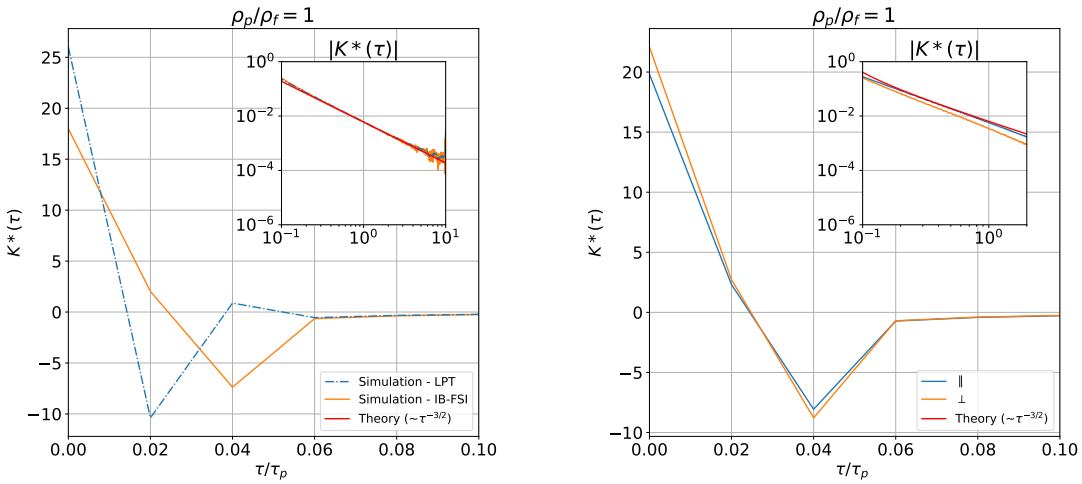


Figure 5.3: The memory kernels obtained for a neutrally buoyant spherical particle in the unhindered and wall-adjacent cases are shown on the left and right, respectively. The memory kernel plot on the left (unhindered case) contains the kernel values obtained from the LPT simulations and DNS based on the IB-FSI framework. The plot on the right (wall-adjacent case)contains only values from DNS based on the IB-FSI framework. The kernel is split into normal ($\perp$) and parallel ($\parallel$) components to indicate the influence from the directions normal and parallel to the wall on the hydrodynamic force.

simulations can be attributed to the difference in the way the hydrodynamic force is calculated in each case (i.e. mathematical model in LPT and surface integral in DNS). The explicit time-stepping scheme used in LPT and the implicit Newmark scheme used in IB-FSI are also a source of difference in the kernels. The second peak in the LPT kernel is due to the history force model used, which gives a positive contribution to the kernel in its discredited form as used in the simulation. The tails of the kernels in both cases show good agreement with the theoretical tail that scales with $\tau^{-3/2}$.

The wall-adjacent case, shown on the right of figure 5.3, contains the memory kernels associated with the wall-normal and wall-perpendicular directions obtained from the DNS simulations. The peaks of the kernels in this case are higher than their DNS counterparts in the unhindered case, indicating the increased drag and added mass effect that is felt by the particle due to the presence of the wall[44], [54] . The effect is more pronounced in the normal direction indicated by the higher peak in that direction. The tails also show a deviation from the theoretical trends, with a more pronounced deviation in the normal direction, where the presence of the wall reduces the memory effects on the particle.

The fluctuation-dissipation relationship states that the covariance of the Brownian force is directly proportional to the memory kernel. Thus, the high peaks in the beginning of the kernel indicate the strength of the Brownian force, while the remaining values indicate the strength of the influence on the current force from past forces. In the wall-adjacent case, the Brownian forces

decorrelate much faster than in the unhindered case, as seen from the memory kernel graph.

The differences in the memory kernels in the unhindered and wall-bounded cases indicate the different hydrodynamic effects brought about by the variation in the domain surrounding the particle. Therefore, the memory kernel is a good continuum representation of the molecular phenomena that cause these different hydrodynamic effects. For a more detailed discussion of how the drag, added mass and history effects change as the particle-fluid density ratio changes, refer to paper A[28] .

5.2 Explorations using the Multiscale Method

In this section, the multiscale LPT framework simulations are first compared against DNS simulations of the same situation to show the accuracy of the framework. The multiscale LPT framework is then used to explore the migration of a particle towards a wall under the influence of an attractive force. The memory kernel library developed for this purpose is showcased with a brief discussion on the information that can be gathered from it. The simulations showing the motion of the particle over similar distances in the presence and absence of hindrance are also shown to understand the various factors that can influence particle migration in such a scenario.

5.2.1 Verification of Method

In the first verification study, an unhindered spherical particle is moved under the influence of a constant force and in the absence of any Brownian motion or forcing. Simulations are done using the memory kernel based LPT method and the pure DNS (based on the IB-FSI framework as described in section 5.1). The hydrodynamic force acting in response to the particle motion is recorded in each case and normalised using the constant force used to move the particle. The normalised hydrodynamic force ($\mathbf{F}^*_{\mathbf{H}}$) from the multiscale LPT is compared against the same from DNS on the left side in figure 5.4 . The memory kernel associated with an unhindered particle is used to generate the hydrodynamic force in the case of the LPT simulations, while the surface integral of pressure and viscous stresses is used to measure the same in DNS. The LPT simulations can be seen to be in good agreement with the results from the DNS.

Pure Brownian motion simulations of an unhindered spherical particle are also run in the multiscale LPT framework and using DNS. The VACF (C_V) calculated as depicted in equation 5.2 is shown for both the simulations on the right of figure 5.4 . The random colored Brownian force is generated from the memory kernel associated with an unhindered particle. The VACF from the kernel based LPT simulations are seen to closely follow the same from the DNS. Thus, the multiscale kernel based LPT can be seen to be able to accurately model hydrodynamic forces and Brownian motion.

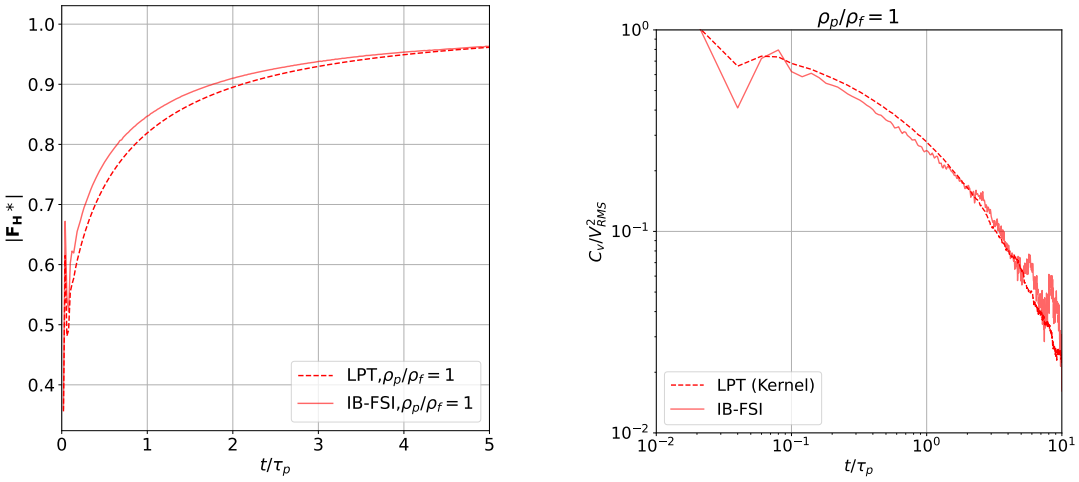


Figure 5.4: In the image on the left, the hydrodynamic force generated using the memory kernel based LPT simulation is compared against the same from the DNS using IB-FSI framework. On the right, the VACF of an unhindered Brownian motion simulation of a spherical particle as simulated by the memory kernel based LPT is compared with the same from a DNS.

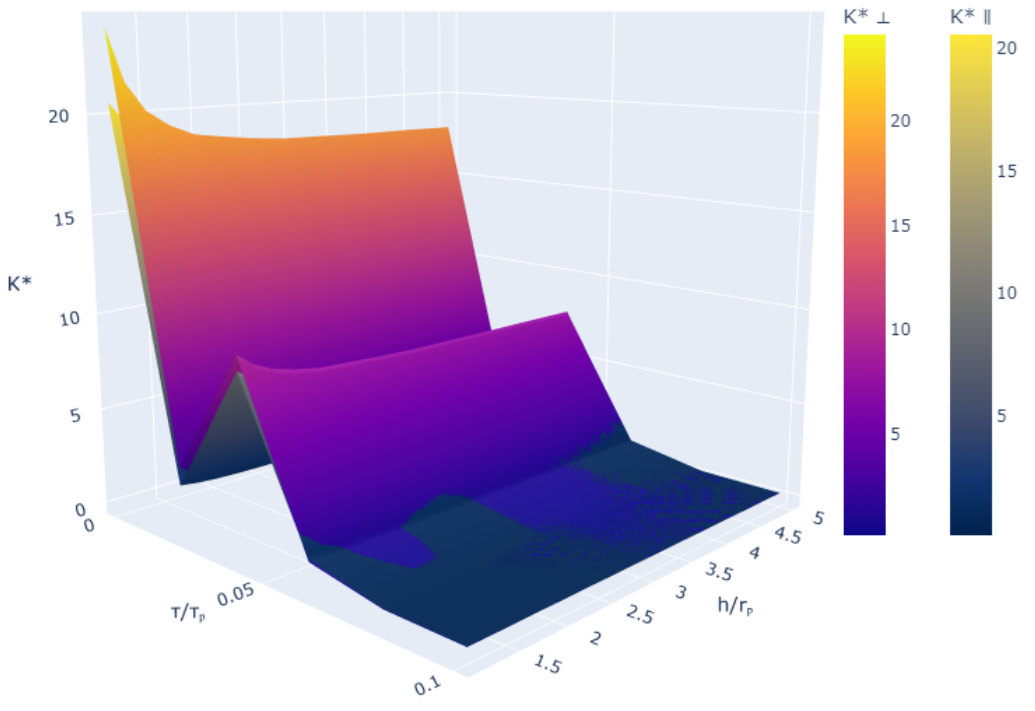
5.2.2 Wall Bounded Simulations

The multiscale LPT simulations of a particle moving towards a wall under the influence of an attractive force are discussed here. Double-walled domains are used, with the distance between the walls varied between $10r_p$ and $30r_p$. The attractive force pulling the particle towards the wall varies between $m_p g$ and $5m_p g$. The particle motion is also simulated in an unhindered scenario where the wall is considered to be far away, making its influence on the particle motion negligible. In this case, the same attractive force is used to move the particle over the same distance it would have traversed by the time it struck the wall.

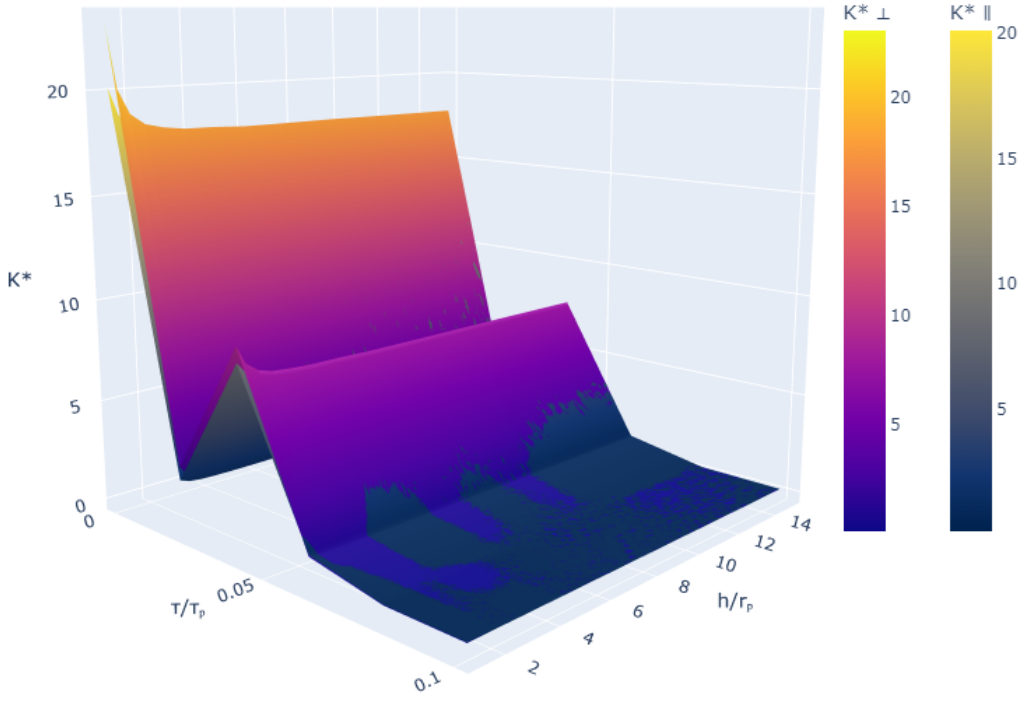
The memory kernel library developed for these simulations is presented first. This is followed by a discussion on particle motion during the simulations.

5.2.2.1 Memory Kernel Library

The memory kernels required for the simulations in the $10r_p$ and $30r_p$ domain were obtained from short DNS simulations of the particle as described in section 4.2.1 . The positions selected for obtaining the kernels are shown in table 4.1 . The initial part of the memory kernels obtained in this manner is shown in figure 5.5 . All the kernels in the memory kernel library obtained for both domains are shown here, mapped against the positions at which they are obtained to show how the kernel varies across the domain. It is interesting to note here that the peaks of the kernels shift towards higher values as the particle is shifted closer to the wall. The peak indicates the strength of the drag and added mass effect acting on the particle. Thus, the hydrodynamic hindrance (drag and added mass) increases closer to the wall as expected from



(a) $10r_p$



(b) $30r_p$

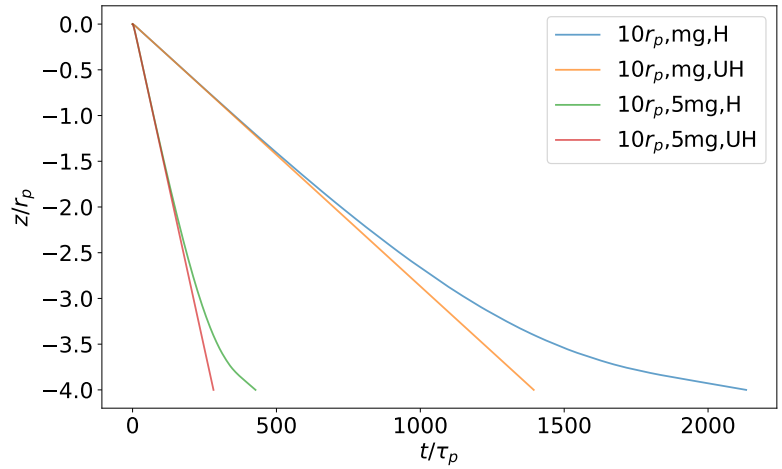
Figure 5.5: The initial parts of the 12 memory kernels ($K^* = K(h, \tau)\Delta\tau/2\gamma$) sampled from the bounded domains are shown as surface plots against time (t) and the distance of the sampling point from the wall (h). There are two surfaces plotted in each figure corresponding to the kernel associated with the directions normal ($\perp$) and parallel ($\parallel$) to the wall. As the particle moves closer to the wall, the associated peaks and valleys in the initial part of the memory kernel attain higher values, indicating the increased drag and added mass effect that the particle experiences close to the wall. The increase is more pronounced in the wall-normal direction as compared to that in the parallel direction. Away from the wall, the kernels converge to the same values irrespective of the direction indicating the decreasing influence from the wall.

theory[1] . This was briefly also observed while discussing the wall-adjacent case in the DNS simulations (see sections 5.1.2 and 5.1.3). Here, however, the rate at which the effect increases can be observed starting from the centre of the domain all the way to the wall. The wall normal direction shows more hindrance compared to the parallel direction. This is seen from the divergence of the kernel maps in each direction close to the wall, with the kernels in the normal direction having higher values. The tails of the kernels (not shown here) also depict the nature of the nonlocal history effects acting on the particle. The memory kernels are a rich source of information about the hydrodynamic interactions that the particle has with its surroundings. Thus, the kernels provide a way of modelling the hydrodynamic force in LPT simulations without the need for analytic models.

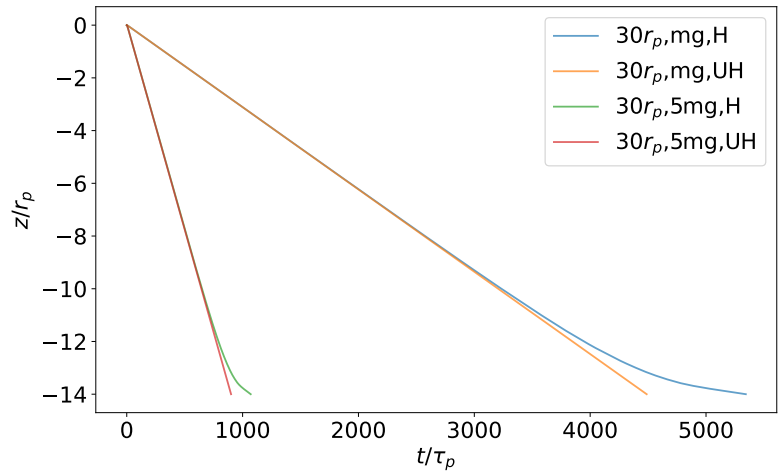
5.2.2.2 Particle Migration in Microfluidic Systems

The particle position (z) as it migrates towards the wall is shown in figure 5.6 . The position data has been normalised with the particle radius (r_p). While the top figure shows the position as the particle migrates in the $10r_p$ domain, the bottom figure shows the same in the $30r_p$ domain. The forces used in each case and the presence or absence of a wall are also indicated. The increased hydrodynamic hindrance to the particle motion can be seen from the difference in the particle trajectories between the unhindered (UH) and hindered (H) cases. The deviation only occurs as the particle gets closer to the wall, indicating that the wall does not significantly affect the motion close to the centre of the domain. This is especially noticeable in the bigger domain where the hindered and unhindered trajectories overlap for a considerable time. The increased hindrance from the wall increases the time required for the particle to cover the same distance. The strength of the attracting force is also another factor that controls the approach time, with the stronger force resulting in a quicker approach.

The velocity (v_z) of the particle while it approaches the wall is shown in a similar manner in figure 5.7 . The particle velocity is normalised with the RMS velocity for the Brownian motion of the particle (V_{RMS}). The oscillations seen in the velocity plot are caused by the Brownian motion of the particle. The particle initially approaches a terminal velocity in all cases associated with the hydrodynamic resistance felt in the centre of the domain. In the hindered cases, the velocity, however, quickly starts decreasing due to increased hydrodynamic force from the wall. In contrast, in the unhindered cases, where the wall is considered to be far from the particle, the same terminal velocity is maintained till the end of the simulations. As expected, a higher attractive force results in a higher velocity for the particle. Close to the wall, the particles in the hindered simulations can be seen to attain a new terminal velocity instead of continuing to decrease in magnitude. Once the particle crosses the closest sampling point to the wall shown in table 4.1, the memory kernel at that point is used without any extrapolation for the rest of the simulations, creating the steady terminal velocity effect close to the wall. This could be avoided by using more sampling points for the kernel.

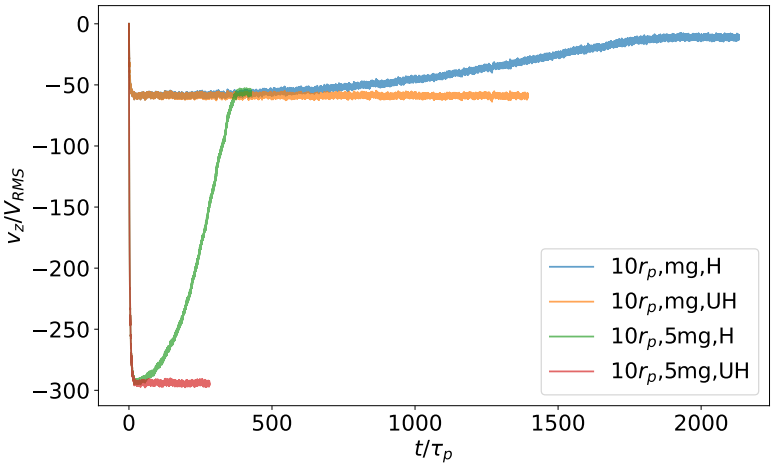


(a)

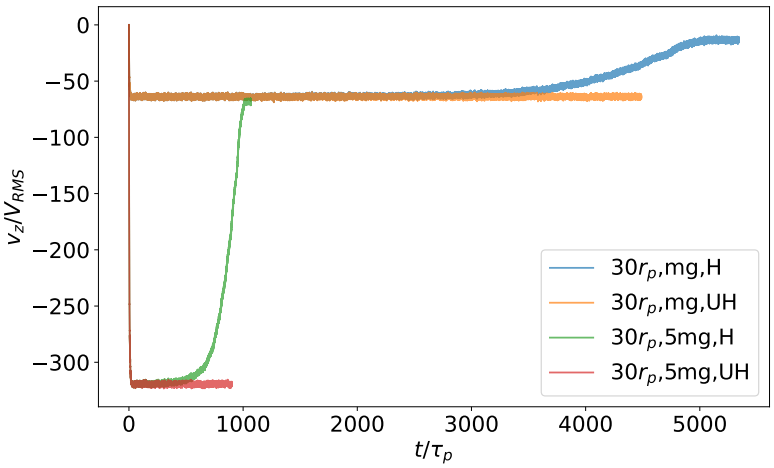


(b)

Figure 5.6: The position in the z direction of a particle migrating in unhindered and hindered scenarios towards a target wall: (a) distance between the walls in the domain is $10r_p$, (b) distance between the walls is $30r_p$. A force equivalent to $m_p g$ and $5m_p g$ is used in each domain as explained in the legends to each figure. The letters H and UH in the legends indicate hindered and unhindered runs of the simulations respectively.



(a)



(b)

Figure 5.7: The velocity in the z direction of a particle migrating in the unhindered and hindered scenarios towards a target wall: (a) the distance between the walls in the domain is $10r_p$, (b) the distance between the walls is $30r_p$. A force equivalent to m_pg and $5m_pg$ is used in each domain as explained in the legends to each figure. The letters H and UH in the legends indicate hindered and unhindered runs of the simulations respectively.

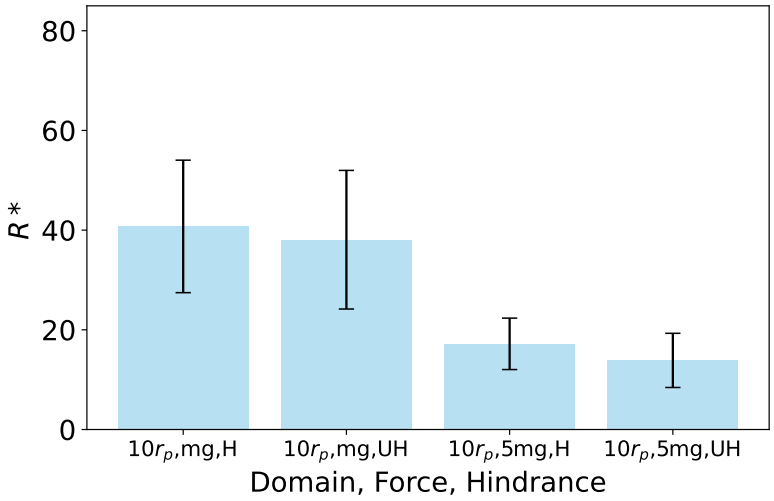
Since the point of impact is an important parameter in many microfluidic applications, the impact point in the xy-plane has been captured in these simulations. The particle undergoes pure diffusion with no external forces acting on it in this plane as it moves along the z-direction towards the wall. The impact radius, defined as $R = \sqrt{x_{ip}^2 + y_{ip}^2}$, is used to indicate the distance the particle has diffused in the xy-plane by the time it hits the wall. Here x_{ip} and y_{ip} are the position of impact on the wall. The ten sets of simulations run for each case using different random number sequences is used to obtain the average impact radius and its variance. The average impact radius and its variance, normalised with $\sqrt{D\tau_p}$, are shown in figure 5.8 for the different cases. The figure on the top shows the impact radius for the smaller domain, while the bottom one shows the same for the larger domain. For the same force and hindrance, it is seen that a larger domain shows a higher impact factor. The longer duration spent diffusing in the domain before striking the wall can be attributed to this increased impact radius.

The increased hydrodynamic resistance from the presence of the wall reduces the diffusivity of the particle as seen from the DNS simulations shown in section 5.1.2 [1] . It is interesting here to note that, contrary to expectations, the hindered simulations show a higher impact radius than their unhindered counterparts, where the particle diffusivity is higher throughout the whole simulation. However, in the larger domain at the higher attractive force, the particle impact radius in the hindered and unhindered cases are seen to be very close. These trends can be explained by the two different factors influencing the impact radius, namely the particle diffusivity and the time duration available for diffusion. The diffusivity of the particle determines the rate at which the particle spreads from its original position to its surroundings. While the diffusivity decreases close to the wall, the time available for diffusion increases since the particle takes longer to approach the wall. Thus, while the decreased diffusivity lowers the impact radius, a longer time before striking the wall increases the impact radius. Playing around with the distance to the wall or the magnitude of attractive force (which affects the time spend in the domain) are, therefore, possible ways to focus a particle onto a certain region in a microfluidic device.

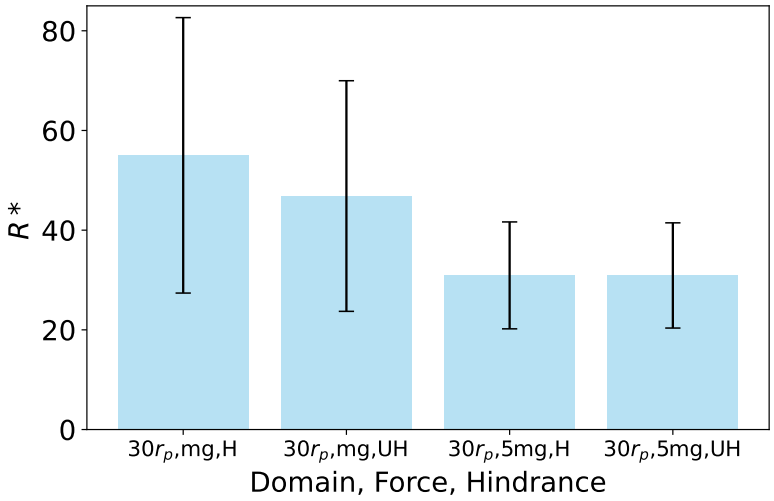
5.2.3 Computational Cost: DNS vs Multiscale

A major cost in running the multiscale LPT method is the generation of the memory kernels using the short DNS simulations. But a decrease in the number of kernels implies a reduction in the information available about the hydrodynamic forces acting at different points in the domain. A study on the simulation accuracy as the number of kernels used to run the multiscale LPT method is varied is shown here initially. This is followed by a comparison of the computational cost associated with the multiscale method and the DNS method to highlight the computational efficiency of the former.

The hindered simulations in the domain with a height of $30r_p$, with an attractive force of $m_p g$, are run using 3 and 6 kernels. The kernels required for these simulations are picked from the 12 kernels sampled at the positions



(a) $10r_p$



(b) $30r_p$

Figure 5.8: Average normalized impact radius on the target wall for the hindered and unhindered particle migration in each domain at two different attracting forces. The variance of the impact radius is shown as error bars in the plots. The letters H and UH in the labels indicate hindered and unhindered runs of the simulations respectively.

Number of Kernels	Error in position (%)		
	x	y	z
6	0.12	0.11	0.92
3	0.38	0.38	4.45

Table 5.1: Error in position for different numbers of memory kernels used to create the simulations. The error has been estimated by comparing against the simulation that is run using 12 memory kernels.

shown in table 4.1 starting with the kernel closest to the wall and uniformly skipping kernels to arrive at the required number of kernels. These simulation results are then compared against the same simulations run using 12 memory kernels taken at the positions shown in table 4.1 . The 12 kernel simulation is considered to be the most accurate since it has the most hydrodynamic information. The position data obtained in each case is then used to calculate the error as:

$$Error(n_K) = \frac{\sum_{i=0}^{n} |(x_{n_K,i+1} - x_{n_K,i}) - (x_{12,i+1} - x_{12,i})|}{\sum_{i=0}^{n} |x_{12,i+1} - x_{12,i}|}. \tag{5.8}$$

Here, n_K is the number of kernels used in the simulation while $x_{n_K,i}$ is the position of the particle at the time step i. The numerator shows the deviation of the particle from its expected position with 12 kernels while the denominator shows the total distance covered by the particle. This error is converted to a percentage value and shown in table 5.1 . It can be seen that the error from the simulations using 6 kernels is less than 1% in all the directions while the error in the simulations using 3 kernels is more than 4% in the z direction. The wall normal direction (z) in both cases show the highest error since this is the direction in which the hindrance effect from the wall is the strongest. It is also the direction in which the particle moves the most due to the influence of the attractive force. A more detailed comparison of the trajectories obtained from the different simulations is available in paper B. The simulation using 6 kernels is seen to be very close to the 12 kernel simulation, while the 3 kernel simulation shows noticeable differences. While the exact error can depend on the nature of the simulations run, this gives an indication of the information lost with successive reduction in the number of kernels used in the simulations.

The computational cost associated with the kernel based simulations is compared against the cost of running pure DNS of the same case in figure 5.9 . The cost of every simulation has been normalised with the cost associated with the pure DNS case. The cost of running pure DNS is more than two orders of magnitude more expensive than running any of the multiscale LPT simulations. This is a significant reduction in the computational requirements while being able to accurately capture all the relevant flow physics associated with a Brownian particle. The major part of the cost associated with running the multiscale method is the determination of the memory kernel library required for the simulations. As seen in the figure, this scales linearly with the number

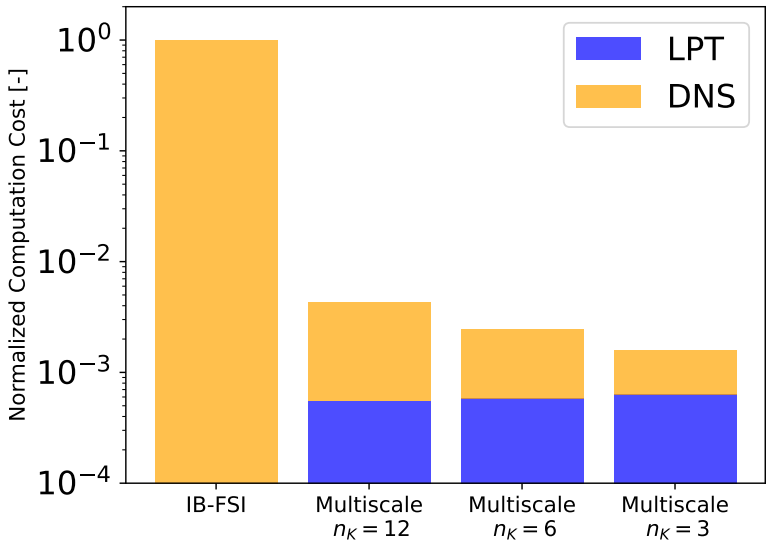


Figure 5.9: Comparison between computational costs associated with running a pure DNS based on IB-FSI framework and the corresponding multiscale simulation using the memory kernel-based LPT framework. Here, n_K refers to the number of memory kernels used to characterize the hydrodynamic effects the fluid in this particular domain has on the particle.

of kernels in the library. So a lower number of kernels would reduce the inherent cost while also reducing the accuracy of the simulation as discussed earlier. A balance between the number kernels needed and the hydrodynamic information required for the simulation needs to be figured out according to the domain. While the changes in the theoretical estimate of the added mass has been used in this work, alternate approaches could be used depending on the simulations scenario. It is also possible to couple the LPT part of the simulation with the DNS required for kernel determination. In this way, new kernels can be determined *on-the-fly* when the particle position has undergone significant changes that would change its hydrodynamic neighbourhood. Alternatively, a machine learning (ML) model could be trained to produce the kernel associated with a particle based on its position in the domain. Here, the cost of training the model then becomes the initial one-time cost to run multiple LPT simulations later.

Chapter 6

Conclusions

In this thesis, two novel methods have been presented that are capable of simulating Brownian phenomena. Both these methods, based on the GLE, are especially useful for simulating Brownian motion in a mixture where the particle-fluid density ratio is very low ($\rho_p/\rho_f \approx 1$). The first method is a pure DNS method that determines the hydrodynamic force acting on the particle based on the numerical solution of the Navier Stokes equation. The hydrodynamic force, along with the velocity history of the particle, is then used to optimise the memory kernel associated with the particle. The colored random Brownian force required for the particle dynamics is generated using the memory kernel. The method is able to accurately capture the Brownian phenomena of a neutrally buoyant particle in an unhindered as well as wall-adjacent scenario. The memory kernel associated with the simulations is shown to capture the drag, added mass and history effects associated with the particle motion.

However, the DNS method can be computationally quite expensive, leading to the development of the multiscale LPT method. In this method, both the hydrodynamic force and the random Brownian force required for the GLE are obtained using the memory kernel associated with the current particle position. The method relies on the ability of the memory kernel to capture the non-local hydrodynamic effects associated with the particle dynamics. The memory kernels required for the simulations are initially created by running short DNS simulations at different positions in the domain. The changes in the shape of the memory kernels can be used to interpret the strength of the hydrodynamic effects in different parts of the domains. The capabilities of the method in analysing microfluidic devices are demonstrated by simulating the migration of a Brownian particle towards a wall under the influence of an attractive force. The position of impact of Brownian particles and ways to control them are of relevance in multiple microfluidic applications[6]–[10] . While the increased residence time in the domain due to the hindrance of the wall provides more time for diffusion, the decreased particle diffusivity close to the wall reduces the rate of diffusion close to the wall. The balance between the two determines the final impact region of the particle. Finally, it is shown that the multiscale

LPT method is significantly more efficient than the pure DNS method, while still capturing the relevant flow physics required for low particle-fluid density mixtures.

6.1 Outlook for the Future

While significant progress has been made towards achieving the goals listed initially in chapter 2, there is still more that can be done. The methods in their current form are capable of simulating spherical particles in symmetric geometries since only the diagonal terms of the memory kernel are currently used for the simulations. The method can be extended in the future to non-spherical particles and irregular geometries by optimising for the non-diagonal terms as well. The short DNS used in the multiscale method is especially suitable for this, since a steady force is used throughout the simulation while optimising the kernel, thereby reducing the amount of numerical noise that will creep into the kernel values. The simulation of rotation also becomes more relevant for non-spherical particles, as the orientation of the particle can influence the Brownian dynamics. The memory kernel tensor then needs to be extended to a 6×6 symmetric matrix where the correlations between the rotational and translational motion in every direction can be accounted for, while modelling the Brownian phenomena. In this case, the hydrodynamic torque experienced by the body combined with the angular velocity history can be used in similar manner, along with their translational counterparts, to optimise the net hydrodynamic memory kernel. As previously mentioned, the memory kernel shape can depend on the fluid properties, the particle geometry and the domain shape. These factors can be used to train an ML model capable of predicting the memory kernel associated with a particle depending on the simulation scenarios. Such a pretrained model would open up the doors to simulating more complex microfluidic phenomena without much effort from a new user of the method.

The simulation methods in their current form already allow for the study of many microfluid phenomena, as depicted using the study of particle migration. The "fingerprint" nature of the memory kernel in showcasing the hydrodynamic nature of the particle domain is especially useful in understanding the strength of various effects the particle undergoes. A multiparticle simulation is quite straightforward in this case, since the memory kernel through the hydrodynamic force is capable of accounting for the hindrances introduced to the particle dynamics from the presence of neighbouring particles. The presence of multiple fluids can also be accounted for since changes in the density ratio and viscosity would cause changes in hydrodynamics, which will reflect in the kernel shape as well. The hydrodynamic effect of chemical reactions and the associated changes to concentration gradients in the system can also be studied using the kernel based methods. Thus, the kernel based methods presented in this work open up the possibilities to study more complex microfluidic systems in the future while accounting for accurate flow physics relevant to the system.

Chapter 7

Summary of Included Papers

7.1 Paper A

A. J. Michael, A. Mark, S. Sasic, H. Ström. *Generalized Langevin dynamics in multiphase direct numerical simulations using hydrodynamically optimized memory kernels*, Physics of Fluids 37 (2025) 033317.
DOI: https://doi.org/10.1063/5.0254930

Contribution

A novel pure Direct Numerical Simulation (DNS) method based on the generalised Langevin equation (GLE) is presented in this paper capable of accurately simulating Brownian motion at a wide range of particle-fluid density ratios ($\rho_p/\rho_f \in [1, 1000]$). The method is capable of capturing the effects of the added mass and history force, which are of relevance at lower particle-fluid density ratios. The GLE requires the total hydrodynamic force acting on the particle along with a random colored Brownian force. The hydrodynamic force is calculated by numerically solving the Navier-Stokes equations for the flow around the particle. Alternately, the hydrodynamic force can also be expressed as the convolution integral between a memory kernel and the velocity history of the particle. This relation is used to obtain the memory kernel associated with the particle through an optimisation routine. The random Brownian force is then generated from the fluctuation-dissipation relation that relates the covariance of the Brownian forces with the memory kernel. The net force, combining the hydrodynamic and Brownian contributions, is used to evolve the particle trajectory.

Results and Discussions

The method is used to simulated unhindered and wall-adjacent Brownian motion for a spherical particle. The Mean Square Displacement (MSD) and Velocity

Autocorrelation Function (VACF) are calculated in each case to compare with theoretical and experimental trends. In the unhindered case, the particle-fluid density ratio is varied from 1 to 1000 in multiples of 10. Here, simulations using the classical Langevin equation (CLE) is also used to show the improvements brought about by a GLE based method. The MSD is seen to closely follow theoretical trends with a clear ballistic regime in the short term that tends to a pure diffusion regime in the long term. The VACF also matches well with the theoretical values, with the expected slower decay in correlations that is expected at lower particle fluid density ratios. The CLE based simulations, in contrast, show under predications of the MSD and VACF at low particle fluid density ratios, while at higher ones they match well with the theory and GLE based simulations. The wall-adjacent simulations are carried at density ratios of 1 and 10 at a distance of $1.25r_p$ from the wall. The MSD shows a decrease in the normal and parallel direction that matches well with the theoretically expected diffusion. The VACF also shows faster decay in the normal and parallel directions, which matches the theoretical and experimental trends associated with wall-adjacent diffusion. The changes to the MSD and VACF are more pronounced in the wall-normal direction compared to the wall-parallel direction. Finally, the memory kernels obtained in the different cases as part of the simulation process are discussed to highlight how their shapes capture the change in the density of the fluid and particle position. Also, the association of different parts of the kernel with different hydrodynamic effects (drag, added mass and history effects) is discussed. Thus, the kernels can be used as a "fingerprint" for the hydrodynamic effects associated with complex fluid flow scenarios.

7.2 Paper B

A. J. Michael, A. Mark, S. Sasic, H. Ström.*A nonlocal multiscale model for Brownian particles: application to hindered deposition in microfluidic systems*, Submitted to a scientific journal

Contribution

A novel multiscale Lagrangian Particle Tracking (LPT) method based on the GLE is presented. The method uses hydrodynamic memory kernels to generate the hydrodynamic force and the random colored Brownian force required for GLE. The generation of the hydrodynamic force from the memory kernel removes the need for explicit analytical models for the force, especially while accounting for the hindrance effects from features in the domains such as walls. The method is used to study the migration of a particle towards a wall under the influence of an attractive force, which is of relevance in many microfluidic applications. The memory kernel library mapping the hydrodynamic effects at different positions in the simulation domain are obtained by running short DNS at the required positions. The kernels used during the LPT simulation of the Brownian particle migration is then obtained by linear interpolation

between the kernels in the library. The hydrodynamic force and Brownian force generated from the kernel are then used to propagate the particle using the GLE.

Results and Discussions

The method is used to simulate Brownian particle migration in domains with two walls, where the separation between the walls is set to $10r_p$ and $30r_p$. The memory kernel libraries for the two domains obtained for the two domains is presented. The variation in the kernel shape across the different positions in the domain and their association with wall effects are presented. LPT simulations are run in both domains, with an attractive force pulling the particle towards one of the walls from the centre of the domain. The strength of the pulling force is set to $m_p g$ and $5m_p g$ in different runs of each domain. The impact radius of the particle on the domain wall is seen to increase with the domain size, since the particle has a longer duration to diffuse parallelly before striking the wall. Meanwhile, an increase in the pulling force decreases the impact radius since the particle has less time to diffuse parallelly. The effects of hindrance on the particle migration are also analysed by simulating the motion of the particle in an unhindered scenario, where the attracting wall is considered to be far from the particle. In this case, the impact radius is higher in the hindered case compared to the unhindered case, but one of the cases shows a very close value between the two. This lack of a clear pattern can be explained by the increased hydrodynamic resistance introduced by the wall. Although the increased hydrodynamic resistance from the wall reduces the diffusion close to it, it also increases the duration for diffusion, since the particle takes longer to cover the same distance. The multiscale LPT method is also shown to be computationally less expensive than the pure DNS method while still capturing accurate flow physics.

Bibliography

[1] X. Bian, C. Kim and G. E. Karniadakis, "111 years of brownian motion," *Soft Matter*, vol. 12, no. 30, pp. 6331–6346, 2016 (cit. on pp. 3, 4, 10, 11, 14, 23, 33, 40, 43).

[2] P. Hänggi and F. Marchesoni, "Artificial brownian motors: Controlling transport on the nanoscale," *Reviews of Modern Physics*, vol. 81, no. 1, pp. 387–442, 2009 (cit. on pp. 3, 4).

[3] L. Fan, Y. Wu, E. Jian *et al.*, "Diffusiophoresis of a highly charged dielectric fluid droplet induced by diffusion potential," *Physics of Fluids*, vol. 34, no. 4, 2022 (cit. on pp. 3, 5).

[4] S. Shin, "Diffusiophoretic separation of colloids in microfluidic flows," *Physics of Fluids*, vol. 32, no. 10, 2020 (cit. on p. 3).

[5] H. Liu and A. A. Pahlavan, "Diffusioosmotic reversal of colloidal focusing direction in a microfluidic t-junction," *Physical Review Letters*, vol. 134, no. 9, p. 098 201, 2025 (cit. on p. 3).

[6] B. Çetin and D. Li, "Dielectrophoresis in microfluidics technology," *Electrophoresis*, vol. 32, no. 18, pp. 2410–2427, 2011 (cit. on pp. 3, 26, 47).

[7] C. Midelet, B. Le Pioufle and M. H. Werts, "Brownian motion and large electric polarizabilities facilitate dielectrophoretic capture of sub-200 nm gold nanoparticles in water," *ChemPhysChem*, vol. 20, no. 24, pp. 3354–3365, 2019 (cit. on pp. 3, 26, 47).

[8] T. P. Forbes and S. P. Forry, "Microfluidic magnetophoretic separations of immunomagnetically labeled rare mammalian cells," *Lab on a Chip*, vol. 12, no. 8, pp. 1471–1479, 2012 (cit. on pp. 3, 26, 47).

[9] E. P. Furlani, "Magnetic biotransport: Analysis and applications," *Materials*, vol. 3, no. 4, pp. 2412–2446, 2010 (cit. on pp. 3, 5, 26, 47).

[10] H. Ma, E. F. Pazmino and W. P. Johnson, "Gravitational settling effects on unit cell predictions of colloidal retention in porous media in the absence of energy barriers," *Environmental science & technology*, vol. 45, no. 19, pp. 8306–8312, 2011 (cit. on pp. 3, 5, 26, 47).

[11] M. Lavaud, T. Salez, Y. Louyer and Y. Amarouchene, "Stochastic inference of surface-induced effects using brownian motion," *Physical Review Research*, vol. 3, no. 3, p. L032011, 2021 (cit. on p. 3).

[12] B. J. Alder and T. Wainwright, "Decay of the velocity autocorrelation function," *Physical review A*, vol. 1, no. 1, p. 18, 1970 (cit. on pp. 3, 11, 13).

[13] H. K. Shin, C. Kim, P. Talkner and E. K. Lee, "Brownian motion from molecular dynamics," *Chemical Physics*, vol. 375, no. 2-3, pp. 316–326, 2010 (cit. on p. 3).

[14] J. C. Chen and A. S. Kim, "Brownian dynamics, molecular dynamics, and monte carlo modeling of colloidal systems," *Advances in colloid and interface science*, vol. 112, no. 1-3, pp. 159–173, 2004 (cit. on pp. 3, 4).

[15] P. Langevin *et al.*, "Sur la théorie du mouvement brownien," *CR Acad. Sci. Paris*, vol. 146, no. 530-533, p. 530, 1908 (cit. on pp. 4, 9).

[16] A. S. Kannan, V. Naserentin, A. Mark *et al.*, "A continuum-based multiphase dns method for studying the brownian dynamics of soot particles in a rarefied gas," *Chemical Engineering Science*, vol. 210, p. 115 229, 2019 (cit. on pp. 4, 9, 14).

[17] J. F. Brady, G. Bossis *et al.*, "Stokesian dynamics," *Annual review of fluid mechanics*, vol. 20, no. 1, pp. 111–157, 1988 (cit. on p. 4).

[18] Y. J. Lee, H. Jin, D. Y. Kim, S. Kang and K. H. Ahn, "Lagrangian-based simulation method using constrained stokesian dynamics for particulate flows in microchannel," *Journal of Fluid Mechanics*, vol. 990, A3, 2024 (cit. on p. 4).

[19] T. Iwashita, Y. Nakayama and R. Yamamoto, "A numerical model for brownian particles fluctuating in incompressible fluids," *Journal of the Physical Society of Japan*, vol. 77, no. 7, p. 074 007, 2008 (cit. on pp. 4, 10).

[20] F. Mainardi, A. Mura and F. Tampieri, "Brownian motion and anomalous diffusion revisited via a fractional langevin equation," *arXiv preprint arXiv:1004.3505*, 2010 (cit. on pp. 4, 9, 10, 13, 14, 23, 32, 33).

[21] E. Hauge and A Martin-Löf, "Fluctuating hydrodynamics and brownian motion," *Journal of Statistical Physics*, vol. 7, pp. 259–281, 1973 (cit. on pp. 4, 11, 12, 15, 18, 19, 27).

[22] A. B. Basset, *A treatise on hydrodynamics: with numerous examples*. Deighton, Bell and Company, 1888, vol. 2 (cit. on pp. 4, 11).

[23] G. Jung, M. Hanke and F. Schmid, "Iterative reconstruction of memory kernels," *Journal of chemical theory and computation*, vol. 13, no. 6, pp. 2481–2488, 2017 (cit. on p. 4).

[24] G. Jung, M. Hanke and F. Schmid, "Generalized langevin dynamics: Construction and numerical integration of non-markovian particle-based models," *Soft matter*, vol. 14, no. 46, pp. 9368–9382, 2018 (cit. on pp. 4, 12, 20).

[25] M. Kerr Winter, I. Pihlajamaa, V. E. Debets and L. Janssen, "A deep learning approach to the measurement of long-lived memory kernels from generalized langevin dynamics," *The Journal of Chemical Physics*, vol. 158, no. 24, 2023 (cit. on p. 4).

[26] P. Xie, R. Car and W. E, "Ab initio generalized langevin equation," *Proceedings of the National Academy of Sciences*, vol. 121, no. 14, e2308668121, 2024 (cit. on p. 4).

[27] Z. Li, X. Bian, X. Li and G. E. Karniadakis, "Incorporation of memory effects in coarse-grained modeling via the mori-zwanzig formalism," *The Journal of chemical physics*, vol. 143, no. 24, 2015 (cit. on p. 4).

[28] A. J. Michael, A. Mark, S. Sasic and H. Ström, "Generalized langevin dynamics in multiphase direct numerical simulations using hydrodynamically optimized memory kernels," *Physics of Fluids*, vol. 37, no. 3, 2025 (cit. on pp. 5, 11, 12, 17, 31, 37).

[29] J. W. Dufty, "Gaussian model for fluctuation of a brownian particle," *Physics of Fluids*, vol. 17, no. 2, pp. 328–333, 1974 (cit. on pp. 9, 10, 12, 19, 35).

[30] G. G. Stokes *et al.*, "On the effect of the internal friction of fluids on the motion of pendulums," 1851 (cit. on p. 10).

[31] H. Ounis, G. Ahmadi and J. B. McLaughlin, "Brownian diffusion of submicrometer particles in the viscous sublayer," *Journal of Colloid and Interface Science*, vol. 143, no. 1, pp. 266–277, 1991 (cit. on p. 10).

[32] A. Li and G. Ahmadi, "Dispersion and deposition of spherical particles from point sources in a turbulent channel flow," *Aerosol science and technology*, vol. 16, no. 4, pp. 209–226, 1992 (cit. on p. 10).

[33] A. Einstein, "Ber die von der molekularkinetischen theorie der w rme geforderte bewegung von in ruhenden fl ssigkeiten suspendierten teilchen," *Annalen der physik*, vol. 322, no. 8, pp. 549–560, 1905 (cit. on pp. 11, 32).

[34] A. Jannasch, M. Mahamdeh and E. Schäffer, "Inertial effects of a small brownian particle cause a colored power spectral density of thermal noise," *Physical Review Letters*, vol. 107, p. 228 301, 2011 (cit. on pp. 11, 12).

[35] T. Franosch, M. Grimm, M. Belushkin *et al.*, "Resonances arising from hydrodynamic memory in brownian motion," *Nature*, vol. 478, no. 7367, pp. 85–88, 2011 (cit. on pp. 11, 12).

[36] D. Bedeaux and P. Mazur, "Brownian motion and fluctuating hydrodynamics," *Physica*, vol. 76, no. 2, pp. 247–258, 1974 (cit. on p. 11).

[37] R. Kubo, "The fluctuation-dissipation theorem," *Reports on progress in physics*, vol. 29, no. 1, p. 255, 1966 (cit. on p. 12).

[38] G. Jung and F. Schmid, "Frequency-dependent hydrodynamic interaction between two solid spheres," *Physics of Fluids*, vol. 29, no. 12, 2017 (cit. on p. 12).

[39] M. Maxey and J. Riley, "Equation of motion for a small rigid sphere in a nonuniform flow," *Physics of Fluids*, vol. 26, no. 4, pp. 883–889, 1983 (cit. on pp. 14, 23).

[40] A. J. Michael, A. Mark, S. Sasic and H. Ström, "A nonlocal multiscale model for brownian particles: Application to hindered deposition in microfluidic systems," *Submitted for publication*, 2025 (cit. on p. 17).

[41] A. Mark and B. G. van Wachem, "Derivation and validation of a novel implicit second-order accurate immersed boundary method," *Journal of Computational Physics*, vol. 227, no. 13, pp. 6660–6680, 2008 (cit. on p. 18).

[42] A. Mark, R. Rundqvist and F. Edelvik, "Comparison between different immersed boundary conditions for simulation of complex fluid flows," *Fluid dynamics & materials processing*, vol. 7, no. 3, pp. 241–258, 2011 (cit. on p. 18).

[43] E. Lauga and T. M. Squires, "Brownian motion near a partial-slip boundary: A local probe of the no-slip condition," *Physics of Fluids*, vol. 17, p. 103 012, 2005 (cit. on pp. 19, 33).

[44] A. Simha, J. Mo and P. J. Morrison, "Unsteady stokes flow near boundaries: The point-particle approximation and the method of reflections," *Journal of Fluid Mechanics*, vol. 841, pp. 883–924, 2018 (cit. on pp. 19, 23, 27, 29, 36).

[45] D. Liu and J Nocedal, "On the limited memory method for large scale optimization," *Mathematical Programming B*, vol. 45, no. 3, 503–528, 1989 (cit. on p. 20).

[46] R. H. Byrd, P. Lu, J. Nocedal and C. Zhu, "A limited memory algorithm for bound constrained optimization," *SIAM J. Sci. Comput.*, vol. 16, no. 5, 1190–1208, 1995 (cit. on p. 20).

[47] C. Zhu, R. H. Byrd, P. Lu and J. Nocedal, "L-bfgs-b: Algorithm 778: L-bfgs-b, fortran routines for large scale bound constrained optimization," *ACM Transactions on Mathematical Software*, vol. 23, no. 4, 550–560, 1997 (cit. on p. 20).

[48] P. Virtanen, R. Gommers, T. E. Oliphant *et al.*, "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python," *Nature Methods*, vol. 17, pp. 261–272, 2020. DOI: 10.1038/s41592-019-0686-2 (cit. on p. 20).

[49] N. Newmark, "A method of computation for structural dynamics," *J. Eng. Mech. Divis.*, vol. 85, pp. 67–94, 1959 (cit. on p. 22).

[50] P. S. Foundation, *The python standard library: Random — generate pseudo-random numbers*, 2025. [Online]. Available: https://docs.python.org/3/library/random.html (cit. on p. 29).

[51] A. Goldman, R. Cox and H Brenner, "Slow viscous motion of a sphere parallel to a plane wall—i motion through a quiescent fluid," *Chemical Engineering Science*, vol. 22, pp. 637–651, 1967 (cit. on p. 33).

[52] M. Bevan and D. Prieve, "Hindered diffusion of colloidal particles very near to a wall: Revisited," *Journal of Chemical Physics*, vol. 113, pp. 1228–1236, 2000 (cit. on p. 33).

[53] A. Banerjee and K. D. Kihm, "Experimental verification of near-wall hindered diffusion for the brownian motion of nanoparticles using evanescent wave microscopy," *Physcial Review E*, vol. 72, p. 042 101, 2005 (cit. on p. 33).

[54] A. S. Kannan, A. Mark, D. Maggiolo, G. Sardina, S. Sasic and H. Ström, "A hydrodynamic basis for off-axis brownian diffusion under intermediate confinements in micro-channels," *International Journal of Multiphase Flow*, vol. 143, p. 103 772, 2021 (cit. on pp. 33, 36).

[55] B. Felderhof, "Effect of the wall on the velocity autocorrelation function and long-time tail of brownian motion," *The Journal of Physical Chemistry B*, vol. 109, no. 45, pp. 21 406–21 412, 2005 (cit. on pp. 34, 35).

[56] S. Jeney, B. Lukic, J. A. Kraus, T. Franosch and L. Forró, "Anisotropic memory effects in confined colloidal diffusion," *Physical Review Letters*, vol. 100, p. 240 604, 2008 (cit. on pp. 34, 35).